Modeling and simulating three phases of steel

Austenite, ferrite and cementite

T. Verbeek

Modeling and simulating three phases of steel

Austenite, ferrite and cementite

by

T. Verbeek

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday April 19, 2018 at 13:15 PM.

Student number:	4102002	
Project duration:	January 1, 2017 to April 19, 2018	
Thesis committee:	Prof. dr. ir. C. Vuik,	TU Delft, Full professor
	Dr. ir. D.R. van der Heul,	Associate Professor
	Dr. ir. W.T. van Horssen,	Associate Professor
	Dr. ir. D. den Ouden- van der Horst,	Lecturer

Abstract

Abstract: In the work of Den Ouden [18] the level-set method has been used to model the growth and dissolution in a steel alloy of the precipitate cementite and the diffusive phase austenite. The movement of the interface between the two phases is controlled by the diffusion of carbon and a reaction on the interface between the two phases. In this thesis research a similar model and method is given for a three-phased model with an extra diffusive phase. This third phase is ferrite, which nucleates on the interface of austenite and cementite when the temperature is dropped below the eutectoid temperature for a high carbon steel alloy. The difference between austenite and ferrite is the matrix structure of the iron atoms. This influences how much carbon the phase can contain. The difference in structure also results in different properties of the steel with respect to strength, hardness, etc. Modeling the growth of this three-phased steel alloy could eventually lead to useful insights for high-carbon steel production.

Extra difficulties arise by the introduction of a second diffusive phase. These difficulties concern defining sufficient interface boundary conditions between the different phases, capturing the different phases with the level-set method, creating a qualitative good mesh capturing these phases, recovering gradients on an interface and getting correct carbon concentration value in the triple points because of the interface boundary conditions. Except for the latter problem, the results of this research show decent results. The effect of cooling on the growth of ferrite is also shown and explained.

Contents

1

Introduction

Steel is an important material used in the construction of many different things. Steel is originally an alloy of iron (Fe) and a small amount of carbon (C), normally up to about 2.1% of the total weight. Steel with 0.05% to 0.3% carbon content is considered as low-carbon steel, it is used in a big variety of objects ranging from flat sheets to structural beam support. Steel with 0.3% up to 0.6% carbon content is called medium carbon steel and has its uses in for example high pressure containers, railway wheels and rails and structures like bridges that forces high pressure and torque on the structure. This type of steel is stronger than low carbon steel, but more difficult to handle and is more brittle, meaning it has to be properly insulated to prevent it being affected by high or low temperatures. High carbon steel with 0.6% to 2.1% carbon content is even stronger and becomes very hard and brittle once heat treated. It is used for things as springs, swords, cutleries, gardening tools and other high wear equipment. To improve certain properties of steel other alloying elements such as manganese, nickel, chromium, molybdenum, boron, titanium, vanadium, tungsten, cobalt, and niobium can be added.
In this thesis research we will look at the phase transformation in steel with a high carbon content (above 0.75% C). At temperatures above 1000 K (known as the eutectoid temperature and denoted as A_1), the alloy in equilibrium state will consist of the phases austenite and cementite, whereas just below 1000 K the equilibrium state is an alloy of ferrite and cementite. The different equilibrium states that exist at certain temperatures and carbon contents can be found in the phase diagram in Figure 1.1. In the austenite phase the iron atoms are structured in a face-centered cubic (FCC) crystalline form and ferrite has a body-centered cubic (BCC) crystalline form (see Figure 1.2). The carbon atoms are located at the octahedral sites between the iron atoms. Because of the difference in structure, austenite and ferrite contain different quantities of carbon. Both the structure and the carbon concentration influence the hardness, ductility and tensile strength of the steel. For steel alloys with a high percentage of carbon content, cementite (θ) can also be present. Cementite is a stoichiometric compound with formula Fe_3C, meaning 6.67% of its weight is carbon and 93.3% is iron. In the austenite-cementite alloy the cementite is located as small grains between larger austenite grains (See Figure 1.3). When the temperature is lowered, free energy builds up on the interfaces between austenite and cementite. Once the temperature passes the eutectoid temperature enough energy builds up for a ferrite nucleus to nucleate. The ferrite and cementite will both grow and alternately cementite and ferrite nucleus will be created to make up for the carbon movement and forms pearlite. This two-phased structure is build up of narrow alternating sheets of ferrite and cementite (see Figure 1.4). The pearlite will grow further into the austenite until all austenite has dissolved or the temperature is too low to continue the process.

The transition from austenite to ferrite can be modeled by a so-called Stefan problem. It describes the diffusion of carbon in the different phases and the movement of the interfaces between different phases. When numerically solving the Stefan problem, keeping track of the interfaces can be complicated, especially when there are multiple phases connected. In this project we model the mechanism of ferrite growth just after nucleating at the interface between austenite and cementite (see Figure 1.5). The goal of this thesis project is to make a two-dimensional implementation using the level-set method. Some choices regarding the numerical implementation and the physical parameters will be based upon the results from the one-dimensional model studied in [26].

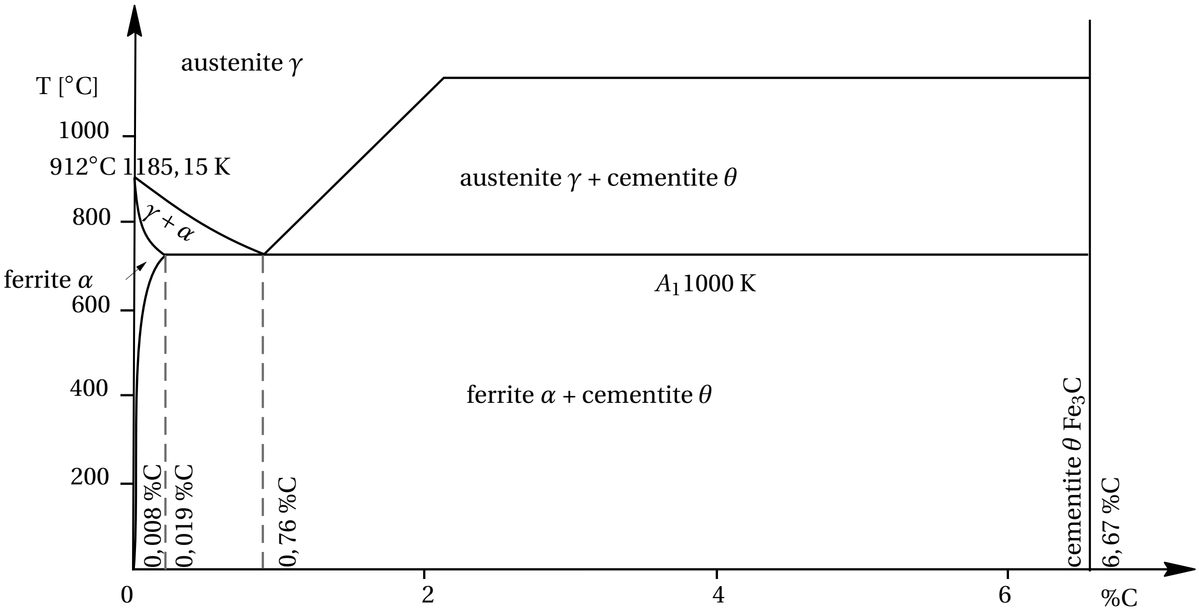


Figure 1.1: Partial phase diagram of steel, showing the different equilibrium states depending on temperature (°C) and carbon weight percentage (wt %). The lines are called the lines of equilibrium or phase boundaries. Crossing these lines means phase transformations occur.

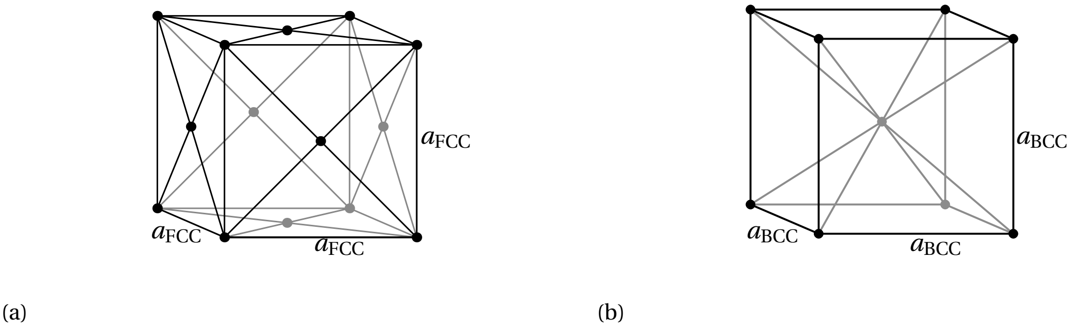


Figure 1.2: FCC (figure (a)) and BCC (figure (b)) crystal structures with a the lattice length.

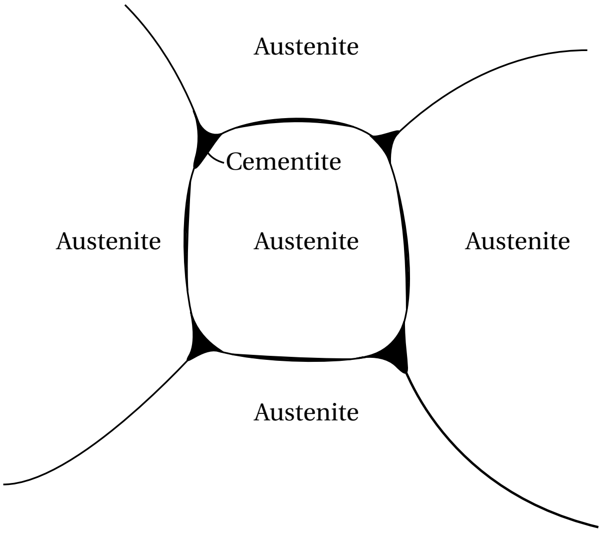


Figure 1.3: Visualization of cementite located in austenite. The darker areas represent the cementite, which are located on the austenite-austenite grains.

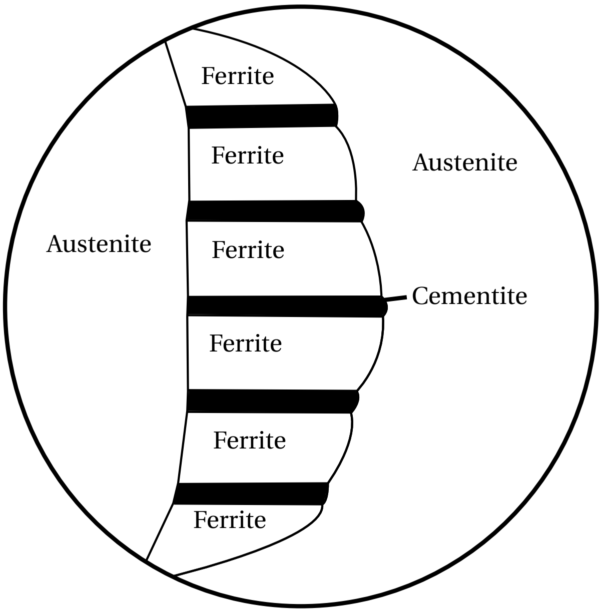


Figure 1.4: Zoomed in visualization of a pearlite plate growing in austenite. Recreated from the image in [22].

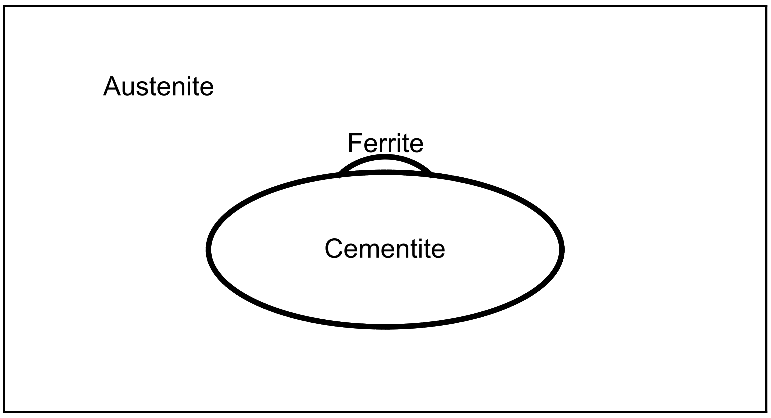


Figure 1.5: Sketch of the nucleation of ferrite on the interface of austenite and cementite.

In Chapter 2 the two-dimensional model is given. It is obtained by defining the partial differential equations of the carbon diffusion in the different phases and the boundary conditions of the different phase interfaces and the outer boundaries. These boundary conditions are derived by using several physical properties and assumptions. Furthermore the initial shapes of the austenite, ferrite and cementite domains are given based on the interface energies present at the eutectoid temperature. In Chapter 3 the level-set method for two and multiple phases is shown. This method uses a signed-distance function to implicitly give the location of an interface and the changing phases over time. With three or more phases extra difficulties arise compared to two phases when using this method. These difficulties will be shown and handled here. In Chapter 4 a discretization of the model and the level-set update and reinitialization is given using the Finite Elements Method. The more practical details of the discretization are handled in Chapter 5. This concerns mesh generation, fixing a dimensional mismatch, approximating gradients and using the Algebraic Flux Correction Method to stabilize the Finite Elements approximation of the model. In Chapter 6 the physical parameters of the model are either given or derived when not already available in literature. Then in Chapter 7 the results are shown and discussed. Finally, in Chapter 8 the conclusions of this report are presented and in Chapter 9 ideas for future work is given.

2
Model

The model presented here is based on the Stefan problem and is a continuation on the model described in Den Ouden's PhD thesis [18]. His work models a two-phase domain with interaction between a matrix and a precipitate, austenite (γ) and cementite (θ) respectively. In this research we work with a second matrix phase ferrite (α). We will assume we have a ferrite nucleus at $T = 995$ K in a domain with a single cementite particle in austenite. Over time, if the temperature stays high enough for the reaction to continue, all austenite should dissolve and ferrite should mostly replace it. This dispersion/growth is controlled by carbon diffusion in the matrices and the behavior of the three interfaces between austenite and ferrite, austenite and cementite and ferrite and cementite. Each of these interfaces have different reactions rates between the phases and need a corresponding physical boundary condition. The partial differential equation of the Stefan problem described in the phase domains, prescribing the evolution of the carbon concentration, is the following:

$$\begin{cases} \frac{\partial c_\gamma}{\partial t}(\boldsymbol{x},t) & = & \nabla\cdot\big(D_\gamma(\boldsymbol{x},t)\nabla c_\gamma(\boldsymbol{x},t)\big) & \text{, for } \boldsymbol{x}\in\Omega_\gamma(t), t\geq t_0, \\ c_\gamma(\boldsymbol{x},t) & = & c_\gamma^0 & \text{, for } \boldsymbol{x}\in\Omega_\gamma(t), t = t_0, \\ \frac{\partial c_\alpha}{\partial t}(\boldsymbol{x},t) & = & \nabla\cdot(D_\alpha(\boldsymbol{x},t)\nabla c_\alpha(\boldsymbol{x},t)) & \text{, for } \boldsymbol{x}\in\Omega_\alpha(t), t\geq t_0, \\ c_\alpha(\boldsymbol{x},t) & = & c_\alpha^0 & \text{, for } \boldsymbol{x}\in\Omega_\alpha(t), t = t_0, \\ c_\theta(\boldsymbol{x},t) & = & c_\theta & \text{, for } \boldsymbol{x}\in\Omega_\theta(t), t\geq t_0, \end{cases} \tag{2.1}$$

where $\Omega_\gamma(t)$ and $\Omega_\alpha(t)$ are the diffusion domain of austenite and ferrite matrix phases respectively. $\Omega_\theta(t)$ is the precipitate domain of cementite, where the concentration of carbon is assumed constant. This assumption is made because the carbon atoms in cementite are able to diffuse almost instantly compared to austenite and ferrite and thus believed to have the same carbon concentration level over its domain. $c_\gamma^0 := c_\gamma(\boldsymbol{x}, t_0)$ and $c_\alpha^0 := c_alpha(\boldsymbol{x}, t_0)$ are the initial solutions at $t = t_0$.

The three interfaces between the three domains will have their own boundary conditions, which are considered in the following section. The interfaces between austenite-ferrite, austenite-cementite and ferrite-cementite will be denoted as $\Gamma^{\gamma\alpha}(t), \Gamma^{\gamma\theta}(t)$ and $\Gamma^{\alpha\theta}(t)$ respectively. The order in which the two symbols are notated will also define the normal vectors' direction on these interfaces. Meaning the normal vector $\boldsymbol{n}^{\gamma\alpha}$ on $\Gamma^{\gamma\alpha}(t)$ points from $\Omega_\gamma(t)$ to $\Omega_\alpha(t)$ and $\boldsymbol{n}^{\gamma\alpha} = -\boldsymbol{n}^{\alpha\gamma}$. The total domain is the union of $\Omega_\gamma(t), \Omega_\alpha(t), \Omega_\theta(t)$, the interfaces $\Gamma^{\gamma\alpha}(t), \Gamma^{\gamma\theta}(t), \Gamma^{\alpha\theta}(t)$ and the outer boundary $\partial\Omega(t)$. It is defined as $\overline{\Omega(t)} = \overline{\Omega_\gamma}(t) \cup \overline{\Omega_\alpha}(t) \cup \overline{\Omega_\theta}(t)$, where $\overline{\Omega}(t)$ is defined as the domain including its boundaries.

2.1. Boundary Conditions

2.1.1. Outer Boundary

At the outer boundary $\partial\Omega(t)$ no carbon atoms can get in or out of the system. Which means:

$$\frac{\partial c_k}{\partial \boldsymbol{n}}(\boldsymbol{x},t) = 0, \text{ for } \boldsymbol{x}\in\partial\Omega^k(t), k=\gamma,\alpha, t\geq t_0. \tag{2.2}$$

This also means the shape of $\Omega(t)$ does not change, which implies $\Omega(t) = \Omega$ and $\partial\Omega(t) = \partial\Omega$. A homogeneous Neumann boundary condition is practical and realistic, as it will imply no change in mass over Ω. Another option for boundary condition on $\partial\Omega$ is a periodic extension of c_γ and c_α, which will represent an infinitely large domain, but this will not be considered in this thesis.

2.1.2. Stefan Condition

An important condition on the model is the conservation of mass M. As the flux over the outer boundary is zero, we know that the total change in mass over Ω should also be zero. If we look at the domains individually, the changes in mass should add up to zero, meaning:

$$\frac{dM(t)}{dt} = \frac{d\left(M_\gamma(t) + M_\alpha(t) + M_\theta(t)\right)}{dt} = 0. \tag{2.3}$$

This condition is commonly known as the Stefan condition and for this model gives (see [26] for the derivation):

$$\begin{aligned}
\left(c_\gamma(\boldsymbol{x},t) - c_\theta\right) v_n^{\gamma\theta}(\boldsymbol{x},t) + D_\gamma(\boldsymbol{x},t)\frac{\partial c_\gamma}{\partial \boldsymbol{n}}(\boldsymbol{x},t) &= 0, && (2.4)\\
&\quad \text{for } \boldsymbol{x} \in \Gamma^{\gamma\theta}(t), t \geq t_0. \\
\left(c_\alpha(\boldsymbol{x},t) - c_\theta\right) v_n^{\alpha\theta}(\boldsymbol{x},t) + D_\alpha(\boldsymbol{x},t)\frac{\partial c_\alpha}{\partial \boldsymbol{n}}(\boldsymbol{x},t) &= 0, && (2.5)\\
&\quad \text{for } \boldsymbol{x} \in \Gamma^{\alpha\theta}(t), t \geq t_0. \\
\left(c_\gamma(\boldsymbol{x},t) - c_\alpha(\boldsymbol{x},t)\right) v_n^{\gamma\alpha}(\boldsymbol{x},t) + D_\gamma(\boldsymbol{x},t)\frac{\partial c_\gamma}{\partial \boldsymbol{n}}(\boldsymbol{x},t) - D_\alpha(\boldsymbol{x},t)\frac{\partial c_\alpha}{\partial \boldsymbol{n}}(\boldsymbol{x},t) &= 0, && (2.6)\\
&\quad \text{for } \boldsymbol{x} \in \Gamma^{\gamma\alpha}(t), t \geq t_0.
\end{aligned}$$

Note that there are two unknowns within Equation (2.4), $c_\gamma(\boldsymbol{x},t)$ and $v_n^{\gamma\theta}(\boldsymbol{x},t)$. Also two within Equation (2.5), $c_\alpha(\boldsymbol{x},t)$ and $v_n^{\alpha\theta}(\boldsymbol{x},t)$ and three within Equation (2.6), $c_\gamma(\boldsymbol{x},t), c_\alpha(\boldsymbol{x},t)$ and $v_n^{\gamma\alpha}(\boldsymbol{x},t)$. Meaning that there is one more condition required for (2.4) and (2.5) and two more conditions for (2.6) to make the model sufficient. In the remainder of this chapter we will give the boundary conditions and the initial conditions c_γ^0 and c_α^0.

2.1.3. Matrix-Precipitate Boundary

There are three physical phenomena active [18] at the matrix-precipitate interfaces $\Gamma^{\gamma\theta}(t), \Gamma^{\alpha\theta}(t)$:

1. Atoms coming lose from the lattice structure of the precipitate phase.
2. Moving atoms from within the precipitate going into the matrix.
3. Long distance diffusion of atoms in the matrix.

At the matrix-precipitate boundary many models assume that the diffusive behavior limits the growth/dissolution of the model and neglects the possible influence of the first two reaction-like phenomenon given above. In the work of Vermolen [27] it has been shown that these interface reactions can have a significant influence on the dissolution for a plate-like precipitate. When neglecting these effects a simple Dirichlet boundary can be chosen to complement the Stefan condition for the matrix-precipitate interfaces. Physically this means the carbon concentration is at local equilibrium on the interfaces at all time. We will choose the more general reaction boundary condition and show it is a combination of the two effects.

Reaction Boundary Condition

The first and second phenomena given earlier across the interface are assumed to be a first-order reaction in terms of its flux:

$$J_r^{k\theta}(\boldsymbol{x},t) = -K^{k\theta}(\boldsymbol{x},t)\left(c_{k\theta}^{\text{sol}}(\boldsymbol{x},t) - c_k(\boldsymbol{x},t)\right), \text{ for } \boldsymbol{x} \in \Gamma^{k\theta}(t), t \geq t_0, k = \gamma, \alpha. \tag{2.7}$$

Here $K^{k\theta}(\boldsymbol{x},t)$ is the interface-reaction speed and $c^{\text{sol}}_{k\theta}(\boldsymbol{x},t)$ the equilibrium carbon concentration, or also known as the solubility.
The fluxes within the diffusive domains $\Omega_\gamma(t)$ and $\Omega_\alpha(t)$ at the interfaces consist of two parts. The flux $J^{k\theta}_m(\boldsymbol{x},t)$ generated by movement of the interface

$$J^{k\theta}_m(\boldsymbol{x},t) = -c_k(\boldsymbol{x},t)v^{k\theta}_n(\boldsymbol{x},t), \text{ for } \boldsymbol{x} \in \Gamma^{k\theta}(t), t \geq t_0, k = \gamma, \alpha, \tag{2.8}$$

and the diffusive flux

$$J^{k\theta}_d(\boldsymbol{x},t) = -D_k(\boldsymbol{x},t)\frac{\partial c_k}{\partial \boldsymbol{n}}(\boldsymbol{x},t), \text{ for } \boldsymbol{x} \in \Gamma^{k\theta}(t), t \geq t_0, k = \gamma, \alpha. \tag{2.9}$$

Requiring the net flux to be zero, gives the flux boundary condition:

$$-K^{k\theta}(\boldsymbol{x},t)\left(c^{\text{sol}}_{k\theta}(\boldsymbol{x},t) - c_k(\boldsymbol{x},t)\right) = -c_k(\boldsymbol{x},t)v^{k\theta}_n(\boldsymbol{x},t) - D_k(\boldsymbol{x},t)\frac{\partial c_k}{\partial \boldsymbol{n}}(\boldsymbol{x},t), \tag{2.10}$$
$$\text{for } \boldsymbol{x} \in \Gamma^{k\theta}(t), t \geq t_0, k = \gamma, \alpha.$$

Together with the Stefan condition (2.4) and (2.5) we obtain

$$v^{k\theta}_n(\boldsymbol{x},t) = \frac{K^{k\theta}(\boldsymbol{x},t)}{c_\theta}\left(c^{\text{sol}}_{k\theta}(\boldsymbol{x},t) - c_k(\boldsymbol{x},t)\right), \text{ for } \boldsymbol{x} \in \Gamma^{k\theta}(t), t \geq t_0, k = \gamma, \alpha, \tag{2.11}$$

and consequently

$$D_k(\boldsymbol{x},t)\frac{\partial c_k}{\partial \boldsymbol{n}}(\boldsymbol{x},t) = \frac{K^{k\theta}(\boldsymbol{x},t)}{c_\theta}\left(c_\theta - c_k(\boldsymbol{x},t)\right)\left(c^{\text{sol}}_{k\theta}(\boldsymbol{x},t) - c_k(\boldsymbol{x},t)\right), \tag{2.12}$$
$$\text{for } \boldsymbol{x} \in \Gamma^{k\theta}(t), t \geq t_0, k = \gamma, \alpha.$$

Note that from this we can see that

$$c_k(\boldsymbol{x},t) = c^{\text{sol}}_{k\theta}(\boldsymbol{x},t) - \frac{c_\theta}{K^{k\theta}(\boldsymbol{x},t)}v^{k\theta}_n(\boldsymbol{x},t), \text{ for } \boldsymbol{x} \in \Gamma^{k\theta}(t), t \geq t_0, k = \gamma, \alpha. \tag{2.13}$$

This implies that for large $K^{k\theta}(\boldsymbol{x},t)$, $c_k(\boldsymbol{x},t)$ will almost be at local equilibrium. Indicating the process is diffusion controlled. For small $K^{k\theta}(\boldsymbol{x},t)$ the velocity term will have a significant influence on the carbon concentration, resulting in a reaction controlled system. Having both effects influencing the behavior is often called the mixed-mode character of the austenite/ferrite to cementite transformation kinetics.

2.1.4. Matrix-Matrix Boundary

The matrix-matrix boundary $\Gamma^{\gamma\alpha}(t)$ is called a grain boundary. This kind of boundary will try to move to a position such that it reduces the total free energy. A para-equilibrium of the carbon concentration is assumed. This means one of the concentration values is set constant on the interface, while the other is still variable. Then we need one more condition. This will be a reaction boundary condition like the matrix-precipitate boundary condition. In other research for phase transformation in steel, often a mobility condition is prescribed (see for instance [8]). In previous work [26] we showed the similarity between the two relations.

Reaction Boundary Condition

Just like the matrix-precipitate boundary a linear reaction flux across the interface can be applied, with flux

$$J^{\gamma\alpha}_r(\boldsymbol{x},t) = -K^{\gamma\alpha}(\boldsymbol{x},t)\left(c^{\text{sol}}_{\gamma\alpha}(\boldsymbol{x},t) - c_\gamma(\boldsymbol{x},t)\right), \text{ for } \boldsymbol{x} \in \Gamma^{\gamma\alpha}(t), t \geq t_0. \tag{2.14}$$

Together with the Stefan condition there is still one more condition required. A Dirichlet condition for $c_\alpha(\boldsymbol{x},t)$ is set as

$$c_\alpha(\boldsymbol{x},t) = c^{\text{sol}}_{\alpha\gamma}(\boldsymbol{x},t), \text{ for } \boldsymbol{x} \in \Gamma^{\gamma\alpha}(t), t \geq t_0. \tag{2.15}$$

This assumption is made because the diffusion coefficient $D_\alpha(\boldsymbol{x},t)$ is significantly bigger than $D_\gamma(\boldsymbol{x},t)$ ($8.7 \cdot 10^{-11}$ m^{2}s^{-1} versus $5.6 \cdot 10^{-13}$ m^{2}s^{-1} at $T = A_1$ respectively), meaning that the carbon in ferrite will diffuse

faster to its equilibrium concentration (on the austenite-ferrite interface).
Filling in the Dirichlet condition in the Stefan condition gives:

$$D_\gamma(\boldsymbol{x},t)\frac{\partial c_\gamma}{\partial \boldsymbol{n}}(\boldsymbol{x},t) - D_\alpha(\boldsymbol{x},t)\frac{\partial c_\alpha}{\partial \boldsymbol{n}}(\boldsymbol{x},t) = v_n^{\gamma\alpha}(\boldsymbol{x},t)\left(c_{\alpha\gamma}^{\text{sol}}(\boldsymbol{x},t) - c_\gamma(\boldsymbol{x},t)\right), \quad \text{for } \boldsymbol{x}\in\Gamma^{\gamma\alpha}(t), t\geq t_0. \tag{2.16}$$

The flux terms are:

$$J_r^{\gamma\alpha}(\boldsymbol{x},t) = -K^{\gamma\alpha}(\boldsymbol{x},t)\left(c_{\gamma\alpha}^{\text{sol}}(\boldsymbol{x},t) - c_\gamma(\boldsymbol{x},t)\right), \tag{2.17}$$

$$J_m^{\gamma\alpha}(\boldsymbol{x},t) = -c_\gamma(\boldsymbol{x},t)v_n^{\gamma\alpha}(\boldsymbol{x},t), \tag{2.18}$$

$$J_d^{\gamma\alpha}(\boldsymbol{x},t) = -D_\gamma(\boldsymbol{x},t)\frac{\partial c_\gamma}{\partial \boldsymbol{n}}(\boldsymbol{x},t), \quad \text{for } \boldsymbol{x}\in\Gamma^{\gamma\alpha}(t), t\geq t_0. \tag{2.19}$$

Which give the flux boundary condition:

$$-K^{\gamma\alpha}(\boldsymbol{x},t)\left(c_{\gamma\alpha}^{\text{sol}}(\boldsymbol{x},t) - c_\gamma(\boldsymbol{x},t)\right) = -c_\gamma(\boldsymbol{x},t)v_n^{\gamma\alpha}(\boldsymbol{x},t) - D_\gamma(\boldsymbol{x},t)\frac{\partial c_\gamma}{\partial \boldsymbol{n}}(\boldsymbol{x},t), \quad \text{for } \boldsymbol{x}\in\Gamma^{\gamma\alpha}(t), t\geq t_0. \tag{2.20}$$

Combining the Stefan condition with the flux boundary condition results in:

$$D_\gamma(\boldsymbol{x},t)\frac{\partial c_\gamma}{\partial \boldsymbol{n}}(\boldsymbol{x},t) = \frac{D_\alpha(\boldsymbol{x},t)}{c_{\alpha\gamma}^{\text{sol}}(\boldsymbol{x},t)}\frac{\partial c_\alpha}{\partial \boldsymbol{n}}(\boldsymbol{x},t)c_\gamma(\boldsymbol{x},t) + \frac{K^{\gamma\alpha}(\boldsymbol{x},t)}{c_{\alpha\gamma}^{\text{sol}}(\boldsymbol{x},t)}\left(c_{\gamma\alpha}^{\text{sol}}(\boldsymbol{x},t) - c_\gamma(\boldsymbol{x},t)\right)\left(c_{\alpha\gamma}^{\text{sol}}(\boldsymbol{x},t) - c_\gamma(\boldsymbol{x},t)\right), \text{ for } \boldsymbol{x}\in\Gamma^{\gamma\alpha}(t), t\geq t_0. \tag{2.21}$$

and

$$v_n^{\gamma\alpha}(\boldsymbol{x},t) = -\frac{D_\alpha(\boldsymbol{x},t)}{c_{\alpha\gamma}^{\text{sol}}(\boldsymbol{x},t)}\frac{\partial c_\alpha}{\partial \boldsymbol{n}}(\boldsymbol{x},t) + \frac{K^{\gamma\alpha}(\boldsymbol{x},t)}{c_{\alpha\gamma}^{\text{sol}}(\boldsymbol{x},t)}\left(c_{\gamma\alpha}^{\text{sol}}(\boldsymbol{x},t) - c_\gamma(\boldsymbol{x},t)\right), \quad \text{for } \boldsymbol{x}\in\Gamma^{\gamma\alpha}(t), t\geq t_0. \tag{2.22}$$

Also note that from this we can see a similar mixed-mode character as the matrix-precipitate boundary condition:

$$c_\gamma(\boldsymbol{x},t) = c_{\gamma\alpha}^{\text{sol}}(\boldsymbol{x},t) - \frac{1}{K^{\gamma\alpha}(\boldsymbol{x},t)}\left(v_n^{\gamma\alpha}(\boldsymbol{x},t)c_{\alpha\gamma}^{\text{sol}}(\boldsymbol{x},t) - D_\alpha(\boldsymbol{x},t)\frac{\partial c_\alpha}{\partial \boldsymbol{n}}(\boldsymbol{x},t)\right), \quad \text{for } \boldsymbol{x}\in\Gamma^{\gamma\alpha}(t), t\geq t_0. \tag{2.23}$$

If $K^{\gamma\alpha}(\boldsymbol{x},t)$ is large, $c_\gamma(\boldsymbol{x},t)$ will be close to local equilibrium and the process is diffusion controlled. For small $K^{\gamma\alpha}(\boldsymbol{x},t)$ the velocity term, together with the diffusive flux in $\Omega_\alpha(t)$ over $\Gamma^{\gamma\alpha}(t)$, will have a significant influence on the carbon concentration. Meaning the system is reaction controlled, just like the reaction based boundary condition on $\Gamma^{\gamma\theta}(t)$ and $\Gamma^{\alpha\theta}(t)$.

2.1.5. Matrix-Matrix-Precipitate Boundary

In the points where the two interfaces $\Gamma^{\gamma\theta}(t)$ and $\Gamma^{\alpha\theta}(t)$ meet, we need the condition $v_n^{\gamma\theta}(\boldsymbol{x},t) = v_n^{\alpha\theta}(\boldsymbol{x},t)$ in order to have a continuous interface velocity for $\Gamma^{\theta}(t) = \Gamma^{\gamma\theta}(t)\cup\Gamma^{\alpha\theta}(t)$. These points are exactly at the triple points $\boldsymbol{x}^{\perp}$ where all three interfaces $\Gamma^{\gamma\alpha}(t)$,$\Gamma^{\gamma\theta}(t)$ and $\Gamma^{\alpha\theta}(t)$ meet. We can see from the Dirichlet condition for the ferrite carbon concentration (2.15) that the concentration is fixed by $c_\alpha(\boldsymbol{x},t) = c_{\alpha\gamma}^{\text{sol}}(\boldsymbol{x},t)$, for $\boldsymbol{x}\in\Gamma^{\gamma\alpha}(t)$. This means the interface velocity $v_n^{\alpha\theta}(\boldsymbol{x},t)$ (2.11) is also fixed by:

$$v_n^{\alpha\theta}(\boldsymbol{x},t) = \frac{K^{\alpha\theta}(\boldsymbol{x},t)}{c_\theta}\left(c_{\alpha\theta}^{\text{sol}}(\boldsymbol{x},t) - c_{\alpha\gamma}^{\text{sol}}(\boldsymbol{x},t)\right), \text{ for } \boldsymbol{x}\in\Gamma^{\gamma\theta}\cap\Gamma^{\alpha\theta}(t), t\geq t_0. \tag{2.24}$$

Which then implies that the austenite carbon concentration should also be fixed, with

$$c_\gamma(\boldsymbol{x},t) = c^{\text{sol}}_{\gamma\theta}(\boldsymbol{x},t) - \frac{K^{\alpha\theta}(\boldsymbol{x},t)}{K^{\gamma\theta}(\boldsymbol{x},t)}\left(c^{\text{sol}}_{\alpha\theta}(\boldsymbol{x},t) - c^{\text{sol}}_{\alpha\gamma}(\boldsymbol{x},t)\right), \text{ for } \boldsymbol{x} \in \Gamma^{\gamma\theta}(t) \cap \Gamma^{\alpha\theta}(t), t \geq t_0. \tag{2.25}$$

Note that if $c_\alpha(\boldsymbol{x},t)$ would have been at its solubility $c^{\text{sol}}_{\alpha\theta}(\boldsymbol{x},t)$, $c_\gamma(\boldsymbol{x},t)$ would be equal to $c_{\gamma\theta}(\boldsymbol{x},t)$. This is what you would expect as it implies $v^{\alpha\theta}_n(\boldsymbol{x},t) = v^{\gamma\theta}_n(\boldsymbol{x},t) = 0$. The problem is that this difference in solubilities $c^{\text{sol}}_{\alpha\theta}(\boldsymbol{x},t) - c^{\text{sol}}_{\alpha\gamma}(\boldsymbol{x},t)$ can be quite significant in relation to $c^{\text{sol}}_{\gamma\theta}(\boldsymbol{x},t)$ and causes the austenite carbon concentration to attain unrealistic values and has a radical effect on the behavior of the model. We will look into this effect later on in Section 6.3 and see how to cope with this.

2.2. Initial Solution

In order to have a valid initial solution, we choose c^0_γ and c^0_α satisfying the stationary model, as proposed by Den Ouden in [18]. For $\Omega_\gamma(t_0)$ we obtain the system:

$$\left\{\begin{array}{rcll} \nabla\left(D_\gamma(\boldsymbol{x})\nabla\tilde{c}_\gamma(\boldsymbol{x})\right) & = & 0 & \text{, for } \boldsymbol{x} \in \Omega_\gamma(t_0), \\ \tilde{c}_\gamma(\boldsymbol{x}) & = & c^{\partial\Omega_\gamma}_\gamma & \text{, for } \boldsymbol{x} \in \partial\Omega_\gamma(t_0), \\ D_\gamma(\boldsymbol{x})\frac{\partial\tilde{c}_\gamma}{\partial\boldsymbol{n}}(\boldsymbol{x}) & = & \frac{K^{\gamma\theta}(\boldsymbol{x})}{c_\theta}\left(c_\theta - \tilde{c}_\gamma(\boldsymbol{x})\right)\left(c^{\text{sol}}_{\gamma\theta}(\boldsymbol{x}) - \tilde{c}_\gamma(\boldsymbol{x})\right) & \text{, for } \boldsymbol{x} \in \Gamma^{\gamma\theta}(t_0), \\ D_\gamma(\boldsymbol{x})\frac{\partial\tilde{c}_\gamma}{\partial\boldsymbol{n}}(\boldsymbol{x}) & = & \frac{D_\alpha(\boldsymbol{x})}{c^{\text{sol}}_{\alpha\gamma}(\boldsymbol{x})}\frac{\partial c_\alpha}{\partial\boldsymbol{n}}(\boldsymbol{x}) + & \\ & & \frac{K^{\gamma\alpha}(\boldsymbol{x})}{c^{\text{sol}}_{\alpha\gamma}(\boldsymbol{x})}\left(c^{\text{sol}}_{\gamma\alpha}(\boldsymbol{x}) - \tilde{c}_\gamma(\boldsymbol{x})\right)\left(c^{\text{sol}}_{\alpha\gamma}(\boldsymbol{x}) - \tilde{c}_\gamma(\boldsymbol{x})\right) & \text{, for } \boldsymbol{x} \in \Gamma^{\gamma\alpha}(t_0). \end{array}\right. \tag{2.26}$$

Where $c^{\partial\Omega_\gamma}_\gamma$ is some chosen initial austenite carbon concentration on the outer boundary $\partial\Omega_\gamma(t_0) = \Omega_\gamma(t_0) \cap \partial\Omega$. For $\Omega_\alpha(t_0)$:

$$\left\{\begin{array}{rcll} \nabla\left(D_\alpha(\boldsymbol{x})\nabla\tilde{c}_\alpha(\boldsymbol{x})\right) & = & 0 & \text{, for } \boldsymbol{x} \in \Omega_\alpha(t_0), \\ \tilde{c}_\alpha(\boldsymbol{x}) & = & c^{\partial\Omega_\alpha}_\alpha(\boldsymbol{x}) & \text{, for } \boldsymbol{x} \in \partial\Omega_\alpha(t_0), \\ D_\alpha(\boldsymbol{x})\frac{\partial\tilde{c}_\alpha}{\partial\boldsymbol{n}}(\boldsymbol{x}) & = & \frac{K^{\alpha\theta}(\boldsymbol{x})}{c_\theta}\left(c_\theta - \tilde{c}_\alpha(\boldsymbol{x})\right)\left(c^{\text{sol}}_{\alpha\theta}(\boldsymbol{x}) - \tilde{c}_\alpha(\boldsymbol{x})\right) & \text{, for } \boldsymbol{x} \in \Gamma^{\alpha\theta}(t_0), \\ \tilde{c}_\alpha(\boldsymbol{x}) & = & c^{\text{sol}}_{\alpha\gamma}(\boldsymbol{x}) & \text{, for } \boldsymbol{x} \in \Gamma^{\alpha\gamma}(t_0). \end{array}\right. \tag{2.27}$$

As the ferrite is placed between the austenite and cementite it has no boundary coinciding with $\partial\Omega(t_0)$, meaning $\partial\Omega_\alpha(t_0) = \emptyset$, and thus $c_\alpha(\boldsymbol{x},t_0) = \tilde{c}_\alpha(\boldsymbol{x})$. For the austenite carbon concentration $c_\gamma(\boldsymbol{x},t)$, we set our initial condition as

$$c_\gamma(\boldsymbol{x},t_0) = \left(\tilde{c}_\gamma(\boldsymbol{x}) - c^{\partial\Omega_\gamma}_\gamma\right)H(\boldsymbol{x}) + c^{\partial\Omega_\gamma}_\gamma, \tag{2.28}$$

where the function $H(\boldsymbol{x})$ is defined as

$$H(\boldsymbol{x}) = \max_{\boldsymbol{x}\in\Omega_\gamma(t_0)}\{H^{\gamma\alpha}(\boldsymbol{x}), H^\theta(\boldsymbol{x})\},$$

and

$$\begin{array}{rcl} H^k(\boldsymbol{x}) & = & \left\{\begin{array}{llrcl} \frac{1}{2}\left(1 - \sin\left(\frac{\phi^k(\boldsymbol{x},t_0) - \sigma^k}{2\sigma^k}\pi\right)\right) & \text{if} & |\phi^k(\boldsymbol{x},t_0) + \sigma^k| & \leq & \sigma^k, \\ 0 & \text{if} & \phi^k(\boldsymbol{x},t_0) & < & -2\sigma^k, \end{array}\right. \\ \sigma^k & = & \frac{1}{2}\left|\max_{\boldsymbol{x}\in\Omega_\gamma(t_0)}\phi^k(\boldsymbol{x},t_0)\right|, k \in \{\gamma\alpha,\theta\}. \end{array} \tag{2.29}$$

$\phi^{\gamma\alpha}(\boldsymbol{x},t_0)$ and $\phi^\theta(\boldsymbol{x},t_0)$ are signed distance functions that we will use to define the different phases and interfaces. More about these functions can be found in Chapter 3. This way $c_\gamma(\boldsymbol{x},t_0)$ both holds for the no-flux boundary condition on $\partial\Omega_\gamma(t_0)$ and its interface boundary conditions.

2.3. Initial Domain

The model is initialized at the moment ferrite has just nucleated on the austenite-cementite interface. This nucleus is shaped by the surface tensions between the different phases. This effect can be related to the wetting of a liquid maintaining contact with a solid surface, like a water drop on a window. See also [1]. As

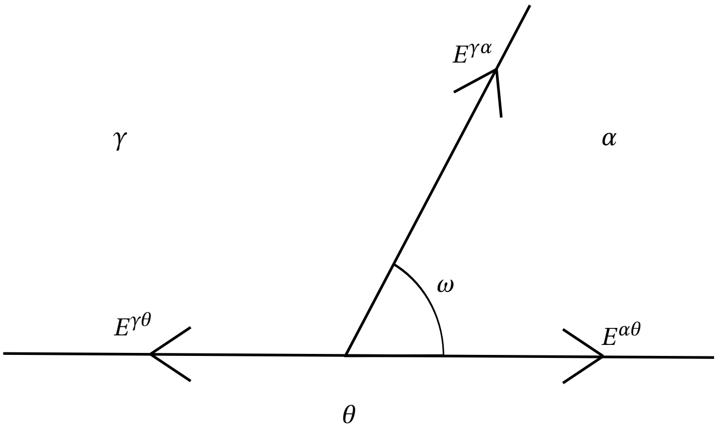


Figure 2.1: Angle ω of $\Gamma^{\gamma\alpha}$ interface with $\Gamma^{\alpha\theta}$ and $\Gamma^{\gamma\theta}$ depending on the interface energies E.

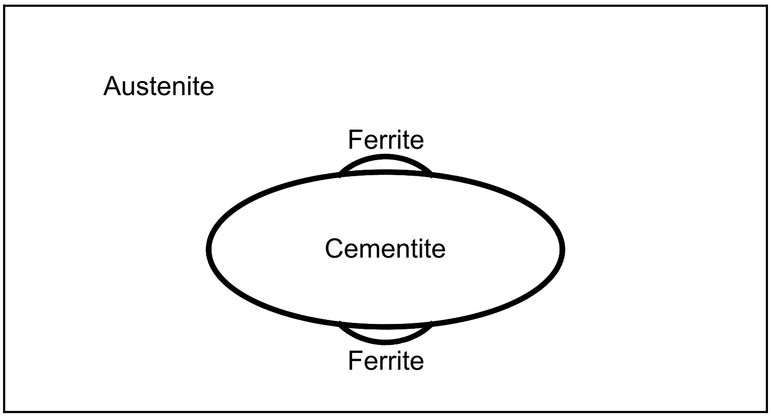


Figure 2.2: Sketch of a double ferrite nucleation on the interface of austenite and a cementite particle.

the ferrite nucleus will be quite small compared to the cementite particle, we assume the ferrite-cementite interface to be flat, see Figure 2.1. This results in Young's relation [29]:

$$E^{\alpha\theta} = E^{\gamma\alpha}\cos(\omega) + E^{\gamma\theta}, \tag{2.30}$$

where $E^k, k = \gamma\theta, \gamma\alpha$ and $\alpha\theta$ are the interface energies and ω the angle of the interface $\Gamma^{\gamma\alpha}(t_0)$ with $\Gamma^{\alpha\theta}(t_0)$ in the triple point $\boldsymbol{x}^{\perp}$ where the three interfaces meet. From the work of Savran [20] we find $E^{\gamma\alpha} = 0.39$ Jm^{-2}, $E^{\gamma\theta} = 0.94$ Jm^{-2} and $E^{\alpha\theta} = 1.05$ Jm^{-2}. This gives $\omega = 0.41\pi$.

If we choose the θ-ellipse's origin, set the triple points $\boldsymbol{x}_0^{\perp}$ and $\boldsymbol{x}_1^{\perp}$ vertically symmetric of this point on the θ-boundary, set the angle ω on which the triple points are positioned on the θ ellipse, we can calculate the ellipse parameters for both θ as α and we get an initial domain as in Figure 1.5.

2.3.1. Two Particles

We will also look at a two ferrite particle model, where also a nucleus appeared at the bottom of the cementite particle. See also Figure 2.2. The same construction for the one ferrite particle is used in the construction of the two particle model.

3

Level-Set Method

The biggest problem in approximating Stefan problems is how to keep track of the interface(s), which is/are part of the models description. There are two types of methods being used in today's research. The first type are implicit tracking methods, also called front tracking methods. Several very well known methods are the level-set method, the enthalpy method and the phase-field method, which do not use the interface itself but describe it implicitly. The other type of moving boundary methods are explicit tracking methods, called front capturing methods, which keep track of the interface itself. The advantage of the implicit methods is that capturing different interface/domains is handled more naturally then with explicit methods. Which has big problems when for example two interfaces meet. The reason to pick the level-set method for this thesis over other implicit methods is studied in the literature study for this thesis [26].
We will first give a definition for the level-set method with just one interface (two phases) and then extend it to multiple interfaces/domains, as this needs a bit more attention.

3.1. One Level-Set Function

The level-set method with only one level-set function captures the movement of an interface $\Gamma^{kl}(t)$ between the two domains $\Omega_k(t)$ and $\Omega_l(t)$, by keeping track of a signed-distance function $\phi^{kl}(\boldsymbol{x},t)$ defined as:

$$\phi^{kl}(\boldsymbol{x},t)=\begin{cases} +\min\limits_{\boldsymbol{y}\in\Gamma^{kl}(t)}\left\|\boldsymbol{y}-\boldsymbol{x}\right\|_2 & \text{, if } \boldsymbol{x}\in\overline{\Omega_k}(t)\backslash\Gamma^{kl}(t), \\ 0 & \text{, if } \boldsymbol{x}\in\Gamma^{kl}(t), \\ -\min\limits_{\boldsymbol{y}\in\Gamma^{kl}(t)}\left\|\boldsymbol{y}-\boldsymbol{x}\right\|_2 & \text{, if } \boldsymbol{x}\in\overline{\Omega_l}(t)\backslash\Gamma^{kl}(t). \end{cases} \tag{3.1}$$

Note that $\phi^{kl}(\boldsymbol{x},t)=0$ implicitly implies that $\boldsymbol{x}$ is located on the interface. It is also worthwhile to note that for a signed-distance function a special case of the Eikonal equation holds, namely

$$\left\|\nabla\phi^{kl}(\boldsymbol{x},t)\right\|_2=1,\text{ for } \boldsymbol{x}\in\overline{\Omega}_k(t), t\geq t_0, \tag{3.2}$$

at coordinates $\boldsymbol{x}$ in which $\nabla\phi^{kl}(\boldsymbol{x},t)$ is defined. The set of points where $\nabla\phi^{kl}(\boldsymbol{x},t)$ is not defined is called the skeleton of the level-set function $\phi^{kl}(\boldsymbol{x},t)$. These points are the points $\boldsymbol{x}$ for which at least two points on the interface are equally close. For example the center of a circle. However, in most numerical approximations, the level-set function will deter from the Eikonal property over time. More about this later in Section 3.3.
From the definition of $\phi^{kl}(\boldsymbol{x},t)$, the normal vector of the interface is easily derived,

$$\boldsymbol{n}^{kl}(\boldsymbol{x},t)=\frac{\nabla\phi^{kl}(\boldsymbol{x},t)}{\left\|\nabla\phi^{kl}(\boldsymbol{x},t)\right\|_2}. \tag{3.3}$$

The movement of the interface is captured by solving the convection equation,

$$\frac{\partial\phi^{kl}}{\partial t}(\boldsymbol{x},t)+v_n^{\text{ext},kl}(\boldsymbol{x},t)\left\|\nabla\phi^{kl}(\boldsymbol{x},t)\right\|_2=0,\text{ for } \boldsymbol{x}\in\Omega, t\geq t_0. \tag{3.4}$$

$v_n^{\text{ext},kl}(\boldsymbol{x},t)$ is the normal speed $v_n^{kl}(\boldsymbol{x},t)$ of the interface extended over the whole domain. Multiple possible extensions exist for this normal velocity (see [7]), but to keep this extension simple we solve the Laplace equation

$$\Delta v_n^{\text{ext}}(\boldsymbol{x},t) = 0, \text{ for } \boldsymbol{x} \in \Omega, t \geq t_0, \tag{3.5}$$

with homogeneous Neumann boundary condition on $\partial\Omega$ and Dirichlet condition

$$v_n^{\text{ext},kl}(\boldsymbol{x},t) = v_n^{kl}(\boldsymbol{x},t), \text{ for } \boldsymbol{x} \in \Gamma^{kl}(t), t \geq t_0. \tag{3.6}$$

Where $v_n^{kl}(\boldsymbol{x},t)$ is given by the chosen boundary conditions in Section 2.1 for $kl \in \{\gamma\alpha, \gamma\theta, \alpha\theta\}$.

3.2. Multiple Level-Set Functions

In order to capture the domains of multiple phases, multiple level-sets are required to describe them. A first option is to describe each domain by a separate level-set function, meaning $\phi^\gamma(\boldsymbol{x},t), \phi^\alpha(\boldsymbol{x},t)$ and $\phi^\theta(\boldsymbol{x},t)$. Of course, when two of these functions are known, the third is derived as the complement of the other two (where both level-set functions are negative). This choice ensures the level-set functions are initially defined over closed domains. Over time however, the different interfaces will move apart with different velocities. It will be difficult to make sure no voids are created or domains will overlap each other.

Another way to describe the domains is using only level-set functions describing the different interfaces $\Gamma^{\gamma\alpha}(t), \Gamma^{\gamma\theta}(t)$ and $\Gamma^{\alpha\theta}(t)$ as in the case of two phases. Then the different phases can be found by combining the level-sets, for example $\phi^{\gamma\alpha}(\boldsymbol{x},t)$ and $\phi^{\gamma\theta}(\boldsymbol{x},t)$ will form the austenite domain. The problem however is that interfaces do not have to be closed, while the level-set function has to be defined on a closed domain in order to define a signed-distance function. To solve this the interfaces have to be closed somehow, connecting the ends, but still resulting in the correct domains when the level-set functions are combined. It will be difficult updating these level-set functions without losing the correct domains of the different phases.

A third way is to combine these two techniques. If we take $\phi^\gamma(\boldsymbol{x},t)$ and $\phi^\theta(\boldsymbol{x},t)$ to describe the austenite and cementite domain respectively, we only need to find a way to define the ferrite domain. The only interface that is not used in this definition is $\Gamma^{\gamma\alpha}$(t). If we take $\phi^{\gamma\alpha}(\boldsymbol{x},t)$, the level-set function describing this interface, we can find the ferrite domain by taking the intersection of $\phi^\theta(\boldsymbol{x},t) < 0$ and $\phi^{\gamma\alpha}(\boldsymbol{x},t) > 0$, see Figure 3.1. The triple points can now be found as the points for which $\phi^\theta(\boldsymbol{x},t)$ and $\phi^{\gamma\alpha}(\boldsymbol{x},t)$ are zero. This means the triple points are defined by the intersection of two lines in stead of three (the intersection of the three interfaces $\Gamma^{\gamma\alpha}(t), \Gamma^{\gamma\theta}(t)$ and $\Gamma^{\alpha\theta}(t)$). Which also makes it a lot more easy to detect and capture the triple points. It will also avoid voids from appearing at the triple points. The only problem is how to close the domain on which $\phi^{\gamma\alpha}(\boldsymbol{x},t)$ is positive and how to make sure the overlapping part of the domain on certain parts does not become bigger than the domain of $\phi^\theta(\boldsymbol{x},t)$. We will call the overlapping part $\Omega_{\text{ghost}}(t)$ of $\phi^{\gamma\alpha}(\boldsymbol{x},t)$ with

$$\Omega_{\text{ghost}}(t) = \{\boldsymbol{x} | \phi^{\gamma\alpha}(\boldsymbol{x},t) > 0 \cap \phi^\theta(\boldsymbol{x},t) > 0\},$$

the ghost part of $\phi^{\gamma\alpha}(\boldsymbol{x},t)$ and its interface is defined as $\Gamma^{\gamma\alpha}_{\text{ghost}}(t)$. As initial domain we use an ellipse inside the ellipse of $\Omega_\theta(t)$ as described in Section 2.3.

The mathematical description of the two level-set functions $\phi^\theta(\boldsymbol{x},t)$ and $\phi^{\gamma\alpha}(\boldsymbol{x},t)$ are:

$$\phi^\theta(\boldsymbol{x},t) = \begin{cases} +\min\limits_{\boldsymbol{y}\in\Gamma^\theta(t)} \left\|\boldsymbol{y}-\boldsymbol{x}\right\|_2 & \text{, if } \boldsymbol{x} \in \overline{\Omega_\theta}(t) \backslash \Gamma^\theta(t), \\ 0 & \text{, if } \boldsymbol{x} \in \Gamma^\theta(t), \\ -\min\limits_{\boldsymbol{y}\in\Gamma^\theta(t)} \left\|\boldsymbol{y}-\boldsymbol{x}\right\|_2 & \text{, if } \boldsymbol{x} \in \{\overline{\Omega_\gamma}(t) \cup \overline{\Omega_\alpha}(t)\} \backslash \Gamma^\theta(t), \end{cases} \tag{3.7}$$

and

$$\phi^{\gamma\alpha}(\boldsymbol{x},t) = \begin{cases} +\min\limits_{\boldsymbol{y}\in\{\Gamma^{\gamma\alpha}(t)\cup\Gamma^{\gamma\alpha}_{\text{ghost}}(t)\}} \left\|\boldsymbol{y}-\boldsymbol{x}\right\|_2 & \text{, if } \boldsymbol{x} \in \overline{\Omega_\alpha}(t) \cup \overline{\Omega}_{\text{ghost}}(t) \backslash \{\Gamma^{\gamma\alpha}(t) \cup \Gamma^{\gamma\alpha}_{\text{ghost}}(t)\}, \\ 0 & \text{, if } \boldsymbol{x} \in \{\Gamma^{\gamma\alpha}(t) \cup \Gamma^{\gamma\alpha}_{\text{ghost}}(t)\}, \\ -\min\limits_{\boldsymbol{y}\in\{\Gamma^{\gamma\alpha}(t)\cup\Gamma^{\gamma\alpha}_{\text{ghost}}(t)\}} \left\|\boldsymbol{y}-\boldsymbol{x}\right\|_2 & \text{, if } \boldsymbol{x} \in \overline{\Omega_\gamma}(t) \cup \{\overline{\Omega_\theta}(t) \backslash \Omega_{\text{ghost}}(t)\} \backslash \{\Gamma^{\gamma\alpha}(t) \cup \Gamma^{\gamma\alpha}_{\text{ghost}}(t)\}. \end{cases} \tag{3.8}$$

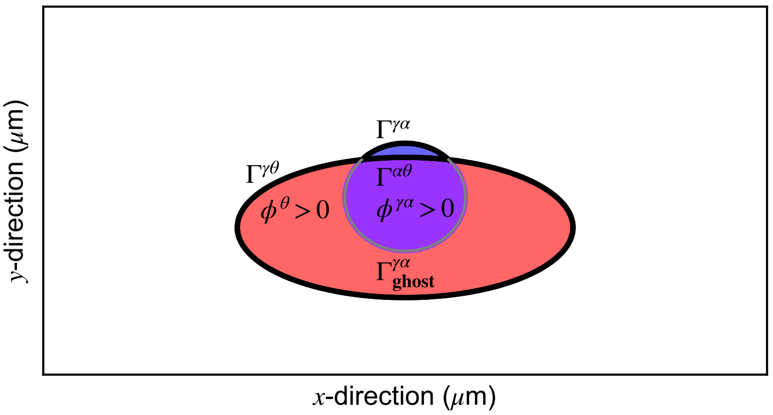


Figure 3.1: Sketch of the nucleated ferrite (α) on the interface of austenite (γ) and cementite (θ) with the overlapping (purple) level-set functions $\phi^{\gamma\alpha}$ (blue) and ϕ^{θ} (red).

Then $\Omega_\gamma(t), \Omega_\alpha(t)$ and $\Omega_\theta(t)$ can be identified as:

$$\Omega_\gamma(t) = \left\{ \boldsymbol{x} | \phi^{\gamma\alpha}(\boldsymbol{x}, t) < 0 \cap \phi^{\theta}(\boldsymbol{x}, t) < 0 \right\}, \tag{3.9}$$

$$\Omega_\alpha(t) = \left\{ \boldsymbol{x} | \phi^{\gamma\alpha}(\boldsymbol{x}, t) > 0 \cap \phi^{\theta}(\boldsymbol{x}, t) < 0 \right\}, \tag{3.10}$$

$$\Omega_\theta(t) = \left\{ \boldsymbol{x} | \phi^{\theta}(\boldsymbol{x}, t) > 0 \right\}. \tag{3.11}$$

3.2.1. Interface Velocity Ghost Interface

Because we have an interface velocity on $\Gamma^{\gamma\alpha}(t)$, we also need to have an velocity over $\Gamma^{\gamma\alpha}_{\text{ghost}}(t)$. The interface velocity will be extended by assuming the velocity is zero on the point $\boldsymbol{x}_{1/2}$ that is located furthest away from both triple points at $t = t_0$ on $\Gamma^{\gamma\alpha}_{\text{ghost}}(t)$ and using the known velocity in the triple points. Next to fixing the velocity of the ghost interface in the triple points and $\boldsymbol{x}_{1/2}$, the change of the velocity along the ghost boundary is known in the triple points from the interface velocity on the austenite-ferrite interface and we set it to be zero in $\boldsymbol{x}_{1/2}$. We can denote this as:

$$\begin{cases} v_n^{\gamma\alpha-\text{ghost}}(s_0) = v_n^{\gamma\alpha}(\boldsymbol{x}_0^\perp), \\ v_n^{\gamma\alpha-\text{ghost}}(s_1) = v_n^{\gamma\alpha}(\boldsymbol{x}_1^\perp), \\ v_n^{\gamma\alpha-\text{ghost}}(s_{1/2}) = 0, \\ \left.\dfrac{dv_n^{\gamma\alpha-\text{ghost}}}{ds}\right|_{s=s_0} = C \left.\dfrac{dv^{\gamma\alpha n}}{ds}\right|_{s=\boldsymbol{x}_0^\perp}, \\ \left.\dfrac{dv_n^{\gamma\alpha-\text{ghost}}}{ds}\right|_{s=s_1} = C \left.\dfrac{dv^{\gamma\alpha n}}{ds}\right|_{s=\boldsymbol{x}_1^\perp}, \\ \left.\dfrac{dv_n^{\gamma\alpha-\text{ghost}}}{ds}\right|_{s=s_{1/2}} = 0, \end{cases} \tag{3.12}$$

where $s(\boldsymbol{x})$ is the parametrization of the austenite-ferrite ghost interface between the triple points starting in $s(\boldsymbol{x}_0^\perp) = s_0$ and ending in $s(\boldsymbol{x}_1^\perp) = s_1$, defined as the arc length from a point $\boldsymbol{x}$ to $\boldsymbol{x}_0^\perp$. $C \in [0,1]$ is a constant to limit the ghost-interface velocity to make sure the ghost-interface will not create a new ferrite particle. Depending on the shape of the cementite domain this could always happen even with $C = 0$.

This extension gives a smooth extension of the interface velocity over the zero line of the austenite-ferrite level-set function as can be seen in Figure 3.2. Also this way the interface will not grow too much towards the cementite interface where ferrite is not located because $\boldsymbol{x}_{1/2}$ will stay fixed over time, but will follow the austenite-ferrite interface close to the triple points.

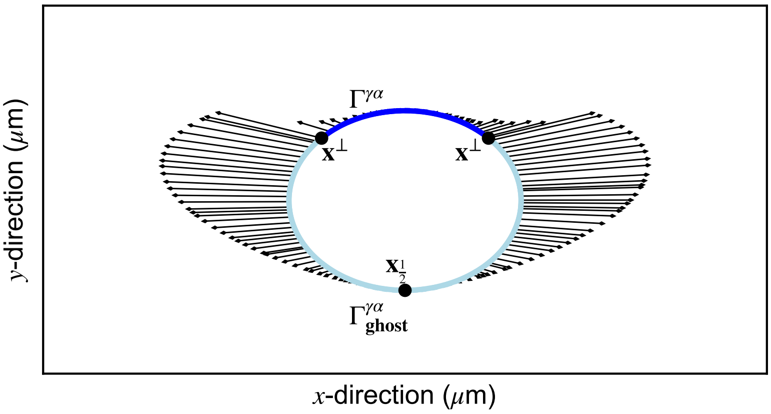


Figure 3.2: Sketch of the velocity over the whole zero contour of $\phi^{\gamma\alpha}(\boldsymbol{x}, t)$ at time $t = t_0$.

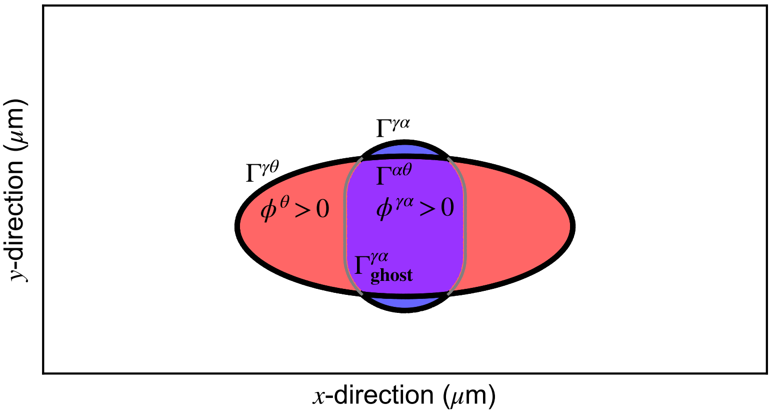


Figure 3.3: Sketch of two ferrite (α) nucleus on the interface of austenite (γ) and cementite (θ) with the overlapping (purple) level-set functions $\phi^{\gamma\alpha}$ (blue) and ϕ^{θ} (red).

3.2.2. Two Particles

In Chapter 2 we mentioned a two particle model, where on one cementite particle two symmetric ferrite nuclei nucleated. To capture this with the level-set function we combine three level-set functions. The ellipse for the one particle method, the mirrored ellipse capturing the second particle and a vertical infinite long quadrilateral connecting the horizontal vertices of the ellipses. See also Figure 3.3.
The interface velocity of the ghost interface is extended the same way as in the one particle model per pair of connected triple points.

3.3. Reinitialization of the Level-Set Function

During the process of updating the level-set functions in Equation (3.4), the signed-distance property $\left\|\nabla\phi(\boldsymbol{x}, t)\right\|_2 = 1$ of the level-set function $\phi(\boldsymbol{x}, t)$ can be lost. In the work of Sussman et al. [23] it has been shown that this property is necessary, at least in a narrow band around the interface $\Gamma(t)$, in order to get proper results (mass conservation and correct shape). To solve this problem, Sussman et al. proposed reinitializing the obtained level-set function $\tilde{\phi}(\boldsymbol{x}, t^{n+1})$ after updating by solving the hyperbolic system

$$\begin{cases} \frac{\partial\phi}{\partial\tau}(\boldsymbol{x},\tau) + \boldsymbol{w}(\boldsymbol{x},\tau,t^{n+1})\cdot\nabla\phi(\boldsymbol{x},\tau) &= \operatorname{sign}\left(\tilde{\phi}\left(\boldsymbol{x},t^{n+1}\right)\right) \quad \text{, for } \boldsymbol{x}\in\overline{\Omega}(t^{n+1}), \\ \phi(\boldsymbol{x},\tau) &= 0 \quad \text{, for } \boldsymbol{x}\in\Gamma(t^{n+1}), \\ \phi(\boldsymbol{x},\tau_0) &= \tilde{\phi}\left(\boldsymbol{x},t^{n+1}\right) \quad \text{, for } \boldsymbol{x}\in\overline{\Omega}(t^{n+1}), \end{cases} \tag{3.13}$$

where $\boldsymbol{w}(\boldsymbol{x},\tau,t^{n+1}) = \operatorname{sign}\left(\tilde{\phi}\left(\boldsymbol{x},t^{n+1}\right)\right)^{\nabla\phi}/_{\|\nabla\phi\|_2}(\boldsymbol{x},\tau)$. This system will converge to a steady state satisfying $\left\|\nabla\phi(\boldsymbol{x},\tau)\right\|_2 = 1$ in pseudo-time τ. The signed-distance property will be forced on $\phi(\boldsymbol{x},\tau)$ starting on the interface $\Gamma(t)$ and moving with a speed of one micrometer per pseudo second $[\tau]$ in both directions perpendicular to the interface $(\pm\boldsymbol{n})$. This method is called hyperbolic reinitialization, after the hyperbolic system that has to be solved.

Some other hyperbolic reinitialization methods are the fast sweeping methods (see for example [25]) and fast marching (see [21]) methods. An alternative to hyperbolic reinitialization is parabolic (see [12]) or elliptic (see [12]) reinitialization, which, as their names imply, reinitialize the level-set function by solving a parabolic/elliptic equation in stead of the hyperbolic Eikonal equation.

4 Discretization

Galerkins finite elements method is used to approximate the solution of the Stefan problem. In this section we will first derive the weak form for the two diffusive phases $\Omega_\gamma(t)$ and $\Omega_\alpha(t)$ with only the Neumann condition on $\partial\Omega(t)$. Then the interface boundary conditions will be applied to get the boundary element equations. The implicit Euler time discretization is used to approximate the time derivative and a resulting dimensional problem is treated using a simple extension technique. Additionally, we use an algebraic flux correction (AFC) method [10] to limit the numerical anti-diffusive fluxes created by the numerical method that result in unrealistic oscillations or violations of the maximum principle, which holds for the Equations (2.1). Finally to update and reinitialize the level-set functions $\phi^{\gamma\alpha}(\boldsymbol{x},t)$ and $\phi^{\theta}(\boldsymbol{x},t)$ we also use the Galerkins method and a total-variation-diminishing (TVD) Runge-Kutta-third-order (RK3) time integration method on the convection and reinitialization equation for both level-set functions.

4.1. Model

4.1.1. Weak Form

The governing equations of the model are

$$\left\{\begin{array}{lll} \frac{\partial c_k}{\partial t}(\boldsymbol{x},t) & = \nabla\cdot(D_k(\boldsymbol{x},t)\nabla c_k(\boldsymbol{x},t)) & \text{, for } \boldsymbol{x}\in\Omega_k(t), k=\gamma,\alpha, t\geq t_0, \\ c_k(\boldsymbol{x},t) & = c_k^0 & \text{, for } \boldsymbol{x}\in\Omega_k(t), k=\gamma,\alpha, t=t_0, \\ c_\theta(\boldsymbol{x},t) & = c_\theta & \text{, for } \boldsymbol{x}\in\Omega_\theta(t), t\geq t_0, \\ \frac{\partial c}{\partial \boldsymbol{n}}(\boldsymbol{x},t) & = 0 & \text{, for } \boldsymbol{x}\in\partial\Omega, t\geq t_0, \end{array}\right. \tag{4.1}$$

where boundary conditions for $\Gamma^{\gamma\alpha}(t), \Gamma^{\gamma\theta}(t), \Gamma^{\alpha\gamma}(t)$ and $\Gamma^{\alpha\theta}(t)$ still have to be set to make the model sufficient.

For sake of clarity we do not write down the dependence on $\boldsymbol{x}$ and t in the following equations.
To get the weak form we multiply the right hand side of the first equation of (4.1) by a test function ψ, which is zero on a boundary if c_k has a Dirichlet boundary condition on this boundary, and integrate over domain Ω_k.

$$\int\limits_{\Omega_k} \psi \frac{\partial c_k}{\partial t} d\Omega = \int\limits_{\Omega_k} \psi \nabla\cdot(D_k \nabla c_k) d\Omega. \tag{4.2}$$

Using Green's first identity we get:

$$\int\limits_{\Omega_k} \psi \frac{\partial c_k}{\partial t} d\Omega = \int\limits_{\partial\Omega_k} \psi D_k \frac{\partial c_k}{\partial \boldsymbol{n}} d\Gamma - \int\limits_{\Omega_k} \nabla\psi\cdot D_k \nabla c_k d\Omega. \tag{4.3}$$

For $k=\gamma$, splitting the boundary integrals to each different interface and using the Neumann condition at the outer boundary gives:

$$\int_{\Omega_\gamma} \psi \frac{\partial c_\gamma}{\partial t} d\Omega = \int_{\Gamma^{\gamma\theta}} \psi D_\gamma \frac{\partial c_\gamma}{\partial \boldsymbol{n}} d\Gamma + \int_{\Gamma^{\gamma\alpha}} \psi D_\gamma \frac{\partial c_\gamma}{\partial \boldsymbol{n}} d\Gamma - \int_{\Omega_\gamma} \nabla\psi \cdot D_\gamma \nabla c_\gamma d\Omega. \tag{4.4}$$

For $k = \alpha$ we get a similar equation:

$$\int_{\Omega_\alpha} \psi \frac{\partial c_\alpha}{\partial t} d\Omega = \int_{\Gamma^{\alpha\theta}} \psi D_\alpha \frac{\partial c_\alpha}{\partial \boldsymbol{n}} d\Gamma - \int_{\Omega_\alpha} \nabla\psi \cdot D_\alpha \nabla c_\alpha d\Omega. \tag{4.5}$$

Equations (4.4) and (4.5) define the weak form of (4.1).

The different boundary conditions will now be given to make the weak form sufficient.

Matrix-Precipitate Boundary

For the matrix-precipitate boundaries, austenite-cementite and ferrite-cementite, an interface reaction boundary condition is taken. This will fix the interface velocity, leaving the austenite and ferrite carbon concentration as variables on the austenite-cementite and ferrite-cementite interfaces respectively.
The $k\theta$-interface term of the weak forms (4.4) and (4.5) become

$$\int_{\Gamma^{k\theta}} \psi D_k \frac{\partial c_k}{\partial \boldsymbol{n}} d\Gamma = \int_{\Gamma^{k\theta}} \psi \frac{K^{k\theta}}{c_\theta} (c_\theta - c_k) \left(c^{\text{sol}}_{k\theta} - c_k \right) d\Gamma, k = \gamma, \alpha. \tag{4.6}$$

Matrix-Matrix Boundary

Also for the austenite-ferrite boundary an interface reaction is assumed on the interface $\Gamma^{\gamma\alpha}$, with para-equilibrium for the ferrite carbon concentration. The boundary integral over $\Gamma^{\gamma\alpha}$ in equation (4.4) becomes

$$\int_{\Gamma^{\gamma\alpha}} \psi D_\gamma \frac{\partial c_\gamma}{\partial \boldsymbol{n}} d\Gamma = \int_{\Gamma^{\gamma\alpha}} \psi \left(\frac{D_\alpha}{c^{\text{sol}}_{\alpha\gamma}} \frac{\partial c_\alpha}{\partial \boldsymbol{n}} c_\gamma + \frac{K^{\gamma\alpha}}{c^{\text{sol}}_{\alpha\gamma}} \left(c^{\text{sol}}_{\gamma\alpha} - c_\gamma \right) \left(c^{\text{sol}}_{\alpha\gamma} - c_\gamma \right) \right) d\Gamma. \tag{4.7}$$

4.1.2. Space Discretization

To calculate the concentration $c_k(\boldsymbol{x}, t), k = \gamma, \alpha$, we need to find $c_k(\boldsymbol{x}, t) \in \Sigma$, where

$$\Sigma = \left\{ c_k(\boldsymbol{x}, t) \text{ sufficiently smooth }, \boldsymbol{x} \in \Omega_k(t) \,\middle|\, c_k(\boldsymbol{x}, t) = c^{\text{sol}}(\boldsymbol{x}, t), \text{for } \boldsymbol{x} \in \Gamma^D_k(t), t \geq t_0 \right\},$$

is the solution space and $\Gamma^D_k(t)$ are the interfaces with a Dirichlet boundary condition, in our case $\Gamma^D_k(t) = \emptyset$ for $k = \gamma$ and $\Gamma^D_k(t) = \Gamma^{\gamma\alpha}(t)$ for $k = \alpha$, such that the weak form (4.4) and (4.5) holds for all

$$\psi(\boldsymbol{x}) \in \Sigma_0 = \left\{ \psi(\boldsymbol{x}), \boldsymbol{x} \in \Omega_k(t), t \geq t_0 \,\middle|\, \psi(\boldsymbol{x}) = 0, \text{for } \boldsymbol{x} \in \Gamma^D_k(t) \right\}.$$

We choose piecewise linear basis functions $\psi_l(\boldsymbol{x}) \in \Sigma_0, l = 1, \ldots, N_k(t), k = \gamma, \alpha$ defined on a mesh $T_k(t)$, which is a union of disjunct elements $\Omega^m_k(t), m = 1 \ldots N^{\text{el}}_k(t)$. Next we choose an arbitrary, but known function, $c_D(\boldsymbol{x}, t)$ that satisfies the Dirichlet boundary conditions, like:

$$c_D(\boldsymbol{x}, t) = c^{\text{sol}}(\boldsymbol{x}, t), \text{for } \boldsymbol{x} \in \Gamma^D_k(t), t \geq t_0. \tag{4.8}$$

We approximate $c_k(\boldsymbol{x}, t)$ by a finite dimensional subset of Σ as:

$$c_k(\boldsymbol{x}, t) \approx c_k^{N_k(t)}(\boldsymbol{x}, t) = \sum_{l=1}^{N_k(t)} \psi_l(\boldsymbol{x}) \left(c_k(t) \right)_l + c_D(\boldsymbol{x}, t). \tag{4.9}$$

Substituting (4.9) in the weak form will result in a system defined as:

$$M_k(t) \frac{d\boldsymbol{c}_k(t)}{dt} = S_k\left(t, \boldsymbol{c}_k(t)\right) \boldsymbol{c}_k(t) + \boldsymbol{f}_k\left(t, \boldsymbol{c}_k(t)\right), k = \gamma, \alpha, t \geq t_0, \tag{4.10}$$

where the matrix $M_k(t)$ is called the mass matrix, the matrix $S_k(t, \boldsymbol{c}_k(t))$ the stiffness matrix and $\boldsymbol{f}_k(t, \boldsymbol{c}_k(t))$ the right hand side vector.

4.1.3. Time Integration

First order implicit Euler time integration on equation (4.10) gives the system for $\boldsymbol{c}_k(t)$ as

$$\left(M_k\left(t^{n+1}\right) - \Delta t S_k\left(t^{n+1}, \boldsymbol{c}_k^{n+1}\right)\right)\boldsymbol{c}_k^{n+1} = M_k\left(t^{n+1}\right)\boldsymbol{c}_k^{n} + \Delta t \boldsymbol{f}_k\left(t^{n+1}, \boldsymbol{c}_k^{n+1}\right). \tag{4.11}$$

Δt is the chosen time step from t^n to t^{n+1}. This approximation results in a first order accurate time integration. The implicit Euler method, or also called the backward Euler method, is unconditionally stable, making it a perfect candidate. Furthermore the implicit Euler method preserves positivity if Equation (4.11) is solved exactly [11].

Satisfying Dimensions

In Equation (4.11) we have a multiplication between the mass matrix M_k at time t^{n+1} with the concentration values $\boldsymbol{c}_k$ on time t^n. As the amount of nodes in the discretized domain $\Omega_k(t)$ can change over time, this multiplication can have a mismatch in dimensions. To fix this mismatch we will extend the solution $\boldsymbol{c}_k^n$ from the mesh T_k^n to T_k^{n+1}. In the literature study of this thesis [26] three different techniques were investigated for the one-dimensional problem. These techniques extend the solution vector by assuming the new arisen point, or even all points in the new mesh, has/have moved from somewhere in the old mesh to the new mesh. By this movement of points, convection is introduced to the system. To take in account this convection we must replace the partial derivative in our system of Equations (4.1) with a material derivative. The material derivative is defined as

$$\frac{Dc_k}{Dt}(\boldsymbol{x}, t) = \frac{\partial c_k}{\partial t}(\boldsymbol{x}, t) + \frac{d\boldsymbol{x}}{dt}(t)\cdot\nabla c_k(\boldsymbol{x}, t), \tag{4.12}$$

which will turn Equation (4.1) into

$$\begin{cases} \frac{Dc_k}{Dt}(\boldsymbol{x}, t) & = \nabla\cdot\left(D_k(\boldsymbol{x}, t)\nabla c_k(\boldsymbol{x}, t)\right) + \frac{d\boldsymbol{x}}{dt}(t)\cdot\nabla c_k(\boldsymbol{x}, t) & \text{, for } \boldsymbol{x}\in\Omega_k(t), k=\gamma,\alpha, t\geq t_0 \\ c_k(\boldsymbol{x}, t) & = c_k^0 & \text{, for } \boldsymbol{x}\in\Omega_k(t), k=\gamma,\alpha, t=t_0, \\ c_\theta(\boldsymbol{x}, t) & = c_\theta & \text{, for } \boldsymbol{x}\in\Omega_\theta(t), t\geq t_0, \\ \frac{\partial c_k}{\partial \boldsymbol{n}}(\boldsymbol{x}, t) & = 0 & \text{, for } \boldsymbol{x}\in\partial\Omega_k(t), t\geq t_0. \end{cases} \tag{4.13}$$

The velocity of a point $\boldsymbol{x}$ at time t^{n+1} will be approximated by the mesh velocity $\boldsymbol{v}_k^{\text{mesh}}(t^{n+1})$, created by moving the grid points as described above. This new term will be included in the stiffness matrix $S_k\left(t, \boldsymbol{c}_k(t)\right)$. The system we will need to solve is:

$$\left(M_k\left(t^{n+1}\right) - \Delta t S_k\left(t^{n+1}, \boldsymbol{c}_k^{n+1}\right)\right)\boldsymbol{c}_k^{n+1} = M_k\left(t^{n+1}\right)\boldsymbol{c}_{k,\text{ex}}^{n} + \Delta t \boldsymbol{f}_k\left(t^{n+1}, \boldsymbol{c}_k^{n+1}\right). \tag{4.14}$$

The extension technique which we use will set the concentration on new internal grid points by interpolating in the element it is located in and new interface grid points will be projected on the old interface. More details can be found in Section 5.2.

4.1.4. Fixed-Point Iteration

Because the reaction boundary conditions are non-linear, the stiffness matrix $S_k\left(t, \boldsymbol{c}_k(t)\right)$ and the right hand side vector $\boldsymbol{f}_k\left(t, \boldsymbol{c}_k(t)\right)$ depend on the solution vector $\boldsymbol{c}_k(t)$. The terms

$$\int\limits_{\Gamma^{k\theta}} \psi \frac{K^{k\theta}}{c_\theta}\left(c_\theta - c_k\right) c_{k\theta}^{\text{sol}} d\Gamma, \text{ for } k=\gamma,\alpha,$$

and

$$\int\limits_{\Gamma^{\gamma\alpha}} \psi c_{\gamma\alpha}^{\text{sol}} \frac{K^{\gamma\alpha}}{c_{\alpha\gamma}^{\text{sol}}}\left(c_{\alpha\gamma}^{\text{sol}} - c_\gamma\right) d\Gamma, \text{ for } k=\gamma,$$

are included in the right hand side vector $\boldsymbol{f}_k\left(t, \boldsymbol{c}_k(t)\right)$. The remaining terms

$$-\int\limits_{\Gamma^{k\theta}} \psi \frac{K^{k\theta}}{c_\theta}\left(c_\theta - c_k\right) c_k d\Gamma, \text{ for } k=\gamma,\alpha,$$

and

$$\int_{\Gamma^{\gamma\alpha}} \psi \left(\frac{D_\alpha}{c^{\text{sol}}_{\alpha\gamma}} \frac{\partial c_\alpha}{\partial \boldsymbol{n}} - \frac{K^{\gamma\alpha}}{c^{\text{sol}}_{\alpha\gamma}} \left(c^{\text{sol}}_{\alpha\gamma} - c_\gamma \right) \right) d\Gamma, \text{ for} k = \gamma,$$

are placed in the stiffness matrix $S_k\left(t, \boldsymbol{c}_k\left(t\right)\right)$.

In order to solve the resulting non-linear system we have a natural fixed-point problem (4.14), just like in the two phase model of Den Ouden's research [18], which we solve by using the same fixed-point iteration technique

$$\begin{cases} \boldsymbol{c}^{n+1}_{k,0} & = & \boldsymbol{c}^{n}_{\text{ex}}, \\ A\left(c^{n+1}_{k,r}\right) \boldsymbol{c}^{n+1}_{k,r+1} & = & g\left(\boldsymbol{c}^{\boldsymbol{n+1}}_{\boldsymbol{k,r}}\right), \end{cases} \tag{4.15}$$

where

$$\begin{array}{rcl} A\left(\boldsymbol{q}\right) & = & \left(M_k\left(t^{n+1}\right) - \Delta t S_k\left(t^{n+1}, \boldsymbol{q}\right)\right), \\ \boldsymbol{g}\left(\boldsymbol{q}\right) & = & M_k\left(t^{n+1}\right) \boldsymbol{c}^{n}_{k,\text{ex}} + \Delta t \boldsymbol{f}_k\left(t^{n+1}, \boldsymbol{q}\right). \end{array} \tag{4.16}$$

We iterate until the error ϵ_r defined as

$$\epsilon_r = \frac{\left\| A\left(\boldsymbol{c}^{n+1}_{k,r+1}\right) \boldsymbol{c}^{n+1}_{k,r+1} - \boldsymbol{g}\left(c^{n+1}_{r,r+1}\right) \right\|_\infty}{\left\| \boldsymbol{g}\left(c^{n+1}_{k,r+1}\right) \right\|_\infty}, \tag{4.17}$$

becomes lower than some chosen value FPT.

4.2. Level-Set Update and Reinitialization

The level-set functions are updated by solving Equation (3.4) and reinitialized by Equation (3.13). Both equations are convection equations. Where in the first case the convection parameter $\boldsymbol{v}(\boldsymbol{x}, t)$ is the extended interface velocity and no source term is present, in the latter case with the normal velocity of size one and a source term of plus or minus one (depending on the sign of the level-set function) to enforce the Eikonal equation.

We will approximate both equations by deriving the weak form and applying Galerkins Finite Element Method for the space discretization and a TVD RK3 scheme for the time integration. As Kuzmin remarks in [10] we can not use FCT for updating and reinitializing the level-set functions, because FCT applied in combination with forward Euler time integration will result in clipping and terracing. In stead Kuzmin recommends using a TVD RK scheme to stabilize the time integration for high convective transport problems.

Most reinitialization methods give a good approximation for the signed-distance property, but by doing this they will change the location of the interface because the piecewise linear approximation of level set function describing the interface can not exactly suffice the signed-distance property. In this research we want to keep the interface location fixed while preserving the Eikonal property as best as possible. This extra Dirichlet boundary condition will however result in some issues at later times in the model when reinitializing the level-set function as we will see in the results. That is why we will also use a least squares solve to give the system more freedom in satisfying the Eikonal property (but allowing the interface to change). As for now there is no known working method that can reinitialize the level-set function while retaining the interface location without fixing any function values in the elements near the interface. For example the fast marching method [21] fixes the interface by restraining the value of the level-set function in the elements the interface crosses. However these are the elements where you really want to do the reinitialization in order to fix the Eikonal property.

4.2.1. Weak Form

For a general convection equation of the form

$$\frac{\partial \phi}{\partial t}(\boldsymbol{x}, t) + \boldsymbol{v}(\boldsymbol{x}, t) \cdot \nabla \phi(\boldsymbol{x}, t) = S(\boldsymbol{x}, t), \tag{4.18}$$

the weak form is given as

$$\int_\Omega \psi(\boldsymbol{x}) \frac{\partial \phi}{\partial t}(\boldsymbol{x}, t) + \psi(\boldsymbol{x}) \boldsymbol{v}(\boldsymbol{x}, t) \cdot \nabla \phi(\boldsymbol{x}, t) d\Omega = \int_\Omega \psi(\boldsymbol{x}) S(\boldsymbol{x}, t) d\Omega. \tag{4.19}$$

Where $\psi(\boldsymbol{x})$ is a test function.
For the update Equation (3.4) $\boldsymbol{v}(\boldsymbol{x},t)$ is the extended interface speed $v_n^{\mathrm{ex}}(\boldsymbol{x},t)$ and $S(\boldsymbol{x},t)$ is zero. For the reinitialization Equation (3.13) $\boldsymbol{v}(\boldsymbol{x},t)$ is $\boldsymbol{w}(\boldsymbol{x},\tau) = \operatorname{sign}\left(\tilde{\phi}\right)(\boldsymbol{x},\tau)^{\nabla\phi}/_{\|\nabla\phi\|_2}$ and $S(\boldsymbol{x},t)$ is $\operatorname{sign}\left(\tilde{\phi}(\boldsymbol{x},t)\right)$. For the reinitializing equation a homogeneous Dirichlet boundary condition is present on the interface boundaries of the phases. For the reinitialization of $\phi^{\gamma\alpha}(\boldsymbol{x},t)$

$$\phi^{\gamma\alpha}(\boldsymbol{x},t) = 0, \text{ for } \boldsymbol{x} \in \Gamma^{\gamma\alpha}(t) \cup \Gamma^{\gamma\alpha}_{\text{ghost}}(t), t \geq t_0, \tag{4.20}$$

and for $\phi^{\theta}(\boldsymbol{x},t)$

$$\phi^{\theta}(\boldsymbol{x},t) = 0, \text{ for } \boldsymbol{x} \in \Gamma^{\gamma\theta}(t) \cup \Gamma^{\alpha\theta}(t), t \geq t_0. \tag{4.21}$$

4.2.2. Space Discretization

Again linear basis functions $\psi(\boldsymbol{x})$ will be used for the space discretization, where $\psi(\boldsymbol{x})$ is zero on the interface for reinitialization. We get the system

$$M(t)\frac{\partial \boldsymbol{\phi}(t)}{\partial t} = A(t)\boldsymbol{\phi}(t) + \boldsymbol{g}(t), \text{ for } t \geq t_0, \tag{4.22}$$

where $A(t)$ is the convection matrix with velocities $v_{\boldsymbol{n}}^{\mathrm{ex}}(\boldsymbol{x},t)$ and $\boldsymbol{w}(\boldsymbol{x},\tau)$ for the update and reinitialization equations respectively. $\boldsymbol{g}(\boldsymbol{x},t)$ is zero for the update equation and the right hand side vector with the sign term for the reinitialization equation. Furthermore to impose the Dirichlet condition $\phi(\boldsymbol{x},t) = 0$ on the interface $\Gamma(t)$ we use Lagrange Multipliers with the implicit Dirichlet matrix F containing the linear combinations of $\phi(\boldsymbol{x},t)$ values that imply the location of the interface points. More details will be shown in Section 5.3.1.

4.2.3. Total Variation Diminishing Runge-Kutta-Third-Order Time Integration

In some problems linearly stable Runge-Kutta time integration methods can still generate oscillations even though TVD spatial discretization is used [5]. We will use a TVD Runge-Kutta-third-order to ensure no oscillations occur. The optimal third order TVD Runge-Kutta-third-order scheme (see also [5]) is given by

$$\left\{\begin{array}{rcl} M_L\boldsymbol{u}^{(1)} & = & M_L\boldsymbol{u}^n + \Delta t L(\boldsymbol{u}^n), \\ M_L\boldsymbol{u}^{(2)} & = & \frac{3}{4}M_L\boldsymbol{u}^n + \frac{2}{3}M_L\boldsymbol{u}(1) + \frac{1}{4}\Delta t L\left(\boldsymbol{u}^{(1)}\right), \\ M_L\boldsymbol{u}^{n+1} & = & \frac{1}{3}M_L\boldsymbol{u}^n + \frac{2}{3}M_L\boldsymbol{u}^{(2)} + \frac{2}{3}\Delta t L\left(\boldsymbol{u}^{(2)}\right). \end{array}\right. \tag{4.23}$$

Where in our case L is the discrete convection operator combined with the source vector of Equation (4.19), M_L the lumped mass matrix and $\boldsymbol{u} = \boldsymbol{\phi}^{\gamma\alpha}$ or $\boldsymbol{\phi}^{\theta}$. The CFL coefficient for this method is one, meaning the time step Δt must limited by the spatial length Δx as

$$\Delta t \leq \Delta x. \tag{4.24}$$

5

Technical Details

5.1. Mesh Generation

In order to approximate the diffusion equations (2.1) we see that we need to be able to distinguish the three different phases austenite, ferrite and cementite. Furthermore the interface velocities are calculated from the concentration values on the interfaces. To keep track of the domains and interfaces we define several meshes. A mesh T is defined on Ω with triangular elements and is used as the background mesh, which remains the same over time. Next to T, we define T_E^n as the enriched mesh, which is T extended with nodes on the interfaces $\Gamma^{\gamma\alpha}(t_n), \Gamma^{\gamma\theta}(t_n)$ and $\Gamma^{\alpha\theta}(t_n)$ at discrete time t_n. Then T_γ^n and T_α^n are the meshes containing the elements of T_E^n which contain only nodes of the austenite domain $\Omega_\gamma(t_n)$ and ferrite domain $\Omega_\alpha(t_n)$ respectively, including their boundaries $\partial\Omega_\gamma(t_n) = \Gamma^{\gamma\alpha}(t_n) \cup \Gamma^{\gamma\theta}(t_n)$ and $\partial\Omega_\alpha(t_n) = \Gamma^{\alpha\gamma}(t_n) \cup \Gamma^{\alpha\theta}(t_n)$.

At an edge e where a level-set function $\phi(\boldsymbol{x}, t)$ has a positive and a negative value, we know an interface is present which crosses this edge in the point $\boldsymbol{x}^e$ due to continuity of $\phi(\boldsymbol{x}, t)$ on this edge. For this point it holds that $\phi(\boldsymbol{x}^e, t_n) = 0$. To capture the interface we need to add these points to the mesh. To add these points we need to change the triangles that contain these edges. So for each triangle $\Delta_i, i = 1, \dots, N_\Delta$ we check the three edges whether

$$\phi^k(\boldsymbol{e}_1, t_n)\phi^k(\boldsymbol{e}_2, t_n) < 0, \text{ for } e \in \Delta_i, k \in \{\gamma\alpha, \theta\}, \tag{5.1}$$

where $\boldsymbol{e}_1, \boldsymbol{e}_2$ are the first and second vertex of the edge respectively. In order to avoid ill shaped triangles (those containing small and/or large angles), we do not add the point $\boldsymbol{x}^e$ to T_E^n, for which $\phi^k(\boldsymbol{x}^e, t_n) = 0$, but shift a mesh point of T_E^n to $\boldsymbol{x}^e$, if $\boldsymbol{x}^e$ is close to a mesh point. To check if $\boldsymbol{x}^e$ is close to a mesh point we use the linear approximation of $\phi^k(\boldsymbol{x}, t)$, by defining τ^e for each edge as

$$\tau^e = \frac{-\phi^k(\boldsymbol{e}_1, t_n)}{\phi^k(\boldsymbol{e}_2, t_n) - \phi^k(\boldsymbol{e}_1, t_n)}, k \in \{\gamma\alpha, \theta\}. \tag{5.2}$$

With τ^e we can approximate $\boldsymbol{x}^e$ along edge e by

$$\boldsymbol{x}^e = \left(1 - \tau^e\right)\boldsymbol{e}_1 + \tau^e \boldsymbol{e}_2. \tag{5.3}$$

If τ^e is either close to zero or one we see that $\boldsymbol{x}^e$ is close to $\boldsymbol{e}_1$ or $\boldsymbol{e}_2$ respectively. So if τ^e is smaller than some δ or bigger than $1 - \delta$, with $\delta \in [0, 1/2]$, we shift the mesh point $\boldsymbol{e}_1, \boldsymbol{e}_2$ to $\boldsymbol{x}^e$ respectively. So

$$\boldsymbol{e}_1 := \boldsymbol{x}^e \text{ and } \phi^k(\boldsymbol{x}^e, t_n) = 0 \text{ if } \tau^e < \delta, \tag{5.4}$$

and

$$\boldsymbol{e}_2 := \boldsymbol{x}^e \text{ and } \phi^k(\boldsymbol{x}^e, t_n) = 0 \text{ if } \tau^e > 1 - \delta. \tag{5.5}$$

When a point $\boldsymbol{x}^e$ is not shifted, it is added to T_E^n. This process will be called cutting, as this new point cuts the edge e in two. When a point is added to the mesh, a re-triangulation is needed. The triangle will have to be split into two new triangles with the new point as an extra vertex.

If a point is able to shift to several locations, we choose the location that is closest to it. Also, when three points of a triangle are able to shift to the same interface, the one that is located furthest will not be shifted. This is required to avoid a change in rotation of the triangle, which results in overlapping elements and also to avoid ill shaped triangles. See Section 5.1.1 for some more insight on this occurrence. Obviously nodes located on the outer boundary $\partial\Omega$ are only allowed to shift in the direction of other outer boundary nodes which share the same outward normal, so the shape of Ω is never lost.

Prior to cutting interface points we add the triple points to the enriched mesh, which are located before any change has been made to the enriched mesh. How to add these triple points and how to re-triangulate will be shown below.

On an edge where both level-set functions change sign, meaning multiple interfaces cross this edge, a mesh point is allowed to shift to one of the interfaces, but at the same time the other intersection point has to be added to the enriched mesh. For easier implementation we choose to always shift to the austenite interfaces if a mesh point can shift to multiple interfaces. We cannot avoid small or large angles in the triangles in these regions, so we have to check our results to be accurate enough.

The value of δ will be set by looking at the quality and accuracy of the resulting mesh, which depend on the different mesh adjustments mentioned above and in the handling of multiple interfaces. In all cases where different re-triangulations can be done, we will choose the triangles that have smallest variance in their angles. The closer the angles are to $\pi/3 = 60°$, the better the quality of the mesh.

5.1.1. Three Shifts in One Triangle

When three points of a triangle are able to shift to the same interface, the interface will have a sharp kink. If we allow all three points to shift, it takes some effort to find out to which domain this triangle belongs and which edge is not part of the interface, although for the solving process one only needs to know which element this triangle is part of. For now we will fix this issue by allowing only two points to shift to the same interface. The third point, that is not allowed to shift, is the one that is located farthest away from the interface. This of course results in some small and bad shaped triangles, as we cut multiple interface points which would normally want to shift.

One could allow for three points of a triangle to shift as long as it does not result in a change of rotation and the triangle with three shifted points is not allowed to have low accuracy. Where we define low accuracy with the equiangular skewness value (see Section 5.1.4). This will give better qualitative triangles, but increases the effort in administration.

A third way to handle these triangles is by assuming they have transitioned to the other phase and forget about the part where the level-set function is zero. By this you lose some information about the interface location, but it gives less instability in the interface growth and a more qualitative mesh. We will not consider these two latter options, but they should be considered if the quality of the mesh becomes too low.

5.1.2. Handling Multiple Interfaces

When two interfaces cross the same triangle there are two options. One, the interfaces intersect each other (in this triangle) or they do not. In the first case we have found a triple point.

To see if two line segments intersect we parametrize the lines in the following way:

$$\begin{array}{rclcl} \boldsymbol{x}^{\gamma\alpha}(t^{\gamma\alpha}) & = & (1-t^{\gamma\alpha})\boldsymbol{x}_1^{\gamma\alpha} & + & t^{\gamma\alpha}\boldsymbol{x}_2^{\gamma\alpha}, \\ \boldsymbol{x}^{\theta}(t^{\theta}) & = & (1-t^{\theta})\boldsymbol{x}_1^{\theta} & + & t^{\theta}\boldsymbol{x}_2^{\theta}, \end{array} \tag{5.6}$$

where $\boldsymbol{x}_1^{\gamma\alpha}, \boldsymbol{x}_1^{\theta}$ and $\boldsymbol{x}_2^{\gamma\alpha}, \boldsymbol{x}_2^{\theta}$ are the two crossing points of the interfaces $\Gamma^{\gamma\alpha}(t)$ and $\Gamma^{\gamma\theta}(t) \cup \Gamma^{\alpha\theta}(t)$ with the triangle respectively. We solve the system that results from $\boldsymbol{x}^{\gamma\alpha}(t^{\gamma\alpha}) = \boldsymbol{x}^{\theta}(t^{\theta})$ for $\boldsymbol{t} = [t^{\gamma\alpha}, t^{\theta}]^T$. If this system has no solution, the lines are parallel but not overlapping. When the system has infinitely many solutions, they are in line with each other. In the first case the lines do not intersect and no triple point is present. In the second case it just means the phase between these two interfaces has been dissolved within this triangle. If $t^{\gamma\alpha}$ and t^{θ} are both between zero and one, we know the two line segments intersect in this triangle, otherwise the point of intersection is outside this triangle. First we will show how to deal with two interface segments crossing the same triangle and intersecting within the triangle. Then we show what to do when the interfaces do not intersect.

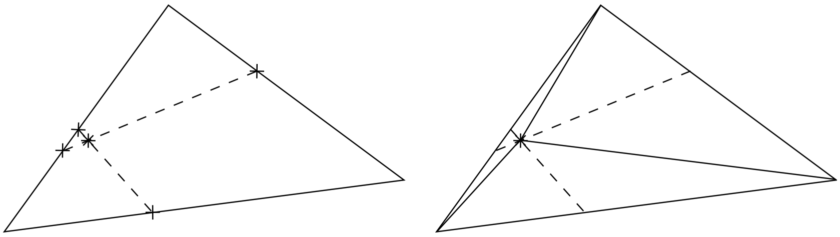


Figure 5.1: When an intersection of two interfaces is close to an edge of the triangle, narrow triangles have to be created (see left figure). The added lines in the right figures represent the most optimal splitting of the quadrilaterals, which gives ill defined triangles. The +-marks here are the cross points of the interfaces with the triangle and the star mark represents the triple point.

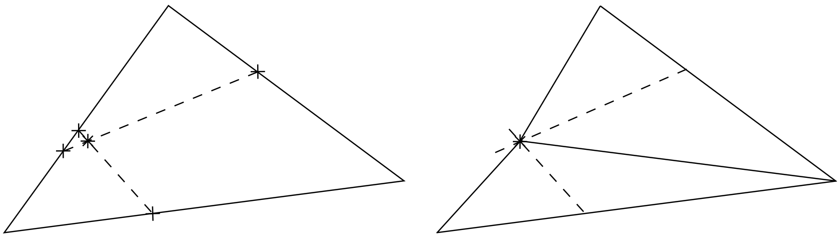


Figure 5.2: 'Bending' the edge to the triple point.

Intersecting Interfaces

A triple point is where the interfaces $\Gamma^{\gamma\alpha}(t), \Gamma^{\gamma\theta}(t)$ and $\Gamma^{\alpha\theta}(t)$ connect. These points have to be added to the mesh in all cases, but also small or odd shaped triangles have to be avoided as best as possible. We deal with these points by, again, using the shift and cut techniques, but also introduce a third technique we will call 'bending'. The shift and cut techniques are the same as shifting and cutting for only one interface crossing the triangle, where if the triple point is 'close' to one of the vertices, that vertex shifts to the triple point and if the triple point is far enough from every vertex, it is added and the triangle is split up in three new triangles including the triple point. However, once a triple point is close to one of the edges and is not shifted, the triangle that is created with this edge becomes oddly shaped, see for instance Figure 5.1 on the right. In stead of only adding this point, this edge is 'bend' to the triple point (see Figure 5.2 on the right). The edge is replaced by two edges connecting its end points with the triple point. A possible disadvantage of this technique is that a bit of the interface position might be lost, but avoiding bad triangles is deemed more important.

To quantify if a triple point $\boldsymbol{x}^{\perp}$ is close to a vertex, close to an edge or none of these two, we use barycentric coordinates. Barycentric coordinates define a point by a linear combination of the vertices of the triangle. For a point $\boldsymbol{x}^{\perp}$ we write:

$$\boldsymbol{x}^{\perp} = \lambda_1\boldsymbol{x}_1 + \lambda_2\boldsymbol{x}_2 + \lambda_3\boldsymbol{x}_3, \tag{5.7}$$

$$\lambda_1 + \lambda_2 + \lambda_3 = 1, \tag{5.8}$$

where $\boldsymbol{x}_i, i = 1,2,3$ are the vertices of the triangle. If $0 \leq \lambda_i \leq 1$ for all $i = 1,2,3$, then this point is located inside of the triangle. For a triple point $\boldsymbol{x}^{\perp}$ on an edge, we see that one of the three λ_i values must be zero. In this case we get the τ^e-definition used for shifting or cutting points on an edge, thus if for instance $\lambda_1 = 0$ and $\lambda_2 < \delta$, we shift the vertex $\boldsymbol{x}_3$ to $\boldsymbol{x}^{\perp}$, as $\lambda_3 > 1-\delta$ means the triple point is closest to this vertex. For a point inside the triangle, where $\lambda_i > 0, i = 1,2,3$, we shift when $\lambda_i < \delta$ for two i's, cut when all $\lambda_i > \delta$ and bend for the other cases. In Figure 5.3 this is visualized.

Not Intersecting

If the interfaces do not intersect, so both values of $\boldsymbol{t}$ are not between zero and one or if Equation (5.6) has no solution, this gives three cases where extra attention has to be paid to the re-triangulation. These are:

1. One point of the triangle will be shifted, meaning one of the two interfaces goes through this point and the other three intersection points will have to be added.

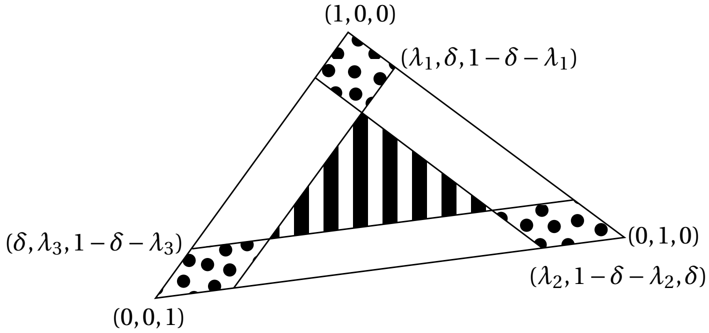

(1,0,0)
$(\lambda_1, \delta, 1-\delta-\lambda_1)$
$(\delta, \lambda_3, 1-\delta-\lambda_3)$
(0,1,0)
$(\lambda_2, 1-\delta-\lambda_2, \delta)$
(0,0,1)

Figure 5.3: Visualization of shift (dotted), cut (striped) and bend (white) areas. Several important points on the triangle are given in their barycentric form $(\lambda_1, \lambda_2, \lambda_3)$.

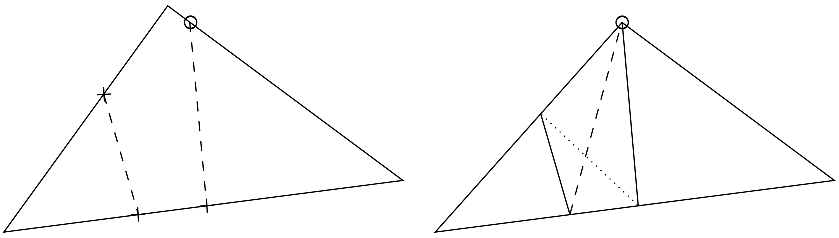


Figure 5.4: When two interfaces cross the triangle such that there is one shift (circle) and three cuts (+-marks in the left figure). This results in four triangles, where two different triangles (dotted and striped in the right figure) can be formed in the middle quadrilateral.

2. When the interfaces cut the triangle along two of the same edges.
3. When the interfaces both cut one similar edge and the two other edges separately.

Below we show these three cases and how to handle the re-triangulation for the added points by cutting.

One Shift and Three Cuts

When one point is shifted, the interface passing through this point will always have to cut the triangle in the opposite edge. For the interfaces not to intersect, the other interface will have to go through this same edge and the edge on the same side of the shifted point, for visualization see Figure 5.4. Any other way would mean the interfaces cross each other. With this one shift and three cuts it is easy to see that four triangles are created from which two are fixed by the interfaces and the two triangles in the middle have to be created from the three cut points and one shifted point. Between these four points two different triangulations can be made. We choose the triangulation that has lowest variance in angle in both the new triangles.

Four Cuts Along Two Edges

If two interfaces cut the triangle such that two edges are cut twice, we get seven triangles where two are fixed by the interfaces. See Figure 5.5. The quadrilateral and the pentagon are also triangulated such that the triangles will have lowest variance angles.

Four Cuts Along Three Edges

The second option for two interfaces to cross the triangle, is when one edge is cut twice and the other edges cut once. Then also seven triangles are created, from which one is fixed and three quadrilaterals have to be re-triangulated as can be seen in Figure 5.6.

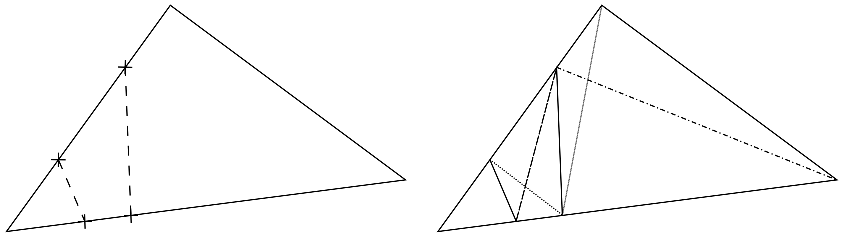


Figure 5.5: When two interfaces cross the triangle such that there are four cuts (X-marks in the left figure) over two edges. This results in five triangles (see right figure), where there are two options for the area between the two interfaces and for the interface and the part of the triangle that is not cut.

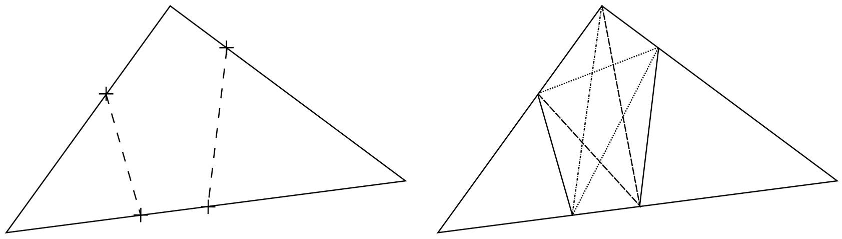


Figure 5.6: When two interfaces cross the triangle such that there are four cuts (X-marks in the left figure) over all three edges. This results in five triangles (see right figure), where there are three different options (dotted, striped and stripe dotted) in the middle pentagon.

5.1.3. Algorithm

The order in which the different techniques are applied to the background mesh will have an influence on the final enriched mesh. In general shifting when possible is preferred, so will be done first. Although shifting also changes the level-set function. That is why before mesh creation we find the position of the triple points based on the level-set functions. Only then do we shift. From there we add the triple points and then cut. However, when a pair of triple points come close to each other, the resolution of the mesh can give issues when using the bend technique on the edges of the same element. We assume that when two triple points come close to each other, the triple points will soon disappear and the interface in between will disappear. Numerically, a pair of triple points will vanish when they end up in the same triangle. This resolution issue can however happen just before the collapse. So we take a bigger time step when there is an overlap between neighboring triangles of the triple points by doubling the CFL number to see if the triple points will disappear. If not, we double the CFL-number again and check the triple points again. If still the triple points do not disappear, we run with the original CFL-number and must check if the mesh creation goes correctly. To be able to double the CFL number twice and suffice the CFL condition, we set CFL = 0.2 such that the two times doubled CFL value 0.8 is still below one. The algorithm for creating the enriched mesh is written in pseudo-code in Algorithm 5.1 and an example is shown in Figure 5.7. This shows the meshing of the initial domain with $h_{\max} = {}^1\!/_4$ zoomed in on the ferrite particle.

Because the mesh changes after shifting vertices, it is possible that vertices which could not shift before, can do now. For this reason we keep on shifting as long as we can identify vertices that can be shifted. How many iterations are needed depends on δ, as it will determine how likely a vertex will shift, but most of the time only one iteration is needed.

Algorithm 5.1 The enriched mesh creation algorithm in pseudo-code.

```
Find triple points;
while points shift do
    Shift vertices to interfaces;
end while
Shift/cut/bend triple points to found triple points;
Cut triangles where one interface cuts, or two interfaces cut but do not interfere;
Cut triangles where two interfaces cut and a different treatment is needed;
```

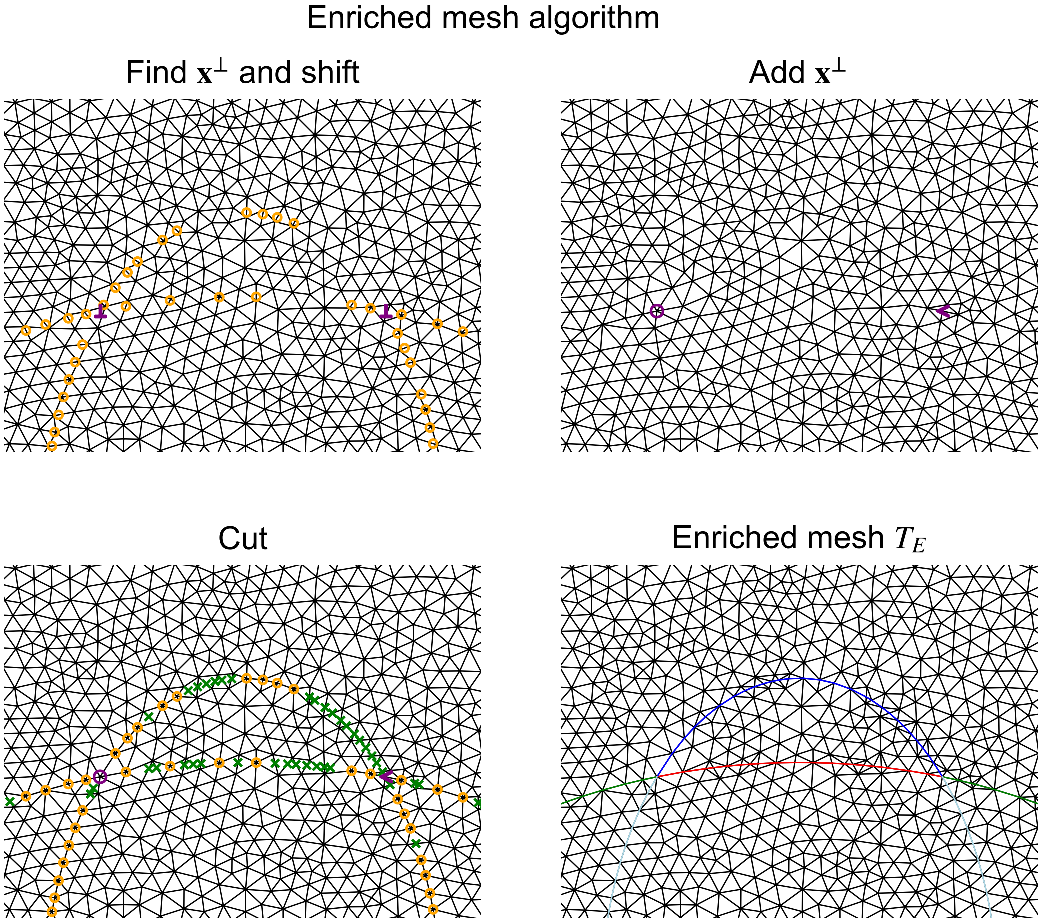


Figure 5.7: The different steps of the mesh creation algorithm. The top left shows the first 2 steps of the algorithm, finding the triple points (purple ⊥-marks) and shifting background nodes (orange ∘-marks) to the interfaces if possible. Then the triple points are added (shifted: ∘-mark, cut: x-mark) in the top right. In the bottom left the cut points (green x-marks) are also added and we end up with the enriched mesh including the interfaces (blue: $\Gamma^{\gamma\alpha}$, green: $\Gamma^{\gamma\theta}$, red: $\Gamma^{\alpha\theta}$ and light blue: $\Gamma^{\gamma\alpha}_{\text{ghost}}$) in the right bottom figure. All the domains are already quite accurately approximated even though a low amount of background points is used for this mesh.

5.1.4. The Quality and Accuracy of the Mesh

In the creation of the mesh we considered two things. One, the quality. A qualitatively perfect mesh is if all triangle elements have 60, or close to 60, degree angles ($60° = \pi/3$). The background mesh T we use is a mesh created such that all edges are approximately $h_{\max}$ or less in length and approaches this optimal quality using the Delaunay triangulation method. However, when we create the enriched mesh T_E^n at any time t^n, points are shifted and added to the mesh to capture the interfaces/domains. This dilutes the quality of the background mesh. How much the quality is lost, with fixed interfaces, depends on δ and the specific locations of the interfaces.

To check the quality of the mesh we will look at the equiangular skewness of all triangles. The lower the skewness is, the more triangles will have angles close to 60 degrees. The equiangular skewness of a triangle Δ is defined as

$$\text{Skew}(\Delta) = \max\left\{\frac{\theta_{\max} - \frac{\pi}{3}}{\pi - \frac{\pi}{3}}, \frac{\frac{\pi}{3} - \theta_{\min}}{\frac{\pi}{3}}\right\}, \tag{5.9}$$

where $\theta_{\max}$ and $\theta_{\min}$ are the maximal/minimal angle of the triangle respectively. Skew is a value between zero and one, where zero means a perfect qualitative triangle with only 60 degrees angles. A value above 0.8 is to be avoided, as the closer the skewness is to one, the closer it is to the 'triangle' with angles $(\pi, 0, 0)$. The closer the skewness is to zero, the closer the triangle is to the triangle with angles $(\pi/3, \pi/3, \pi/3)$.

The second property of the mesh is the accuracy. With accuracy we mean how accurate the enriched mesh captures the interfaces. This also depends on δ, but also on the accuracy of the level-set functions. As we approximate the interfaces with a piecewise linear function we expect a second order accurate approximation of the interfaces and domains. For $\delta = 0$ all intersections of the interface with the mesh will be added as grid points, this means the approximation of the interface is at its best. But it will also mean we create triangles where two vertices can be very close to each other and the third vertex on $h_{\max}$ distance. To avoid these ill-posed triangles we want to shift these points, although if $\delta = 1/2$ and we always shift, the position of the interface can be lost. See for instance Figure 5.8 for both extremes $\delta = 1/2$ (a) and $\delta = 0$ (b) and the value $\delta = 0.3$ (c) in between.

The accuracy of the interface approximation can be qualified by looking at the calculated and analytical length of the different interfaces and the area of the different domains. The analytical arc length on an ellipse can be calculated with the elliptic integral of the second kind and has to be numerically approximated. The analytical surface area of an ellipse is $ab\pi$. To find the analytical area of the α-domain, we use the technique described in [6].

Clearly the value in between the two extremes should result in an accurate but also qualitative good mesh. In Section 7.2 we try to find an optimal value for $\delta \in [0, 1/2]$.

5.1.5. Ghost Interface Extension

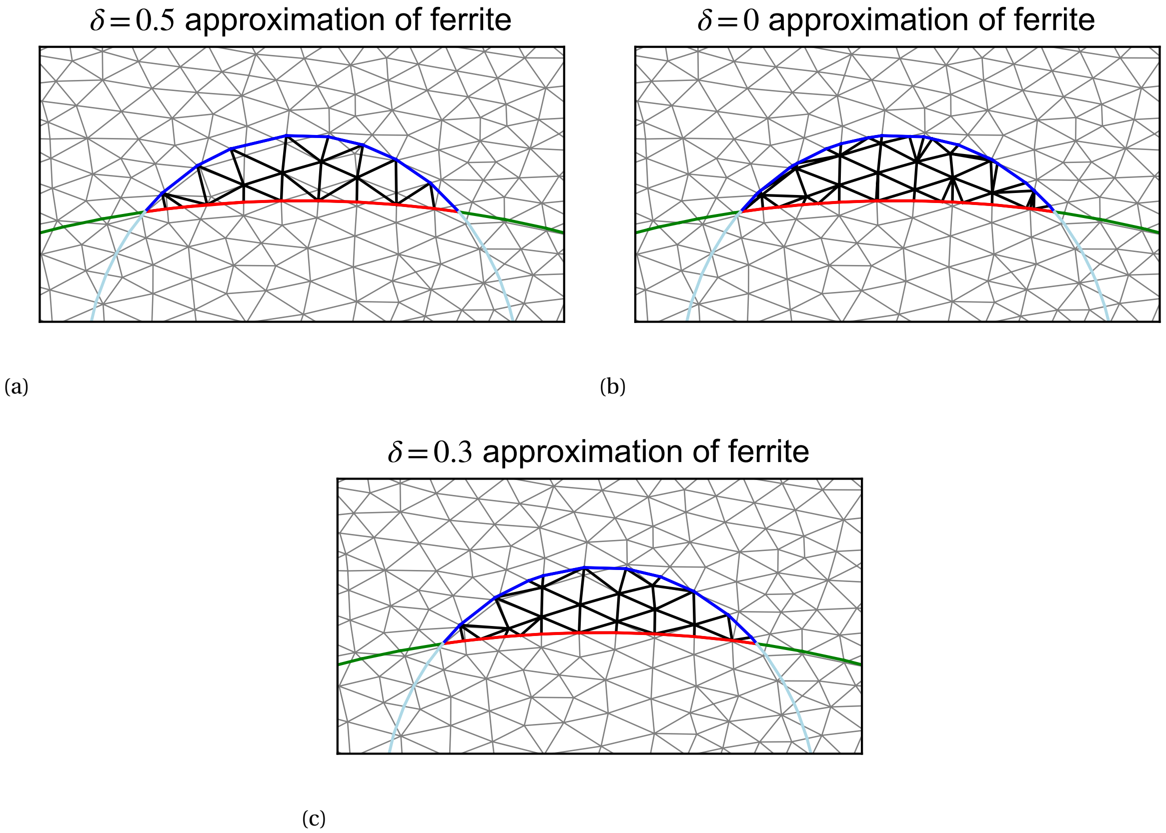
Even though we set the ghost interface velocity to be zero in the point $\boldsymbol{x}_{1/2}$ as described in Section 3.2.1, at a certain moment in time the ghost interface could come close to itself and connect with itself. When this happens, the fixed point is moved to the point on the ghost interface such that the ghost interface is a line again. Meaning every point on the ghost interface has two neighboring ghost interface points except for the triple points, which are the end of the line. The other ghost interface points are deleted and $\phi^{\gamma\alpha}(\boldsymbol{x}, t)$ has to be reinitialized without these interface points.

5.2. Extending Concentration to New Time Step

As shown in Section 4.1.3 a dimensional mismatch can occur because of the matrix-vector multiplication $M_k^{n+1}\boldsymbol{c}_k^n, k = \gamma, \alpha$. In order to fix this mismatch we extend $\boldsymbol{c}_k^n$ to $\boldsymbol{c}_{k,\text{ext}}^n$ from the mesh T^n to the new mesh T^{n+1}.

For new internal grid points $\boldsymbol{x}_i^{n+1} \in T_k^{n+1}$ we find the element $\Omega_k^{m,n} = \Delta_k^{m,n}$ in which this point is located in the old mesh and obtain the carbon concentration value by interpolation

$$\left(\boldsymbol{c}_{k,\text{ext}}^n\right)_i = \sum_{j=1}^{3} \phi_j(\boldsymbol{x}_i)\left(\boldsymbol{c}_k^n\right)_j, \tag{5.10}$$



$\delta = 0.5$ approximation of ferrite

(a)

$\delta = 0$ approximation of ferrite

(b)

$\delta = 0.3$ approximation of ferrite

(c)

Figure 5.8: (a): The α-domain when points are always shifted. The faded points and triangles show the background mesh. (b): The α-domain when points are always cut. (c): The α-domain when points are both cut and shifted with $\delta = 0.3$. The faded points and triangles in the back show the background mesh. Clearly this value of δ combines the better properties of the other two values. Where always cut results in small, odd shaped triangles when the intersection is close to a mesh points, shift gives a good approximation of the interface, but does not lose quality. When an intersection is more in between two grid points the always shift technique loses the accuracy of capturing the interface, whereas the cut technique captures it almost perfectly.

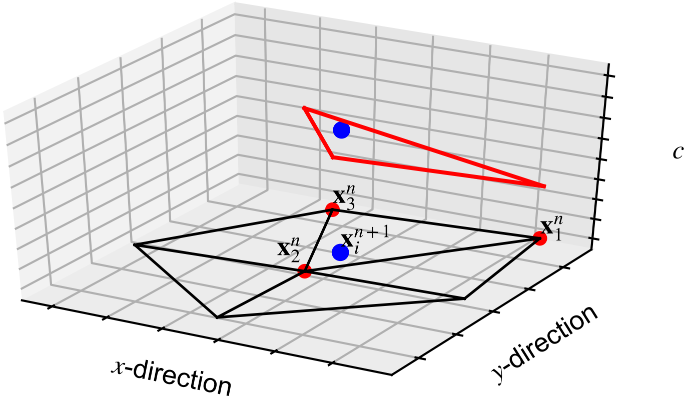


Figure 5.9: Concentration value in new point $\boldsymbol{x}_i^{n+1}$ located in triangular element is found by interpolation from nodes x_j, $j = \{1,2,3\}$ of the old mesh.

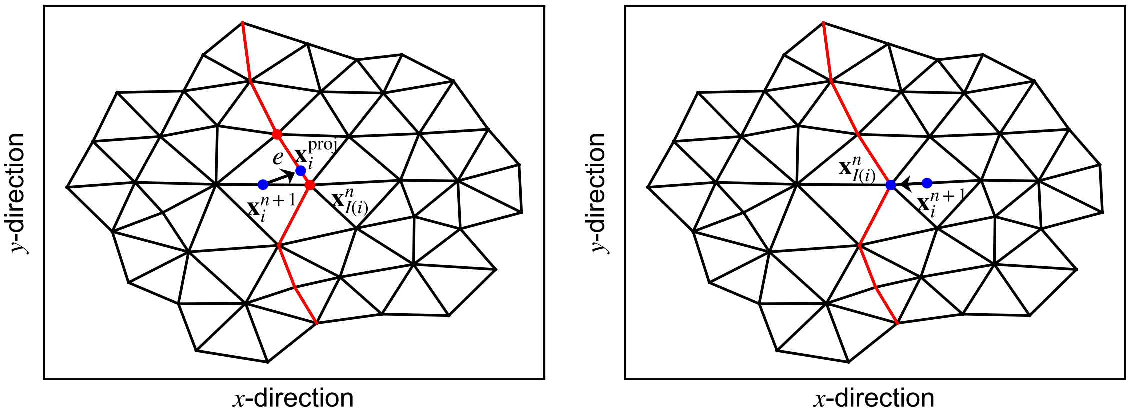


Figure 5.10: On the left the carbon concentration in new interface node i is obtained by projecting it on the edge $\boldsymbol{e}$ of the old mesh. On the right the carbon concentration is set equal to the carbon concentration in the nearest node $\boldsymbol{x}_{I(i)}$. The red edges represent the interface in the neighborhood (up unto the third neighbors) of $\boldsymbol{x}_{I(i)}$.

where j are the indices of the nodes in the element $\Delta_k^{m,n}$. See Figure 5.9 as example for the extension for an internal point with a triangulated mesh and linear basis functions.

For new grid points $\boldsymbol{x}_i^{n+1}$ on an interface, we find the closest point or edge of the old interface in a band of third neighbor nodes in T_k^n around $\boldsymbol{x}_{I(i)}^n$, denoted as $N_3^n(\boldsymbol{x}_{I(i)}^n)$. If an edge $e = (\boldsymbol{e}_1, \boldsymbol{e}_2)$ is closest we project $\boldsymbol{x}_i^{n+1}$ on this edge in point $\boldsymbol{x}^{\text{proj}}$. We set

$$\left(\boldsymbol{c}_{k,\text{ext}}^n\right)_i = \sum_{j\in\{e_1,e_2\}} \phi_j\left(\boldsymbol{x}^{\text{proj}}\right)\left(\boldsymbol{c}_k^n\right)_j. \tag{5.11}$$

If the point $\boldsymbol{x}_{I(i)}^n$ of the old interface is closest, the concentration is extended by giving it the concentration value of this point:

$$\left(\boldsymbol{c}_{k,\text{ext}}^n\right)_i = \left(\boldsymbol{c}_k^n\right)_{I(i)}. \tag{5.12}$$

See Figure 5.10 as example for the extension of a new interface point projected on an edge and on the closest point.

5.3. Zhangs and Naga's Gradient Approximation

To calculate the normal vectors and velocities, respectively gradients of the level-set function and the ferrite carbon concentration have to be calculated. In order to give an accurate approximation we will use the approximation method of Zhang and Naga [31], which is super convergent if the mesh is regular in the neighborhood of the point in which the gradient is required. This method will give a second order accurate approximation in a point $\boldsymbol{x}^p$ using a patch with at least ${}^1\!/_2 d^2 + {}^3\!/_2 d + 1$ mesh points for linear elements in $\mathbb{R}^d$. This patch is used to create a quadratic fit $a_{h^2}(\boldsymbol{x})$ to $a_h(\boldsymbol{x})$, where $a = \phi(\boldsymbol{x}, t)$ or $c_\alpha(\boldsymbol{x}, t)$ in our case. The quadratic fit in general form is given as

$$a_{h^2}(\boldsymbol{x}) = A + B^T\boldsymbol{\eta} + \eta^T C\boldsymbol{\eta},$$

with A a scalar, B a $d \times 1$ vector and C a $d \times d$ symmetric matrix and $\boldsymbol{\eta} = \left(\eta_x, \eta_y\right)$, for $d = 2$, the scaled centralized coordinate

$$\boldsymbol{\eta} = \frac{\boldsymbol{x} - \boldsymbol{x}^p}{h_p},$$

where h_p is the length of the longest edge attached to $\boldsymbol{x}^p$. Then in the point $\boldsymbol{x}^p$ the gradient is given as

$$\nabla a_{h^2}(\boldsymbol{x}) = h_p^{-1}B.$$

In case of the normal vector we have, with $a_{h^2} = \phi_{h^2}$,

$$\boldsymbol{n}_h(\boldsymbol{x}) = \frac{\nabla\phi_{h^2}(\boldsymbol{x})}{\|\nabla\phi_{h^2}(\boldsymbol{x})\|_2} = \frac{B}{\|B\|_2}.$$

To find A, B and C we approximate a solution of the following system with the least square technique

$$M_\eta \left[A \;\; B_1 \;\; B_2 \;\; C_{12} \;\; C_{11} \;\; C_{22}\right]^T = a(\boldsymbol{x}),$$

where

$$M_\eta = \begin{bmatrix} 1 & \eta_{x_1} & \eta_{y_1} & \eta_{x_1}\eta_{y_1} & \eta_{x_1}^2 & \eta_{y_1}^2 \\ 1 & \ddots & & & & \eta_{y_2}^2 \\ \vdots & & & & \ddots & \vdots \\ 1 & \eta_{x_n} & \eta_{y_n} & \eta_{x_n}\eta_{y_n} & \eta_{x_n}^2 & \eta_{y_n}^2 \end{bmatrix},$$

n is the amount of nodes in the patch and

$$a(\boldsymbol{x}) = \left[a(\boldsymbol{x}_1) \ldots a(\boldsymbol{x}_n)\right]^T.$$

Although this method has the super convergence property for internal grid points on a regular mesh, approximating the gradient at the boundaries of the domain can be a problem, as was also shown in the work of Zhang and Naga [31]. Especially for domains where the boundary forms sharp corners there is a loss of accuracy even with local grid refinement. The patch that is used to recover the gradient at a boundary point $\boldsymbol{x}^{\mathrm{B}}$ is found by looking for neighbors of $\boldsymbol{x}^{\mathrm{B}}$ until you have found one or more internal nodes and adding the patch of all these internal nodes to the nodes found so far. This technique gives a decent approximation if the mesh is regular, but because our model has sharp corners near the triple points in the ferrite domain, the quality of the mesh in this region is quite low and an internal point is often located multiple neighbors away. This can give an unsymmetrical patch because of different edge lengths (mostly because of cutting). In turn, because the ferrite carbon concentration has a steep gradient in the neighborhood of the triple points, a patch that contains more points towards the austenite-ferrite or the ferrite-cementite interface will give a very different result. With this technique we observe irregularities in our approximation of the normal gradient of the ferrite carbon concentration on the austenite ferrite interface, which results in an inaccurate approximation of the interface velocity. Which in turn disturbs the movement of the interface.

We make a slight adjustment to the technique of Zhang and Naga [31], which will give slightly better result for our case. In stead of directly taking the neighbors of $\boldsymbol{x}^{\mathrm{B}}$, we use the same spheres around point $\boldsymbol{x}^{\mathrm{B}}$ of radius $kh_p, k = 1, 2, \ldots$ as defined for internal points until we have found at least one internal point and then add the patches of all the internal points. This results in a more symmetrical patch for boundary points that has only boundary points for its first several layers of neighbors.

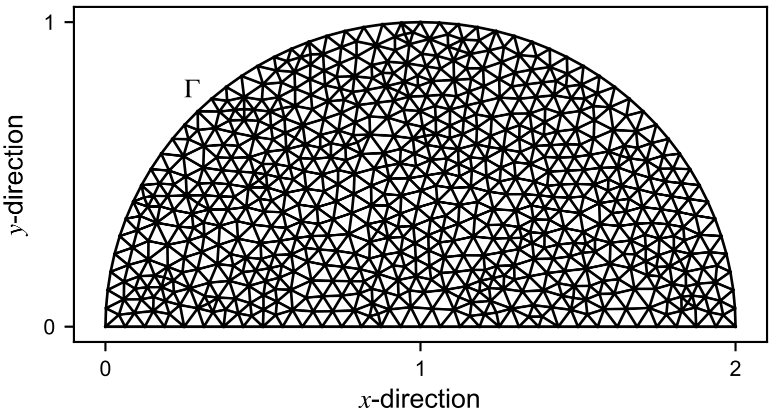


Figure 5.11: Delaunary triangulated half circle with center $(1,0)$, where we recover the gradient in the points on Γ.

In the results (Chapter 7) we will apply this gradient approximation method for two test cases. The first test case is a half circle with center $(1,0)$ and radius one (see Figure 5.11), which is discretized with a Delaunay triangulation. We recover the gradient on the arc of the circle, with only the values of the points inside the half circle. The function f for which we approximate the gradient is defined as the shortest distance to the arc of the circle, which is the radius minus the distance from the node to the center. For the second test case we use the initial ferrite domain and recover the gradient of the initial level-set function $\phi^{\gamma\alpha}(\boldsymbol{x},t)$ on the enriched mesh T_E^{α} at time $t=t_0$. For this second test case we will also show the ferrite carbon concentrations and the austenite-ferrite interface velocity that contains the recovered ferrite carbon concentration gradient on the austenite-ferrite interface.

5.3.1. Level-Set Reinitialization

We do note in our results that the above discussed method is still not a perfect solution for the gradient approximation and has to be taken into account for the validation of this model. Because we need the gradient of the ferrite carbon concentration on the austenite-ferrite interface and we only have ferrite carbon concentration values on the ferrite domain, we are forced to use the ferrite part of the enriched mesh T_{α}^{n} for the gradient approximation. However for the level-set function update and reinitialization we are allowed to update and reinitialize on the background mesh T. This mesh is a regular Delaunay triangulated mesh, which means the gradient approximation should be good with Zhang and Naga's gradient approximation method.

To impose the Dirichlet boundary condition $\phi(\boldsymbol{x},t)=0$ on the interface $\Gamma(t)$ we use Lagrange Multipliers with the implicit Dirichlet matrix F containing the τ^e values of Equation (5.2) on the nodes of all the edges where $\phi(\boldsymbol{x},t)$ changes sign. So for an edge where $\phi(\boldsymbol{x},t)$ changes sign we get

$$(1-\tau)\,\phi(\boldsymbol{e}_1,t)+\tau\phi(\boldsymbol{e}_2,t)=0. \tag{5.13}$$

This gives the system

$$\begin{bmatrix} S & F^T \\ F & \emptyset \end{bmatrix}\begin{bmatrix} \boldsymbol{\phi} \\ \boldsymbol{\Lambda} \end{bmatrix}=\begin{bmatrix} \text{RHS} \\ \boldsymbol{0} \end{bmatrix}. \tag{5.14}$$

Here S and RHS are obtained from Equation (4.22) together with time integration for the reinitialization process. $\boldsymbol{\Lambda}$ contains the Lagrange multipliers which can be discarded after the calculations.

Reinitialization is only needed when $\left\|\nabla\phi(\boldsymbol{x},t)\right\|_2$ is no longer close to one in the neighborhood of the interface. Because the initial level-set functions $\phi^{\gamma\alpha}(\boldsymbol{x},t)$ and $\phi\theta(\boldsymbol{x},t)$ are known (close to) analytically we have a very good approximation of $\left\|\nabla\phi(\boldsymbol{x},t)\right\|_2=1$ at $t=t_0$. We define the error

$$\epsilon_n^k=\sqrt{\sum_{\boldsymbol{x}\in N_3^T(\Gamma^k(t))}\left(\left\|\phi^k(\boldsymbol{x},t)\right\|_2-1\right)^2},\ \text{for}\ ,t=t_n,k\in\left\{\gamma\alpha,\theta\right\}, \tag{5.15}$$

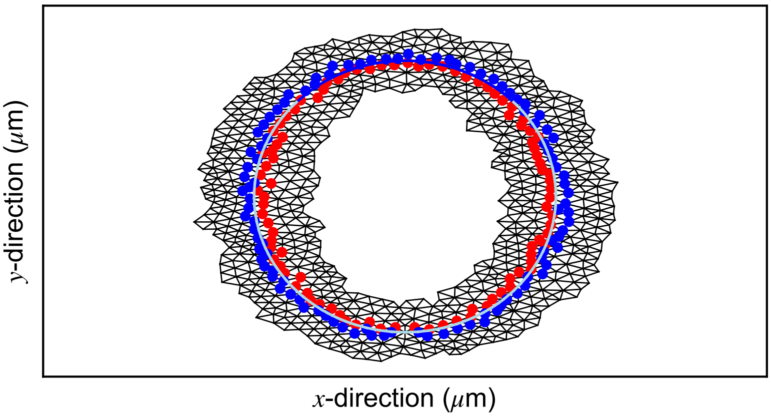


Figure 5.12: The band of all third neighbors around the nodes from the edges where $\phi^{\gamma\alpha}$ changes sign. The blue and red points represent the points of the edges with negative/positive $\phi^{\gamma\alpha}$ respectively. The blue line is the austenite/ferrite interface and the light blue line is the ghost interface.

where $N_3^T\left(\Gamma^k(t)\right)$ is the band of nodes in T around the interface $\Gamma^k(t)$ of third neighbors of the nodes which are located on the edges where $\phi^k(\boldsymbol{x},t)$ changes sign (see Figure 5.12). $\phi^k(\boldsymbol{x},t_n)$ will be reinitialized if ϵ_n^k is bigger than $C_R\epsilon_0^k$, with $C_R > 1$ some constant.

We will also look at reinitialization with the following system:

$$\begin{bmatrix} S \\ F \end{bmatrix} \boldsymbol{\phi} = \begin{bmatrix} \text{RHS} \\ \mathbf{0} \end{bmatrix}. \tag{5.16}$$

A least square approximation is done to obtain the reinitialized level-set function. This does change the location of the zero-contour (the interfaces) slightly, but gives the reinitialization process more freedom in changing the level-set function around the interface.
We will notice in the results (Chapter 7) that for both methods the constraint to keep the interface fixed gives problems when reinitializing. Thus we set $C_R = \infty$ for simulations, meaning reinitialization is never done.

5.4. Algebraic Flux Correction

In a system with too strong convective effects or when the diffusion tensor is anisotropic, the standard Galerkin discretization will fail certain discrete maximum principles. Algebraic Flux Correction (AFC) [10] can be used to correct these failures. The idea of AFC is a generalization of the flux-correction transport (FCT) methods.

In a discrete system written as

$$(M - \Delta t(K+L))\,\boldsymbol{c}^{n+1} = M\boldsymbol{c}^n, \tag{5.17}$$

where M is the mass matrix, K is the discrete convective operator and L the discrete diffusion operator, the discrete maximum principles can be violated by

- positive off-diagonal entries of the mass matrix M,
- negative off-diagonal entries of the discrete convection operator K,
- negative off-diagonal entries of the discrete diffusion operator L.

To resolve these violations, the AFC method constrains these entries by adding a certain amount of discrete diffusion. It does this by decomposing the matrices in its 'good' and 'bad' parts and then limits on this 'bad' part of the system. This includes using the lumped mass matrix M_L, adding a discrete diffusion operator D to K, $\tilde{K} = K + D$, with

$$d_{ij} = \max\left\{-k_{ij}, 0, -k_{ji}\right\}, d_{ii} = -\sum_{j\neq i} d_{ij}, \tag{5.18}$$

and change L with $\tilde{L} = L - L^{+}$, where L^{+} is the anti-diffusive part of L. These are the off-diagonal entries that are strictly negative, i.e.

$$l_{ij}^{+} = \min\{0, l_{ij}\}, l_{ii}^{+} = -\sum_{i \neq j} l_{ij}^{+}. \tag{5.19}$$

From these definitions we can rewrite Equation (5.17) to

$$M_L \frac{d\boldsymbol{c}}{dt} = \left(\tilde{K} + \tilde{L}\right)\boldsymbol{c} + \boldsymbol{f}(\boldsymbol{c}), \tag{5.20}$$

where

$$\boldsymbol{f}(\boldsymbol{c}) = (M_L - M)\frac{d\boldsymbol{c}}{dt} + \left(K - \tilde{K}\right)\boldsymbol{c} + \left(L - \tilde{L}\right)\boldsymbol{c}, \tag{5.21}$$

is the anti-diffusive part of Equation (5.17). The AFC method limits the numerical fluxes from which this anti-diffusive term $\boldsymbol{f}(\boldsymbol{c})$ is composed of, such that there is no violation of the (semi-)discrete maximum principle. In areas where these fluxes violate this principle, the flux is limited by some coefficient $\alpha \in [0,1)$ ($\alpha = 1$ if no violation is present).

The flux $\boldsymbol{f}(\boldsymbol{c})$ can also be written as:

$$\boldsymbol{f} = (M_L - M)\frac{d\boldsymbol{c}}{dt} - D\boldsymbol{c} + L^{+}\boldsymbol{c}, \tag{5.22}$$

where we used the fact that $D = \tilde{K} - K$ and $L^{+} = L - \tilde{L}$. The flux from node j to i can be rewritten using the zero row sum property of $M_L - M, D$ and L as

$$f_{ij} = m_{ij}\frac{d}{dt}\left(c_i - c_j\right) + \left(d_{ij} - l_{ij}\right)\left(c_i - c_j\right), \forall j \neq i. \tag{5.23}$$

The diagonal terms are set as the net-flux of node i

$$f_i = \sum_{j \neq i} f_{ij}. \tag{5.24}$$

With $\tilde{L}, \tilde{K}$ and the lumped mass matrix M_L, a low order approximation $\boldsymbol{c}^L$ of $\boldsymbol{c}$ can be calculated without any anti-diffusive effects.

$$M_L \frac{d\boldsymbol{c}^L}{dt} = \left(\tilde{K} + \tilde{L}\right)\boldsymbol{c}. \tag{5.25}$$

This solution certainly respects the discrete maximum principle and is used to approximate the flux. There are two easy ways in which ${}^{dc^L}/{}_{dt}$ can be approximated. With the simple low order system (5.25):

$$M_L \frac{d\boldsymbol{c}^L}{dt} = \left(\tilde{K} + \tilde{L}\right)\boldsymbol{c}^L, \tag{5.26}$$

or by solving:

$$M \frac{d\boldsymbol{c}^L}{dt} = (K + L)\boldsymbol{c}^L. \tag{5.27}$$

The second option is a less diffusive approximation, but results in a more difficult system to solve. We will use the more diffusive approximation for the concentrations c_γ and c_α calculations.

The flux f_{ij} becomes

$$f_{ij} = m_{ij}\left(\frac{dc_i^L}{dt} - \frac{dc_j^L}{dt}\right) + \left(d_{ij} - l_{ij}\right)\left(c_i^L - c_j^L\right), \forall j \neq i, f_{ii} = \sum_{j \neq i} f_{ij}. \tag{5.28}$$

We will then limit this raw flux with the multidimensional FCT limiter using f_{ij}, which returns coefficients α_{ij} and gives the limited flux

$$\overline{f}_i = \sum_{j \neq i} \alpha_{ij} f_{ij}, 0 \leq \alpha_{ij} \leq 1. \tag{5.29}$$

With this limited flux we calculate the new concentration $\boldsymbol{c}$

$$M_L \boldsymbol{c}^{n+1} = M_L \boldsymbol{c}^L + \Delta t \overline{\boldsymbol{f}}. \tag{5.30}$$

Algorithm 5.2 Zalesak's multidimensional FCT limiter.

```
1: Compute the sums of positive/negative anti-diffusive fluxes into node $i$
       $$P_i^+ = \sum_{j\neq i} \max\{0, f_{ij}\}, \quad P_i^- = \sum_{j\neq i} \min\{0, f_{ij}\}. \qquad (5.33)$$
2: Calculate the distance from node $i$ to the local maximum/minimum and the bounds
       $$Q_i^+ = \frac{(m_L)_{i,i}}{\Delta t}\left(c_i^{\max} - \tilde{c}_i\right), \quad Q_i^- = \frac{(m_L)_{i,i}}{\Delta t}\left(c_i^{\min} - \tilde{c}_i\right). \qquad (5.34)$$
3: Find the correction factor for the net increase/decrease of flux to node $i$
       $$R_i^+ = \min\left\{1, \frac{Q_i^+}{P_i^+}\right\}, \quad R_i^- = \min\left\{1, \frac{Q_i^-}{P_i^-}\right\}. \qquad (5.35)$$
   $R_i^\pm := 1$ for nodes $i$ that have Dirichlet boundary conditions, as the value $c_i$ does not depend on $\alpha_{ij}$.
4: Check the sign of the anti-diffusive flux $f_{ij}$ and multiply it by
       $$\alpha_{ij} = \begin{cases} \min\left\{R_i^+, R_j^-\right\}, & \text{if } f_{ij} > 0, \\ \min\left\{R_i^-, R_j^+\right\}, & \text{if } f_{ij} < 0. \end{cases} \qquad (5.36)$$
```

To find α_{ij} we will use Zalesak's FCT limiter [30]. This limiters looks at a solution update in the form

$$m_i c_i = m_i \tilde{c}_i + \Delta t \sum_{j\neq i} \alpha_{ij} f_{ij}, \qquad (5.31)$$

like Equation (5.30), where $\tilde{\boldsymbol{c}}$ is a non-oscillatory approximate solution (like (5.25)). Define $c_i^{\min}$ and $c_i^{\max}$ as the local extrema of $\tilde{c}_i$, determined as $\min/\max_{j\in S_i}\left\{\tilde{c}_j\right\}$ respectively, with S_i the set of neighbor nodes of i and itself. The goal is to find the best α_{ij} such that

$$c_i^{\min} \leq c_i \leq c_i^{\max}. \qquad (5.32)$$

This condition promises (5.30) to satisfy the local discrete maximum principle.
Zalesak's algorithm is described in Algorithm 5.2.

Not all FCT limiters have the time step Δt as a restriction in its formulation. As Kuzmin [10] remarks:

> The presence of the time step Δt in the denominator of $Q_i^\pm$ is a blessing or a curse, depending on the purpose of simulation. On the one hand, the LED constraints become less restrictive and, consequently, a larger portion of the raw anti-diffusive flux f_{ij} is retained as the time step is refined. This makes FCT the method of choice for transient computations. On the other hand, the use of large Δt results in a loss of accuracy, and severe convergence problems may occur in the steady state limit.

This means we can not take too large time steps for the backward Euler time discretization, but will be rewarded in accuracy for smaller time steps.

5.4.1. Prelimiting

The goal of AFC is to limit fluxes that can cause local extrema, which violate the discrete maximum principle. There are however also fluxes that help flatten the solution, in stead of steepening it. These fluxes will also be limited by the AFC method without any further adjustment. These fluxes can be found by looking at the sign of $\tilde{c}_j - \tilde{c}_i$ and f_{ij}. If the signs coincide the flux will help flatten the solution. Kuzmin [10] argued that canceling these 'diffusive' fluxes is safer than keeping them in, so we set

$$f_{ij} = 0 \quad \text{if} \quad f_{ij}\left(\tilde{c}_j - \tilde{c}_i\right) > 0. \qquad (5.37)$$

This is done before determining the coefficients α_{ij} and flux limiting.

5.4.2. Reaction Term

Kuzmin's work only mentions the AFC method for convection-diffusion equations. As we also have a reaction (boundary) term, we need to ensure it is taken into account in the application of the AFC. In the work of Gintautas [3] it is noted that without any adjustments the restriction

$$(K+L+R)_{ij} \leq 0, \forall i \neq j, \tag{5.38}$$

where R is the reaction matrix, is not always met because of the extra reaction term

$$\int_{\Omega} K^{kl}(\boldsymbol{x},t)\psi_i(\boldsymbol{x})c_k(\boldsymbol{x},t)d\Omega, k=\gamma,\alpha, l=\{\gamma,\alpha,\theta\}\backslash k.$$

This restriction is one of the restrictions needed to satisfy the discrete maximum principle. An easy way to overcome this is to use the Newton-Cotes approximation on the reaction matrix terms

$$R_{ij} = \int_{\Omega} K^{kl}(\boldsymbol{x},t)\psi_i(\boldsymbol{x})\psi_j(\boldsymbol{x},t)d\Omega \approx K^{kl}(\boldsymbol{x},t)\frac{|\Delta|}{(2+1)!}\delta_{ij}.$$

$K^{kl}(\boldsymbol{x},t)$ are the reaction constants and is taken constant in space. This gives a diagonal matrix for R, thus $R_{ij}=0, \forall i \neq j$.

5.4.3. AFC for Stationary Convection-Diffusion-Reaction Equations

For the stationary problems (2.26) and (2.27) we have the discrete system

$$(K+L+R)\,\boldsymbol{c}_k = \boldsymbol{0}. \tag{5.39}$$

There is no convection present in the stationary equations, so $K=0$. Also with a regular mesh and our diffusion coefficient being constant, no anisotropic anti-diffusion is present. Thus $L^+_{ij}=0, \forall i,j$ in the decomposition of $\tilde{L}=L-L^+$. Which means we only have to approximate the reaction entries for the matrix R with Newton-Cotes and we are left with a zero anti-diffusive flux.

5.5. The Algorithm

To solve the discretized model we use the algorithm as prescribed in the pseudo code Algorithm 5.3. Because we use the implicit RK3 TVD scheme, the Courant-Friedrich-Lewy condition [2] limits the allowed time step by linking it to the mesh size and maximal velocity. We will use CFL = 0.2 such that we are able to double it twice and stay below CFL = 1 for which time integration becomes unstable. Doubling is sometimes necessary due to triple points moving close towards each other in our meshing algorithm introduced in Section 5.1.

Algorithm 5.3 The algorithm in pseudo-code.

```
Create background mesh $T$;
Initialize level-set functions $\boldsymbol{\phi}^{\gamma\alpha}(t_0)$ and $\boldsymbol{\phi}^{\theta}(t_0)$;
Create enriched mesh $T_E^0$;
Create Diffusion mesh $T_\gamma^0$;
Create Diffusion mesh $T_\alpha^0$;
Calculate initial concentrations $\boldsymbol{c}_\gamma(t_0)$ and $\boldsymbol{c}_\alpha(t_0)$;
Calculate initial interface velocities $\boldsymbol{v}^{\gamma\alpha}(t_0), \boldsymbol{v}^{\gamma\theta}(t_0)$ and $\boldsymbol{v}^{\alpha\theta}(t_0)$;
set $t = t_0 = 0$; set $k = 0$; set FPT$= 10^{-6}$;
while $t < t_{\text{end}}$ do
    Calculate $\boldsymbol{v}_n^{\text{ext},\gamma\alpha}, \boldsymbol{v}_n^{\text{ext},\gamma\theta}$ and $\boldsymbol{v}_n^{\text{ext},\alpha\theta}$ on $T_E^k$;
    Set $\Delta t = \text{CFL0}\ {}^{h_{\min}}/_{\max\|\boldsymbol{v}_n^{\text{ext},\gamma\alpha}, \boldsymbol{v}_n^{\text{ext},\gamma\theta}, \boldsymbol{v}_n^{\text{ext},\alpha\theta}\|_\infty}$;
    Calculate $\boldsymbol{\phi}^{\gamma\alpha}(t_{k+1})$ and $\boldsymbol{\phi}^{\theta}(t_{k+1})$ on $T$;
    Find triple points;
    if Overlap in neighboring triangles then
        Set CFL = 2CFL0;
        Calculate $\boldsymbol{\phi}^{\gamma\alpha}(t_{k+1})$ and $\boldsymbol{\phi}^{\theta}(t_{k+1})$ on $T$;
        Find triple points;
        if not any pair of triple points vanished then
            Set CFL = 4CFL0;
            Calculate $\boldsymbol{\phi}^{\gamma\alpha}(t_{k+1})$ and $\boldsymbol{\phi}^{\theta}(t_{k+1})$ on $T$;
            Find triple points;
            if not any pair of triple points vanished then
                Set CFL = CFL0;
                Calculate $\boldsymbol{\phi}^{\gamma\alpha}(t_{k+1})$ and $\boldsymbol{\phi}^{\theta}(t_{k+1})$ on $T$;
            end if
        end if
    end if
    Create enriched mesh $T_E^{k+1}$;
    Create Diffusion mesh $T_\gamma^{k+1}$;
    Create Diffusion mesh $T_\alpha^{k+1}$;
    Calculate concentrations $\boldsymbol{c}_{\gamma,\text{ext}}^k$ and $\boldsymbol{c}_{\alpha,\text{ext}}^k$;
    Calculate interface velocities $\boldsymbol{v}_{\text{mesh}}^{\gamma}(t_{k+1})$ and $\boldsymbol{v}_{\text{mesh}}^{\alpha}(t_{k+1})$;
    Set r = 0; set $\boldsymbol{c}_{\alpha,r}^{k+1} = \boldsymbol{c}_{\alpha,\text{ext}}^k$; Set $\epsilon = \infty$;
    while $\epsilon >$ FPT do
        Calculate $\boldsymbol{c}_{\alpha,r+1}^{k+1}$;
        Calculate $\epsilon$;
        Set $r = r + 1$;
    end while
    Set $\boldsymbol{c}_\alpha^{k+1} = \boldsymbol{c}_{\alpha,r}^{k+1}$;
    Calculate ${}^{\partial c_\alpha}/_{\partial n}$ on $\Gamma^{\gamma\alpha}(t_{k+1})$;
    Set r = 0; set $\boldsymbol{c}_{\gamma,r}^{k+1} = \boldsymbol{c}_{\gamma,\text{ext}}^k$; Set $\epsilon = \infty$;
    while $\epsilon >$ FPT do
        Calculate $\boldsymbol{c}_{\gamma,r+1}^{k+1}$;
        Calculate $\epsilon$;
        Set $r = r + 1$;
    end while
    Set $\boldsymbol{c}_\gamma^{k+1} = \boldsymbol{c}_{\gamma,r}^{k+1}$;
    Calculate initial interface velocities $\boldsymbol{v}^{\gamma\alpha}(t_{k+1}), \boldsymbol{v}^{\gamma\theta}(t_{k+1})$ and $\boldsymbol{v}^{\alpha\theta}(t_{k+1})$;
    Set k = k+1; Set $t = t + \Delta t$;
end while
```

6

Physical Parameters

In our model we have introduced several physical parameters which influence the model's behavior. In order to see if our model resembles the real physical model, these parameters have to be acquired. Some values can be easily found in literature, but some have to be derived under certain assumptions. In this chapter we will give all parameters together with introducing their temperature/time dependency.

Because temperature drops below the eutectoid temperature A_1, austenite will start dissolving and disappear in time with ferrite mostly replacing it. The biggest driving force behind the dissolve/growth process is that the carbon concentrations try to attain equilibrium. The equilibrium concentration $c^{\infty}_{kl}(T)$, between phases k and l, it will want to attain depends on temperature T, making the system temperature dependent. As we decrease the temperature over time, temperature is dependent on time: $T = T(t)$.
The parameters $K^{k\theta}(\boldsymbol{x},t)$ and $K^{\gamma\alpha}(\boldsymbol{x},t)$ are the coefficients that influence the speed of this driving force and are assumed to be only temperature dependent, thus implicitly time-dependent.
The other carbon transporting process in our model is diffusion. Diffusion wants to spread out the carbon concentration evenly over the domain.The rate of this diffusion is dependent of the diffusion coefficient $D_k(\boldsymbol{x},t)$. The higher temperature, the faster atoms can move, thus the diffusion coefficient tends to be bigger at higher temperatures. So $D_k(\boldsymbol{x},t)$ is also implicitly time-dependent. We will assume it has the same value for all $\boldsymbol{x} \in \Omega_k(t)$, meaning it is location independent $D_k(\boldsymbol{x},t) = D_k(t)$.

6.1. Local Equilibrium Concentration

At a temperature just above the eutectoid temperature $A_1 = 1000K$, with a carbon composition between 0.76 and 6.67 wt%, there will be two phases in steel, austenite (γ) and cementite (θ). The carbon equilibrium composition $\text{wt}_{kl}(T(t))\%$ -or concentration $c^{\text{sol}}_{kl}(T(t)), kl \in \{\gamma\alpha, \gamma\theta, \alpha\gamma, \alpha\theta\}$ values can be found in the phase diagram of steel by looking at the given temperature and the equilibrium lines of the different phases (see Figure 6.1).

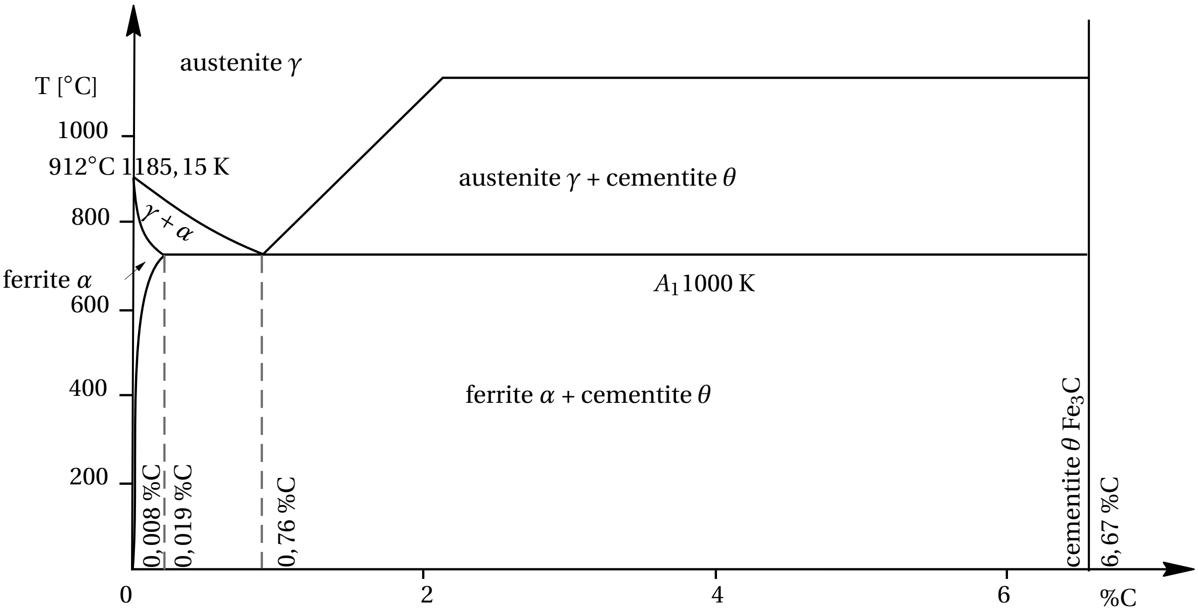


Figure 6.1: Partial phase diagram of steel.

The moment the temperature drops below A_1 it is expected that ferrite will be created, but not immediately as it will need some amount of free energy available on an interface of austenite and cementite to create the BCC structure. This energy threshold is known as the latent heat.

As seen in the phase diagram, there are no equilibrium carbon composition/concentration values for austenite below A_1. This is to be expected, as at this temperature all austenite will be dissolved when equilibrium is attained. We do however need equilibrium concentration values on the interfaces between austenite and ferrite and between austenite and cementite at temperatures below A_1. We can obtain these values by extrapolating the equilibrium lines found in the phase diagram above A_1. These values are obtained by extrapolating data from ThermoCALC [24] assuming an Arrhenius relation

$$c_{kl}^{\infty}(T(t)) = A\exp\left\{\frac{-E_{\Gamma^{kl}}}{R_g T(t)}\right\}, kl \in \{\gamma\alpha, \gamma\theta, \alpha\gamma, \alpha\theta\}, \tag{6.1}$$

where A is a pre-exponential factor, depending on the frequency of collision of the reaction, $E_{\Gamma^{kl}}$ the interface energy, R_g the gas constant and $T(t)$ the temperature. See [26] for more details.

We choose the initial carbon concentration in austenite $c_{\gamma}^{\partial\Omega_\gamma}$ at the austenite-cementite equilibrium concentration such that it is the maximal value over the whole austenite domain. Realistically, austenite was in equilibrium with cementite before the ferrite nucleated and because the austenite equilibrium concentration $c_{\gamma\theta}^{\text{sol}}(T(t))$ decreases with the temperature, the concentration of carbon in austenite should start at a higher value than its interface concentration values as attained in the equilibrium state from the initial solution Equations 2.26. With the equilibrium values defined in [26] we pick the austenite carbon equilibrium at $T = 1010$ K.

6.1.1. From Composition-to Concentration Values

Most databases and research on phase transformations work with carbon component percentages. In this research we want to stick close the International System of Units (SI) to have a better apprehension of the models solution. Component percentages values can be converted to concentration values by the following formula:

$$c_k = N_k^{\text{atoms}} \frac{M_{\text{Fe}}}{a_k^3} \frac{\text{wt}_k}{1-\text{wt}_k}, k = \gamma, \alpha, \tag{6.2}$$

where N_k^{atoms} is the effective number of iron atoms present in a unit cell of steel in phase k, M_{Fe} is the molar mass of iron, a_k is the lattice length of a unit cell in phase k. Austenite its FCC crystalline structure gives

$N_\gamma^{\text{atoms}} = 4$ and ferrite's BCC crystalline structure gives $N_\alpha^{\text{atoms}} = 2$. Cementite does not have a cubic cell, thus it is difficult to find a formula for cementite as presented here. We choose

$$c_\theta = 7730.14 \frac{\text{wt}_\theta}{1 - \text{wt}_\theta} \text{kg m}^{-3},$$

and $\text{wt}_\theta = 6^2/_3\%$, where 7730.14 kg m^{-3} is the density of all cementite including iron.

6.2. Diffusion Coefficient

The diffusion coefficient follows the same Arrhenius relation as the equilibrium concentrations:

$$D_k(T(t)) = D_{0,k} \exp\left\{ \frac{-Q_k}{R_g T(t)} \right\}, \tag{6.3}$$

where $D_{0,k}$ is the diffusion coefficient at infinite temperature, Q_k the activation energy for diffusion, R_g the gas constant and $T(t)$ the temperature at time t. For $k = \gamma$ we have $D_{0,\gamma} = 1.5 \cdot 10^{-5}$ m^{2}s^{-1} and $Q_\gamma = {}^{142.1}/_{M_{\text{Fe}}}$ kJg^{-1}, for $k = \alpha$ we have $D_{0,\alpha} = 2.2 \cdot 10^{-4}$ m^{2}s^{-1} and $Q_\alpha = {}^{122.5}/_{M_{\text{Fe}}}$ kJg^{-1} [13]. The gas constant is an universal constant given by $R_g = {}^{8.31}/_{M_{\text{Fe}}}$ JK^{-1}g^{-1} [16].

6.3. Reaction Velocity

In the research of den Ouden [18], $K^{\gamma\theta}(\boldsymbol{x}, t)$ is chosen proportional to the Debye frequency of carbon multiplied by the distance an atom jumps. The Debye frequency is used in theoretical estimates of rates of diffusion and is considered to be proportional to the diffusivity coefficient $D_k(T(t))$ of carbon divided by the square of the lattice parameter of the phase (γ in his case). The jump distance is proportional with the lattice parameter a_k. This gives

$$K^{k\theta}(T(t)) = k_{0,k\theta} a_k \frac{D_k(T(t))}{a_k^2} = k_{0,k\theta} \frac{D_k(T(t))}{a_k}, k = \gamma, \alpha, \tag{6.4}$$

with $k_{0,k\theta}$ a proportionality constant, which increases the influence of the reaction term in the mixed-mode transformation character of the model. Similarly, the reaction velocity defined on the interface $\Gamma^{\gamma\alpha}(t)$ is the velocity of the carbon atoms in γ, so we get

$$K^{\gamma\alpha}(T(t)) = k_{0,\gamma\alpha} \frac{D_\gamma(T(t))}{a_\gamma}. \tag{6.5}$$

In Section 2.1 we noticed that because of the continuity constraint for the interface speed $v_n^\theta(\boldsymbol{x}, t)$ in the triple point and the Dirichlet boundary condition for the ferrite carbon concentration, a Dirichlet boundary condition is imposed for the austenite carbon concentration. It is not clear what the austenite carbon concentration should be in the triple point and we need to tweak some of the parameters to get more acceptable concentration values. We can tweak the proportionality constants of $k_{0,\gamma\theta}$ and $k_{0,\alpha\theta}$ in order to change the Dirichlet boundary as can be seen from Equation (2.25):

$$c_\gamma^\perp(\boldsymbol{x}, t) = c_{\gamma\theta}^{\text{sol}}(T(t)) - \frac{k_{0,\alpha\theta}}{k_{0,\gamma\theta}} \left(c_{\alpha\theta}^{\text{sol}}(T(t)) - c_{\alpha\gamma}^{\text{sol}}(T(t)) \right), \text{ for } \boldsymbol{x} \in \Gamma^{\gamma\alpha}(t) \cap \Gamma^{\alpha\theta}(t), t = t_0. \tag{6.6}$$

As we want the concentration value to change with temperature, we lock the fraction ${}^{k_{0,\alpha\theta}}/_{k_{0,\gamma\theta}}$ and let $c_\gamma(\boldsymbol{x}, t)$ change with the temperature, depending on the solubility concentrations $c_{\gamma\theta}^{\text{sol}}(T(t)), c_{\alpha\gamma}^{\text{sol}}(T(t))$ and $c_{\alpha\theta}^{\text{sol}}(T(t))$. The concentration value $c_\gamma^\perp(\boldsymbol{x}, t)$ in the triple points defined at $t = t_0$ at $T_0 = 995K$ we choose by looking at the concentration profile for carbon in austenite for several different fixed values on a fine mesh and choose the one that seems to fit 'best'. Where best means that we expect the maximum concentration value over the interfaces to be in the triple points to counter the high ferrite carbon concentration gradient in these points, but not a lot higher than the other concentration values. This will give the relation (6.7) between $k_{0,\alpha\theta}$ and $k_{0,\gamma\theta}$. This relation is a very unsubstantiated assumption, as there is no literature on the carbon concentration in the triple point where austenite, ferrite and cementite meet. This should be further investigated in further research of this model for more realistic results.

$$k_{0,\alpha\theta} = k_{0,\gamma\theta} \frac{c_{\gamma\theta}^{\text{sol}}(T_0) - c_\gamma^\perp}{c_{\alpha\theta}^{\text{sol}}(T_0) - c_{\alpha\gamma}^{\text{sol}}(T_0)} \frac{a_\alpha D_\gamma(T_0)}{a_\gamma D_\alpha(T_0)}. \tag{6.7}$$

6.4. Temperature Cooling Rates

At low carbon iron steels (below 0.6 carbon wt %), at the start of the carbon transport process, the interface is found to be controlling the transformation of austenite to ferrite. Over time the diffusion gradually takes over. The rate of transition of interface to diffusion control depends on the cooling rate ${}^{dT}/_{dt}(t)$ [13]. We will consider different cooling rates to see if the same holds for higher carbon steels (0.6 to 2.0 %). 0.05 Ks^{-1} is considered a low cooling rate, 0.4 Ks^{-1} medium and 10 Ks^{-1} high. We will assume that the temperature decreases linearly from T_0 to T_{end} with the medium and high cooling rate and stays constant at T_{end} for some time to let the model attain equilibrium. Remark that a high cooling rate is not comparable with quenching of steel, which is rapidly cooling steel by putting it in a relatively cold medium like water or oil. With quenching the rate of cooling is a lot higher which results in other reactions of the phases.

7

Results and Discussion

The results for the proposed model and test cases will be shown and discussed in this chapter. First we will look at the results of our model and the different settings (cooling rate, two ferrite particles and interface ghost velocity extension). Then other more specific parts of the solving method are tested. This includes the mesh accuracy and quality, the gradient approximation method, reinitialization and convergence results.

In this research the parameters in Table 7.1 will be used when not specifically specified otherwise. Furthermore we will not write down the dependencies of most quantities for sake of clarity.

7.1. Ferrite Nucleus Growth

With a constant temperature of T = 995 K ferrite is expected to grow into the austenite phase, although at a lower rate than for a lower temperature. Figure 7.1 shows the interface location over time and constant temperature T = 995 K. We can see that the ferrite nucleus grows very rapidly around the cementite particle. This can easily be clarified by the fact that the austenite carbon concentration in the triple points is much higher than on the rest of the interfaces (and austenite domain) and the carbon concentration gradient in ferrite on the interface is not big enough to nullify this increase. Although it probably should as we will see later in the results. Once the triple points meet and the austenite-cementite interface has disappeared, the reaction slows down and the interfaces move with a lower velocity, more resembling the results we found in the one-dimensional model [26]. We do not however get to see the layered like behavior as pearlite has, nor do we see a conservation of the wetting angle determined by the interface energies. If the ferrite and cementite were to behave as pearlite we would expect ferrite to be growing in the same order of speed as the cementite is growing into austenite. These interface energy effects have not in any way been included in the models' boundary conditions and thus the triple point movement is not constrained by any interface energies that are present on the interfaces. Also the austenite carbon concentration in the triple points together with the inaccurate ferrite carbon concentration gradient approximation result in a high triple point velocity. For a finer mesh with $h_{\max} = {}^{1}/_{16}\ \mu$m we will see in Section 7.3 that the gradient becomes a lot steeper in the triple points, slowing down the interface movement.
When the temperature is lowered at a constant rate up until t = 2.5 s we see the same behavior as with constant temperature, but at a higher rate. For the cooling rate 0.4 Ks^{-1}, with temperature T = 994 K at t = 2.5 s, there is only a slight increase in the ferrite growth (see also Figure 7.2), but in Figure 7.3 for cooling rate 10 Ks^{-1}, with temperature T = 970 K at t = 2.5 s, we can clearly see that the austenite-ferrite interface gets to grow further into the austenite phase in the same time compared to the other two cooling rates. With lower temperatures the austenite will be replaced faster by ferrite as the system is further away from the original austenite-cementite equilibrium. This is resembled by the change in equilibrium concentration values and the reaction coefficients which are directly related to the diffusion coefficients (see Equations (6.4) and (6.5)). The diffusion coefficients decrease when temperature is lowered, so the reaction coefficients also decrease, implying we have a more reaction-controlled system as was predicted for the reaction boundary conditions. There is unstable growth of the ferrite phase in the austenite phase, which shows after the triple points have

Quantity	Parameter	Value	Unit	Reference
Corner coordinates	x_0	0	μm	Chosen
	x_N	25	μm	Chosen
	y_0	0	μm	Chosen
	x_N	25	μm	Chosen
Max edge size background mesh	h_{max}	$^1/_4$	μm	Chosen
Start time	t_0	0	s	Chosen
Start temperature	T_0	995	K	Chosen
End temperature	T_{end}	800	K	Chosen
Cooling rates	$^{dT}/_{dt}$	$0, 0.4, 10$	Ks^{-1}	[13]
Proportionality constants	$k_{0,\gamma\alpha}, k_{0,\gamma\theta}$	1	1	Chosen
	$k_{0,\alpha\theta}$	Equation (6.7)	1	Section 6.3
Carbon concentration austenite triple points	$c_\gamma^\perp(T_0)$	62800	kgm^{-3}	Chosen at $T = T_0$
Diffusion coefficient austenite	$D_{0,\gamma}$	$1.5 \cdot 10^{-5}$	m^{2}s^{-1}	[13]
Diffusion coefficient ferrite	$D_{0,\alpha}$	$2.2 \cdot 10^{-4}$	m^{2}s^{-1}	[13]
Lattice length austenite	a_γ	$0.36 \cdot 10^{-3}$	μm	[17]
Lattice length ferrite	a_α	$0.29 \cdot 10^{-3}$	μm	[17]
Molecular mass iron	M_{Fe}	55.845	gmol^{-1}	[28]
Activation energy austenite	Q_γ	$^{142.1}/_{M_{\text{Fe}}}$	kJg^{-1}	[13]
Activation energy ferrite	Q_α	$^{122.5}/_{M_{\text{Fe}}}$	kJg^{-1}	[13]
Gas constant	R_g	$^{8.31}/_{M_{\text{Fe}}}$	JK^{-1}g^{-1}	[16]
Weight fraction cementite	wt$_\theta$	6.67	wt % C	Figure 6.1
Cementite carbon concentration	$c_{\theta,\text{Fe3C}}$	7730.14	kgm^{-3}	Chosen
Boundary carbon concentration austenite	$c_\gamma^{\partial\Omega_\gamma}$	$c_{\gamma\theta}^\infty(T = 1010\text{ K})$	gm^{-3}	Chosen
Ghost interface velocity extension constant	C	$^1/_{10}$	1	Chosen
Reinitialization constant	C_R	∞ (no reinit)	1	Chosen

Table 7.1: Physical parameters and initial settings.

disappeared and also later on for the cementite growth into ferrite. Mullins and Sekerka [15] showed that the growth of a particle by diffusion is an unstable process. While Mullins and Sekerka their model has Dirichlet boundary conditions for the carbon concentration, Den Ouden [18] showed that this is also the case for reaction boundary conditions with high reaction values K (which resembles a Dirichlet boundary condition) and believes that small perturbations of the concentration values caused by the topology of the interface amplify during the growth. For the higher cooling rates we have higher interface velocities, so we see the instabilities amplified more in the same time. A way to stabilize this numerical instability amplified by the physical process is to introduce the Gibbs-Thomson effect. More about this in Section 9.3 about future work, as we were not able to implement this effect mostly due the loss of Eikonal property of our level-set functions.
Because of the sharper angles in the austenite-ferrite interface caused by the unstable growth with higher cooling rates, the enriched mesh could contain multiple triangles where the level-set function $\phi^{\gamma\alpha}$ is zero if all three points of a triangle are allowed to shift to the same interface. In the simulation with cooling rate 10 Ks^{-1} (Figure 7.3), we did not allow this to happen. To see what effect this has, we also ran a simulation where this was allowed, but only if the created triangle has the same rotation as the original triangle and has an adequate skewness value. We set 0.65 or lower as adequate. We did not however see any (big) difference in the results. If in further research the mesh quality suffers from cutting a points that would normally shift because of this occurrence, this method should improve the quality.
Another thing to note is that the original ferrite particle is not expanding as much as the rest of the austenite-ferrite interface, also after the triple points disappear. So the other part catches up to the original ferrite particle and the ferrite starts resembling the same ellipse shape (except for the instabilities) as the cementite particle. This is because of the ferrite carbon concentration gradient. The concentration levels of the carbon in ferrite change from around the equilibrium value $c^{\text{sol}}_{\alpha\theta}$ to $c^{\text{sol}}_{\alpha\gamma}$ from the ferrite-cementite -to ferrite/austenite interface. The narrower the band between these interfaces, the steeper the gradient. This should only decrease the outward speed of the austenite-ferrite interface, in stead of increasing it, if you look at the equation for the austenite-ferrite interface velocity (Equation (2.22)). But the ferrite carbon concentration gradient is also present in the calculation of the austenite carbon concentration where it gives an increase of concentration with respect to the equilibrium concentration $c^{\text{sol}}_{\gamma\alpha}$, thus increasing the difference $c^{\text{sol}}_{\gamma\alpha} - c_\gamma$ in the austenite-ferrite interface speed calculation. We can see that the factor ${}^{K^{\gamma\alpha}}/_{c^{\text{sol}}_{\alpha\gamma}}$ is a lot bigger than ${}^{D_\alpha}/_{c^{\text{sol}}_{\alpha\gamma}}$, because $K^{\gamma\alpha} = {}^{D_\alpha}/_{a_\alpha}$ and $0 < a_\alpha \ll 1$. The increase in the austenite carbon concentration gap multiplied by this factor ${}^{1}/_{a_\alpha}$ is bigger than the ferrite carbon concentration gradient, thus giving a higher interface velocity in the outward direction. The final equilibrium state for cooling rate 0 Ks^{-1} can be seen in Figure 7.4. The final part of austenite that was located at the top right and left of the domain dissolved almost at the same time around $t = 430$ s. After this happened, the equilibrium state between ferrite and cementite was obtained after only a few iterations, but with large time steps, as the ferrite-cementite interface velocity was already quite low (maximal normal velocity was about 0.0016 μms^{-1}). This means the carbon concentration values of ferrite were already close to equilibrium, so diffusion was dominating the system. Just as in the one-dimensional case for zero cooling rate. Also the ferrite-cementite interface growth became more visibly unstable. Even with instability we can see there is still some symmetry on the y-axis at $x = 12.5$ μm, the x coordinate of the center of the initial ellipses. Although this can be expected as this is the symmetry axis of the initial domain, by numerical errors and a not symmetrical mesh, this symmetry is sometimes lost. This symmetry helps verifying the validity of the solution.

7.1.1. Two Nuclei

If we start with two nuclei placed symmetrically on the same cementite particle, we see in Figure 7.5 the exact same growth for both nuclei growing toward each other and meeting half way a bit over half the dissolve time of the one particle model with constant temperature. This tells us the triple point velocity along the cementite interface does not change much over time. Which in turn implies the normal gradient of the ferrite carbon concentration does not change much, as this is the only changing coefficient in the triple point velocities. Because the narrow band of ferrite around the cementite is not getting much thicker over time, this is not surprising. The rest of the model behaves similar to the one particle model. The advantage of this model is that the ferrite has less trouble growing around the cementite particle, because it does not have to grow past the major axis of the ellipse and it is possibly a more realistic setting towards a real pearlite growth model.

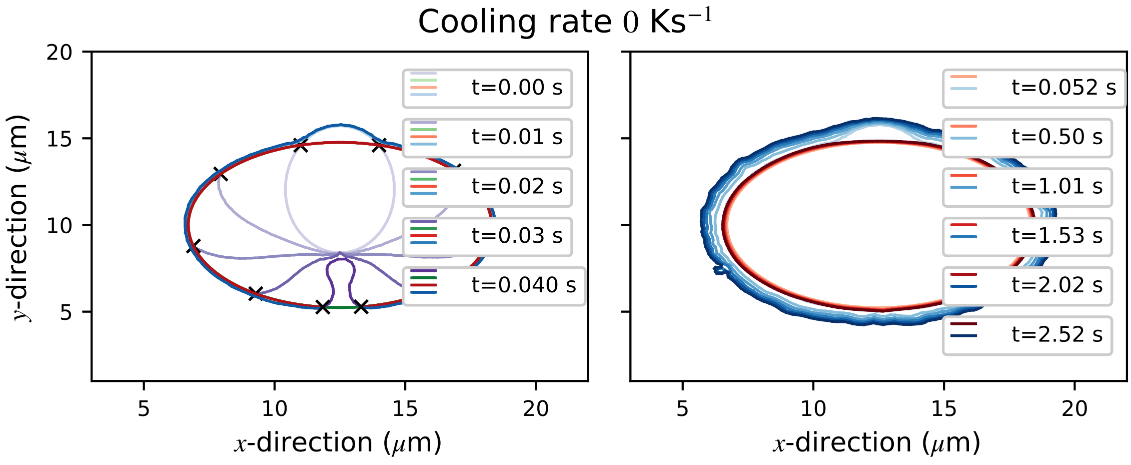


Figure 7.1: Interface locations over time with constant temperature $T = 995$ K. The left image show the interfaces while $\Gamma^{\gamma\alpha}(t)$ has not encapsulated the cementite particle yet. The right image shows the interfaces after encapsulation. Note that the figures are zoomed to $[3,22] \times [1,20]$.

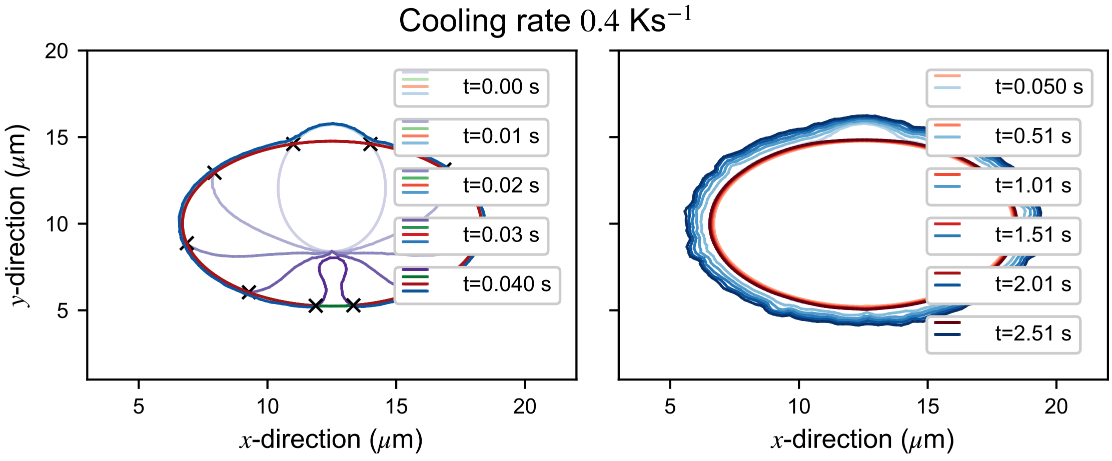


Figure 7.2: Interface locations over time with cooling rate 0.4 Ks^{-1}. The left image show the interfaces while $\Gamma^{\gamma\alpha}(t)$ has not encapsulated the cementite particle yet. The right image shows the interfaces after encapsulation. As the cooling rate is 0.4 Ks^{-1}, the temperature for $t = 2.53$ s is still about 994 K. Note that the figures are zoomed to $[3,22] \times [1,20]$.

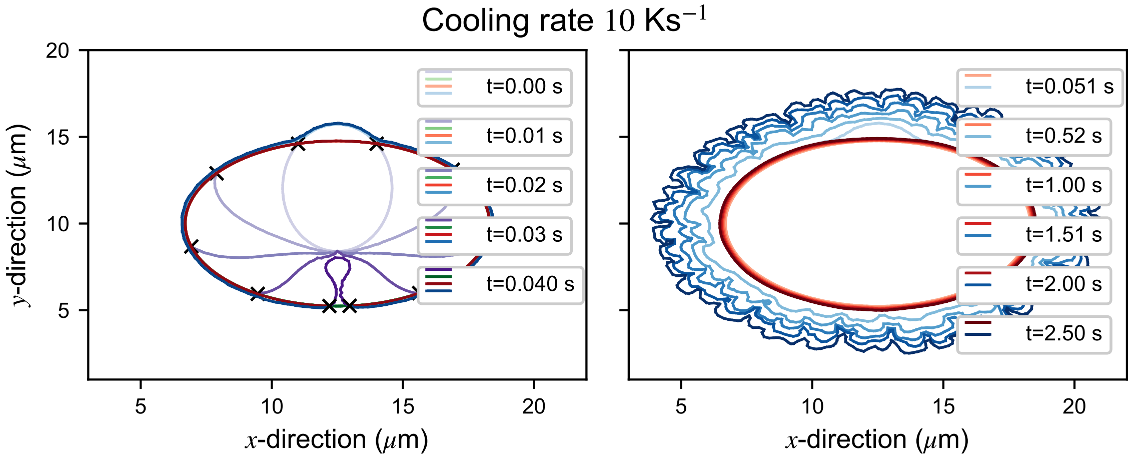


Figure 7.3: Interface locations over time with cooling rate 10 Ks^{-1}. The left image show the interfaces while $\Gamma^{\gamma\alpha}(t)$ has not encapsulated the cementite particle yet. The right image shows the interfaces after encapsulation. Note that the figures are zoomed to $[3,22] \times [1,20]$.

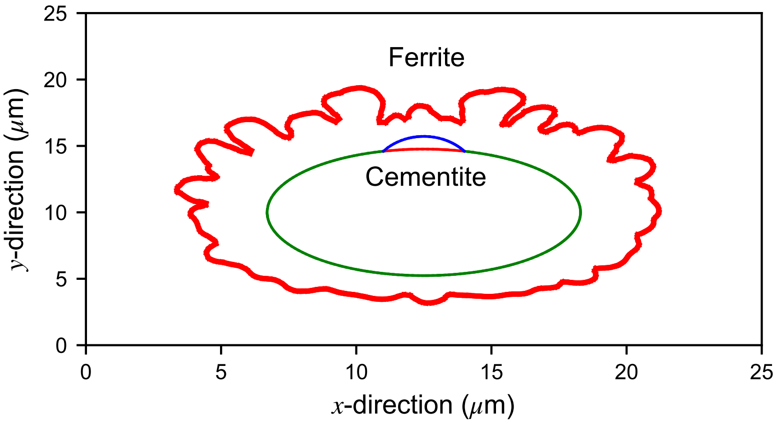


Figure 7.4: Final equilibrium state between ferrite and cementite for cooling rate 0 Ks^{-1} at $t = 430$ s and the initial ferrite and cementite particles with the austenite-ferrite interface in blue, the austenite-cementite interface in green and the ferrite-cementite interface in red (between the other two interfaces).

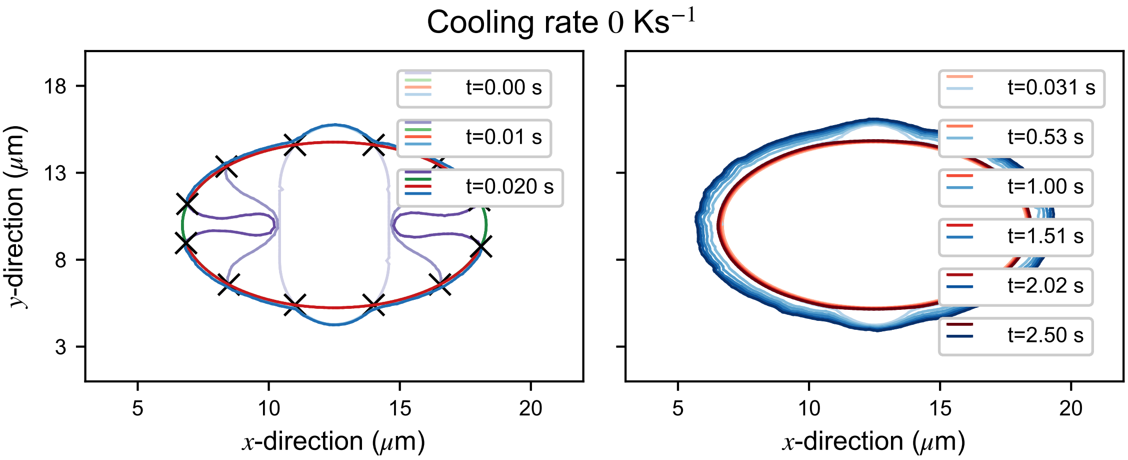


Figure 7.5: Interface locations over time for the two particle model with constant temperature $T = 995$ K. The left image show the interfaces while $\Gamma^{\gamma\alpha}(t)$ has not encapsulated the cementite particle yet. The right image shows the interfaces after encapsulation. Note that the figures are zoomed to $[3, 22] \times [1, 20]$.

7.2. Quality and Accuracy of the Mesh

In Section 5.1.4 we set two restrictions on the creation of the enriched mesh $T^E(t)$. First, the quality of the mesh and secondly the accuracy of the mesh. The quality of the mesh we will test by looking at the equiangular skewness as defined in Equation (5.9) and the variance of all angles in all triangles in the enriched mesh compared to the background mesh, which should have a high quality. The accuracy of the mesh will be observed by calculating the area each phase has on the enriched mesh $T^E(t_0)$ and compare it to the analytical area of the ellipses. The same will be done for the interface lengths.
The only parameter which we can change, next to the maximal edge length $h_{\max}$ of the background mesh T, is the parameter δ that says when to shift or when to cut an interface point based on the τ value of Equation (5.2). Based on the quality and the accuracy we will choose a δ value which gives both a good quality as accuracy. In this section we will first show results for the quality and the for the accuracy. From these results we conclude that a value somewhere in the middle of $(0, 0.5)$ results in a combination of good accuracy in approximating the domains and interfaces and quality of the enriched mesh. We will choose $\delta = 0.3$ for this research, just like in the work of Den Ouden [18].

7.2.1. Quality

Figure 7.6 and 7.7 show histograms of the frequency of all angles for the mesh with $h_{\max} = 1/4$. To be able to compare the different values for δ, the histograms are normalized. The red bars (which are more visible when zoomed, see Figure 7.7) show the enriched mesh and the black bars are the background mesh. As only a part of all triangles are changed, the frequency is not changed much. However if you look more closely to the histograms, you can see the tails becoming bigger and longer depending on the value of δ. The longer the tail, the more extreme the angles of the triangles are. The bigger the tail, the more these angles are present. As expected when $\delta = 0$ and all points are cut, a lot more of these ill-posed triangles are present. But also $\delta = 0.5$ shows a tail that contains low and high angles, but not as much. These histograms imply a value somewhere between $\delta = 0.3$ and 0.5 as the best option. In Figure 7.8 the mean variance of all angles relative to the mean variance of the background mesh is shown versus the different values of δ. It shows that $\delta = 0.45$ has the lowest mean variance (for this particular background and enriched mesh). This also shows that always shifting ($\delta = 0.5$) does not always result in better triangles.

7.2.2. Accuracy

If $\delta = 0$ we expect the interfaces and domains to be approximated by the accuracy of the level-set functions. In Figure 7.9 we show results for $\delta = 0, 0.3$ and 0.5 and different $h_{\max}$. We can clearly see a second order convergence of capturing the different interfaces and domains for $\delta = 0$, which we expected from using piecewise linear basis functions for the level-set functions. But also for $\delta = 0.3$ we have similar convergence. Even for $\delta = 0.5$, which has the worst approximation of the different interfaces and domains, the error decreases with second order accuracy.

In Figure 7.10 the error in area (top figures) and in length (bottom figures) are given for the different domains and interfaces respectively for the same background mesh with maximal edge size $h_{\max} = 1/4$. As expected a lower δ results in a lower error, although the difference seems to be small. Interesting result is the positive deviation in length of the θ-interfaces for $\delta = 0.3$ compared to the other values. Looking at the error in total length of the θ-domain boundary however, this deviation is not present as the relative error is dominated by the ferrite-cementite interface error. Different techniques (shift/bend/cut) were used for the triple points in these cases which could lead to a loss of the exact interface length.
Looking at the approximation of the interfaces for different δ values shows $\delta = 0$ gives the best results, but overall there is not much difference.

For $\delta = 0, 0.3$ and 0.5 we show the equiangular skewness as defined in Section 5.1.4 of all triangles in the neighborhood of the initial ferrite domain for the background mesh T and the initial mesh T_E^0 in Figures 7.11, 7.12 and 7.13. The skew value 0.48 is the maximal skewness found in the whole background mesh. Clearly the quality of the enriched mesh became worse by cutting and shifting of the interface points, but the maximal skewness value 0.78 of the enriched mesh is not bigger than 0.8. So for the initial domain and $\delta = 0.3$, the enriched mesh is a qualitative good mesh. Figures 7.12 and 7.13 show the skewness values when the enriched mesh is made with $\delta = 0$ and 0.5 respectively. Clearly always cutting results in very ill-shaped triangles as can be seen by the very high skewness value of 0.99. Also, as expected, when points are always

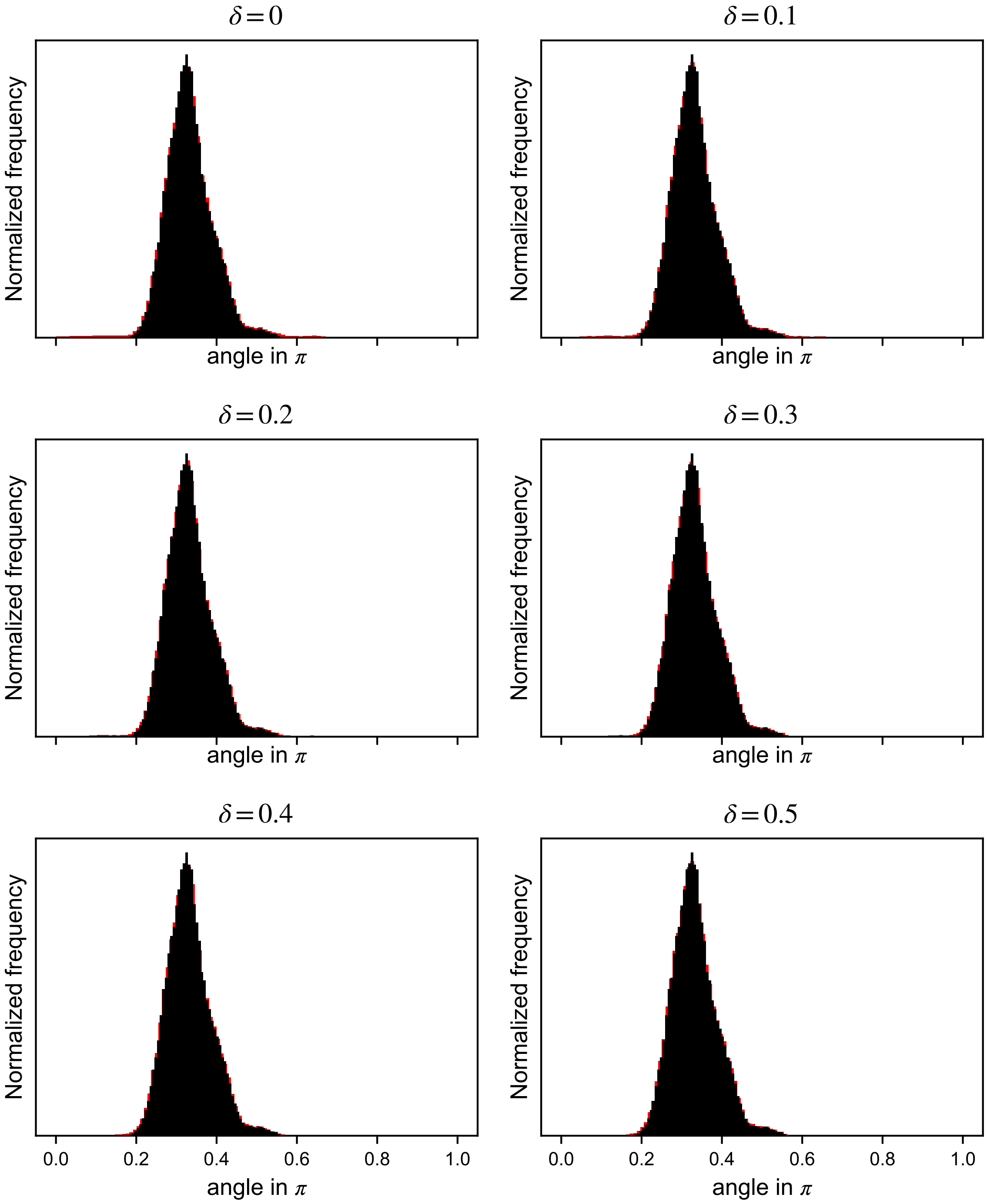


Figure 7.6: Normalized histogram showing the frequency of the angles present over all triangles in the enriched meshes (in red) for different values of δ and the background mesh (in black).

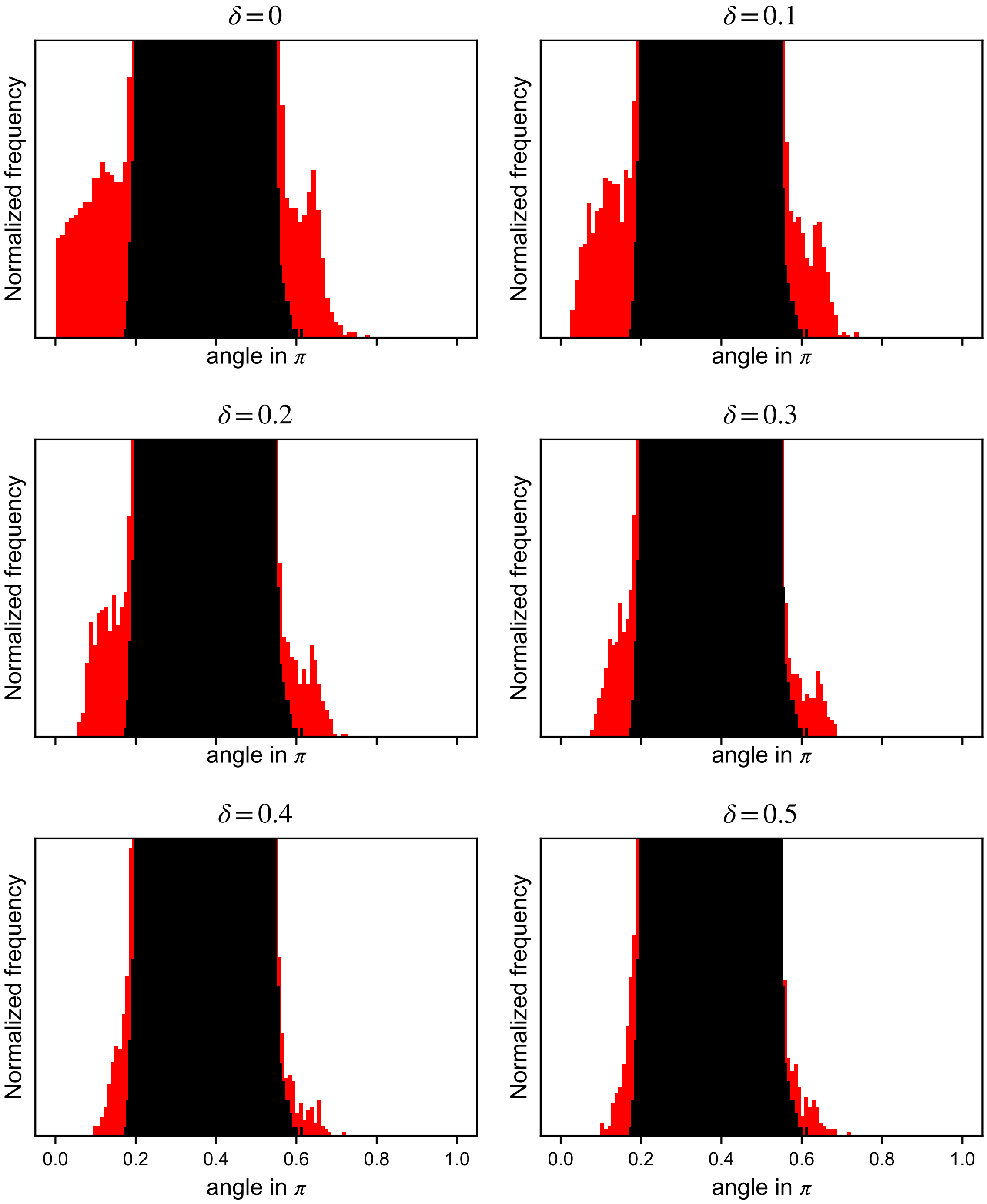


Figure 7.7: A zoom of the normalized histogram showing the frequency of the angles present over all triangles in the enriched meshes (in red) for different values of δ and the background mesh (in black).

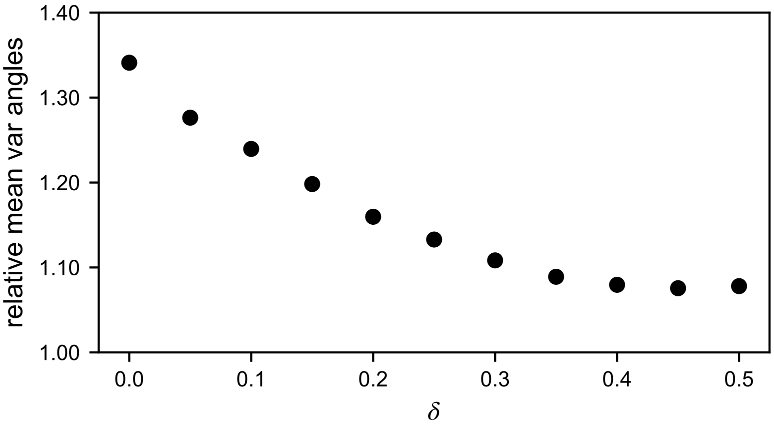


Figure 7.8: Mean variance of the enriched mesh relative to the background mesh mean variance for different values of δ.

shifted the skewness remains lower.

7.3. Concentration Gradient Approximation

In Section 5.3 we defined two test cases for the gradient recovery method. The first test case is to recover the gradient of the distance-to-arc function f on the arc of a half circle with center $(1,0)$ and radius one. In Table 7.2 the one, two and infinity norm of several $h_{\max}$ is shown. By Richardson extrapolation order approximations we see a first order convergence for the two and infinity norm (p^2 and p^∞ respectively) and close to first order for the one norm (p^1).

$h_{\max}$	N	$\lVert\lVert\nabla f\rVert_2-1\rVert_1$	$\lVert\lVert\nabla f\rVert_2-1\rVert_2$	$\lVert\lVert\nabla f\rVert_2-1\rVert_\infty$	p^1	p^2	p^∞
0.125	190	0.1046	$0.3496\cdot10^{-1}$	$0.2840\cdot10^{-1}$			
0.0625	699	$0.6238\cdot10^{-1}$	$0.1766\cdot10^{-1}$	$0.1047\cdot10^{-1}$			
0.03125	2850	$0.3437\cdot10^{-1}$	$0.6872\cdot10^{-2}$	$0.3097\cdot10^{-2}$	0.59	0.68	1.28
0.015625	11311	$0.1958\cdot10^{-1}$	$0.2690\cdot10^{-2}$	$0.7873\cdot10^{-3}$	0.92	1.37	1.67
0.0078125	44576	$0.9695\cdot10^{-2}$	$0.9492\cdot10^{-3}$	$0.2156\cdot10^{-3}$	0.58	1.27	2.02
0.00390625	177909	$0.4845\cdot10^{-2}$	$0.3557\cdot10^{-3}$	$0.9282\cdot10^{-4}$	1.03	1.55	2.22

Table 7.2: One, two and infinity norm errors for the length of the gradient of distance function for different mesh sizes (max edge size $h_{\max}$ and N grid nodes on a half circle.

For the second test case we test the gradient recovery of the level-set function $\phi^{\gamma\alpha}$ on $\Gamma^{\gamma\alpha}$ of the initial ferrite domain. We get the one, two and infinity norm errors as shown in Table 7.3. We get the same behavior in two and infinity norm, but seem to have insufficient data to say anything about convergence in the one norm.

$h_{\max}$	N_α	$\lVert\lVert\nabla\phi^{\gamma\alpha}\rVert_2-1\rVert_1$	$\lVert\lVert\nabla\phi^{\gamma\alpha}\rVert_2-1\rVert_2$	$\lVert\lVert\nabla\phi^{\gamma\alpha}\rVert_2-1\rVert_\infty$	p^1	p^2	p^∞
0.25	31	0.3768	0.1367	$0.8480\cdot10^{-1}$			
0.125	96	0.2879	$0.6401\cdot10^{-1}$	$0.3667\cdot10^{-1}$			
0.0625	292	0.1462	$0.3129\cdot10^{-1}$	$0.1512\cdot10^{-1}$	−0.67	1.15	1.16
0.03125	1033	$0.9383\cdot10^{-1}$	$0.1344\cdot10^{-1}$	$0.4843\cdot10^{-2}$	1.44	0.87	1.07

Table 7.3: One, two and infinity norm errors for the length of the gradient of distance function for different mesh sizes (max edge size $h_{\max}$) and N_α grid nodes for the initial ferrite domain.

If we however look at the ferrite concentration gradients close to the left and the right triple points in Fig-

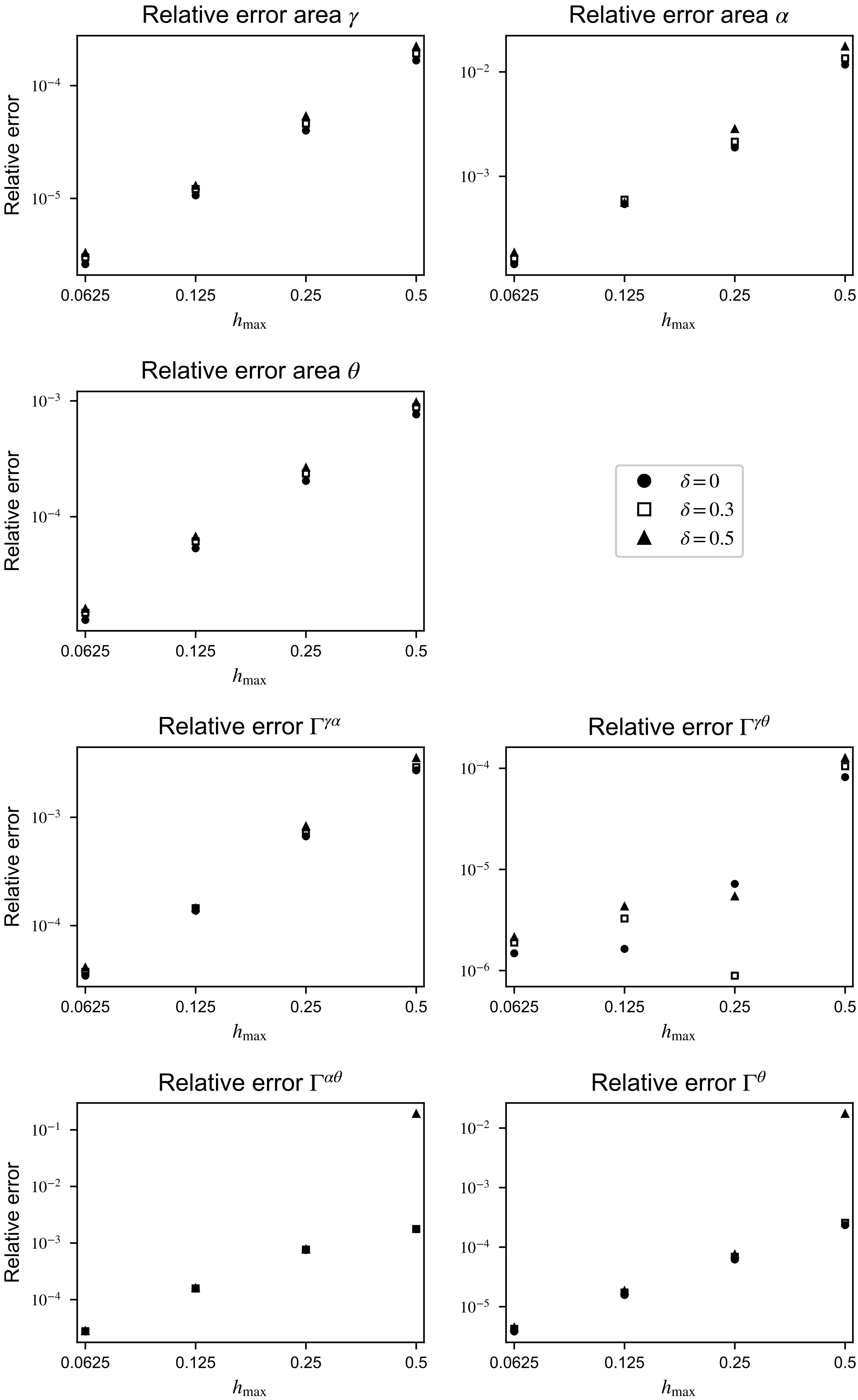


Figure 7.9: Relative error shown for $\delta = 0, 0.3$ and 0.5 and different $h_{\max}$. Second order of convergence is attained for both convergence of the areas of the different domains as the length of the different interfaces for all δ. Although $\delta = 0$ has a slightly lower error.

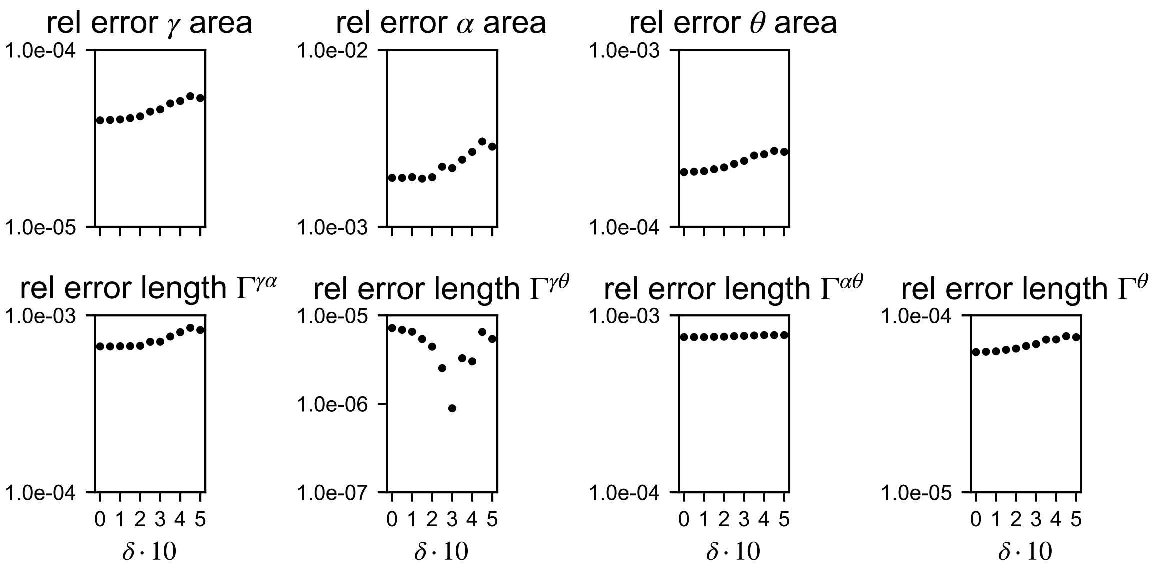


Figure 7.10: Relative error in area of the different domains and in length of the interfaces shown for $\delta = 0, 0.05, 0.1, \ldots, 0.5$ and fixed $h_{\max} = 1/4$.

ure 7.14 (upper and lower right figures), we see that the gradients differ quite a lot depending on $h_{\max}$. Because the concentration is constant on the austenite-ferrite interface, the concentration levels on the ferrite-cementite interface will determine how steep the concentration function is. Because the resolution is relatively low for the lower $h_{\max}$, it does not capture the rate of change of the ferrite carbon concentration as good as the finer mesh with $h_{\max} = 1/32$ μm. The concentration on the ferrite-cementite interface behaves as a Dirichlet boundary condition, with concentration level close to $c^{\text{sol}}_{\alpha\theta}$, but is able to differ from this value. This makes the transition from $c^{\text{sol}}_{\alpha\theta}$ on the ferrite-cementite interface to $c^{\text{sol}}_{\alpha\gamma}$ on the austenite-ferrite interface very spontaneously, almost behaving as a pulse function near the triple points. Calculating the gradient for such a function is very difficult, thus the gradient recovery for $h_{\max} = 1/4$ μm, with which we got the ferrite particle growth results, is not accurate enough. Because of computation time we were not able to simulate with finer meshes. If we were able to however, we would see that the austenite-ferrite interface speed would not be as high as we have seen, with this specific choice of austenite carbon concentration in the triple points. The resulting austenite-ferrite interface velocities are shown in Figure 7.15. Because the gradient approximation in the triple points is much steeper for the finer meshes, the speed becomes lower and lower, even changing direction (sign) in the right triple point for the finer meshes $h_{\max} = 1/32$ μm. As a negative austenite-ferrite velocity is expected, a higher austenite carbon concentration should be assigned in the triple points than set right now, even though the triple points are moving too fast for the more coarse meshes for this concentration value. It is difficult to say how accurate the current approximation is and how much further the gradient will converge, but the accuracy of the gradient recovery should be the determining factor in the choice of $h_{\max}$.

Even if the gradient approximation is good for the mesh at $t = 0$ s, we have seen in the results that we get a very narrow band of ferrite around the cementite particle. For $h_{\max} = 1/4$ μm this band is about one $h_{\max}$ in width. To get the gradient approximation in an austenite-ferrite interface point, the nearest internal ferrite grid point can sometimes be located somewhere in the initial ferrite domain. This will never give a good approximation of the gradient. If narrow bands still occur for finer meshes this should be further investigated.

7.4. Reinitialization

The level-set function loses its distance function property over time because the piecewise linear approximation can not suffice to the Eikonal property of having length gradient one and because the convection of the level-set functions does not preserve this property.

The reinitialization method does not seem to diverge immediately at the first instance of reinitialization, but

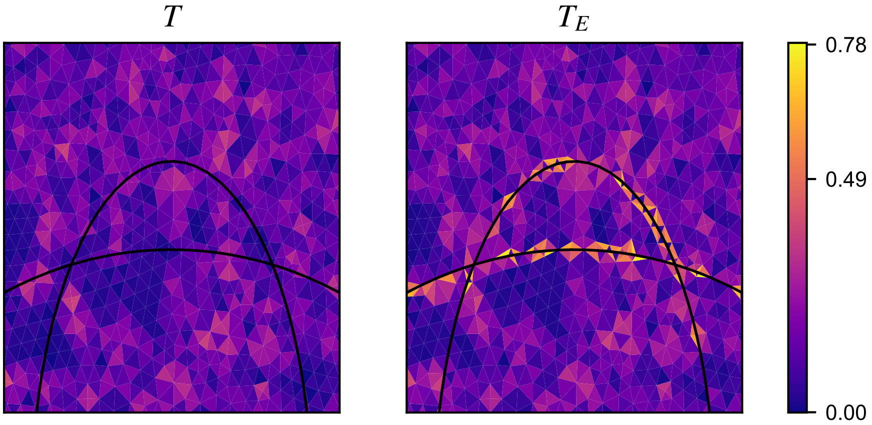


Figure 7.11: Equiangular skewness per triangle between 0 and 1 for the background mesh T (on the left) and T_E (on the right) with $\delta = 0.3$. The maximal skewness for T is 0.49, for T_E it is 0.78. 0 skewness is for a 'perfect' triangle with angles $[\pi/3, \pi/3 \pi/3]$ and skewness 1 represents a 'triangle' with an angle of 0 or π.

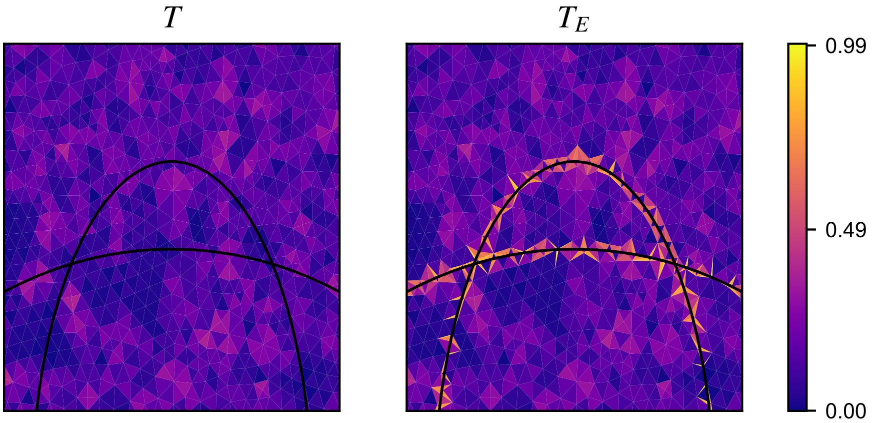


Figure 7.12: Equiangular skewness per triangle between 0 and 1 for the background mesh T (on the left) and T_E (on the right) with $\delta = 0$ when interface points are added as new mesh points. The maximal skewness for T is 0.49, for T_E it is 0.99. That means there are triangles with a skewness value very close to the worst possible 'triangle'.

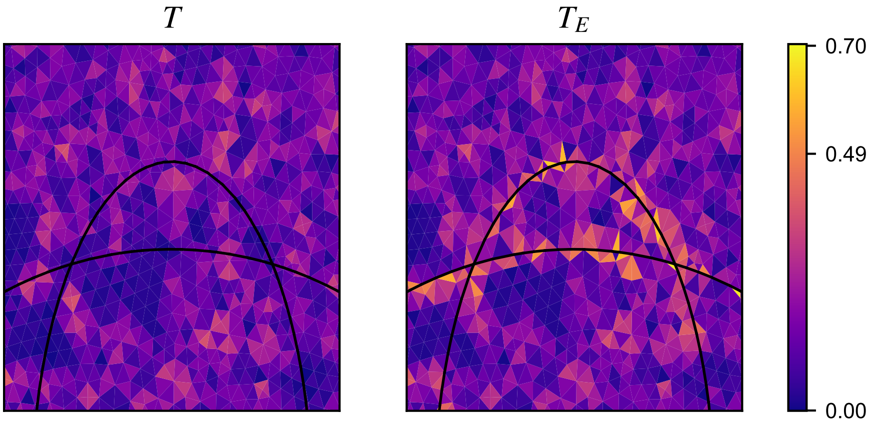


Figure 7.13: Equiangular skewness per triangle between 0 and 1 for the background mesh T (on the left) and T_E (on the right) with $\delta = 0.5$ when interface points are always shifted background mesh points. The maximal skewness for T is 0.49, for T_E it is 0.7. Meaning a better quality of the mesh than $\delta = 0$ and 0.3.

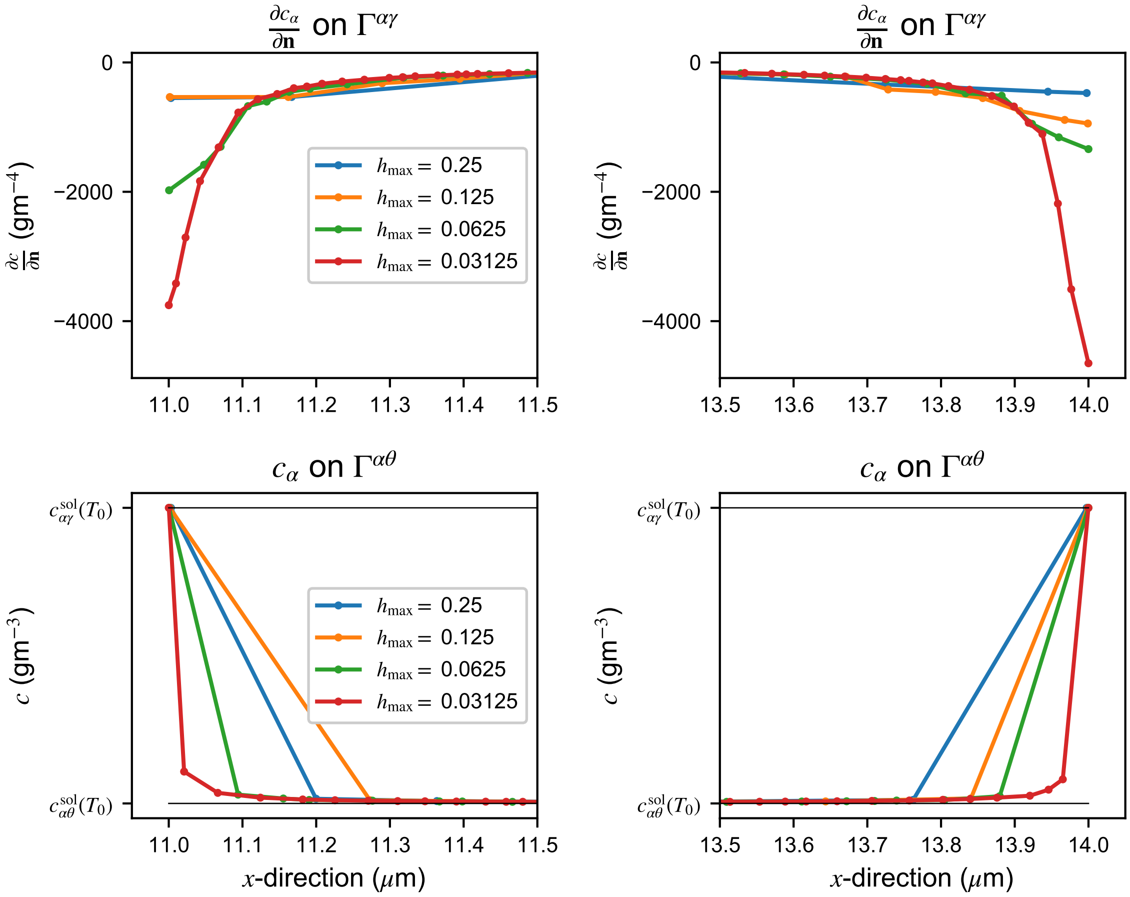


Figure 7.14: Austenite carbon concentration gradient approximation for different $h_{\max}$ close to the left (upper figures) and right (lower figures) triple points seen from the x-axis.

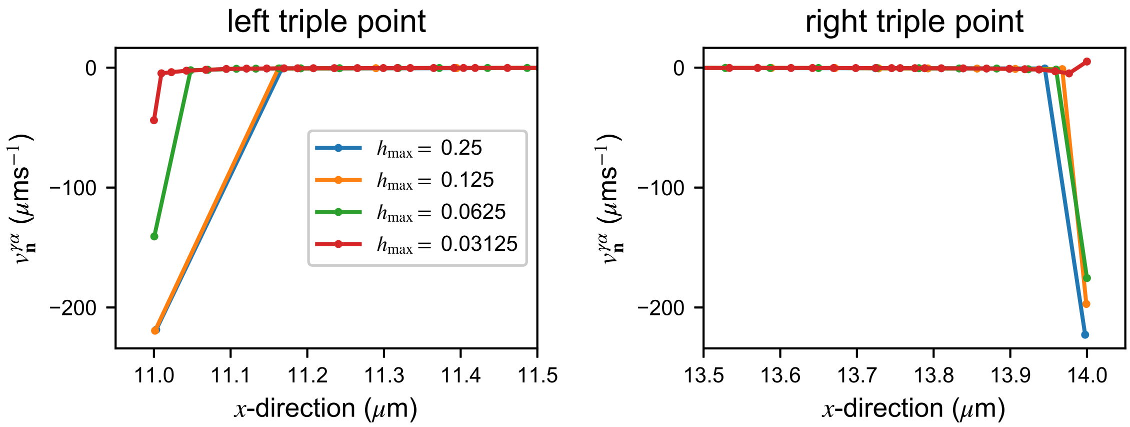


Figure 7.15: austenite-ferrite interface velocity $v^{\gamma\alpha}_{\boldsymbol{n}}$ for different $h_{\max}$ close to the left and right triple point seen from the x-axis.

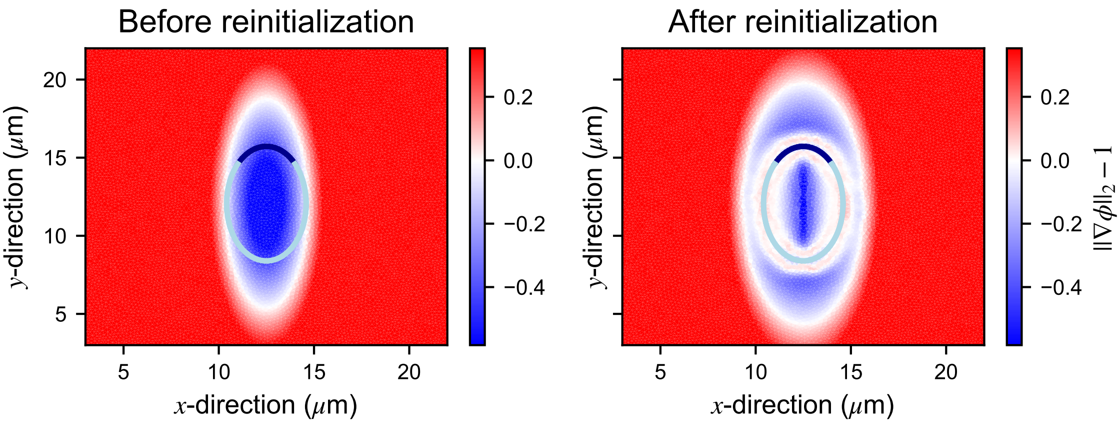


Figure 7.16: Reinitialization of $\boldsymbol{f}$ with Lagrange multiplier for the Dirichlet boundary condition. The dark and light blue lines show the zero contour of f coinciding with the austenite-ferrite and ghost interface respectively.

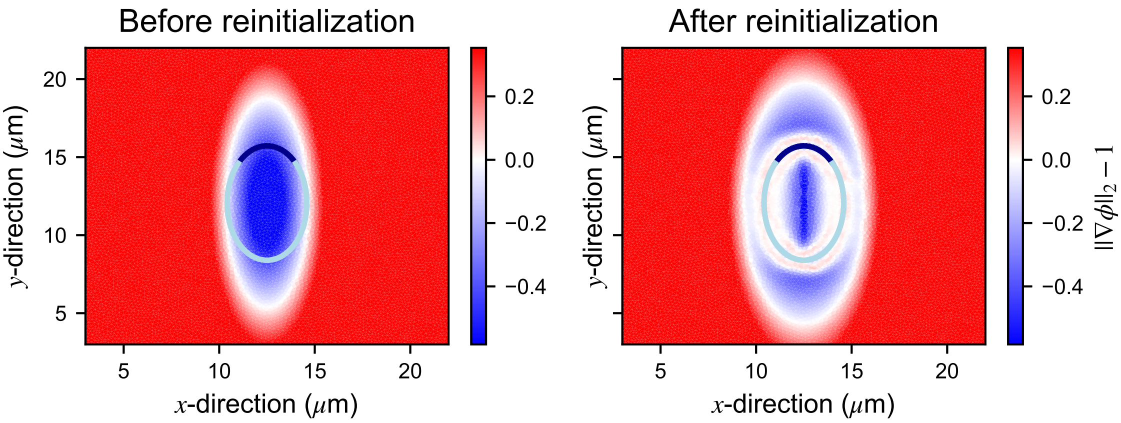


Figure 7.17: Reinitialization of $\boldsymbol{f}$ with least square solving to also hold for the Dirichlet boundary condition. The dark and light blue lines show the zero contour of f coinciding with the austenite-ferrite and ghost interface respectively.

after a few times of reinitializing. First we look at a test case where we define the function

$$f(x,y) = \left(\frac{x - x_C^{\gamma\alpha}}{b_{\gamma\alpha}}\right)^2 + \left(\frac{y - y_C^{\gamma\alpha}}{a_{\gamma\alpha}}\right)^2 - 1. \tag{7.1}$$

This functions has the same zero-contour as the level-set function $\phi^{\gamma\alpha}$, but does not have the Eikonal property $\|\phi^{\gamma\alpha}\|_2 = 1$. The reinitialization algorithm will change the piecewise linear approximation $\boldsymbol{f}$ of f to the signed-distance function $\phi^{\gamma\alpha}$ away from the zero-contour. In Figures 7.16 and 7.17 the reinitialization of f is shown with the Dirichlet condition by Lagrange multiplier and with least square solving respectively. For the first method we get the error $\epsilon_0^{\boldsymbol{f}} = 1.348$ over the band of zero (points itself) and first neighbors of all points laying on edges where f changes sign (which are the same for $\phi^{\gamma\alpha}$), or N_1^T as defined in Section 5.3.1. For the less restrictive least square solving method we get a lower error $\epsilon_0^{\boldsymbol{f}} = 1.053$. There is however movement of the zero-contour with the least square solving method because of the relaxation on the Dirichlet condition. The maximal shift of the interface points can by found by calculating the maximum of $\|F\boldsymbol{f}\|_\infty$ after reinitialization over all edges where $\boldsymbol{f}$ changes sign. We find the distance $\|F\boldsymbol{f}\|_\infty = 0.211 \cdot 10^{-3}$ μm. Which is not that big of a shift compared to the mesh size.

We now show the reinitialization method as given in Sections 4.2 and 5.3.1 for a simulation of the one ferrite particle model with constant temperature $T = 995$ K and a threshold of $C_R = 25$ times $\epsilon_0^k, k = \gamma\alpha, \theta$. In this

$h_{\max}$ (μm)	$m_h(t_0)$ (kg)	p_0	$m_h(t)$ (kg)	p
1	$6.4068 \cdot 10^{-5}$		$6.4017 \cdot 10^{-5}$	
0.5	$6.4065 \cdot 10^{-5}$		$6.4032 \cdot 10^{-5}$	
0.25	$6.4022 \cdot 10^{-5}$	-3.81	$6.4010 \cdot 10^{-5}$	NaN
0.125	$6.3993 \cdot 10^{-5}$	0.52	$6.3991 \cdot 10^{-5}$	0.28
0.0625	$6.3975 \cdot 10^{-5}$	0.73	N/A	N/A

Table 7.4: Mass per mesh size $h_{\max}$ at $t_0 = 0$ s and $t = 0.008$ s.

case

$$\epsilon_0^{\gamma\alpha} = 0.02328, \tag{7.2}$$

$$\epsilon_0^{\theta} = 0.004794. \tag{7.3}$$

The first time reinitialization is applied is after eight iterations. In Figure 7.18 the reinitialization with Lagrange multiplier is shown and in Figure 7.19 the least squares approach for this reinitialization step is shown. Both reinitialization processes seem to increase the signed-distance property of the level-set functions $\phi^{\gamma\alpha}$ on a narrow band (we reinitialize until $\tau = 2.5h^E_{\min}$, where $h^E_{\min}$ is the minimal edge length in the enriched mesh T^E) around the interface. However we see in Figure 7.20 that $\|\nabla\phi^{\gamma\alpha}\|_2$ becomes unstable and the simulation eventually stops because the level-set functions gets odd zero-contours and there even arises a new ghost interface. Because the austenite-ferrite interface moves more during the simulation than the cementite interfaces, $\phi^{\gamma\alpha}$ is reinitialized much earlier and more frequent than ϕ^{θ} (it even happened to be that ϕ^{θ} was not even reinitialized ones during this simulation).
Also for the least squares approach we see the same unstable reinitialization not many iterations later than with the Lagrange multiplier systems. See Figure 7.21 of an unstable reinitialization with the least squares approach.

7.5. Convergence

To show convergence of the algorithm, we look at the error of the total carbon mass of the background meshes with maximal edge lengths $h_{\max} = 1, {}^1/_2, {}^1/_4$ and ${}^1/_8$ μm at $t = 0$ and 0.008 s. We also show the initial mass for $h_{\max} = {}^1/_{16}$ μm at $t = 0$ s. Doing time iterations with this finer mesh takes too much computation time, that is why no additional mass data is available. In Table 7.4 these errors are given and in Figure 7.22 the mass over time is shown for all four $h_{\max} = 1, {}^1/_2, {}^1/_4$ and ${}^1/_8$ μm. For the latter three mesh sizes we see that by Richardson extrapolation order estimation the error barely converges, but for the coarser meshes no convergence is observable as $h_{\max} = {}^1/_2$ μm has more mass at $t = 0.08$ s than $h_{\max} = 1$ μm and $h_{\max} = {}^1/_4$ μm. As can also be seen in the figure, where the mass over time for both $h_{\max} = 1$ and ${}^1/_2$ μm almost have the same mass over time and the latter is higher at $t = 0.008$ s. Even the initial mass $m_h(t_0)$ has trouble with converging, but shows order one convergence for the finer meshes including $h_{\max} = {}^1/_{16}$ μm.
However, if we look at the carbon mass of the different phases at $t = 0$ s and $t = 0.008$ s, we do get convergence. See Tables 7.5, 7.6 and 7.7 for the austenite, ferrite and cementite carbon mass respectively. We see that both the carbon mass in austenite and ferrite has first order convergence. Such a difference in convergence of the total mass and the mass per phase individually must mean that there is some crossing of the errors between the phases. In the one-dimensional model we did also see a dampening effect in the mass gain because of the growth/dissolution of the different phases, which canceled once the austenite dissolved. This dampening is also present in this model, which we will show below. The carbon mass in cementite even converges with second order, because the cementite carbon concentration is constant the error in mass is directly linked to the error in the approximation of the cementite particle, which is second order as we have shown before.

In Figure 7.23 the total mass relative to the initial mass is shown for the simulation with cooling rate Ks^{-1} 0 and $h_{\max} = {}^1/_4$ μm and ${}^1/_2$ μm. We see that overall the total mass slowly increases over time up to about 1.8 and 1.85 times the total initial mass at the end of the simulation for $h_{\max} = {}^1/_4$ and ${}^1/_2$ μm respectively. The mass gain decreases over time as the interface velocities decrease over time. So just like in the one-dimensional model studied in [26] the mass gain is mostly due to the movement of the interfaces. In Figure 7.24 you see the

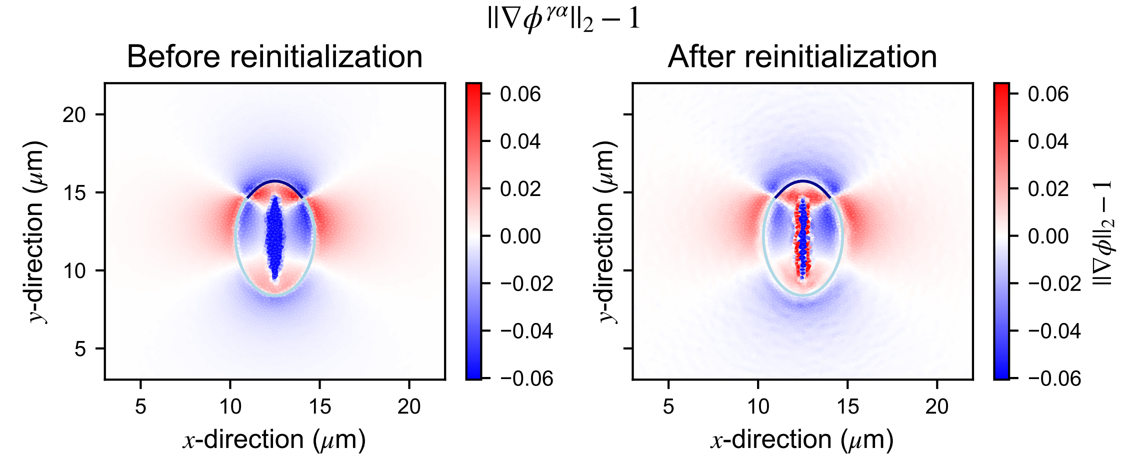


Figure 7.18: Reinitialization after eight iterations with the Lagrange multiplier approach, the first time reinitialization is done. The dark and light blue lines show the austenite-ferrite and ghost interface respectively. The length of the gradient of $\phi^{\gamma\alpha}$ is shown before (left figure) and after (right figure) reinitialization minus one to center the values around zero.

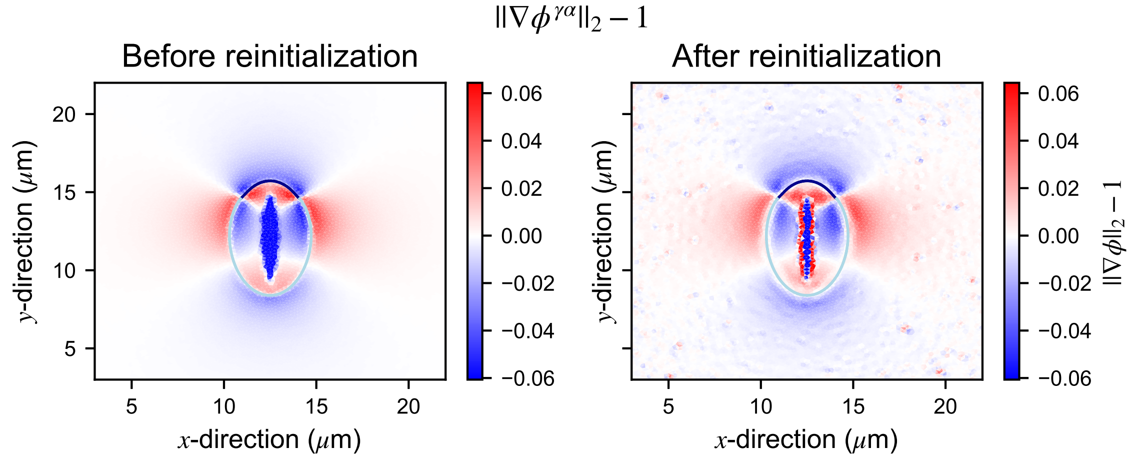


Figure 7.19: Reinitialization after eight iterations with the least squares approach, the first time reinitialization is done. The dark and light blue lines show the austenite-ferrite and ghost interface respectively. The length of the gradient of $\phi^{\gamma\alpha}$ is shown before (left figure) and after (right figure) reinitialization minus one to center the values around zero.

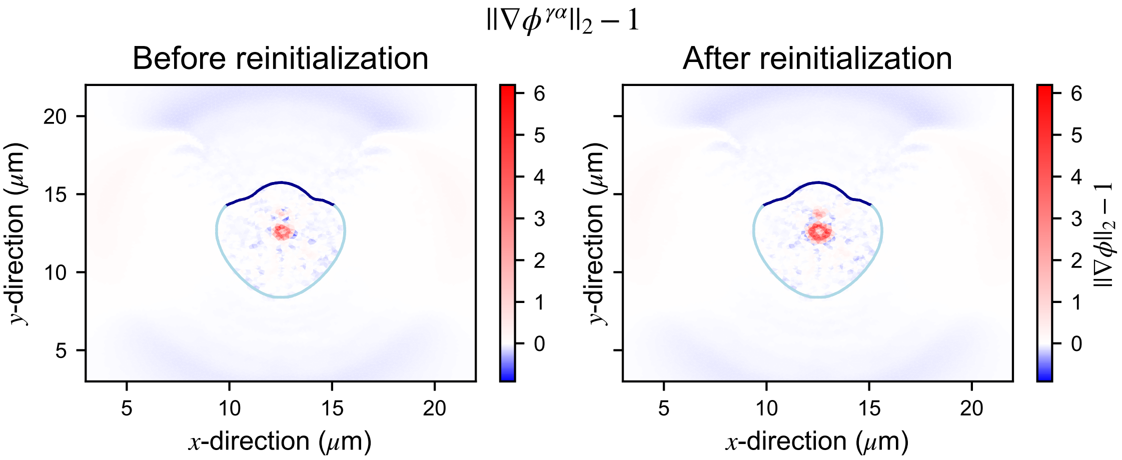


Figure 7.20: Reinitialization after 60 iterations when $\epsilon_{t_{60}}^{\gamma\alpha}$ increases drastically during the reinitialization process. The dark and light blue lines show the austenite-ferrite and ghost interface respectively. The length of the gradient of $\phi^{\gamma\alpha}$ is shown before (left figure) and after (right figure) reinitialization minus one to center the values around zero. Mostly the color bar is showing the bad gradient length levels.

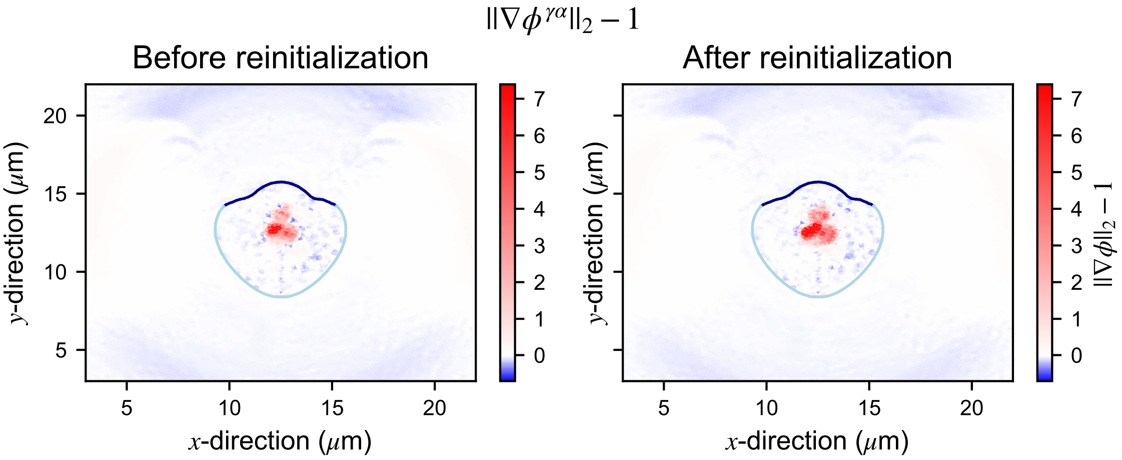


Figure 7.21: Reinitialization after 63 iterations when $\epsilon_{t_{60}}^{\gamma\alpha}$ increases drastically during the reinitialization process. The dark and light blue lines show the austenite-ferrite and ghost interface respectively. The length of the gradient of $\phi^{\gamma\alpha}$ is shown before (left figure) and after (right figure) reinitialization minus one to center the values around zero. Mostly the color bar is showing the bad gradient length levels.

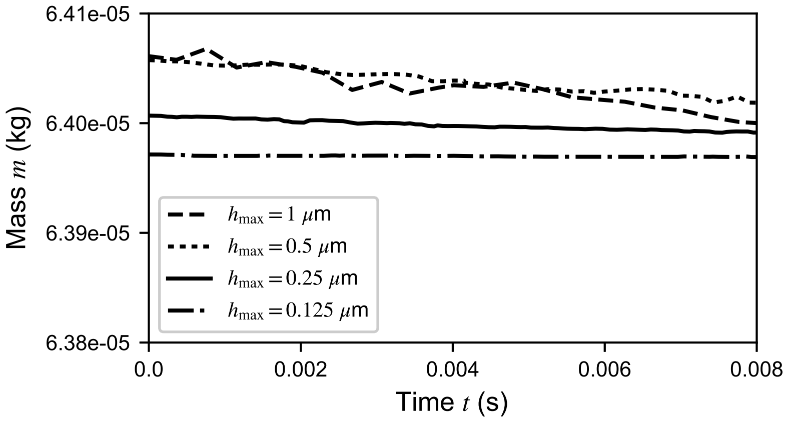


Figure 7.22: Carbon mass over time for different values of $h_{\max}$ = 1/4 μm until t = 0.008 s.

$h_{\max}$ (μm)	$m_h^\gamma(t_0)$ (kg)	p_0^γ	$m_h^\gamma(t)$ (kg)	p^γ
1	$1.6362 \cdot 10^{-5}$		$1.6278 \cdot 10^{-5}$	
0.5	$1.6228 \cdot 10^{-5}$		$1.6174 \cdot 10^{-5}$	
0.25	$1.6156 \cdot 10^{-5}$	0.88	$1.6129 \cdot 10^{-5}$	1.20
0.125	$1.6118 \cdot 10^{-5}$	0.95	$1.6107 \cdot 10^{-5}$	1.05
0.0625	$1.6098 \cdot 10^{-5}$	0.92	N/A	N/A

Table 7.5: Mass for austenite per mesh size $h_{\max}$ at $t_0 = 0$ s and $t = 0.008$ s.

$h_{\max}$ (μm)	$m_h^\alpha(t_0)$ (kg)	p_0^α	$m_h^\alpha(t)$ (kg)	p^α
1	$3.1424 \cdot 10^{-9}$		$6.5611 \cdot 10^{-9}$	
0.5	$2.2087 \cdot 10^{-9}$		$4.1619 \cdot 10^{-9}$	
0.25	$1.7864 \cdot 10^{-9}$	1.14	$2.9121 \cdot 10^{-9}$	0.94
0.125	$1.6156 \cdot 10^{-9}$	1.31	$2.0489 \cdot 10^{-9}$	0.53
0.0625	$1.5228 \cdot 10^{-9}$	0.88	N/A	N/A

Table 7.6: Mass for ferrite per mesh size $h_{\max}$ at $t_0 = 0$ s and $t = 0.008$ s.

$h_{\max}$ (μm)	$m_h^\theta(t_0)$ (kg)	p_0^θ	$m_h^\theta(t)$ (kg)	p^θ
1	$4.7703 \cdot 10^{-5}$		$4.7732 \cdot 10^{-5}$	
0.5	$4.7834 \cdot 10^{-5}$		$4.4785 \cdot 10^{-5}$	
0.25	$4.7865 \cdot 10^{-5}$	2.09	$4.7878 \cdot 10^{-5}$	2.34
0.125	$4.7873 \cdot 10^{-5}$	1.89	$4.7882 \cdot 10^{-5}$	2.51
0.0625	$4.7875 \cdot 10^{-5}$	1.91	N/A	N/A

Table 7.7: Mass for cementite per mesh size $h_{\max}$ at $t_0 = 0$ s and $t = 0.008$ s.

same figure, but zoomed in on the start of the simulation, where the triple points were still present. The mass gain seems to be less consistent and is negative at the start and becomes positive halfway the time when the triple points disappear. There is some dampening on the mass gain because of the movement of the different interfaces, which cancels once the triple points disappear. This dampening is probably one of the reasons that the total mass does not consistently decreases when using finer meshes, but also the resolution problem in the ferrite carbon concentration near the triple points, which gives problems in the gradient approximation, must have negative influences on the convergence. With finer meshes we expect this effect to be less and thus the (first order) convergence to be more visible, also for the total mass.

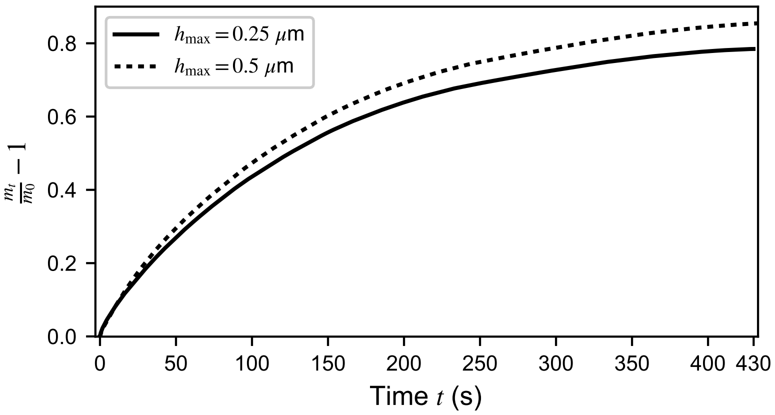


Figure 7.23: Carbon mass over time relative to initial mass for $h_{\max}$ = 1/4 μm (full line) and 1/2 μm (striped line).

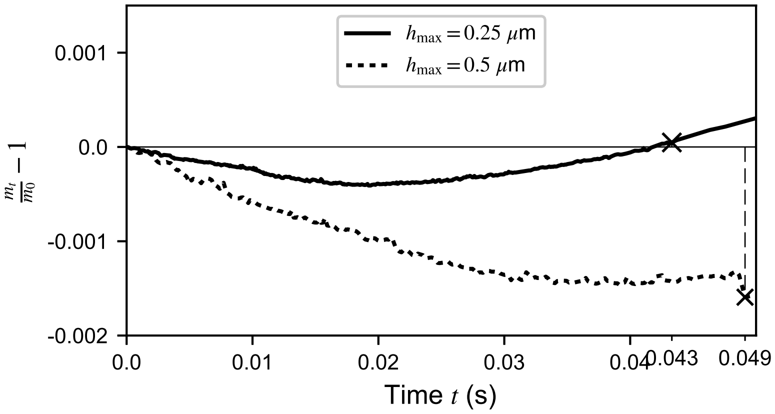


Figure 7.24: Carbon mass relative to initial mass in the time that the triple points are still present (dashed line shows time of disappearing). $h_{\max}$ = 1/4 μm (full line) and 1/2 μm (striped line)

8 Conclusions

To come up with an approximation for the Stefan problem describing the carbon concentration levels in three different phases of steel, we approximated the diffusion equation with mixed mode character interface conditions with Galerkins Finite Element method and the Algebraic Flux Limiter to avoid violation of the discrete maximum principle. To capture the domains and interfaces we used the multi level-set method.

Firstly we derived the different interface boundary conditions from the requirement of mass conservation, called the Stefan condition, in combination with a first order reaction in flux. We also implied a para-equilibrium condition on the austenite-ferrite interface, where the ferrite carbon concentration is assumed in equilibrium on the interface, to make the model sufficient. From this combination of boundary conditions and the assumption that the interface velocities have to be continuous, we indirectly obtained a Dirichlet boundary condition in the triple points where the three phases meet. This condition was however not expected and difficult to deal with as no physical knowledge could substantiate the Dirichlet value of the austenite carbon concentration. We choose a value that seemed to fit the expected concentration levels and interface velocities based on a fine mesh by setting a certain ratio between the proportionality factors in the reaction constants as a combination of equilibrium and diffusion values. There was however another quantity introduced in the model compared to the two phased model with reaction boundary conditions, namely the ferrite carbon concentration gradient. This gradient remained in the austenite-ferrite interface integral of the weak formulation for the austenite carbon concentration approximation and in the austenite-ferrite interface velocity. The numerical approximation of this term is difficult and in combination with the Dirichlet boundary condition in the triple points, it caused unrealistic results for the austenite-ferrite interface velocity.

To obtain an approximation to the solution of the model we defined a background mesh on the domain of interest constructed by a Delaunay triangulation. In order to capture the locations of the domains, we extend this mesh every time step with the points of the interfaces implied by the zero-contour of the piece wise linear approximation of the level-set functions. This extension was done by either shifting nodes of the background mesh to an interface point or by adding the interface point to the background mesh and re-triangulating. As the localization of the interface was the main priority we had to take extra care in the process of adding the triple points. Because shifting of the background mesh points changes the level-set approximation, we located the triple points on the background mesh at the start of the enriched mesh creation and added these to the enriched mesh after shifting. Next to shifting and cutting we introduced a third technique especially for the triple points, called bending. This technique was based on the same principles as when to shift or cut for normal interface points over an edge, but now defined for a triangle. This extra technique was needed in order to avoid ill-shaped triangles. It results in a slight loss of the interface location, which also happens when shifting, but also in a qualitative better mesh.
The accuracy and quality of the mesh was good overall, but most quality issues came from the triple points and the interfaces in their neighborhood. The δ value which tells when to shift, bend or cut was set the same as in the two phase model in the work of Den Ouden [18]. As this value takes a good combination of the accuracy of the enriched mesh capturing the interface and the quality of the mesh by avoiding ill-shaped elements.

Even though the problems introduced because of the model's description, the resulting growth/dissolution of the phases was almost as we expected beforehand, except for the triple points movement. Because of the high austenite carbon concentration in the triple points and the lower ferrite carbon gradient, the triple point movement dominated the movement of the interfaces until they met each other at the bottom of the cementite particle, or at the major axis of the cementite particle for the two particle model, and disappeared. This triple point velocity was not expected to be relatively higher than the other velocities. By adding in the effect of interface energies in the boundary conditions, the pearlite growth behavior could maybe be better recreated. But changing the carbon concentration in the triple points in austenite with respect to better ferrite carbon gradient concentration recovery should also lower the triple point velocity.

After the triple points disappeared, the ferrite phase started to grow into the austenite and the cementite into the ferrite just as in the one-dimensional model we investigated in [26]. The growth of cementite into ferrite was as expected in order to satisfy with mass conservation, because ferrite can hold less carbon than austenite and ferrite replaced austenite. Cementite also had to grew a bit, but only a little as it can contain a lot more carbon, in order to make up for this mass change.

We also saw that the growth of the faster moving interface, the austenite-ferrite interface, started being unstable. This effect has also been seen in the work of Den Ouden and the origin of this growth has been shown in the work of Mullins and Sekerka [15]. The combination of physical growth and small perturbations of the concentration values cause by the topology of the interface cause these inaccuracies to amplify over time, resulting in the wiggly austenite-ferrite interface as seen in Figures 7.1, 7.2, 7.3 and 7.5. Of course higher deviations occur at higher cooling rates where the interfaces move faster. Next to a more realistic growth, the Gibbs-Thomson effect could also stabilize this unstable growth by dampening the carbon concentration gap at highly curved parts of the interface. This decreases the movement of the interface at those locations, letting the parts that fell behind to catch up. We were not able to add the Gibbs-Thomson effect to our model, mostly because of our level-set functions losing the Eikonal property over time and not being able to reinitialize them.

Furthermore for the different cooling rates 0, 0.4 and 10 Ks^{-1} we saw that the movement of the triple points was so fast that they disappeared before the difference in temperature could have much influence. However after the triple points disappeared we saw that the temperature difference did change the rate of growth of ferrite and cementite. A higher cooling rate caused faster reactions and thus faster movement of the interfaces, as was also seen in the one-dimensional model.

We also observed that the original ferrite particle seems to become unrecognizable as the other part of the austenite-ferrite interface catches up with the original particle. We concluded that this was due the ferrite carbon concentration gradient increasing the difference between the austenite carbon concentration and the equilibrium concentration on the austenite-ferrite interface. Although a higher ferrite carbon gradient also directly impacts the austenite-ferrite interface velocity in the opposite direction, its impact was lower than the change in austenite carbon concentration. Thus the steeper gradient on the narrow band of ferrite results in higher outward velocities on the austenite-ferrite interface which was not present at the start of the model, making it catch up with the original nucleus.

Finally for the zero cooling rate, when austenite dissolved, the ferrite and cementite were already close to equilibrium. This means the model was diffusion controlled. The final domain in equilibrium was also still symmetric around the initial x-coordinate of the ellipses centers $x = 12.5$ μm even with the unstable growth.

We also simulated for a two ferrite particle model, but we saw that there was not much difference with the one particle setup. The triple points meet at around half of the time at which the triple points in the one particle model disappeared, meaning the triple point velocity over the cementite interface is almost constant in time. After the triple points disappear, the same unstable growth occurs and the original particles tend to become unrecognizable as the rest of the austenite-ferrite interface catches up.

It was shown in [23] that reinitialization of the level-set function is needed if the level-set function deters too much from the distance function property. We used the same reinitialization method as used by Den Ouden [18], but even with the TVD RK3 stabilizing time integration method we were not able to get a converging reinitialization method for all times during the simulation. We showed the method to work reasonably for a test case and for the first few instances in the simulation, but it fails after a certain amount of time during the simulation. It is unsure what exactly makes the method diverge. One problem could be the combination of fixing the interface, but also wanting to change the gradient at the same time.

Because our algorithm was able to produce the results we showed without using the reinitialization method, it seems the method is not as needed as thought beforehand. Although its error might not be showing because of bigger errors still present in the model. For example the low resolution of the mesh that shows in the gradient approximation of the ferrite carbon concentration. Reinitialization will also be direly needed if one wants to add the Gibbs-Thomson effect to this model, as the curvature has to be recovered from the level-set functions.
Some more ideas on the reinitialization methods are shown in Section 9.4.

The biggest problem in this research was trying to have an accurate approximation of the gradient of the ferrite carbon concentration. Because the ferrite domain has sharp angles with the cementite particle, it is difficult to recover an accurate gradient approximation. It even becomes more difficult during the simulation when the triple points start moving and the gradient has to be approximated over a narrow band of ferrite.
Although the ferrite carbon concentration gradient influences the austenite-ferrite interface velocity and the austenite carbon concentration, at least we still got reasonable results for the phase transformations. Except for the fact that the gradient should have a more dampening influence on the triple point velocity as we saw for the finer meshes. In Section 9.2 we discuss several improvements and alternatives to the current approach.

As there is still a problem present in the model and with resolution in the gradient approximation method, it can be expected for the approximation of the carbon concentrations not to converge. We did see this in the mass change over time for different meshes. However, per phase individually we did see convergence. Some crossing error must be present in the algorithm for this to happen, or the meshes we ran the simulations with were not fine enough to get proper results. Because the computation time for a time step with $h_{\max} = {}^{1}/_{16}$ took too much time, we were not able to get more refined results. In further research this could be improved with a local refinement option by leaving the mesh coarser where not much is happening (away from the interfaces) and refining only in the neighborhoods of the interfaces. More about this in Section 9.2.1.

All in all, even though there are some major flaws still present in the model and the approximation of its solution, the current method showed reasonable results and convergence. However, before this model can be used accurately, several improvements should be made, but the overall idea of the model seems to be correct. In the next Chapter 9 several improvements and other options for the model are discussed for further research.

9 Future Work

In this chapter we discuss some different ideas on how to fix most of the issues we found in the model which we were not able to investigate/apply.

9.1. Triple Point Boundary Condition

The combination of boundary conditions and a continuous interface velocity indirectly lead to a Dirichlet boundary condition for the austenite carbon concentration in the triple points. Because there is no literature about the exact carbon concentration values in the triple points like we have with the equilibrium concentration lines in the phase diagram, we fixed the concentration at the starting temperature T_0 by looking for a reasonable concentration value for a fine mesh. The concentration is fixed by choosing the fraction between proportionality constants in the constants $K^{\gamma\theta}$ and $K^{\alpha\theta}$. The only logical assumption which could be made was the fact that the concentration value needs to be above the equilibrium concentration $c_{\gamma\alpha}^{\text{sol}}$ to have a velocity in the $\boldsymbol{n}^{\alpha\gamma}$ direction (not taking the ferrite carbon concentration gradient in consideration). We saw that the concentration value in the triple points has a big influence on the growth of the ferrite nucleus as it works as a counterbalance to the ferrite carbon concentration gradient. Either other boundary conditions should be derived or this value should be derived from physical measurements or other physical knowledge about austenite/ferrite/cementite triple point behavior. For this research we choose a value that seemed to fit for a fine mesh, but caused unrealistic triple point velocities for the more coarse mesh. This means choosing this concentration value must be done very carefully to obtain realistic model behavior.

This is the biggest open question that is left from this research. If one is to continue this work, solving this problem should be the first priority.

9.2. Gradient Approximation

The second big obstacle that was found in this research, is the gradient recovery of the ferrite carbon concentration. This gradient can only be recovered from within the ferrite domain itself, as the ferrite carbon concentration is only defined on this domain. We showed that for the initial ferrite nucleus, the gradient recovery will converge, but because of the mesh resolution, the gradient was not recovered accurate enough to capture the correct behavior of the model. Because of the computation time per time step, simulations with finer meshes were not an option. With this in mind we propose the following solutions mentioned in this section.

9.2.1. Local Mesh Refinement

For one of the test cases in the work of [31] a local mesh refinement is applied near the singularity of a L-shaped mesh, because their gradient recovery method is not able to get good converging approximations in this region. Even with the refinement they could only obtain a relative low order of convergence. Although this does not seem to be a problem for our initial ferrite domain, it is good to keep this in mind. Local mesh refined could be applied in the neighborhood of all interfaces to increase the accuracy of the algorithm and the gradient recovery, but also to decrease computation time. Far away from the interfaces there is almost no change in carbon concentrations, so the mesh can be more coarse the farther away nodes are from the interfaces. Although the idea of local mesh refinement sounds good, the location of the interfaces change over time. This means the mesh refinement has to be redefined over time, which is computationally heavy. A solution for this comes with adaptive mesh refinement. In the work of Möller [14], the AFC method of Kuzmin [10] is used to determine where high gradients are located. In this neighborhood the mesh is refined and the mesh is coarsened in neighborhoods with low gradients. The same mesh refinement/coarsening technique can be used for our model, but then refinement will take place in the neighborhood of the interfaces based on the level-set function values. The farther away from the interfaces, the coarser the mesh is allowed to be.

9.2.2. Coupled System

In stead of having to approximate the ferrite carbon concentration gradient $\frac{\partial c_\alpha}{\partial \boldsymbol{x}}$, we could define this quantity as a variable. The following statements are directly derived from [4].
Define

$$\boldsymbol{q}(\boldsymbol{x},t) = -D_\alpha(\boldsymbol{x},t)\nabla c_\alpha(\boldsymbol{x},t), \tag{9.1}$$

and using a proposed generalized Cattaneo's law, we get the system of equations

$$\left\{\begin{array}{rcl} \frac{\partial c_\alpha}{\partial t}(\boldsymbol{x},t) + v_n^{\text{mesh}}(\boldsymbol{x},t)\cdot\nabla c_\alpha(\boldsymbol{x},t) + \nabla\cdot\boldsymbol{q}(\boldsymbol{x},t) & = & 0, \\ \boldsymbol{q}(\boldsymbol{x},t) + \boldsymbol{\tau}\left(\frac{\partial \boldsymbol{q}}{\partial t}(\boldsymbol{x},t) + \nabla(\boldsymbol{q}(\boldsymbol{x},t))\,v_n^{\text{mesh}}(\boldsymbol{x},t)\right) & = & -D_\alpha(\boldsymbol{x},t)\nabla c_\alpha(\boldsymbol{x},t). \end{array}\right. \tag{9.2}$$

Where $\boldsymbol{\tau}$ is a relaxation tensor. Under the assumption of incompressibility and homogeneity, i.e. $\boldsymbol{\tau} = \tau I$, with I the identity matrix, we get the hyperbolic system

$$\left\{\begin{array}{rcl} \frac{\partial c_\alpha}{\partial t}(\boldsymbol{x},t) + \nabla\cdot\left(v_n^{\text{mesh}}(\boldsymbol{x},t)c_\alpha(\boldsymbol{x},t) + \boldsymbol{q}(\boldsymbol{x},t)\right) & = & 0, \\ \frac{\partial \tau\boldsymbol{q}}{\partial t}(\boldsymbol{x},t) + \nabla\cdot\left(\tau\boldsymbol{q}(\boldsymbol{x},t)\left(v_n^{\text{mesh}}(\boldsymbol{x},t)\right)^T + kc_\alpha(\boldsymbol{x},t)I\right) & = & -\boldsymbol{q}(\boldsymbol{x},t). \end{array}\right. \tag{9.3}$$

For $\tau = 0$ we get back the equations we started with. The effect of Cattaneo's law is said to introduce anti-diffusion to the system. The assumption of incompressibility of the mesh velocity might not be reachable though.
Now to make sure no oscillations occur, we can apply AFC for coupled systems as shown in [9]. A hyperbolic system can be rewritten from the form

$$\frac{\partial U}{\partial t} + \sum_{d=\{x,y\}} \frac{\partial F^d}{\partial d} = S, \tag{9.4}$$

to

$$\frac{\partial U}{\partial t} + \sum_{d=\{x,y\}} A^d \frac{\partial U}{\partial d} = S. \tag{9.5}$$

In our case we have

$$U = \left[c_\alpha\ \tau q_x\ \tau q_y\right]^T, \tag{9.6}$$

$$F = \left[F_x\ F_y\right]^T, \tag{9.7}$$

$$F = \begin{bmatrix} \boldsymbol{q} + v_n^{\text{mesh}} c_\alpha \\ \tau\boldsymbol{q}\left(v_n^{\text{mesh}}\right)^T + D_\alpha c_\alpha I \end{bmatrix}, \tag{9.8}$$

$$A^x = \begin{bmatrix} v_{n,x}^{\text{mesh}} & \frac{1}{\tau} & 0 \\ D_\alpha & v_{n,x}^{\text{mesh}} & 0 \\ 0 & 0 & v_{n,x}^{\text{mesh}} \end{bmatrix}, \tag{9.9}$$

$$A^y = \begin{bmatrix} v_{n,y}^{\text{mesh}} & 0 & \frac{1}{\tau} \\ 0 & v_{n,y}^{\text{mesh}} & 0 \\ D_\alpha & 0 & v_{n,y}^{\text{mesh}} \end{bmatrix}, \tag{9.10}$$

$$S = \left[0\ -\boldsymbol{q}\right]^T. \tag{9.11}$$

With A^d we can decouple the hyperbolic system by diagonalization of the matrix A^d. We get the eigenvalues for $d = \{x, y\}$

$$\lambda_d^1 = v_{n,d}^{\text{mesh}}, \tag{9.12}$$

$$\lambda_d^{2,3} = v_{n,d}^{\text{mesh}} \pm \sqrt{\frac{D_\alpha}{\tau}}. \tag{9.13}$$

The decoupled system can then be flux corrected per decoupled variable and transformed back afterwards. For this model local mesh refinements can also be used just as proposed in the previous Section 9.2.1.

9.2.3. One Diffusion Domain

A third way of circumventing the problem with the gradient approximation is to make only one diffusion domain for both austenite and ferrite. Because the austenite-ferrite interface is no longer a boundary of the new domain, the interface velocity on this interface should be derived by another assumption, but we lose the problem of recovering the ferrite carbon concentration gradient and the austenite carbon concentration in the triple points. Also the need for the carbon concentration to be close to the equilibrium value $c_{\gamma\alpha}^{\text{sol}}$ for the austenite domain and $c_{\alpha\gamma}^{\text{sol}}$ for the ferrite domain should somehow be incorporated, with either a continuous or discontinuous transition of the concentration levels.

9.3. Gibbs-Thomson Effect

The equilibrium concentration values can be extended with the Gibbs-Thomson effect [19]. This effect adds an exponential factor to the solubility which depends on temperature and curvature of the domain. It is written as:

$$c_{kl}^{\text{sol}}(\boldsymbol{x}, t) = c_{kl}^{\infty}(t) \exp\left\{\frac{E_{kl} V_m}{R_g T(t)} \kappa^{kl}(\boldsymbol{x}, t)\right\}, \text{for } \boldsymbol{x} \in \Gamma^{kl}(t), t > t_0, kl = \{\gamma\alpha, \gamma\theta, \alpha\theta\}, \tag{9.14}$$

where $c_{kl}^{\infty}(t)$ is the equilibrium concentration without space dependency, of phase k with respect to phase l. E_{kl} is the interface energy, V_m the molar volume of phase l, R_g the gas constant, $T(t)$ the temperature at time t and $\kappa^{kl}(\boldsymbol{x}, t)$ the sum of the principle curvatures of $\Gamma^{kl}(t)$. This curvature effect will cause the matrix/precipitate to grow/dissolve to a formation where overall surface tension is lowest, i.e. the total energy of the system will be minimized [15].

The sum of the principle curvatures in 2D can be obtained from the normal vector $\boldsymbol{n}^{kl}(\boldsymbol{x}, t)$:

$$\kappa^{kl}(\boldsymbol{x}, t) = -\nabla \cdot \boldsymbol{n}^{kl}(\boldsymbol{x}, t) \left(= -\nabla \cdot \frac{\nabla \phi^{kl}(\boldsymbol{x}, t)}{\|\nabla \phi^{kl}(\boldsymbol{x}, t)\|_2} \right). \tag{9.15}$$

The curvature quantifies how sharp the interface is as at a given point $\boldsymbol{x}$. The curvature is the value $\pm\frac{1}{R}$ for R the radius of the biggest circle that touches point $\boldsymbol{x}$ but no other points of the interface. A convex part of the interface, seen from phase k to l, has positive curvature value and a concave interface a negative value (see also Figure 9.1). Without specifying direction between two phases, the curvature is always observed from the domain where $\phi(\boldsymbol{x}, t) > 0$. Because the convex parts of the interface have positive curvature the exponential in the Gibbs-Thomson effect has a positive power, meaning it is bigger than one and the gap between the concentration and the equilibrium concentration will be smaller. Thus implying a lower interface velocity. Whereas the concave parts of the interface will have a negative curvature, meaning the gap in concentration will be bigger resulting in a higher velocity. This means that the parts of the interface that have fallen behind, which are concave, will catch up to the convex parts that also slow down because of the positive curvature. This will stabilize the unstable growth introduced by the amplification of small errors in the interface. The stabilizing influence of the Gibbs-Thomson was shown in Den Ouden's work [18].

9.3.1. Curvature Approximation

With the gradient recovery method of Zhang and Naga [31] the curvature can be approximated. As the curvature has a second order derivative, the recovery can be done in two ways. Either by assuming an n'th order

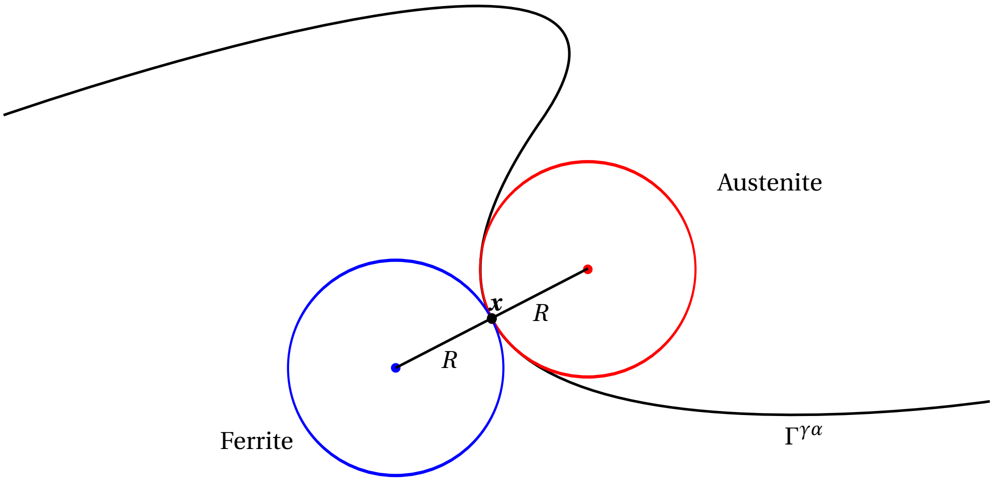


Figure 9.1: Negative valued curvature $\kappa^{\gamma\alpha}(\boldsymbol{x}) = -1/R$ in austenite on the interface $\Gamma^{\gamma\alpha}$ in point $\boldsymbol{x}$. The curvature in ferrite in the same point has the same value but positive, i.e. $\kappa^{\alpha\gamma}(\boldsymbol{x}) = 1/R$.

polynomial for $\phi(\boldsymbol{x}, t)$ and then calculating

$$\kappa(\boldsymbol{x}, t) = -\nabla \cdot \frac{\nabla\phi(\boldsymbol{x}, t)}{\left\|\nabla\phi(\boldsymbol{x}, t)\right\|_2}, \tag{9.16}$$

or by assuming a n'th order polynomial for both components of $\boldsymbol{n}(\boldsymbol{x}, t)$ and then recovering

$$\kappa(\boldsymbol{x}, t) = -\nabla \cdot \boldsymbol{n}(\boldsymbol{x}, t). \tag{9.17}$$

In the work of Den Ouden [18] it was shown that the latter option is more accurate. The normal vector is assumed piece-wise linear in its components n_x and n_y. Den Ouden also showed that recovering the curvature from the enriched mesh is more accurate than interpolating it from the background mesh. So the curvature can directly be calculated in the interface points using the already known normals.

9.4. Reinitialization

As reinitialization was not in the scope of this research we did not put much effort into this, even when eventually it did not seem to work. Reinitialization is needed to fix the loss of the signed-distance function during the simulation. We can show that the length of the gradient changes as follows:

$$\begin{aligned}
\frac{\partial\left\|\nabla\phi\right\|^2}{\partial t}(\boldsymbol{x}, t) &= 2\frac{\partial\nabla\phi}{\partial t}(\boldsymbol{x}, t)\cdot\nabla\phi(\boldsymbol{x}, t) && (9.18)\\
&= -2\nabla\left(v_n^{\text{ex}}(\boldsymbol{x}, t)\cdot\nabla\phi(\boldsymbol{x}, t)\right)\cdot\nabla\phi(\boldsymbol{x}, t) && (9.19)\\
&= -2\left(J_{v_n^{\text{ex}}}(\boldsymbol{x}, t)\nabla\phi(\boldsymbol{x}, t) + J_{\nabla\phi}(\boldsymbol{x}, t)\, v_n^{\text{ex}}(\boldsymbol{x}, t)\right)\cdot\nabla\phi(\boldsymbol{x}, t), && (9.20)
\end{aligned}$$

where $J_{\boldsymbol{y}}$ is the Jacobian of the vector $\boldsymbol{y}$. For this to be zero the following should hold

$$\left(J_{v_n^{\text{ex}}}(\boldsymbol{x}, t)\nabla\phi(\boldsymbol{x}, t) + J_{\nabla\phi}(\boldsymbol{x}, t)\, v_n^{\text{ex}}(\boldsymbol{x}, t)\right)\cdot\nabla\phi(\boldsymbol{x}, t) = 0, \tag{9.21}$$

which is not the case for the Laplace Equation (3.5) we defined in Chapter 3. One could try to solve/approximate this differential equation for $v_n^{\text{ex}}(\boldsymbol{x}, t)$, but generally this is more difficult then reinitializing the level-set function.

Although there are several methods present, none of them can make sure the interface location is preserved while reinitializing all nodes in a band around the zero-contour of the level-set function. We applied the reinitialization method of hyperbolic re-distancing with a TVD time integration scheme and combined with a Dirichlet condition by Laplace multipliers to fix the zero-contour on the background mesh. We showed it to succeed for a non-signed distance function defined with the same zero-contour as our starting austenite-ferrite ellipse and it also worked for the first few times reinitialization was done in the simulation. It however seemed to diverge later on during the simulation. It is unclear what exactly caused the method to diverge, but an alternative should be found. Our results did not seem to really suffer under the slight loss of the signed-distance property, but it should be done especially when the effect of the interface curvatures is added to the model.

9.5. Extending Interface Velocity over the Ghost Domain

Because the ghost interface of the austenite-ferrite level set function is an artificial interface with no physical meaning, there is no known interface velocity on this interface. However, close to the triple points the ghost interface should be moving along with the austenite-ferrite interface, because the movement of the ghost interface will influence the location of the new triple points. That is is why we extended the interface velocity from the known austenite-ferrite interface velocity to the ghost interface by extrapolating the austenite-ferrite interface velocity. The rest of the ghost interface was allowed to move any way it wanted to, except it should never cross the austenite-cementite interface where it would make a new ferrite particle. Because in our model the cementite particle has a smooth shape, the ghost interface did not gave any troubles. Though for more sharp defined domains, this same treatment could not always work. A way to circumvent this is by redefining the ghost interface after updating the level-set function, giving it a ghost interface that does not cross any interfaces.

9.6. Three-Dimensional Model

When all of the above issues of this model have been fixed, one could also implement this model for the three-dimensional case. The meshing algorithm 5.1 for a two-dimensional mesh can be transitioned to a three-dimensional setting. In stead of the interface being a line that cuts a triangle, you will get a plane that cuts a tetrahedra. Although in the two-dimensional case you almost always have two intersection points (only when the interface goes directly through one of the vertices you will have only one intersection), a tetrahedron can have either three or four intersection points when cut by a plane. But the same shift and cut strategy can be applied as in the two-dimensional case, where background mesh nodes can shift to any of the intersection points of the interface in an element when it is close to the intersection point. See for example [7] for a three-dimensional particle dissolution in binary alloys. In the three phase model there will also be tetrahedron where two interfaces will cut, which will give ‘triple’ lines. One could imagine that shifting and cutting techniques for the vertices of these triple lines can be extended in the same way as the two-dimensional case. Although some more research should be done to see if a technique like bending is needed and/or possible.

Bibliography

[1] H.I. Aaronson, M. Enomoto, and J.K. Lee. *Mechanisms of Diffusional Phase Transformations in Metals and Alloys.* CRC Press, Taylor & Francis Group, 2010. ISBN 978-1-4200-6299-1.

[2] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen differenzengleichungen der mathematischen physik. *Mathematische Annalen*, 100:32–74, 1928.

[3] E. Gintautas. Algebraic stabilizations for scalar convection-diffusion equations. Master's thesis, Freie Universität Berlin Fachbereich für Mathematik und Informatik Berlin, Germany, 2016.

[4] H. Gómez, I. Colominas, F. Navarrina, and M. Casteleiro. A hyperbolic model for convection-diffusion transport problems in cfd: Numerical analysis and applications. *Revista de la Real Academia de Ciencias Exactas, Físicas y Naturales. Serie A Matemáticas (RACSAM).*, 102(2):319–334, 2008.

[5] S. Gottlieb and C. Shu. Total variation diminishing runge-kutta schemes. Technical report, Mathematics of Computation, 1998.

[6] G.B. Hughes and M. Chraibi. Calculating ellipse overlap areas. *Computing and Visualization in Science*, 15:291–301, 2012.

[7] E. Javierre. *Numerical methods for vector Stefan models of solid-state alloys.* PhD thesis, Delft University of Technology, 2006.

[8] G.P. Krielaart, J. Sietsma, and S. van der Zwaag. Ferrite formation in fe-c alloys during austenite decomposition under non-equilibrium interface conditions. *Materials Science and Engeneering*, 237:216–223, 1997.

[9] D. Kuzmin. Algebraic flux correction for finite element discretizations of coupled systems. University of Dortmund, 2007.

[10] D. Kuzmin and M. Möller. *Algebraic Flux Correction I. Scalar Conservation Laws.* Springer Berlin Heidelberg, 2005. ISBN 978-3-540-23730-3.

[11] D. Kuzmin, M. Möller, and S. Turek. Multidimensional fem-fct schemes for arbitrary time-stepping. *International Journal for Numerical Methods in Fluids*, 42:265–295, 2003.

[12] C. Li, C. Xu, C. Gui, and M.D. Fox. Distance regularized level set evolution and its application to image segmentation. *IEEE Transactions on Image Processing*, 19:3243–3254, 2010.

[13] M.G. Mecozzi. Phase field modelling of the austenite to ferrite transformation in steels. Technical report, Delft University of Technology, 2006.

[14] M. Möller. *Adaptive High-Resolution Finite Element Schemes.* PhD thesis, Technischen Universität Dortmund, 2008.

[15] W.W. Mullings and R.F. Sekerka. Morphological stability of a particle growing by diffusion or heat flow. *Journal of Applied Physics*, 34(2):323–329, 1963.

[16] US National Institute of Standards and Technology. Codata value: molar gas constant. `http://physics.nist.gov/cgi-bin/cuu/Value?r`, 2015.

[17] M. Onink, C.M. Brakrnan, F.D. Tichelaar, E.J. Mittemeijer, and S. van der Zwaag. The lattice parameters of austenite and ferrite in fec alloys as functions of carbon concentration and temperature. *Scripta Metallurgica*, 29:1011–1016, 1993.

[18] D. den Ouden. *Mathematical Modelling of Nucleating and Growing Precipitates: Distributions and Interfaces.* PhD thesis, Delft University of Technology, 2015.

[19] E. Javierre Perez. Gibbs-thomson effects in phase transformations. *Scripta Materialia*, 52(8):709–712, 2005.

[20] V.I. Savran. *Austenite formation in C-Mn steel.* PhD thesis, Delft University of Technology, 2009.

[21] J.A. Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences of the United States of America.*, 93(4):1591–1595, 1996.

[22] K.P. Shah. Formation and growth of pearlite. `http://practicalmaintenance.net/wp-content/uploads/Formation-and-Growth-of-Pearlite.jpg`.

[23] M. Sussman, P. Smereka, and S.J. Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114:146–159, 1994.

[24] Thermo-Calc Software Inc. Thermocalc. URL `http://www.thermocalc.com`.

[25] Y.H. Tsai, L.T. Cheng, S. Osher, and H.K. Zhao. Fast sweeping algorithms for a class of hamilton- jacobi equations. *SIAM Journal on Numerical Analysis*, 41:673–694, 2003.

[26] T. Verbeek. Literature study for thesis project modelling and simulating three phases of steel: austenite, ferrite and cementite, June 2017.

[27] F.J. Vermolen, E. Javierre, C. Vuik, L. Zhao, and S. van der Zwaag. A three-dimensional model for patricle dissolution in binary alloys. *Computational Materials Science*, 39:767–744, 2007.

[28] M.E. Wieser. Atomic weights of the elements. *Pure and Applied Chemistry*, 78(11):2051–2066, 2005.

[29] T. Young. An essay on the cohesion of fluids. *Philosophical Transactions of the Royal Society*, 95:65–87, 1805.

[30] S.T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics*, 31:335–362, 1978.

[31] Z. Zhang and A. Naga. A new finite element gradient recovery method: Superconvergence property. *SIAM Journal on Scientific Computing*, 26:1192–1213, 2005.