

# A multigrid solver for phase field simulation of microstructure evolution

*Liesbeth Vanherpe*

*Frank Wendler*

*Britta Nestler*

*Stefan Vandewalle*

*Report TW 542, July 2009*



Katholieke Universiteit Leuven  
Department of Computer Science

Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

# A multigrid solver for phase field simulation of microstructure evolution

*Liesbeth Vanherpe*

*Frank Wendler*

*Britta Nestler*

*Stefan Vandewalle*

*Report TW 542, July 2009*

Department of Computer Science, K.U.Leuven

## **Abstract**

This paper presents a semi-implicit numerical method for the simulation of grain growth in two dimensions with a multi phase field model. To avoid the strong stability condition of traditional explicit methods, a first-order, semi-implicit discretisation scheme is employed, which offers a good compromise with regard to memory intensity and computational requirements. A nonlinear multigrid solver based on the Full Approximation Scheme is implemented to solve the equations resulting from this discretisation. Simulations with the multigrid solver show that the solver has grid size independent convergence properties and is faster than a standard first-order explicit solver. To validate the implementation, the results of specific test cases are studied.

**Keywords :** Phase field simulation, Nonlinear multigrid method, Grain growth  
**MSC :** Primary : 35Q99, Secondary : 65M55.

# A multigrid solver for phase field simulation of microstructure evolution

Liesbeth Vanherpe <sup>a,\*</sup>, Frank Wendler <sup>b</sup>, Britta Nestler <sup>b</sup>,  
Stefan Vandewalle <sup>a</sup>

<sup>a</sup>*Department of Computer Science, Katholieke Universiteit Leuven,  
Celestijnenlaan 200A, 3001 Leuven, Belgium*

<sup>b</sup>*Department of Computer Science, Karlsruhe University of Applied Sciences,  
Moltkestraße 30, 76133 Karlsruhe, Germany*

---

## Abstract

This paper presents a semi-implicit numerical method for the simulation of grain growth in two dimensions with a multi phase field model. To avoid the strong stability condition of traditional explicit methods, a first-order, semi-implicit discretisation scheme is employed, which offers a good compromise with regard to memory intensity and computational requirements. A nonlinear multigrid solver based on the Full Approximation Scheme is implemented to solve the equations resulting from this discretisation. Simulations with the multigrid solver show that the solver has grid size independent convergence properties and is faster than a standard first-order explicit solver. To validate the implementation, the results of specific test cases are studied.

*Key words:* Phase field simulation, Nonlinear multigrid method, Grain growth

---

## 1 Introduction

Metallic alloys form an important group of materials in industrial applications. Many macroscopic properties, such as strength and toughness, depend on the materials microstructure. Therefore, the study of the microstructure formation

---

\* Corresponding author. Tel.: +32 16 327663

*Email addresses:* [liesbeth.vanherpe@cs.kuleuven.be](mailto:liesbeth.vanherpe@cs.kuleuven.be) (Liesbeth Vanherpe),  
[frank.wendler@hs-karlsruhe.de](mailto:frank.wendler@hs-karlsruhe.de) (Frank Wendler),  
[britta.nestler@hs-karlsruhe.de](mailto:britta.nestler@hs-karlsruhe.de) (Britta Nestler),  
[stefan.vandewalle@cs.kuleuven.be](mailto:stefan.vandewalle@cs.kuleuven.be) (Stefan Vandewalle).

of these materials during manufacturing is of great technological importance. A large number of theoretical, experimental and computational studies has been performed on this topic. Computer simulations in particular are essential. They allow to study separately the role of different parameters. In combination with information obtained from detailed simulated structures, these studies provide valuable insight in microstructure formation and evolution.

A well established, flexible tool to simulate microstructural evolution phenomena is phase field modelling. Phase field models allow the prediction of complex morphologies, while including different thermodynamic driving forces and different transport processes. Instead of explicitly tracking moving boundaries, a phase field model employs a diffuse interface description.

In [6,15] and [4,13], phase field models are proposed which represent a multi phase microstructure by a set of phase field variables that are continuous in space and time. Inside a bulk phase, a phase field variable takes a non-zero equilibrium value, while the other variables assume the value zero. Across the phase interfaces, all variables change in a smooth continuous fashion to their equilibrium values in the neighbouring phases. The spatial and temporal evolution of the set of phase field variables is governed by a coupled set of partial differential equations of reaction-diffusion type.

One of the multi phase evolution phenomena that can be modelled by the evolution of a set of phase field variables is grain growth. Grain growth occurs in polycrystalline materials at elevated temperatures to reduce the energy associated with grain boundaries. Typically, the larger grains grow, while the smaller grains shrink and eventually disappear. Each of the phase field variables included in a multi phase field simulation of grain growth represents a crystallographic orientation.

For realistic structures, the required computing time and memory of a phase field model impose limitations on any practical implementation. First, to capture the correct dynamics, a sufficient amount of phase field variables has to be included in a simulation. Second, the resolution of the employed numerical techniques has to be fine enough to represent the interfaces correctly. Also, the use of large domains is appropriate to prevent boundary effects. As a consequence, realistic computer simulations of microstructural evolution with a phase field model can demand significant amounts of computation power. Especially when modelling anisotropic systems, it is important that this orientation is well resolved. Moreover, for grain growth in particular, systems with a statistically significant number of grains have to be considered.

Several algorithms and methods have been designed and tested to address these computational challenges and to overcome the current limitations of the phase field method. Parallel computing [14,19], adaptive meshing [16,17], and

moving mesh techniques [5], have been used to increase computational possibilities. Furthermore, the characteristics of the solutions have been exploited in sparse techniques [21,11,7,20]. In particular, the use of an efficient solver is very important.

In [6] and [15], a multi phase field model is presented for grain growth and simulation results are discussed in [14]. The solvers employed in these works are constructed using explicit discretisation schemes. Unfortunately, these discretisation methods suffer from a limited time step due to a strict stability condition. In contrast, implicit discretisation methods allow the choice of a larger time step. However, implicit discretisation schemes result in systems which need to be solved. To make sure that this extra computing time is compensated by the possibility of a large time step, it is necessary to look for dedicated solvers. For the phase field model of [6,15] for grain growth in particular, the nonlinear systems resulting from this discretisation are very large.

Multigrid algorithms accelerate the convergence of standard iterative solvers. When the components of these algorithms are carefully selected, they can scale linearly with the number of spatial unknowns. A number of studies has been performed on the development of multigrid methods for phase field simulations. In [17], a nonlinear multigrid solver is implemented to solve a phase field model for binary alloy solidification based on the Full Approximation Scheme (FAS) with an adaptive grid approach. In the work of [12], monotone multigrid solvers are constructed to solve vector-valued Allen-Cahn equations, which are extensively used for phase field modelling. In [9], an extension of the method of [12] is presented to solve the Cahn-Hilliard equation, employed in conservative phase field models. Furthermore, Uzawa-type multigrid algorithms [1], linear multigrid methods [3] and FAS methods [10,22] have been developed to solve the latter equation type. However, in all of these studies, only a few phase field variables were considered.

In this work, a nonlinear multigrid solver based on the FAS scheme is proposed for two-dimensional phase field simulations with the multi phase field model according to [6,15]. This is the first time that a FAS scheme is implemented for this type of phase field models. The model is discretised with a semi-implicit discretisation scheme. The solver of the resulting equations has grid size independent properties and scales linearly with the number of spatial unknowns. The employed phase field model is presented in section 2. Section 3 deals with the discretisation of the multi phase field model. In section 4, the nonlinear multigrid solver is constructed. The multigrid components are described and the properties of the solver are tested and discussed. Simulation results are presented in section 5. The text ends with a conclusion in section 6.

## 2 Phase field model

In [6,15], a multi phase field model is proposed which models a polycrystalline microstructure by a vector-valued order parameter  $\boldsymbol{\phi}$  with  $N$  components  $(\phi_1, \dots, \phi_\alpha, \dots, \phi_N)$ . Each component is called a phase field variable and represents a different crystallographic orientation. The values of the  $N$  components are continuous in space and time. Inside a grain, one component  $\phi_\alpha$  takes the value 1, while the other phase field variables assume the value zero. Across the grain boundaries, the component values vary continuously from their equilibrium value in the grain to their equilibrium value in the neighbouring grains. In accordance with [6], the condition is imposed that the components  $\phi_\alpha$  must lie in the Gibbs-simplex  $\mathcal{G}$ , which is defined as

$$\mathcal{G} = \{\boldsymbol{\phi} \in \mathbb{R}^N \mid \phi_\alpha \geq 0, \sum_{\alpha=1}^N \phi_\alpha = 1\}. \quad (1)$$

The free energy of the microstructure is a function of the order parameter  $\boldsymbol{\phi}$  and is described by

$$\tilde{\mathcal{F}} = \int_V \left( \epsilon a(\boldsymbol{\phi}, \nabla \boldsymbol{\phi}) + \frac{1}{\epsilon} w(\boldsymbol{\phi}) \right) dV, \quad (2)$$

with  $a(\boldsymbol{\phi}, \nabla \boldsymbol{\phi})$  the gradient free energy density and  $w(\boldsymbol{\phi})$  the potential. The parameter  $\epsilon$  is a measure for the width of the diffuse interfaces. The term  $a(\boldsymbol{\phi}, \nabla \boldsymbol{\phi})$  models the interfacial contribution to the free energy and is given by:

$$a(\boldsymbol{\phi}, \nabla \boldsymbol{\phi}) = s \sum_{\substack{\alpha, \beta=1 \\ \alpha < \beta}}^N \sigma_{\alpha\beta} \|\phi_\alpha \nabla \phi_\beta - \phi_\beta \nabla \phi_\alpha\|^2, \quad (3)$$

with  $\sigma_{\alpha\beta}$  the surface energy density of the corresponding  $\alpha$ - $\beta$  interface [18,6]. A scaling parameter  $s$  is introduced for technical reasons: it allows to formulate the phase field model on the unit square. For  $N = 2$ , formulation (3) reduces to  $a(\boldsymbol{\phi}, \nabla \boldsymbol{\phi}) = \|\nabla \phi_\alpha\|^2$ , taking into account that  $\phi_\alpha + \phi_\beta = 1$ . For the bulk potential  $w(\boldsymbol{\phi})$ , the multiple obstacle potential is chosen. It is formulated as

$$w(\boldsymbol{\phi}) = w_{ob}(\boldsymbol{\phi}) = \frac{16}{\pi^2} \sum_{\alpha < \beta} \sigma_{\alpha\beta} \phi_\alpha \phi_\beta + \sum_{\alpha < \beta < \chi} \sigma_{\alpha\beta\chi} \phi_\alpha \phi_\beta \phi_\chi, \quad (4)$$

where  $w_{ob}(\boldsymbol{\phi})$  is defined to be infinite whenever  $\boldsymbol{\phi}$  is not in the Gibbs-simplex.

To ensure that the constraint  $\sum_{\alpha=1}^N \phi_\alpha = 1$  is preserved, a Lagrange multiplier function is added to the free energy formula (2). The free energy functional is now described by

$$\mathcal{F} = \int_V \left( \epsilon a(\boldsymbol{\phi}, \nabla \boldsymbol{\phi}) + \frac{1}{\epsilon} w(\boldsymbol{\phi}) \right) dV + \lambda \left( \sum_{\alpha=1}^N \phi_\alpha - 1 \right). \quad (5)$$

The evolution of the phase field variables  $\phi_\alpha$  is governed by gradient flow dynamics of the form

$$\omega \epsilon \frac{\partial \phi_\alpha}{\partial t} = - \frac{\delta \mathcal{F}}{\delta \phi_\alpha}, \quad \alpha = 1, \dots, N, \quad (6)$$

where the parameter  $\omega$  is related to a physical grain boundary property:  $\frac{1}{\omega}$  equals the interfacial mobility  $\mu$ . The notation  $\frac{\delta}{\delta \phi_\alpha}$  is used to denote the variational derivative with respect to  $\phi_\alpha$  and is computed as

$$\frac{\delta}{\delta \phi_\alpha} = \frac{\partial}{\partial \phi_\alpha} - \nabla \cdot \frac{\partial}{\partial (\nabla \phi_\alpha)}, \quad (7)$$

in accordance with [6]. The second part of the right-hand side of this formulation is the divergence of a vector, which can be written as

$$\nabla \cdot \frac{\partial}{\partial (\nabla \phi_\alpha)} = \nabla \cdot \begin{pmatrix} \frac{\partial}{\partial (\partial \phi_\alpha / \partial x)} \\ \frac{\partial}{\partial (\partial \phi_\alpha / \partial y)} \end{pmatrix}. \quad (8)$$

Substitution of (3) and (4) in the free energy functional (5), and elaborating the kinetic equations (6) results in:

$$\omega \epsilon \frac{\partial \phi_\alpha}{\partial t} = \epsilon \left( \nabla \cdot \frac{\partial a}{\partial (\nabla \phi_\alpha)} - \frac{\partial a}{\partial \phi_\alpha} \right) - \frac{1}{\epsilon} \frac{\partial w}{\partial \phi_\alpha} - \lambda, \quad \alpha = 1, \dots, N. \quad (9)$$

Here, we shall not follow the approach suggested in [15], where the function  $\lambda$  is eliminated explicitly. Instead, the system of equations (9) is extended with the constraint  $\sum_{\alpha=1}^N \phi_\alpha = 1$ . Hence the model consists of  $N$  partial differential equations plus one algebraic equation.

The gradient free energy component in (9) is derived as follows:

$$\nabla \cdot \frac{\partial a}{\partial(\nabla\phi_\alpha)} = -2 \sum_{\beta \neq \alpha} \sigma_{\alpha\beta} \nabla\phi_\beta \cdot (\phi_\alpha \nabla\phi_\beta - \phi_\beta \nabla\phi_\alpha) \quad (10a)$$

$$- 2 \sum_{\beta \neq \alpha} \sigma_{\alpha\beta} \phi_\beta (\phi_\alpha \nabla^2 \phi_\beta - \phi_\beta \nabla^2 \phi_\alpha),$$

$$\frac{\partial a}{\partial\phi_\alpha} = 2 \sum_{\beta \neq \alpha} \sigma_{\alpha\beta} \nabla\phi_\beta \cdot (\phi_\alpha \nabla\phi_\beta - \phi_\beta \nabla\phi_\alpha), \quad (10b)$$

and thus:

$$\nabla \cdot \frac{\partial a}{\partial(\nabla\phi_\alpha)} - \frac{\partial a}{\partial\phi_\alpha} = -4 \sum_{\beta \neq \alpha} \sigma_{\alpha\beta} \nabla\phi_\beta \cdot (\phi_\alpha \nabla\phi_\beta - \phi_\beta \nabla\phi_\alpha) - 2 \sum_{\beta \neq \alpha} \sigma_{\alpha\beta} \phi_\beta (\phi_\alpha \nabla^2 \phi_\beta - \phi_\beta \nabla^2 \phi_\alpha). \quad (11)$$

### 3 Discretization

To obtain statistically relevant results from a grain growth simulation, the use of a large domain with a large number of grains is required. Since this requirement makes model (9) memory intensive, it is appropriate to employ a two-step scheme for the discretisation in time, rather than a multi-step scheme, where the values of all phase field variables have to be retained for several time steps or stages simultaneously. Central differences are used for the discretisation of the first and second order spatial derivatives. No artificial oscillations were observed that would have required the use of one-sided differences for the first order derivative term. To avoid spurious oscillations near the sharp grain boundaries, we have decided not to use the Crank-Nicolson method, which is not stiffly stable.

Model equations (9) are discretised with a first-order semi-implicit discretisation scheme that treats the gradient energy part implicitly and the potential part explicitly:

$$\frac{\phi_\alpha^{n+1} - \phi_\alpha^n}{\Delta t} = \frac{1}{\omega} \left( \nabla \cdot \frac{\partial a}{\partial(\nabla\phi_\alpha)} - \frac{\partial a}{\partial\phi_\alpha} \right)^{n+1} - \frac{1}{\omega\epsilon^2} \left( \frac{\partial w}{\partial\phi_\alpha} \right)^n - \lambda^{n+1}, \quad (12a)$$

$$\alpha = 1, \dots, N,$$

$$\sum_{\alpha=1}^N \phi_\alpha^{n+1} = 1. \quad (12b)$$

This way, all terms involving spatial derivatives are treated implicitly and the stability condition on the time step proved to be independent of the spatial grid size. We also experimented with fully implicit schemes. However, this led to a convergence deterioration of the Newton solver. After some rearrangements, the following numerical scheme is obtained:

$$\phi_\alpha^{n+1} - \frac{\Delta t}{\omega} \left( \nabla \cdot \frac{\partial a}{\partial(\nabla\phi_\alpha)} - \frac{\partial a}{\partial\phi_\alpha} \right)^{n+1} + \lambda^{n+1} = \phi_\alpha^n - \frac{\Delta t}{\omega\epsilon^2} \left( \frac{\partial w}{\partial\phi_\alpha} \right)^n, \quad (13a)$$

$$\alpha = 1, \dots, N,$$

$$\sum_{\alpha=1}^N \phi_\alpha^{n+1} = 1. \quad (13b)$$

The spatial discretisation of the gradient free energy component (11) is given by

$$\begin{aligned} \left( \nabla \cdot \frac{\partial a}{\partial(\nabla\phi_\alpha)} - \frac{\partial a}{\partial\phi_\alpha} \right)_{ij} &\simeq -4 \sum_{\beta \neq \alpha} \sigma_{\alpha\beta} \left\{ \phi_{\alpha ij} \left[ \frac{(\Delta_{0x}\phi_{\beta ij})^2}{\Delta x^2} + \frac{(\Delta_{0y}\phi_{\beta ij})^2}{\Delta y^2} \right] \right. \\ &\quad \left. - \phi_{\beta ij} \left[ \frac{\Delta_{0x}\phi_{\alpha ij} \Delta_{0x}\phi_{\beta ij}}{\Delta x^2} + \frac{\Delta_{0y}\phi_{\alpha ij} \Delta_{0y}\phi_{\beta ij}}{\Delta y^2} \right] \right\} \\ &- 2 \sum_{\beta \neq \alpha} \sigma_{\alpha\beta} \left\{ \phi_{\alpha ij} \phi_{\beta ij} \left[ \frac{\phi_{\beta i+1j} - 2\phi_{\beta ij} + \phi_{\beta i-1j}}{\Delta x^2} + \frac{\phi_{\beta ij+1} - 2\phi_{\beta ij} + \phi_{\beta ij-1}}{\Delta y^2} \right] \right. \\ &\quad \left. - \phi_{\beta ij}^2 \left[ \frac{\phi_{\alpha i+1j} - 2\phi_{\alpha ij} + \phi_{\alpha i-1j}}{\Delta x^2} + \frac{\phi_{\alpha ij+1} - 2\phi_{\alpha ij} + \phi_{\alpha ij-1}}{\Delta y^2} \right] \right\}, \quad (14) \end{aligned}$$

with  $\Delta_{0x}\phi_{\alpha ij} = (\phi_{\alpha i+1j} - \phi_{\alpha i-1j})/2$  and  $\Delta_{0y}\phi_{\alpha ij} = (\phi_{\alpha ij+1} - \phi_{\alpha ij-1})/2$ .

## 4 Multigrid solver

### 4.1 Nonlinear multigrid solver

Spatial discretisation of scheme (13) yields a nonlinear system of equations that has to be solved at each time step. To solve the nonlinear system, a multigrid solver is implemented based on the Full Approximation Scheme (FAS) [2]. For the intergrid operators, full weighting is chosen as the restriction operator for the residual, injection for the restriction of the solution, and bilinear interpolation as the interpolation operator for both the solution and the residual. The convergence criterion of the multigrid cycle uses the norm of the residual scaled by the norm of the right-hand side. The smoothers are implemented using the collective Gauss-Seidel method. This method sweeps over all grid points in a lexicographic order, solving the spatially discretised equations (13) for all phase field variables and the function  $\lambda$  simultaneously. Therefore, at each grid point, a small nonlinear system of  $N + 1$  equations in  $N + 1$  unknowns has to be solved. For the solution of this nonlinear system, an inner Newton iteration is employed. The equations at a single grid point can be written in the form

$$N(\phi_\lambda^{n+1}) = B, \quad (15)$$

where  $\phi_\lambda$  is the order parameter  $\phi$  extended with the unknown  $\lambda$ . The Newton iteration can now be formulated as

$$\phi_\lambda^{k+1} = \phi_\lambda^k + J_N^{-1} (B - N(\phi_\lambda^k)), \quad k = 0, 1, 2, \dots \quad (16)$$

with  $J_N$  the Jacobian of the nonlinear operator  $N$ . Remark that  $B$  remains constant throughout the Newton iterations: it is the explicit part of the discretisation scheme. Iterations (16) are performed until the norm of the residual of (15), scaled by the norm of the right-hand side  $B$ , falls under the desired accuracy. The dense linear system involved in equation (16) is solved with a direct algorithm, Gaussian elimination with partial pivoting. This type of solver scales with the cube of the number of unknowns, which is here  $O((N + 1)^3) \simeq O(N^3)$ . For large, realistic grain growth simulations, ideally, thousands of phase field variables are included and thus  $N \gg 1000$ . However, at every grid point, only a few phase field variables are active [7,11,20,21] and have to be taken into account. Therefore, with an adaptation of the multigrid solver to the tracking of the active phase field variables, the use of a direct solver should not be problematic. For the coarse grid discretisation, the direct coarse grid analog of the fine grid operator is implemented. The associated

system is solved with the same direct solver as in the smoother.

Once the smoother is defined, the optimal cycle type and number of pre-smoothing and postsmoothing steps can be determined. Table 1 shows the convergence rate, the number of cycles and the execution time, averaged over 10 time steps, for different multigrid cycle types on a  $[0, 1] \times [0, 1]$  domain with  $N = 3$ ,  $s = 1/(6.4)^2$ ,  $\Delta x = 1/256$ , and  $\Delta t = 0.1$ . The other parameter values are chosen  $\omega = 1$ ,  $\epsilon = 1$ , and  $\sigma_{\alpha\beta} = 1.0$ . Unless otherwise mentioned, the latter parameter values are assumed for all following simulation tests. The structure is initialized with a sharp interface Voronoi tessellation: the domain is decomposed into different cells and each cell is assigned to a different phase field variable  $\phi_\alpha$ . Overall, the  $V(1, 1)$ -cycle turned out to be the most efficient.

Table 1

Convergence rate, number of cycles and execution time for different multigrid cycles averaged over 10 time steps for  $N = 3$ ,  $s = 1/(6.4)^2$ ,  $\Delta t = 0.1$ , and  $\Delta x = 1/256$ .

Cycle type	Convergence rate	Number of cycles	Time (s)
V(1,1)	0.2067	11.4	3.2450
V(2,2)	0.0869	7.4	4.0150
W(1,1)	0.0389	6	4.5190
W(2,2)	0.0031	3.1	4.4810

#### 4.2 Convergence properties

We will now verify the convergence properties of the multigrid solver as a function of the grid size,  $\Delta x$ , and as a function of the number of phase field variables,  $N$ . Table 2 shows the computing time and the convergence factor averaged over ten time steps of simulations on a  $[0, 1] \times [0, 1]$  domain with  $\Delta x = 1/64$ ,  $1/128$ ,  $1/256$  and  $1/512$ , for  $s = 1/(6.4)^2$ ,  $\Delta t = 0.1$  and  $N = 10$ . It can be seen that the convergence properties of multigrid solver are independent of  $\Delta x$ ; the computing time scales linearly with the number of spatial unknowns. This is confirmed by both Fig. 1 and Figs. 2. In Fig. 1, the relative computing time per time step is plotted during the first ten time steps of the simulations, for the different numbers of spatial unknowns averaged. The computing times of the simulation for  $\Delta x = 1/64$  are chosen as reference. The sharp interfaces of the initial Voronoi tessellation cause a large initial residual, which increases for larger systems. Therefore, the initial time step takes relatively more computing time for larger systems. Figure 2(a) shows the behaviour of the residual norms scaled by the norm of the right-hand side during the first time step, while Fig. 2(b) shows the behaviour during the tenth time step. It can be seen that the multigrid convergence is grid size independent. The larger initial residual for smaller grid sizes is reflected in the

graph of Fig. 2(a).

Table 2

Average computing time and convergence factor for different numbers of spatial unknowns during the ten first time steps of simulations on a  $[0, 1] \times [0, 1]$  domain with  $\Delta x = 1/64, 1/128, 1/256$  and  $1/512$ , for  $s = 1/(6.4)^2$ ,  $\Delta t = 0.1$  and  $N = 10$ .  $V(1, 1)$ -cycles are applied.

$\Delta x$	System size	Computing time (s)	Convergence factor
1/64	$10 \times 64 \times 64$	2.2250	0.3188
1/128	$10 \times 128 \times 128$	9.5110	0.3318
1/256	$10 \times 256 \times 256$	38.5610	0.3366
1/512	$10 \times 512 \times 512$	153.3250	0.3382

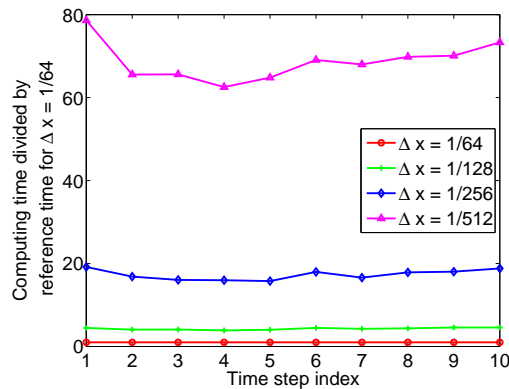


Fig. 1. Computing time per time step for the first 10 time steps of simulations on a  $[0, 1] \times [0, 1]$  domain with  $\Delta x = 1/64, 1/128, 1/256$ , and  $1/512$ , for  $s = 1/(6.4)^2$ ,  $\Delta t = 0.1$  and  $N = 10$ .  $V(1, 1)$ -cycles are applied.

Next, we study the convergence properties and the execution time as a function of  $N$ . In Table 3, the execution time is listed for different values of  $N$ , averaged over the first 10 time steps of simulations on a  $[0, 1] \times [0, 1]$  domain, with  $s = 1/(6.4)^2$ ,  $\Delta x = 1/256$  and  $\Delta t = 0.1$ . It can be seen that the computation work increases slightly faster than  $O(N^2)$ . Note that the smoothers include a direct solver which scales with  $O((N + 1)^3) \simeq O(N^3)$ ; the computing times listed in Table 3 are thus better than expected, probably due to the relatively small values of  $N$  used in our experiments. Figure 3 shows the relative computing time per time step for the first 10 time steps of these simulations. Here, the execution time for  $N = 5$  is taken as a point of reference. Again, the first time step takes more processing time as the initial sharp interfaces cause a larger initial residual. The increase in computing time at the tenth time step for  $N = 20$  is related to the disappearance of a smaller grain. Figure 4 illustrates the evolution of the scaled residual norms during the first and the tenth time step. It can be seen that convergence deteriorates as  $N$  increases. Remark that the convergence behaviour for  $N = 10$  and  $N = 15$  is very similar in Fig. 4(b).

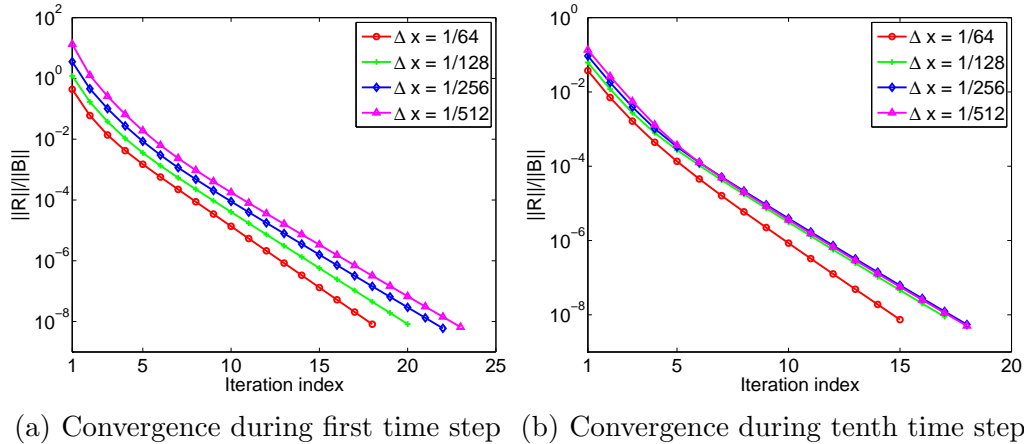


Fig. 2. (a) Convergence rates during the first and (b) the tenth step of simulations on a  $[0, 1] \times [0, 1]$  domain with  $\Delta x = 1/128, 1/256$  and  $1/512$ , for  $s = 1/(6.4)^2$ ,  $\Delta t = 0.1$  and  $N = 10$ . For larger systems, more multigrid cycles are required during the first time step to reduce the larger initial scaled residual.  $V(1, 1)$ -cycles are applied.

Table 3

Computing time and convergence factor for different numbers of phase field variables  $N$  during the tenth time step of simulations on a  $[0, 1] \times [0, 1]$  domain, with  $s = 1/(6.4)^2$ ,  $\Delta x = 1/256$  and  $\Delta t = 0.1$ .  $V(1, 1)$ -cycles are applied.

$N$	System size	Computing time (s)	Convergence factor
5	$5 \times 256 \times 256$	8.0620	0.2457
10	$10 \times 256 \times 256$	38.5610	0.3366
15	$15 \times 256 \times 256$	89.0100	0.3473
20	$20 \times 256 \times 256$	193.3280	0.4342

It should be noted that the initial extra computing work is less significant for smaller values of  $\Delta t$ . Therefore, a solution to the problem of the initial work is to employ a smaller time step at the start of a simulation. Afterwards, a larger time step can be used. Otherwise, it would be impossible to start simulations for very large systems because the initial iterations would take too much time. A possible line of thought is to initialize the structure with smooth interfaces as opposed to a decomposition with sharp interfaces.

### 4.3 Comparison with explicit time stepper

In this section, we will briefly compare the nonlinear multigrid solver with an explicit time stepper. By using an explicit scheme instead of a semi-implicit or a fully implicit scheme, severe restrictions are made with regard to the time step. However, the systems that need to be solved when using an implicit

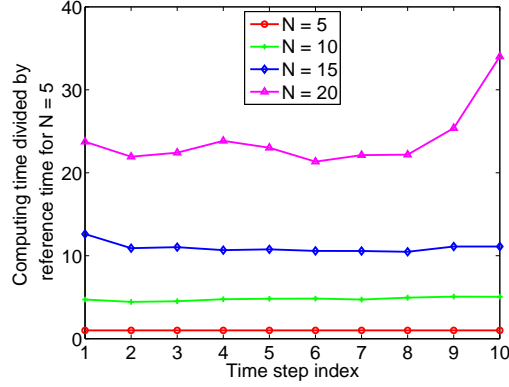
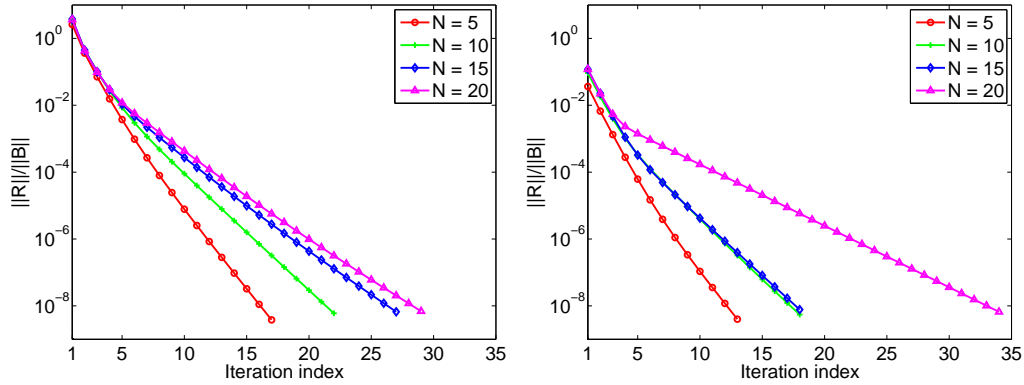


Fig. 3. Computing time during the first ten time steps of simulations on a  $[0, 1] \times [0, 1]$  domain for  $N = 5, 10, 15,$  and  $20$ , with  $s = 1/(6.4)^2$ ,  $\Delta x = 1/256$  and  $\Delta t = 0.1$ .  $V(1, 1)$ -cycles are applied.



(a) Convergence during first time step (b) Convergence during tenth time step

Fig. 4. Convergence behaviour during the first and the tenth time step of simulations on a  $[0, 1] \times [0, 1]$  domain for  $N = 5, 10, 15,$  and  $20$ , with  $s = 1/(6.4)^2$ ,  $\Delta x = 1/256$  and  $\Delta t = 0.1$ .  $V(1, 1)$ -cycles are applied.

scheme are larger and more complex, which costs more computing time.

The employed explicit discretisation scheme is forward Euler. After applying this scheme to model equations (9), the discretised system can be written as:

$$\phi_\alpha^{n+1} = \phi_\alpha^n + \frac{\Delta t}{\omega} \left( \nabla \cdot \frac{\partial a}{\partial (\nabla \phi_\alpha)} - \frac{\partial a}{\partial \phi_\alpha} \right)^n - \frac{\Delta t}{\omega \epsilon^2} \left( \frac{\partial w}{\partial \phi_\alpha} \right)^n - \lambda^{n+1}, \quad (17a)$$

$$\alpha = 1, \dots, N,$$

$$\sum_{\alpha=1}^N \phi_\alpha^{n+1} = 1, \quad (17b)$$

with the spatial discretisation of the gradient free energy component (11) as

in (14). At every time step, system (17) has to be solved for the unknowns  $\phi_\alpha^{n+1}, \phi_\beta^{n+1}, \dots$  and  $\lambda^{n+1}$ . The system matrix is however constant throughout a simulation and its analytically computed inverse is included as such in the algorithm. The solution of the system is thus reduced to a matrix-vector product.

Table 4 contains the computing time required for respectively the explicit and the semi-implicit time stepper to reach the simulation time point  $t = 1.0$  as a function of the chosen time step size. For the chosen parameters  $\Delta x = 1/256$  and  $s = 1/(25.6)^2$ , time step size  $\Delta t = 0.001$  is close to the numerical stability limit for the explicit scheme, while for the semi-implicit scheme,  $\Delta t = 1.0$  is still a stable choice. Simulations show that the semi-implicit solver is more than five times as fast for this particular parameter set. Furthermore, it can be remarked that the stability constraint for the explicit scheme is strongly dependent on the grid size  $\Delta x$ , while the grid size dependent part is treated implicitly by the semi-implicit scheme included in the multigrid solver.

Table 4

Computing time for the explicit and the semi-implicit time stepper to reach  $t = 1.0$ , on a  $[0, 1] \times [0, 1]$  domain, with  $\Delta x = 1/256$  and  $s = 1/(25.6)^2$ , for different time step sizes.

$\Delta t$	Explicit (s)	Semi-implicit (s)
0.001	$4.7435e + 02$	$6.4661e + 03$
0.01	-	$1.5129e + 03$
0.1	-	$2.7978e + 02$
1.0	-	$9.2160e + 01$

## 5 Applications

### 5.1 Evolution of a circular grain in a matrix

For curvature driven growth [8], the evolution of the area of a circular grain of phase  $\alpha$  in a matrix of phase  $\beta$ , illustrated in Fig. 5(a), is described by

$$A_\alpha(t) = A_{\alpha,0} - 2\pi\mu\sigma_{\alpha\beta}t, \quad (18)$$

where  $A_{\alpha,0}$  is the initial grain area, for  $t = 0$ . Simulations are performed on a  $[0, 1] \times [0, 1]$  domain with  $N = 2$ ,  $\omega = 1/\mu = 1$ ,  $\epsilon = 1$ ,  $\sigma_{\alpha\beta} = 1$ ,  $\Delta x = 1/256$ ,  $s = 1/(25.6)^2$  and different time steps  $\Delta t = 0.1, 0.05$  and  $0.025$ . The boundary conditions are periodic. To study the evolution of the microstructure, the grain

area is computed as the sum of  $\phi_\alpha(\Delta x)^2$  over all grid points. Figure 5(b) shows the shrinkage of the grain area during the simulations for the different values of  $\Delta t$ . It can be seen that the results of the simulations correspond well with analytical relation (18). For smaller  $\Delta t$ , the simulations results match the analytical relation better.

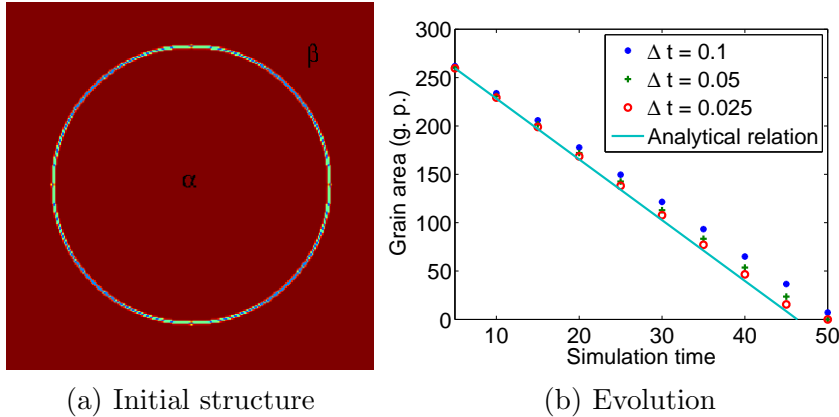


Fig. 5. (a) Circular grain of phase  $\alpha$  within a matrix of phase  $\beta$ . (b) Computed evolution of the area of a circular shaped grain in a matrix in a simulation on a  $[0, 1] \times [0, 1]$  domain with  $N = 2$ ,  $\omega = 1/\mu = 1$ ,  $\epsilon = 1$ ,  $\Delta x = 1/256$ , and  $s = 1/(25.6)^2$ , with different time steps  $\Delta t = 0.1, 0.05$  and  $0.025$ , compared to analytical relation (18).

## 5.2 Grain growth

Another application is the evolution of a polycrystalline microstructure. A grain growth simulation is performed on a  $[0, 1] \times [0, 1]$  domain, with  $N = 15$ ,  $\Delta x = 1/256$ ,  $s = 1/(25.6)^2$ , and isotropic surface energies  $\sigma_{\alpha\beta} = 1.0$ . To initialize the structure, every grid point is assigned to one of the phase field variables  $\phi_\alpha$ , according to a uniform distribution. The domain is thus decomposed into a number of grains that equals the grid size. To visualize the grain growth, we employ a sharp interface representation, where every grid point is assigned to the phase field variable with the highest value at that point. Figures 6(a), 6(b) and 6(c) show the microstructure at  $t = 2$ ,  $t = 20$  and  $t = 160$  respectively.

A second grain growth simulation is performed on a  $[0, 1] \times [0, 1]$  domain, with  $\Delta x = 1/256$ ,  $s = 1/(25.6)^2$  and  $N = 36$ . According to [4], a number of 36 phase field variables is large enough to avoid significant grain coalescence. A small time step  $\Delta t = 0.004$  is chosen to capture all the dynamics of the start of the grain growth. The simulation shows a convergence factor of about 0.09 during these first time steps. Figure 7(a) shows the microstructure at  $t = 0.8$ : the initial grains have already grown into small grains. During grain growth, the smaller grains disappear, while the larger grains grow. This is already

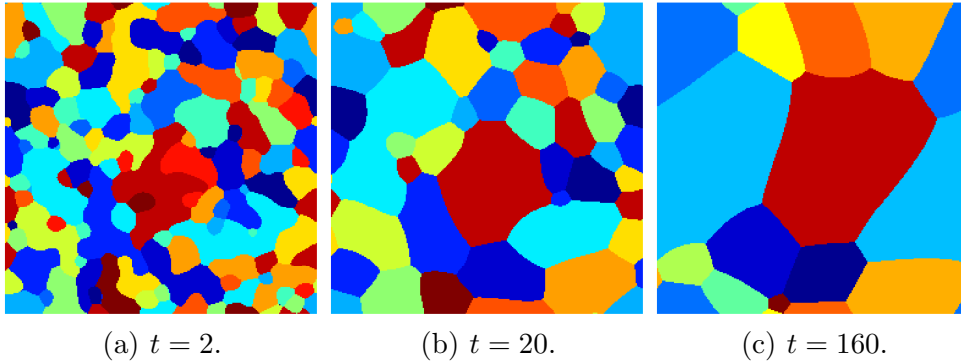


Fig. 6. Results of a simulation on a  $[0, 1] \times [0, 1]$  domain, with  $N = 15$ ,  $\Delta x = 1/256$ ,  $s = 1/(25.6)^2$ , and isotropic  $\sigma_{\alpha\beta} = 1.0$ : (a) Microstructure at  $t = 1$ : several grains are coalescing. (b) Microstructure at  $t = 20$ : some larger grains result from the grain coalescence. (c) Microstructure at  $t = 160$ : only a few grains remain.

clear at  $t = 2.4$ , depicted in Fig. 7(b), where many grains have disappeared in favour of larger grains. From  $t = 2.4$ , a larger time step is chosen:  $\Delta t = 0.1$ . For this time step size, the convergence factor increases to about 0.53.

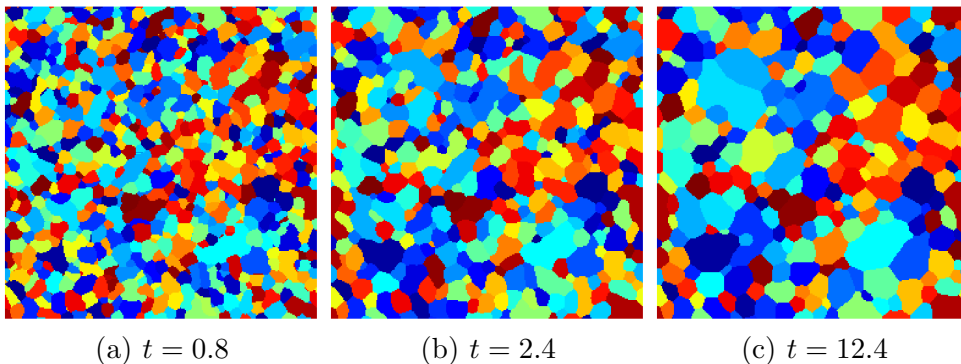


Fig. 7. Results of a simulation on a  $[0, 1] \times [0, 1]$  domain, with  $N = 36$ ,  $\Delta x = 1/256$ ,  $s = 1/(25.6)^2$  and isotropic  $\sigma_{\alpha\beta} = 1.0$  at (a)  $t = 0.8$ , (b)  $t = 2.4$ , and (c)  $t = 12.4$ .

## 6 Conclusion

In this paper, a nonlinear multigrid solver is presented to solve the phase field model of [6,15]. The goal of the solver is to perform grain growth simulations in two dimensions in an efficient way. Until now, only explicit discretisation schemes have been used to solve the phase field model (see e.g. [15]). In this work, a semi-implicit scheme is applied. The semi-implicit discretisation scheme treats the gradient energy part of the model implicitly and the potential part explicitly. It is chosen to be a two-step scheme because of the memory complexity of the model.

To solve the equations describing the discretised system, a nonlinear multigrid

solver based on the FAS scheme is constructed. Experiments with this solver show that the convergence rates are independent of the grid size. However, the convergence properties of the multigrid solver depend on the number of involved phase field variables: for larger numbers, the convergence deteriorates.

To validate the implementation, the results of specific test cases are studied. In the first application, the simulation results for the shrinkage rate of a circular grain within a matrix phase shows very good agreement with the existing analytical relation. In the second application, grain growth simulations are performed.

## Acknowledgements

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its author(s).

This collaboration was made possible through a travel grant awarded by the Fund for Scientific Research – Flanders. The simulations were performed on the computer cluster of the HPC Computing Initiative, funded by the Katholieke Universiteit Leuven Research Council.

## References

- [1] L. Bañas, R. Nürnberg, Finite element approximation of a three dimensional phase field model for void electromigration, *Journal of Scientific Computing* 37 (2) (2008) 202–232.
- [2] A. Brandt, Multi-level adaptive solutions to boundary-value problems, *Mathematics of Computation* 31 (1977) 333–390.
- [3] H. D. Ceniceros, A. M. Roma, A nonstiff, adaptive mesh refinement-based method for the Cahn-Hilliard equation, *Journal of Computational Physics* 225 (2) (2007) 1849 – 1862.
- [4] D. Fan, L.-Q. Chen, Computer simulation of grain growth using a continuum field model, *Acta Materialia* 45 (1997) 611–622.
- [5] W. M. Feng, P. Yu, Z. K. Liu, Q. Du, L.-Q. Chen, Spectral implementation of an adaptive moving mesh method for phase-field equations, *Journal of Computational Physics* 220 (2006) 498–510.

- [6] H. Garcke, B. Nestler, B. Stinner, A diffuse interface model for alloys with multiple components and phases, *SIAM Journal on Applied Mathematics* 64 (2004) 775–799.
- [7] J. Gruber, N. Ma, Y. Wang, A. D. Rollett, G. S. Rohrer, Sparse data structure and algorithm for the phase field method, *Modelling and Simulation in Materials Science and Engineering* 14 (14) (2006) 1189–1195.
- [8] M. Hillert, On the theory of normal and abnormal grain growth, *Acta Metallurgica* 13 (1965) 227–238.
- [9] D. Kay, R. Welford, A multigrid finite element solver for the Cahn-Hilliard equation, *Journal of Computational Physics* 212 (1) (2006) 288–304.
- [10] J. Kim, K. Kang, J. Lowengrub, Conservative multigrid methods for Cahn-Hilliard fluids, *Journal of Computational Physics* 193 (2) (2004) 511–543.
- [11] S. G. Kim, D. I. Kim, W. T. Kim, Y. B. Park, Computer simulations of two-dimensional and three-dimensional ideal grain growth, *Physical Review E* 74 (2006) 061605.
- [12] R. Kornhuber, R. Krause, Robust multigrid methods for vector-valued Allen-Cahn equations with logarithmic free energy, *Computing and Visualization in Science* 9 (2006) 103–116.
- [13] N. Moelans, B. Blanpain, P. Wollants, A phase field model for the simulation of grain growth in materials containing finely dispersed incoherent second-phase particles, *Acta Materialia* 53 (2005) 1771–1781.
- [14] B. Nestler, A 3D parallel simulator for crystal growth and solidification in complex alloy systems, *Journal of Crystal Growth* 275 (2005) e273–e278.
- [15] B. Nestler, H. Garcke, B. Stinner, Multicomponent alloy solidification: Phase-field modeling and simulations, *Physical Review E* 71 (2005) 041609.
- [16] N. Provatas, M. Greenwood, B. Athreya, N. Goldenfeld, J. Dantzig, Multiscale modeling of solidification: Phase-field methods to adaptive mesh refinement, *International Journal of Modern Physics B* 19 (2005) 4525–4565.
- [17] J. Rosam, P. K. Jimack, A. Mullis, A fully implicit, fully adaptive time and space discretisation method for phase-field simulation of binary alloy solidification, *Journal of Computational Physics* 225 (2007) 1271–1287.
- [18] I. Steinbach, F. Pezzolla, B. Nestler, M. Seesselberg, R. Prieler, G. J. Schmitz, J. L. L. Rezende, A phase field concept for multiphase systems, *Physica D* 94 (1996) 135–147.
- [19] Y. Suwa, Y. Saito, H. Onodera, Phase field simulation of grain growth in three dimensional system containing finely dispersed second-phase particles, *Scripta Materialia* 55 (2006) 407–410.
- [20] L. Vanherpe, N. Moelans, B. Blanpain, S. Vandewalle, Bounding box algorithm for three-dimensional phase-field simulations of microstructural evolution in polycrystalline materials, *Physical Review E* 76 (2007) 056702.

- [21] S. Vedantam, B. S. V. Patnaik, Efficient numerical algorithm for multiphase field simulations, *Physical Review E* 73 (2006) 016703.
- [22] S. Wise, J. Kim, J. Lowengrub, Solving the regularized, strongly anisotropic CahnHilliard equation by an adaptive nonlinear multigrid method, *Journal of Computational Physics* 226 (1) (2007) 414–446.