Computational Materials Science 142 (2018) 153–171

Contents lists available at ScienceDirect

Computational Materials Science

journal homepage: www.elsevier.com/locate/commatsci

Quantitative 3D phase field modelling of solidification using next-generation adaptive mesh refinement

Michael Greenwood [a,f,*], K.N. Shampur [b], Nana Ofori-Opoku [c,d], Tatu Pinomaa [e,b], Lei Wang [b,g], Sebastian Gurevich [b], Nikolas Provatas [b]

[a] CanmetMATERIALS, Natural Resources Canada, Hamilton, Canada
[b] Department of Physics and Centre for the Physics of Materials, McGill University, Montreal, Canada
[c] Center for Hierarchical Materials Design, Northwestern University, Evanston, USA
[d] Materials Measurement Laboratory, National Institute of Standards and Technology, Gaithersburg, USA
[e] VTT Technical Research Centre of Finland Ltd, Espoo, Finland
[f] Department of Materials Science and Engineering, McMaster University, Hamilton, Canada
[g] Key Laboratory of Space Applied Physics and Chemistry, Ministry of Education, School of Science, Northwestern Polytechnical University, Xi'an, China

ARTICLE INFO

Article history:
Received 1 May 2017
Received in revised form 22 August 2017
Accepted 14 September 2017

Keywords:
Phase field
Adaptive meshing
Large scale simulation
Solidification
Parallel computing

ABSTRACT

Phase field (PF) models are one of the most popular methods for simulating solidification microstructures due to their fundamental connections to the physics of phase transformations. However, these methods are numerically very stiff due to the multiple length scales in a solidifying material, from the nanoscopic solid-liquid interface, to dendritic structures on the order of hundreds of microns. While this problem can be greatly alleviated by thin-interface analytical treatments of the PF equations, additional numerical methods are required to explore experimentally relevant sample sizes and times scales. It was shown about 18 years ago that the use of dynamic adaptive mesh refinement (AMR) can alleviate this problem by exploiting the simple fact that the majority of the solidification kinetics occur at the solid-liquid interface, which scales with a lower dimensionality than the embedding system itself. AMR methods, together with asymptotic analysis, nowadays provide one of the most efficient numerical strategies for self-consistent quantitative PF modelling of solidification microstructure processes. This paper highlights the latest developments in the AMR technique for 3D modelling of solidification using classical phase field equations. This includes a move away from finite element techniques to faster finite differencing through the use of dynamic *mini-meshes* which are each associated with each node of a 3D *Octree* data structure, and distributed MPI parallelism that uses a new communication algorithm to decompose a 3D domain into multiple adaptive meshes that are spawned on separate cores. The numerical technique is discussed, followed by demonstrations of the new AMR algorithm on select benchmark solidification problems, as well as some illustrations of multi-phase modelling using a recently developed multi-order parameter phase field model.

1. Introduction

Solidification microstructures serve as paradigms for pattern formation in many non-equilibrium processes in materials science, biology and chemistry. In materials science, non-equilibrium phase transformations such as solidification, or solid state precipitation, often produce complex dendritic networks whose patterning and thermo-solutal kinetics establishes the microstructure and properties of the material. In the case of solidification, the as-cast dendritic microstructure impacts the macroscale properties of a material even after significant additional thermal-mechanical processing. Dendritic size, shape and distribution have far-reaching effects on the mechanical, electrical and optical properties of the resultant material [1].

An extensive body of research has been compiled over the years on dendritic solidification. One of the most ubiquitous numerical tools for its numerical modelling is the *phase field* approach. The phase-field (PF) methodology has emerged as a highly robust theoretical formalism to study solidification processes [2–8], but is increasingly expanding its scope for morphological evolution processes beyond materials processes. It is a mesoscopic continuum-level formalism, which has its origins in the diffuse interface theory of van der Waals, Cahn-Hilliard [9,10] and the

* Corresponding author at: CanmetMATERIALS, Natural Resources Canada, Hamilton, Canada.
E-mail addresses: michael.greenwood@canada.ca, michael.james.greenwood@gmail.com (M. Greenwood).

https://doi.org/10.1016/j.commatsci.2017.09.029
0927-0256/

Ginzburg-Landau [11] theories of phase transitions. In the method, smooth order parameter field variables, i.e., *phase-fields*, are introduced to represent order changes between phases, and a free energy functional is constructed in terms of these to account for the thermodynamics of the system. Dynamics then follow from dissipative gradient flow minimization of the free energy functional. The method has evolved past its theoretical origins, being used to describe various microstructure evolution phenomena such as solidification of multiple crystal orientations [12–14], solidification of multiple components and phases [15–17], defect-solute interactions [18], elasticity [19,20] and plasticity [21]. Since the introduction of the phase field technique to mainstream materials science around the early 1990s, the complexity of alloy systems modelled using phase field models has progressively increased. Phase field models are now routinely used to study multiple complex alloy systems, some in a quantitative manner through the use of thin-interface corrections in the various solute fields [22].

While phase field models have become the standard platform for dendritic solidification studies, considerable efforts have been devoted into making their predictions quantitive. The challenges here is grounded in the wide range of relevant length scales, from the width of the solid-liquid interface and its capillary length to the radius of curvature of a growing dendrite arm and the diffusion length of impurities or heat. Efforts to overcome these challenges have generally have followed two, often times complimentary, strategies. The first has been the development of *asymptotic* and *thin-interface* treatments of the PF equations, which prescribe how to self-consistently increase the interface thickness in such a way to recover an appropriate sharp-interface limit of solidification [4,23,17]. The second approach has focused on numerical treatments of phase field equations. Two successful methods include adaptive mesh refinement (AMR) methods[3] and GPU acceleration of fixed grid simulations [24,25]. These two methods also have the potential to complement each other and allow for even greater simulation acceleration. AMR algorithms have been successful in allowing for highly efficient and quantitative simulations of dendritic structures. This technique dynamically establishes a higher density of grid points at the interfacial regions. In doing so, computational resources are allotted mainly around the interface regions where the morphological changes occur. AMR was initially introduced to elucidate kinetics of dendritic growth at low undercooling [3]. However, in the years since it has been primarily used to scale up simulations to length scales relevant to practical experiments, without compromising resolution at the scales of the thin interface [26–31]. Early implementations of AMR further increased computational efficiency by using fairly straightforward shared memory parallelism, which increased the speed of simulations through the simultaneous use of multiple cores. These days, most high performance computing centers typically use distributed memory architectures owing to their low cost compared to shared memory clusters. Moreover, distributed memory AMR algorithm can further scale PF simulations to significantly larger sizes than those allowed by shared memory platforms. Unfortunately, implementation of an efficient AMR algorithm on distributed memory architecture is highly non-trivial. This challenge has generally made AMR less accessible as a tool for common use in phase field simulations of complex materials processes.

This paper introduces a next generation 3D adaptive mesh refinement algorithm written for distributed memory architecture using the Message Passing Interface (MPI). The 3D MPI-AMR algorithm is then benchmarked on two phase field models developed for the study of solidification of practical materials, namely, one for solidification of a pure substance in Ref. [32] and the other, a new multi-order parameter binary alloy solidification model developed in Ref. [33]. Since the focus of this paper is the new AMR algorithm, these models are summarized for the interested reader in Appendix A and references listed there. The remainder of the paper is organized as follows. Section 2 introduces the details of the 3D MPI-AMR algorithm and discusses its implementation for PF equations in two-dimensions (2D) and three-dimensions (3D). Section 3 showcases the accuracy of the algorithm by simulating two well-known numerical benchmarks using the phase field models in Appendix A. These include: (i) the problem of thermally controlled dendritic growth, which is known to follow the analytical solvability theory [32], and (ii) testing the convergence of tip speeds and solute segregation in solutal dendrite growth analogously to what was done in Ref. [34]. Section 4 looks at the efficiency and numerical performance of the 3D adaptive mesh, with and without MPI. Finally, Section 5 concludes with some additional illustrations on the application of our new 3D MPI-AMR algorithm on quantitative solidification modelling.

2. Parallel 3D adaptive meshing

Adaptive mesh refinement (AMR) is an automated process of modifying a simulation mesh during the course of a simulation. The mesh dynamically increases the refinement resolution in regions which require higher resolution and decrease the refinement resolution in regions which no longer require such resolution. This affords significant computational efficiency in free boundary problems, which require higher resolution at the interface and lower resolution in the bulk.

An open source project has been developed over approximately the last ten years in Canada, based off the original concept and design of the adaptive mesh refinement (AMR) code designed by Provatas et al. [3,35] using the Finite Element method in 2 dimensions. The code developed around this open source project by Provatas (McGill), Greenwood (CanmetMATERIALS) and Ofori-Opoku (McGill/NU-NIST) [36] has significantly streamlined the original the AMR mesh technique to take advantage of the simplicity of the finite difference and finite volume methods of simulation. The AMR code has been used in numerous applications at experimentally relevant parameters and processing conditions [37–40,28,41,29,42]. These versions were limited to 2-dimensions and shared memory parallelization.

The prohibitive cost of scaling up shared memory (e.g. OpenMP) codes to large 3D domains and a very large number of cores has recently necessitated the development of an AMR algorithm for distributed memory (e.g. message passing interface (MPI)) applications. This has led to a new collaboration between McGill and CanmetMATERIALS to develop a 3D MPI AMR extension to the aforementioned 2D code family. Specifically, CanmetMATERIALS has recently designed an MPI wrapper software module to implement communication of 3D simulation domains, each of which is itself an adaptive 3D data structure of *mini-meshes* within which finite difference and/or finite volume methods are used to locally update PF type equations. The wrapper uses the MPI for handling communication between AMR data structures. Benchmarks of this new 3D AMR code are found to scale excellently up to 2048 processor cores on CanmetMATERIALS Cray™ supercomputer.

This section first describes the core components of the AMR algorithm, focusing on the mesh representation and the method by which the mesh is broken down and adapted. Second the software organization is discussed, focusing on some of the high level algorithm flows and the object design. Thirdly, the parallel communication used in the method is discussed.

2.1. Core components of the AMR algorithm

The AMR method represents a global simulation volume by a dynamically changing mesh that adjusts to resolve volumes of