Computational Mechanics (2018) 61:543–558
https://doi.org/10.1007/s00466-017-1535-8

ORIGINAL PAPER

A parallelized three-dimensional cellular automaton model for grain growth during additive manufacturing

Yanping Lian[1] · Stephen Lin[1] · Wentao Yan[1] · Wing Kam Liu[1] · Gregory J. Wagner[1]

Received: 4 October 2017 / Accepted: 22 December 2017 / Published online: 12 January 2018

Abstract
In this paper, a parallelized 3D cellular automaton computational model is developed to predict grain morphology for solidification of metal during the additive manufacturing process. Solidification phenomena are characterized by highly localized events, such as the nucleation and growth of multiple grains. As a result, parallelization requires careful treatment of load balancing between processors as well as interprocess communication in order to maintain a high parallel efficiency. We give a detailed summary of the formulation of the model, as well as a description of the communication strategies implemented to ensure parallel efficiency. Scaling tests on a representative problem with about half a billion cells demonstrate parallel efficiency of more than 80% on 8 processors and around 50% on 64; loss of efficiency is attributable to load imbalance due to near-surface grain nucleation in this test problem. The model is further demonstrated through an additive manufacturing simulation with resulting grain structures showing reasonable agreement with those observed in experiments.

Keywords Additive manufacturing · Grain structure · Cellular automaton · High performance computing

1 Introduction

Additive manufacturing (AM) technology, sometimes referred to as 3D printing, is the operation of building a 3D object from computer-aided design model by successively adding material layer by layer. This technology allows rapid fabrication, precise geometric control, and flexibility to create or repair without the use of any die or mold, giving the potential to revolutionize the global part manufacturing landscape [21]. Therefore, much effort is being focused on enhancing AM technologies; research areas include understanding metal powder fabrication, influences of process parameters [24], in situ and real time monitoring [27], and the metallurgical and mechanical properties of the final part [1,7] by means of experiment, theory, and computational models. This work is focused on a numerical model to predict the microstructure during the solidification process; this microstructure has a critical influence on the mechanical properties of the resulting material.

✉ Gregory J. Wagner
gregory.wagner@northwestern.edu

[1] Department of Mechanical Engineering, Northwestern University, Evanston, IL 60208, USA

There are two main types of numerical methods, deterministic and stochastic, used to model the complex dendrite growth and grain structures during metal solidification. A typical method of the first type is the phase field method [4,23]. Phase field methods, in which a field variable varies smoothly over a diffuse interface region to demarcate different phases, have been used to model the growth of dendrite structures in pure materials and alloys [1,8]. Another method similar in spirit is the level set method [12,22], which allows a sharp-interface description of the same phenomena, including the detailed treatment of interface conditions. Both of these approaches are valuable in elucidating fine details of solidification structures. However, the small length scales and small numbers of grains attainable using these methods make it difficult to predict microstructures at sizes that can be directly tied to the performance of resulting materials.

Examples of stochastic-based methods include kinetic Monte Carlo (KMC) and Cellular Automaton (CA) methods (although CA methods include deterministic models of growth along with stochastic grain nucleation models). Compared with the methods described above, KMC and CA methods require fewer computer resources and therefore allow simulations of a large numbers of grains within domains at the millimeter scale and beyond. KMC methods represent individual grains by associating each discrete lat-

tice site with a grain, and utilize a Potts model to simulate solidification and grain growth dynamics by modifying cell values at each Monte Carlo step to reduce surface energies between unlike grains. By coupling KMC to a varying temperature field Rodgers et al. have modeled welding [19] and AM [20] processes; the final grain morphologies are in good agreement with observed experimental results. However, the KMC approach as formulated lacks a model to accurately capture the solidification velocity of individual grains. In the CA method, grains are tracked on a discrete grid of cells in which individual cells can be assigned a particular state, similar to the KMC description. Nucleation of new grains is modeled stochastically depending on the local undercooling of cells in the liquid state, while grains grow (effectively "capturing" neighboring cells) according to a dendrite growth model in the form of a physically-based dendrite tip velocity law, which itself is dependent on the local undercooling of each grain site. Like the KMC method, the CA method can be coupled to a time and space-varying temperature field that is either prescribed or obtained from a solution to the energy equation. Because the CA method evolves the grain structure according to a physically meaningful time step, compared with the harder-to-interpret Monte Carlo steps of the KMC method, it is straightforward to couple CA and thermal models together for a given material and process.

The CA method was first developed by Rappaz and Gandin [18], who successfully predicted the columnar-to-equiaxed grain transition during solidification. Soon after, the same authors coupled the CA representation to a finite element simulation of the local temperature field [9,11], which may vary in space and time. The resulting cellular automaton-finite element (CAFE) model has been refined and improved and applied to increasingly complex simulations of grain formation [5,15]. Zinoviev et al. [28] and Zhang et al. [26] have applied CA to model the grain evolution during laser AM in 2D. Dezfoli et al. [7] and Panwisawas et al. [17] utilized a CA model for microstructure prediction and control of metals in 3D laser AM processes, and have demonstrated good qualitative agreement with experimental results.

Although CA has a broad and impactful range of applications for microstructure prediction in traditional casting and advanced AM, the length scale of the problem that CA can reach is limited by the bounding cell size. To accurately model the extension of grains by branching mechanisms, the cell size should be smaller than the length at which the branching mechanisms of the dendrite network take place; these could be on the length scale of the secondary even the tertiary dendrite arm spacing. For simulations of casting aluminum-silicon alloy, Gandin et al. [10] found a cell size $d_{\text{cell}} = 50\,\mu\text{m}$ to be sufficient. However, fine microstructures with dendrite arm spacings as small as a few microns are often observed in AM processes due to the high cooling rate and temperature gradient [25]. For these length scales, a cell size $d_{\text{cell}} < 10\,\mu\text{m}$ may be required for sufficient accuracy, which raises a computational challenge. For example, a 1 cm^3 volume requires 1 billion cubic cells of size 10 μm. It should be noted that the number of cells increases by 2^3 when refining the cell length scale by a factor of 2 for the 3D case. To overcome this difficulty, Gandin et al. [10] have proposed special dynamic allocation algorithms to model solidification grain structures in a representative volume of the casting to save memory size; however, this does not reduce computational burden. Therefore, high-performance parallel computing algorithms for CA models are required to model large scale problems and expedite the simulation process by leveraging parallel computing resources, especially for AM applications.

In this work, a parallelized 3D CA model is proposed using a Message Passing Interface (MPI) system to exchange data between processors. A static decomposition strategy is employed to implement the parallelization. Some issues in the interprocess communications are addressed. The rest of the paper is organized as follows. In Sect. 2, the 3D CA model is introduced in detail including heterogeneous nucleation, dendrite tip growth kinetics, the decentered octahedron growth scheme, and the explicit time integration algorithm. In Sect. 3, an MPI-based parallelism for 3D CA is proposed. In Sect. 4, numerical examples including grain growth under a uniform decreasing temperature field and an AM application are performed to demonstrate the parallel efficiency of the method. Finally, conclusions and perspectives are discussed in Sect. 5.

2 A 3D cellular automaton model

In CA, a material region is discretized by a set of cubic cells as shown in Fig. 1. Each cell may have variables associated with it, such as temperature and a state index associating the cell with a particular grain. In this work, a state index of -1 denotes a cell in the liquid state, while a non-negative integer value denotes a state of solidification with grain information. The two main phenomena in the CA approach are heterogeneous nucleation and grain growth. Models for these two mechanisms are detailed in this section.

2.1 Heterogeneous nucleation

The heterogeneous nucleation model describes the distribution of nucleation sites, the critical undercooling value at which nucleation occurs at each site, and the crystal orientation of newly nucleated grains.

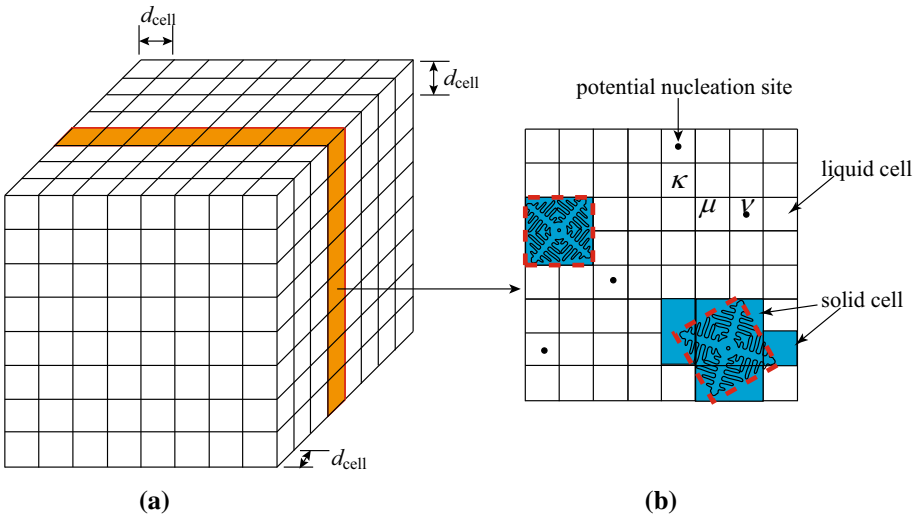

Fig. 1 **a** A 3D cellular automaton network to predict microstructure formation; **b** a cross-sectional view of the network, where ν, μ, and κ are cell IDs discussed in the text, solid dots denotes a nucleation site, and two growing grains are represented over the solid cells in blue. (Color figure online)

2.1.1 Nucleation sites and their critical undercooling distributions

In CA, nucleation may occur both at the surface and in the bulk of the liquid volume. The number density for a population of sites, ϱ, is prescribed as an input parameter, and typically must be fit to experimental measurements. Densities have units of m^{-3} for bulk nucleation sites and m^{-2} for surface nucleation sites. Hereafter, we use ϱ_s and ϱ_v to denote the nucleation density for surface and bulk, respectively. Prior to the beginning of the simulation, the total number of nucleation sites in the bulk and at the surface are calculated as

$$N_v = \varrho_v \cdot V \tag{1}$$

$$N_s = \varrho_s \cdot S \tag{2}$$

where V and S represent the total volume and total surface area, respectively.

At each nucleation site i, nucleation occurs when the undercooling (the liquidus temperature minus the local cell temperature) exceeds a critical value ΔT_i^{crit}. This critical undercooling may vary from site to site, and is assumed to follow a Gaussian distribution. In CA, the values of the mean and standard deviation (ΔT_{max} and ΔT_σ, respectively) of critical undercooling temperatures may differ between surface and bulk nucleation sites, and so subscripts s and v are used to distinguish such quantities as is done for ϱ.

For a given discretization, N_s cells are randomly selected from cells adjacent to the surface of the domain as surface nucleation sites, and N_v cells are selected from all cells in the domain as potential bulk nucleation sites. For each nucleation site created, a random critical undercooling is selected from the corresponding Gaussian distribution function (surface or bulk), and is assigned as the undercooling at which a grain will be nucleated at that cell. It should be noted that a cell may be selected more than once as a possible nucleation site. In this case, only the smallest of the undercooling values randomly selected for this site is assigned.

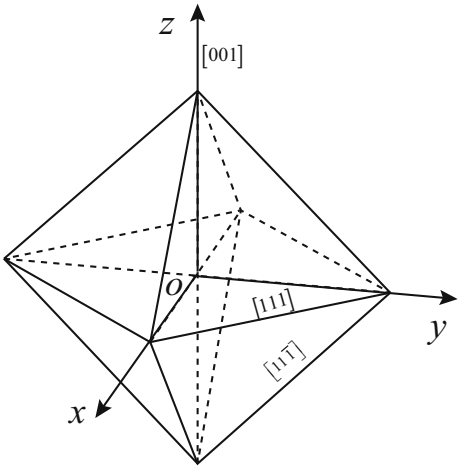

Fig. 2 Regular octahedral envelope

2.1.2 Grain envelope and crystallographic orientation

During the solidification process, a potential nucleation site at a pre-chosen cell ν becomes active (provided it has not already been captured by a neighboring grain) if the undercooling at the center of cell ν exceeds its assigned critical undercooling. Neglecting the incubation time, it is assumed that the dendritic network within cell ν develops as a regular octahedral envelope bound by {111} planes as shown in Fig. 2. The six half-diagonals of the octahedron correspond to the primary dendritic growth directions of the grain [11]. Each grain is characterized by the global orientation of its [001] crystal direction in terms of a set of Euler angles (ϕ_1, ϕ_2, ϕ_3), where $0 \leq \phi_1 \leq 2\pi$, $0 \leq \phi_2 \leq \pi$, and $0 \leq \phi_3 \leq 2\pi$.

Grains that are nucleated either at the surface or in the bulk of the volume are assumed to have a random crystallographic orientation. In order to obtain an isotropically distributed set of random orientations, two auxiliary variables χ_1 and χ_2 are first randomly selected from a uniform distribution over [0,1]. These parameters are then used to generate locations for ϕ_1 and ϕ_2 on the unit sphere:

$$\phi_1 = 2\pi\chi_1 \tag{3}$$

$$\phi_2 = \cos^{-1}(2\chi_2 - 1) \tag{4}$$

It is straightforward to show that this procedure results in a random uniform distribution over the unit sphere by calculating the differential area element over the sphere, dS:

$$\mathrm{d}S = \sin\phi_2 \mathrm{d}\phi_1 \mathrm{d}\phi_2 = -\mathrm{d}\phi_1 \mathrm{d}(\cos\phi_2) = -4\pi \mathrm{d}\chi_1 \mathrm{d}\chi_2 \tag{5}$$

Finally, the third Euler angle, ϕ_3, is selected from a uniform distribution over the range $[0, \pi/2]$ with the symmetry of the octahedral dendrite structure assuring that all unique orientations of the grain are equally likely.

2.2 Growth algorithm for an active grain

The grain growth is governed by the dendrite tip growth kinetics. During the grain propagation process, a decentered octahedron method is used to determine a new envelope; each cell captured by this envelope spawns a new growing octahedron with a crystal orientation inherited from the parent nucleus. The capture of neighboring cells by the growing dendritic network is then equivalent to the overall growth of an octahedral envelope, at least until the envelope impinges on other grains or boundaries.

2.2.1 Dendrite tip growth kinetics

The dendrite tip growth rate, v, is related to its undercooling, ΔT. As presented in [18], the undercooling includes four contributions:

$$\Delta T = \Delta T_c + \Delta T_t + \Delta T_k + \Delta T_r \tag{6}$$

where ΔT_c, ΔT_t, ΔT_k and ΔT_r denote the undercooling contributions associated with solute diffusion, thermal diffusion, attachment kinetics and solid–liquid interface curvature, respectively. Based on dendrite tip kinetics models, such as the KGT model [13] or LGK model [6,14], the relationship between dendrite tip velocity and undercooling ΔT can be determined based on the predominant mechanisms. In order to speed up calculations, such a relationship is usually fitted with a polynomial approximation.

Various polynomial approximations may be fitted with different coefficient values for a specific application. For example, for the Aluminum-Silicon alloy under normal solidification conditions, Gandin and Rappaz [9] proposed a polynomial formulation:

$$v(\Delta T) = \eta_2 \cdot \Delta T^2 + \eta_3 \cdot \Delta T^3 \tag{7}$$

where η_2 and η_3 are coefficients with units of $\mathrm{m}/(\mathrm{s}\cdot\mathrm{K}^2)$ and $\mathrm{m}/(\mathrm{s}\cdot\mathrm{K}^3)$, respectively. For Ti–6Al–4V alloy with high thermal gradient solidification conditions, as in AM processes, Dezfoli et al. [7] have used another polynomial formula given as:

$$v(\Delta T) = \eta_1 \cdot \Delta T + \eta_2 \cdot \Delta T^2 \tag{8}$$

where η_1 is a coefficient with units of $\mathrm{m}/(\mathrm{s}\,\mathrm{K})$.

2.2.2 Grain growth

Since the six half-diagonals of the octahedral envelope represent the ⟨100⟩ crystallographic directions along which the grain grows fastest, growth is simulated by extending these half-diagonals based on the dendrite tip velocity $v(\Delta T)$. Here ΔT is the local undercooling at the center of the cell that owns the envelope. As time proceeds, the envelope grows and eventually engulfs neighboring cells to propagate the grain.

The state of a cell is defined by a set of variables including the state index, grain orientation, envelope center (i.e., the growth center) and envelope size; these are denoted as I, (ϕ_1, ϕ_2, ϕ_3), $\boldsymbol{C}$ and L, respectively. Note that I is an integer associating a cell with a particular grain orientation, so that cells with the same value of I have the same orientation and are considered part of the same grain. For a regular octahedral envelope, the size is defined as the distance from its envelope center to the center of any of its {111} faces. Each solidified cell has a unique octahedral envelope, with a given center and size; the full grain is then given by the union of envelopes with the same orientation and state index.

To detail the grain growth and the capture process, we take two neighboring cubic cells as an example. As shown in Fig. 1b, the two cells are labeled ν and μ with a cell spacing of d_{cell}. The initial state indices of both cells are set as -1 (i.e., $I_\nu = I_\mu = -1$) indicating the liquid state, and the cell ν is pre-chosen as a nucleation site with a critical undercooling $\Delta T_\nu^{\mathrm{crit}}$. At a particular time t_ν, the undercooling $\Delta T_\nu(t_\nu)$ at the center of cell ν becomes larger than $\Delta T_\nu^{\mathrm{crit}}$. At that time, a grain is nucleated at the center of cell ν with a randomly generated orientation as described in Sect. 2.1.2. The state index of cell ν is then set to a unique integer value. Meanwhile, an active envelope associated with cell ν is defined with size $L_\nu(t_\nu) = 0$ and center $\boldsymbol{C}_\nu$ at the cell center. Approximating the temperature as uniform with each cell, the size of the octahedral envelope at a time $t > t_\nu$ is given by

$$L_v(t) = \frac{1}{\sqrt{3}} \int_{t_v}^{t} v\left(\Delta T_v(\tau)\right) \mathrm{d}\tau \tag{9}$$

where v is the dendrite tip velocity (i.e., of the ⟨100⟩ directions or diagonals of the octahedron). As determined by Eq. (7) or (8), the dendrite tip velocity is a function of the local undercooling at the center of the owning cell. Here it is useful again to point out the distinction between the octahedral envelope for a given cell, defined by six equally-sized diagonals, and the grain, which is formed by a union of octahedral envelopes and spans many cells. Thus, the growth velocity of the grain is non-uniform in space and time, and varies as a function of the local temperature throughout the grain.

At some time $t_\mu > t_\nu$, the envelope associated with cell ν engulfs the center of neighboring cell μ. Now cell μ is captured by cell ν and grain information associated with cell ν is used to initialize a new grain envelope associated with cell μ. The state index of cell μ is set as equal to I_ν, while $\boldsymbol{C}_\mu$ and L_μ are calculated relative to those of the parent envelope through a decentered octahedron growth algorithm (described in Sect. 2.2.3). A grain envelope stops growing and is "deactivated" when all of the cells neighboring its owning cells have been captured. In this work, the neighbors of a cell are defined as all face, edge, and corner neighbors, so that a typical interior cell in 3D has 26 neighbors.

2.2.3 A decentered octahedron growth algorithm

When a cell is captured by a growing grain envelope, a new octahedral envelope is created at that cell, the size and center location of which are computed according to a decentered octahedron algorithm [11]. Through this approach, three conditions are met: (1) the new octahedron fits fully inside the old one; (2) a corner of the new octahedron coincides with the corner of the old one that is closest to the captured cell; and (3) the maximum size of the new octahedron is fixed at a value proportional to d_{cell} according to the formulation introduced by Gandin and Rappaz [11]. Details of the implementation of this model following their outline are presented below.

Consider for example the two cells μ and κ shown in Fig. 1b. Assume that the envelope associated with cell μ, with a set of Euler angles (ϕ_1, ϕ_2, ϕ_3), is growing and approaching cell κ. In Fig. 3, the open circle μ and the cross C_μ are used to denote the cell center and envelope center associated with cell μ.

The size of the envelope associated with cell μ at time t is given by

$$L_\mu(t) = L_\mu(t_\mu) + \frac{1}{\sqrt{3}} \int_{t_\mu}^{t} v\left(\Delta T_\mu(\tau)\right) \cdot d\tau \tag{10}$$

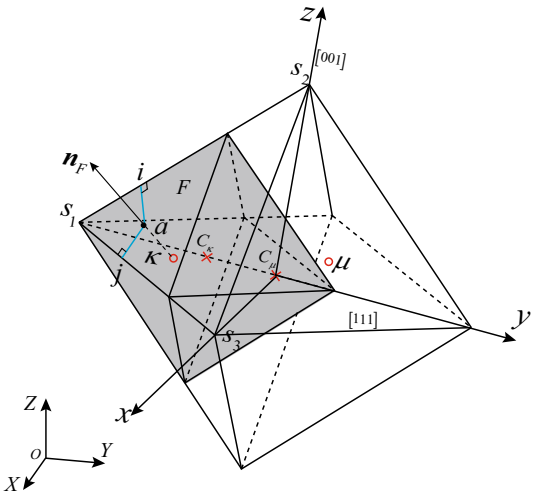

Fig. 3 Schematic illustration of the octahedron envelope truncation

where $L_\mu(t_\mu)$ represents the initial size of the envelope at cell μ, created at the time of its capture t_μ. At each time step, it is determined whether cell κ has been captured by the envelope at cell μ, and if it is, a new envelope at cell κ is initialized using the procedure below:

1. Transform the coordinate of cell center κ to a local coordinate system associated with the envelope:

$$\boldsymbol{x}^\kappa = \mathbf{M}^{-1} \cdot (\boldsymbol{X}^\kappa - \boldsymbol{X}^{C_\mu}) \tag{11}$$

   where uppercase $\boldsymbol{X}$ and lowercase $\boldsymbol{x}$ represent the coordinates of the cell center in global and local coordinate systems, respectively, as shown in Fig. 3. Matrix $\mathbf{M}$ is the coordinate rotation matrix given by the Euler angles (ϕ_1, ϕ_2, ϕ_3).
2. Calculate the octant and nearest octahedral face for the captured cell. In the local coordinate system, space is decomposed into eight octants separated by the local coordinate planes. The signs of the components of $\boldsymbol{x}^\kappa$ then determine in which octant cell center κ is located. The corresponding face of the envelope is denoted by F with its normal direction, $\boldsymbol{n}_F$, denoted by Miller indices $[h, k, l]$ (where all indices are $+1$ or -1).
3. Calculate the distance between κ and the face with normal vector of $\boldsymbol{n}_F$ as follows

$$d = \frac{1}{\sqrt{3}} \left(h x_1^\kappa + k x_2^\kappa + l x_3^\kappa - \lambda_\mu\right) \tag{12}$$

   where $\lambda_\mu = \sqrt{3} L_\mu(t)$ is the half-diagonal length of the envelope associated with cell μ. If $d < 0$, then the cell center κ is engulfed by the face F; otherwise, the cell center is outside of the face F. If the cell κ is captured,

continue with the following steps to obtain the new envelope size and center position associated with cell κ.

3. Calculate the projection of the cell center κ onto face F and denote it as point a:

$$\boldsymbol{x}^a = \boldsymbol{x}^\kappa + \frac{1}{\sqrt{3}}|d|\boldsymbol{n}_F \tag{13}$$

4. Determine the closest corner on face F to a. In Fig. 3, the nearest corner is labeled s_1, with the other two corners of face F denoted by s_2 and s_3, respectively.
5. Calculate the projections of point a onto the edges s_1s_2 and s_1s_3; denote the projected points as i and j, respectively, as follows:

$$\overrightarrow{is_1} = \frac{(\boldsymbol{x}^{s_2} - \boldsymbol{x}^{s_1}) \cdot (\boldsymbol{x}^a - \boldsymbol{x}^{s_1})}{\|(\boldsymbol{x}^{s_2} - \boldsymbol{x}^{s_1})\|^2}(\boldsymbol{x}^{s_2} - \boldsymbol{x}^{s_1}) \tag{14}$$

$$\overrightarrow{js_1} = \frac{(\boldsymbol{x}^{s_3} - \boldsymbol{x}^{s_1}) \cdot (\boldsymbol{x}^a - \boldsymbol{x}^{s_1})}{\|(\boldsymbol{x}^{s_3} - \boldsymbol{x}^{s_1})\|^2}(\boldsymbol{x}^{s_3} - \boldsymbol{x}^{s_1}) \tag{15}$$

6. Determine the new envelope size by first defining two auxiliary variables:

$$L_{12} = \frac{1}{2}\left(\text{Min}\left[is_1, \sqrt{3}d_{\text{cell}}\right] + \text{Min}\left[is_2, \sqrt{3}d_{\text{cell}}\right]\right) \tag{16}$$

$$L_{13} = \frac{1}{2}\left(\text{Min}\left[js_1, \sqrt{3}d_{\text{cell}}\right] + \text{Min}\left[js_3, \sqrt{3}d_{\text{cell}}\right]\right) \tag{17}$$

Here, the lengths of the new envelope edges along the s_1s_2 and s_1s_3 directions are truncated to no more than $\sqrt{3}d_{\text{cell}}$. The new envelope size is then computed as:

$$L_\kappa = \sqrt{2/3}\text{Max}\left[L_{12}, L_{13}\right] \tag{18}$$

The length of the half-diagonal of the new envelope is $\lambda_\kappa = \sqrt{3}L_\mu^t$. Note that by this procedure, if is_1, is_2, js_1, and js_2 are all less than $\sqrt{3}d_{\text{cell}}$ (the maximum spacing between neighboring cell centers), the new envelope has the same size as the original.

7. Based on the new envelope size, calculate its center such that the corner of the envelope corresponds with s_1:

$$\boldsymbol{x}^{C_\kappa} = (\lambda_\mu - \lambda_\kappa)\frac{\boldsymbol{n}_{os_1}}{\|\boldsymbol{n}_{os_1}\|} \tag{19}$$

where $\boldsymbol{x}^{C_\kappa}$ is the coordinate of the new grain envelope associated with cell μ, and $\boldsymbol{n}_{os_1}$ denotes the vector connecting point o to point s_1. Through the coordinate rotation matrix $\mathbf{M}$, the global coordinates of the new envelope center can be obtained as

$$\boldsymbol{X}^{C_\kappa} = \boldsymbol{X}^{C_\mu} + \mathbf{M} \cdot \boldsymbol{x}^{C_\kappa} \tag{20}$$

As a result of this procedure a new envelope is created with envelope center position $\boldsymbol{X}^{C_\kappa}$ and envelope size L_κ. Meanwhile, the orientation of the envelope is identical to that of the parent envelope, and $I_\kappa = I_\mu$.

2.2.4 Time integration scheme

An explicit time integration with variable time step size is applied to integrate grain envelope growth [e.g., Eqs. (9), (10)] for all active grain envelopes. At time t_k, where the subscript k denotes the kth time step, the time step is determined as

$$\delta t_k = \xi \cdot \min_\mu\left(\frac{d_{\text{cell}}}{v\left(\triangle T_\mu(t_k)\right)}\right) \tag{21}$$

where ξ is a time step factor within a range of $(0, 1]$, and $v\left(\triangle T_\mu(t_k)\right)$ is the dendrite tip velocity at cell μ; the minimization is taken over all cells μ with an active growing envelope. Since d_{cell} is a constant, Eq. (21) implies that the time step is determined by the active envelope with the largest growth velocity. At time of $t_{k+1} = t_k + \delta t_k$, the envelope size is then updated for each grain μ as

$$L_\mu(t_{k+1}) = L_\mu(t_k) + \frac{1}{\sqrt{3}}v\left(\Delta T_\mu(t_k)\right)\delta t_k \tag{22}$$

2.3 Numerical implementation

A summary of the CA procedure at a single time step is presented here:

1. Loop over all remaining nucleation sites whose cells have not yet been captured by other grains. At any site whose undercooling exceeds the critical undercooling assigned to that site, nucleate a new active grain envelope with a random orientation.
2. Loop over all active envelopes to determine the current time step size according to Eq. (21).
3. Loop over all the active envelopes to grow the grains:
   (a) Calculate the updated envelope size according to Eq. (22);
   (b) Capture the neighboring cells according to the method described in Sect. 2.2.3.

2.4 Temperature field

As discussed in Sects. 2.1 and 2.2, the heterogeneous nucleation and dendrite tip growth algorithms are functions of the local temperature of the cells. In the original CAFE model [9], a finite element method was used to solve a thermal conduction equation to provide a temperature field at each cell center. Alternatively, such a temperature field may be

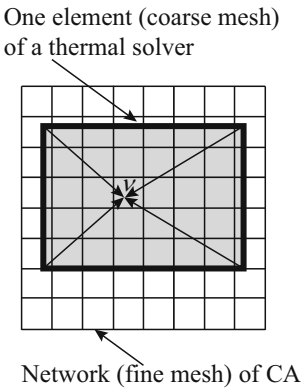

Fig. 4 Schematic illustration of the interpolation method for the 2D case. The temperature at the center of CA cell ν is interpolated from a rectangular element in the thermal solution mesh. The interpolation is computed from the four nodal temperature values via the bilinear shape functions associated with the element

predefined or provided by other numerical methods such as a mesh-based finite volume method (FVM) or a meshfree method.

If a thermal solver is applied, a mesh or grid size coarser then the CA cell network, but overlapping it, is usually adopted to save computational cost. Each CA cell's temperature can be interpolated from the coarse mesh element in which the cell center $\boldsymbol{C}_\nu$ is located as shown in Fig. 4. The one-way coupling method proposed in Ref. [10] is used in this work.

3 A parallelized 3D CA model based on a message-passing paradigm

To accelerate computations and make microstructure simulations feasible on length scales reaching centimeters and beyond, a parallelized C++/MPI implementation of the 3D CA model described in Sect. 2 has been developed. As shown in Sect. 2, most of the computational cost in the CA model is incurred in calculating which cells are captured by growing grain envelopes, and in creating new envelopes. The amount of calculation during this capture process is related to the total number of active grain envelopes distributed throughout the network, and the number of liquid cells surrounding them. In this section, a static domain decomposition and parallel communication scheme is proposed to distribute data and computational burden across multiple processors.

3.1 Domain decomposition strategy

An optimal domain decomposition gives each processor the same amount of computational load while minimizing the interface between subdomains, so that interprocess communication cost is reduced and parallel efficiency enhanced. However, the ideal domain decomposition is problem-dependent, and computational load may be non-uniformly distributed in space. In CA, the memory requirements per processor are related to the number of cells located on each processor, but computational load depends on the distribution of active and growing grain envelopes throughout the domain. To balance memory load in CA, a static domain decomposition strategy is proposed in which the CA cells are evenly divided among processors such that each processor owns approximately same number of cells. For convenient interprocess communications, a Cartesian virtual processor topology is used. Decomposition may be performed along one, two, or all three Cartesian directions; flexibility in selecting a decomposition increases the ability for computational load to be balanced among processors for a given problem geometry.

To illustrate the details of the domain decomposition strategy, we consider a cubical body with eight processors as an example as shown in Fig. 5. A typical decomposition strategy is as shown in Fig. 5a, where the domain is decomposed into two parts along each dimension. Each subdomain is assigned to a unique processor, and the mapping of subdomains to processors is facilitated through the use of a virtual Cartesian topology in MPI [16]. The Cartesian virtual topology matches the physical topology as shown by the dashed arrow lines in Fig. 5b.

In order to grow grains across subdomain interfaces, the state of cells adjacent to the interfaces must be communicated to neighboring processors. To implement this communication, a one-cell-thick ghost layer around each processor subdomain is created. For example, Fig. 5c shows a Y cross-section of a domain with 16 cells on the plane, indexed starting from 0, decomposed among four processors. The corresponding ghost layer cells are shown in Fig. 5d in orange. These ghosting layers provide a convenient way to perform operations, such as grain propagation, involving cells that are not on the current processor. For clarity of the following description, we define the cells in blue within each subdomain as local cells on the current processor as shown in Fig. 5d, while the cells in the ghost layer are referred to as ghost cells.

3.2 Interprocess communications

As listed in Sect. 2.3, there are three major computational steps within each time increment in the CA model. Using the static domain decomposition parallelism, there are four interprocess communications during each time step: global time step size calculation, cross-capture of cells, ghost cell index update, and maintenance of a global orientation list.

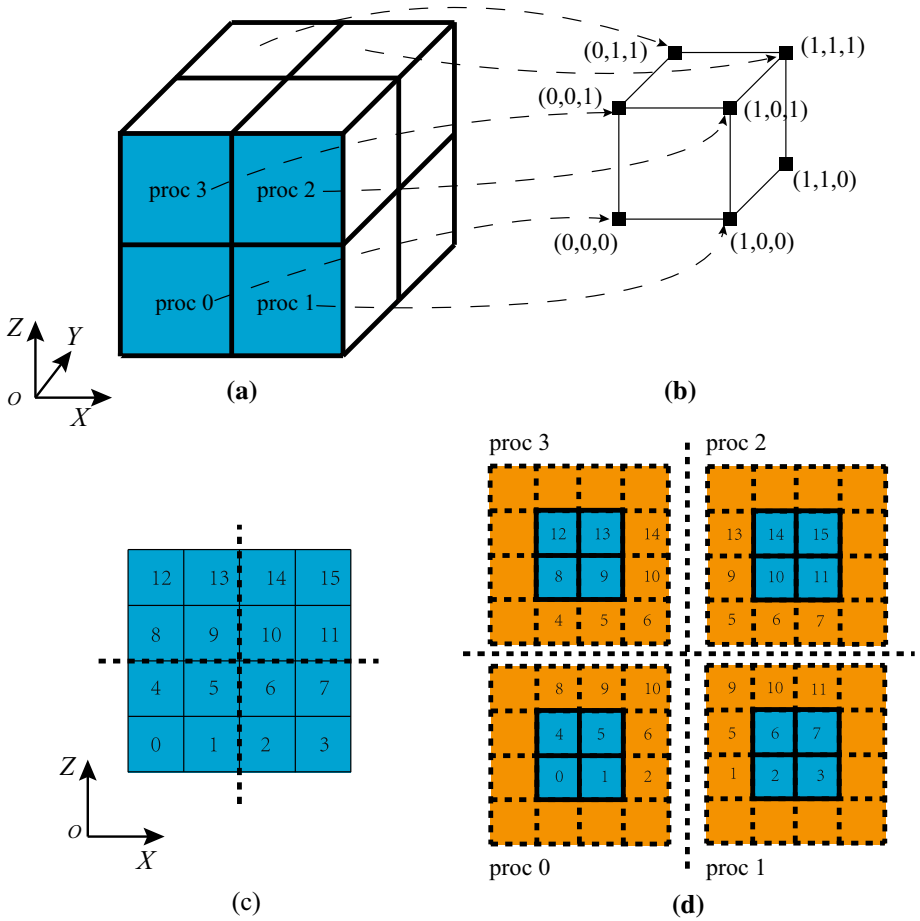

Fig. 5 Illustration of domain decomposition and Cartesian virtual topology: **a** domain decomposition with 8 subdomains; **b** Cartesian coordinates for $2 \times 2 \times 2$ 3D topology, where the solid squares represent the processes and the edges connect processes that communicate with each other; **c** mesh on a typical Y cross-section, assuming 16 cells in the plane; **d** decomposed mesh across 4 processors, where cell indices correspond to those in subfigure (**c**). Blue cells are locally owned, while orange cells indicate ghost layers to facilitate communication. (Color figure online)

3.2.1 Determination of the global time step size

The time step size computed in Eq. (21) is a global minimum over all grain envelopes on all processors. This time step is coordinated among processors with a simple reduction operation (e.g., MPI_Allreduce).

3.2.2 Capturing neighboring cells

During the cell-capturing stage of each time step, each processor loops only over the active envelopes associated with its local cells. For an active envelope associated with a cell that is located at the interface of subdomains, one or more of the neighboring cells will be a ghost cell. In our implementation, if a ghost cell is captured by an active envelope, the new envelope is created by the processor that owns that cell. For example, referring to Fig. 5d, if cell 6 is captured by the envelope associated with cell 5 on processor 0, then the envelope information (size, orientation, center, and captured cell ID) from cell 5 must be communicated to processor 1; a new envelope associated with cell 6 is then created on processor 1 (unless cell 6 has already been captured by a different envelope at the same time step). In this case, the information related to the capturing grain is wrapped and sent to the corresponding processor to complete the capture of neighboring cells. Note that, because a grain envelope may capture any of the 26 neighbors of its owning cell in 3D, care must be taken to communicate not only to immediately adjacent (i.e., face-sharing) processors in the Cartesian topology, but also to processors that own ghost cells along edges and corners. In our implementation, this is handled through multiple communication steps along Cartesian direction until all processors have received data corresponding to local cells.

3.2.3 Updating ghost cell state index

After the cell capture step is complete, if a cell adjacent to a processor interface has been captured, its ghost cell "image" on neighboring processors must be updated. Therefore, another communication operation is required to update the state indices of ghost layers of cells. At the end of this communication step, all processors have updated information about the state of local and ghost cells.

Table 1 Nucleation site parameters used in example simulations

Case	$\Delta T_{s,\max}$ (°C)	$\Delta T_{s,\sigma}$ (°C)	ϱ_s (m^{-2})	$\Delta T_{v,\max}$ (°C)	$\Delta T_{v,\sigma}$ (°C)	ϱ_v (m^{-3})	$\dot{T}$ (°C/s)
I	0.5	0.1	2.5×10^8	8.0	0.1	5.5×10^{10}	−2.3
II	0.5	0.1	2.5×10^8	10.5	0.1	5.5×10^{10}	−2.3

The subscripts v and s denote parameters associated with bulk and surface nucleation sites, respectively

3.2.4 Maintaining a global orientation list

The state index of each cell is an integer that associates the cell with a unique crystallographic orientation (i.e., a set of Euler angles). Cells with the same state index are considered to be part of the same grain. In order to ensure global uniqueness of each index, and to correctly associate each index with a particular orientation, every processor must have a copy of the global list of indices and orientations.

At each time step, new grains (with randomly chosen orientations) may be nucleated locally on any processor. After grains are nucleated, new grain information is gathered globally, and a list of globally unique orientations and state indices is generated. This list is passed back to individual processors, where newly nucleated grains are updated with unique indices.

4 Numerical examples

Two sets of examples are conducted to demonstrate the method and implementation. In the first, a uniform temperature field with a constant cooling rate is applied to show the influence of cell size and time step size while demonstrating the parallel efficiency of the proposed parallelized method. The second set of examples demonstrates an application to the AM process, where a finite volume method is used to provide a thermal field.

4.1 Grain growth in a uniform temperature field

In this section, we conduct microstructure prediction simulations for an Aluminum-7 wt% silicon (Al-7 wt%Si) alloy under a uniform temperature field with a constant cooling rate, $\dot{T}$. Parameters for these examples, including the growth kinetics law, are taken from Refs. [9,18] and are summarized in Tables 1 and 2. It should be noted that the only difference between Case I and Case II in Table 1 is the mean undercooling, $\Delta T_{v,\max}$, for the bulk nucleation; this difference leads to different final grain morphologies.

4.1.1 Mesh size and time step size influence

To study the solution dependence on mesh size and time step size, a cubic domain of $5 \times 5 \times 3$ mm^3 is chosen to represent a 3D Al-7 wt%Si ingot cast in a mold. As reported in [11,18], the secondary dendrite arm spacing for this type of material by casting is on the order of 15 μm.

Table 2 Growth kinetics for Al-7 wt%Si alloy

$v(\Delta T) = \eta_2 \cdot \Delta T^2 + \eta_3 \cdot \Delta T^3$

$\eta_2 = 2.90 \times 10^{-6}$ m/(s K^2) $\quad \eta_3 = 1.49 \times 10^{-6}$ m/(s K^3)

First, two simulations corresponding to the two sets of parameters listed in Table 1 are conducted with the same cell size, $d_{\text{cell}} = 10$ μm and time step factor, $\xi = 0.2$. The final grain morphologies of both cases from the 3D CA model are presented in Fig. 6, where the colors represent the unique ID and orientations for different grains. From Fig. 6, one can see that columnar growth from the surfaces is predicted in both cases, and inspection of our results shows that grains that have ⟨100⟩ crystal orientations perpendicular to the surface are most prevalent because of the relatively faster growth rates in those directions. Since $\Delta T_{v,\max}$ for case II is larger than that of case I, bulk nucleation ahead of the growing columnar grains is postponed in case II, and those columnar grains approach the center of the ingot in case II before a columnar-to-equiaxed transition (CET) can take place. By calculating the expected length that the columnar grains attain before bulk nucleation can occur, we can estimate the column length at the CET in terms of the parameters listed in Tables 1 and 2 as

$$l_{\text{CET}} = \frac{1}{\dot{T}}\left[\frac{\eta_2}{3}\left(\Delta T_{v,\max}^3 - \Delta T_{s,\max}^3\right) + \frac{\eta_3}{4}\left(\Delta T_{v,\max}^4 - \Delta T_{s,\max}^4\right)\right] \tag{23}$$

which gives 0.88 and 2.5 mm for cases I and II, respectively. Because the half-thickness of the domain in the z direction is only 1.5 mm, a CET expected to be present in case I but not in case II. This prediction, along with the approximate position of the CET in case I, is verified by the results shown in Fig. 6.

Case I is used for refined simulations to illustrate the influence of mesh size and time step size. Keeping the time step factor as $\xi = 0.2$, additional cases with cell sizes of $d_{\text{cell}} = 40$, 20 and 5 μm are conducted. Note that even when model parameters in Tables 1 and 2 are kept constant across simulations, results are not expected to be identical for dif-

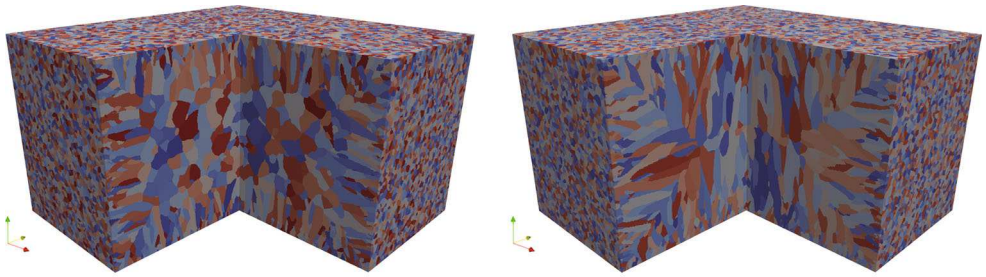

Fig. 6 Grain structure of an Al-7 wt%Si specimen result from 3D CA with cell size $d_{\text{cell}} = 10\,\mu\text{m}$, time step factor $\xi = 0.2$ and other parameters listed in Table 1: **a** for case I with $\Delta T_{v,\max} = 8.0\,^{\circ}\text{C}$, **b** for case II with $\Delta T_{v,\max} = 10.5^{\circ}\text{C}$. Here a corner is cut off to disclose the inner grain morphology

Table 3 Final number of grains versus cell size for spatial refinement study

Cell size (μm)	40	20	10	5
Grain number	24,227	27,970	29,122	29,447

Table 4 Final number of grains versus time step size for temporal refinement study

Step size factor	0.8	0.4	0.2	0.1
Grain number	29,143	29,127	29,122	29,119

ferent values of d_{cell} because of the randomness of nucleation site distribution; results should, however, be similar across simulations and show equivalent morphologies, grain sizes, and grain aspect ratios and orientations. The final number of grains for each case is compared in Table 3. It is seen that the number of grains increases as the cell size decreases. This is mainly attributable to the initial random distribution of nucleation sites in the domain, for which only a single site per cell (that with the smallest critical undercooling for nucleation) is retained; a coarser mesh leads to a larger number of cells with multiple nucleation sites. Effectively, a finer mesh leads to a larger number of small grains. This is evident in Fig. 7a, which shows a histogram of grain volumes (quantified by the radius of the sphere with equal volume) for different choices of cell size. There is a shift toward smaller grain size as the mesh is refined, consistent with the trend in Table 3. A comparison of grain structures on the XY plane for different cell sizes is shown in Fig. 8. All of the results show very similar grain patterns for this case, although smaller cell sizes clearly lead to smoother and higher-resolution representations of grain boundaries.

In a second study, additional cases with time step factors of $\xi = 0.8$, 0.4, and 0.1 are carried out while keeping the cell size fixed at 10 μm. Although the nucleation site distribution is identical across these cases because there is no change in the mesh, a different time step size may influence the competition between neighboring grains and the order in which cells are captured; as a result the final grain structures are expected to be different in their details, though not necessarily in their basic structure. The total number of grains versus time step factor is presented in Table 4; differences are small, but the clear trend is a slight decrease in number as the time step factor decreases. This might be attributable to the ability of a smaller time step to more accurately capture the nonlinear dendrite growth rate in this problem; dendrite envelope size increases slightly more over a given time interval for the finer time resolution, capturing more neighboring cells before new grains can be nucleated there. The comparison of grain size distributions is plotted in Fig. 7b, from which it is can be seen that a time step size factor less than or equal 0.4 has little influence on the grain size distribution. In Fig. 9, the top subfigures are cross-sectional views of the grain structures in the YZ plane and the bottom subfigures are close-ups corresponding to the highlighted rectangular regions. (It should be noted that the same grain for different cases may have different colors; this is because we randomize the numerical values of the state indices to maximize contrast among adjacent grains.) In Fig. 9, differences in details of the grain structure can be observed between all four figures. Some of these differences (e.g., the varying shapes of the 3-grain junction highlighted in the insets) are decreased for the smallest time steps, but other discrepancies remain. These differences are attributed to occurrences in which a cell is captured by two or more different grain envelopes at the same time step, and must be assigned to just one of them; in our algorithm, the assignment is simply to the first grain treated in the loop. These multi-capture events occur less frequently as the time step is reduced, but do not go away completely; each such event may lead to a discrepancy that compounds over time, and time step convergence in the familiar sense cannot be observed. Because the overall quality of the solution is not affected by these differences, we accept that the solution may be time-step dependent, and select a time step factor $\xi = 0.2$ for the ensuing examples in this work.

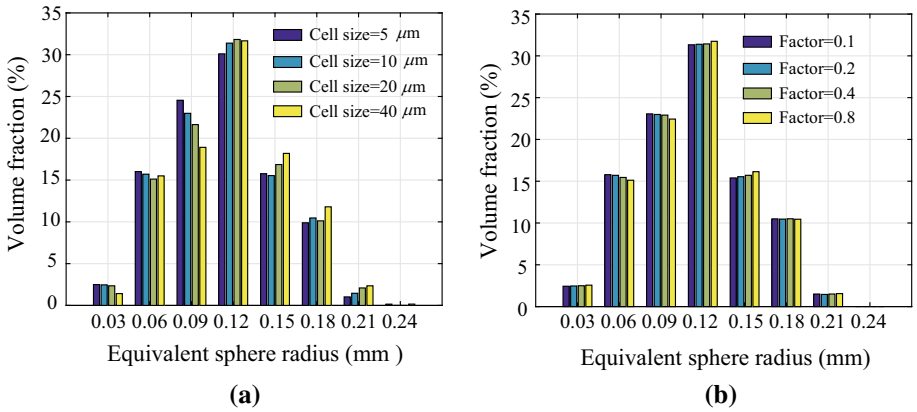

Fig. 7 Grain size (in terms of equivalent sphere radius) distribution: **a** variable cell size, time step factor $\xi = 0.2$; **b** variable time step factor, cell size 10 μm

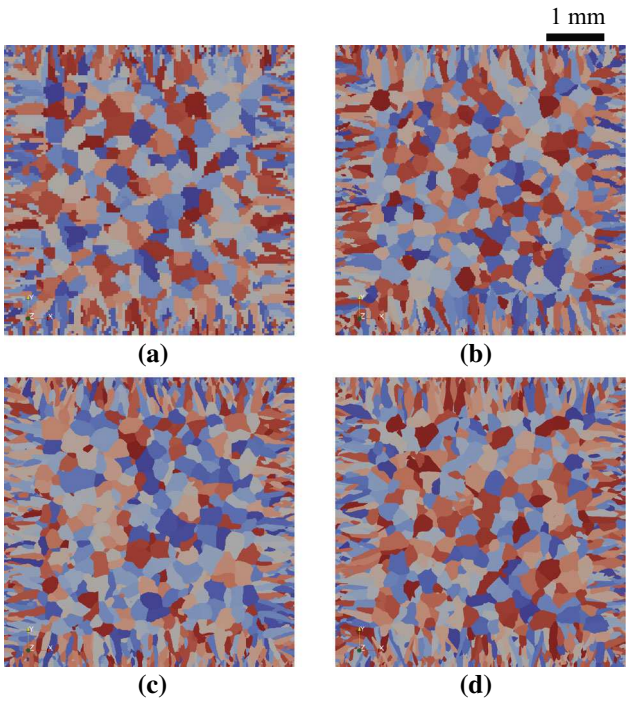

Fig. 8 XY cross-section view of grain structure of an Al-7 wt%Si specimen result from 3D CA for case I (see Table 1) with a time step factor $\xi = 0.2$ and varying cell size: **a** $d_{\text{cell}} = 0.04$ mm, **b** $d_{\text{cell}} = 0.02$ mm, **c** $d_{\text{cell}} = 0.01$ mm and **d** $d_{\text{cell}} = 0.005$ mm

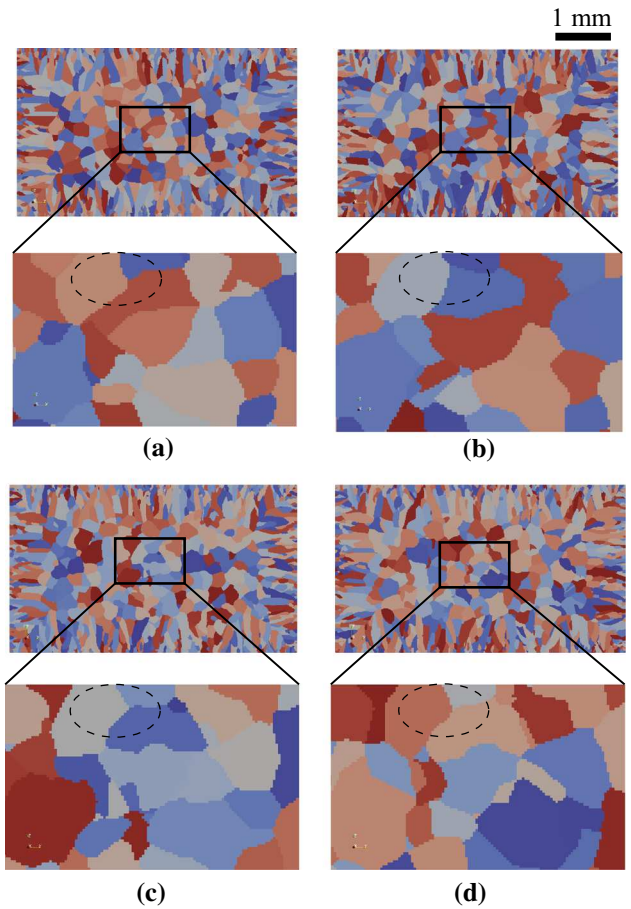

Fig. 9 YZ cross-section view of grain structure of an Al-7 wt%Si specimen result from 3D CA for Case I using $d_{\text{cell}} = 10\,\mu\text{m}$ and various time step factors ξ: **a** $\xi = 0.8$, **b** $\xi = 0.4$, **c** $\xi = 0.2$ and **d** $\xi = 0.1$

4.1.2 Parallel efficiency demonstration

To demonstrate the parallel efficiency of the proposed parallelized 3D CA model, an example with a domain size of $20 \times 5 \times 5$ mm^3, $d_{\text{cell}} = 10\,\mu\text{m}$, $\xi = 0.2$ and others parameters as listed in Case I of Table 1 is run in parallel on different number of processors. For this example, the domain is decomposed only along X dimension.

The overall parallel efficiency with respect to processors number used is listed in Table 5 and plotted in Fig. 10. The parallel efficiency decreases with increasing number of processors for most cases. This drop in efficiency can be attributed in part to load imbalance for this problem. In the CA model, the computational load on each processor p at a given time step, $B_p(k)$, is expected to be dominated by the initialization of new grain envelopes for each cell that is captured at a given time t_k. Therefore, we can approximate the total computational load as the number of liquid cells captured at each time step, $N_p(k)$. The degree of load imbal-

Table 5 Computation time t and parallel efficiency E of the proposed parallelized 3D CA model for the case with a domain size of $20 \times 5 \times 5\,\text{mm}^3$ and $d_{\text{cell}} = 10\,\mu\text{m}$

Processor (n_p)	1	2	4	8	16	32	64
t (hour)	11.11	5.74	3.10	1.59	0.93	0.58	0.38
E (%)	–	96.75	89.49	87.30	74.57	59.88	45.85

Efficiency is computed as $E = t_1/(n_p \times t_p)$, where t_1 and t_p denote the total run time for the serial program and parallel program, respectively, and n_p is the number of processors

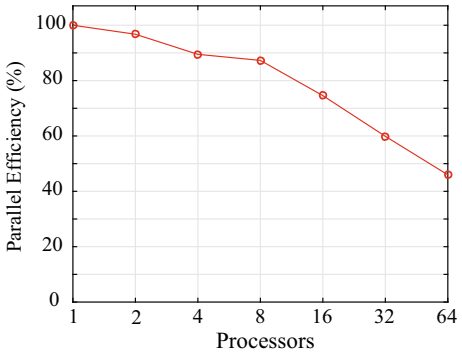

Fig. 10 Parallel efficiency of the proposed parallelized 3D CA model for the case with a domain size of $20 \times 5 \times 5\,\text{mm}^3$ and $d_{\text{cell}} = 10\,\mu\text{m}$

ance in a global sense can be illustrated by comparing $N_p(k)$ across processor as a function of time.

In Fig. 11, $N_p(k)$ for cases with 2, 4, 8 and 16 processors are plotted to show the load balance variation with increasing the number of processors. It can be seen that the load is heavily influenced by differences in the number of surface cells on each processor, because surface nucleation sites are more active early in the simulation compared with the bulk volume, and less active later. For the two-processor case, symmetry leads to an equal number of surface cells on both processors. As a result, the loading curves for this case overlap (Fig. 11a) and load is well balanced; the parallel efficiency is 96.75%. For higher numbers of processors, the total number of cells is equal across processors, but the surface cells are more concentrated on just two processors (because the domain is decomposed only in the x direction). These two processors have a higher workload at early times, when microstructure formation is dominated by surface nucleation, but lower at later times when surface regions have finished solidifying. This discrepancy and the resulting load imbalance, become more pronounced as the total number of processors increases, leading to lower parallel efficiencies.

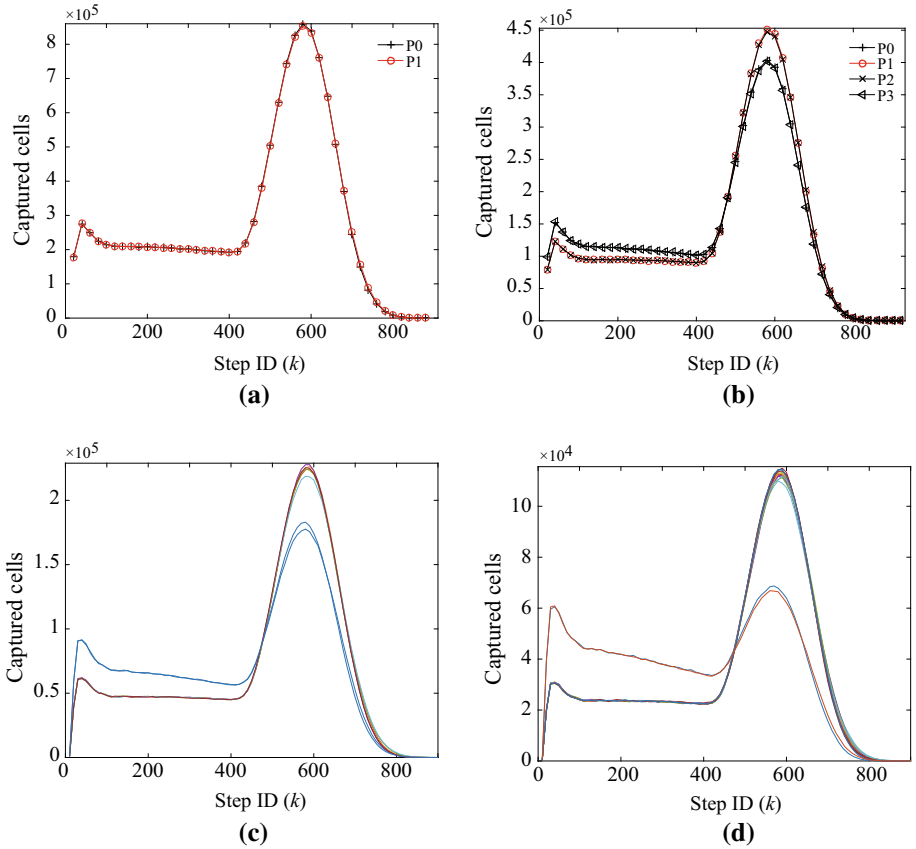

Fig. 11 Computational load for each processor as estimated by the number of liquid cells captured at each time step, $N_p(k)$, for various numbers of processors: **a** 2 processors, **b** 4 processors, **c** 8 processors, and **d** 16 processors

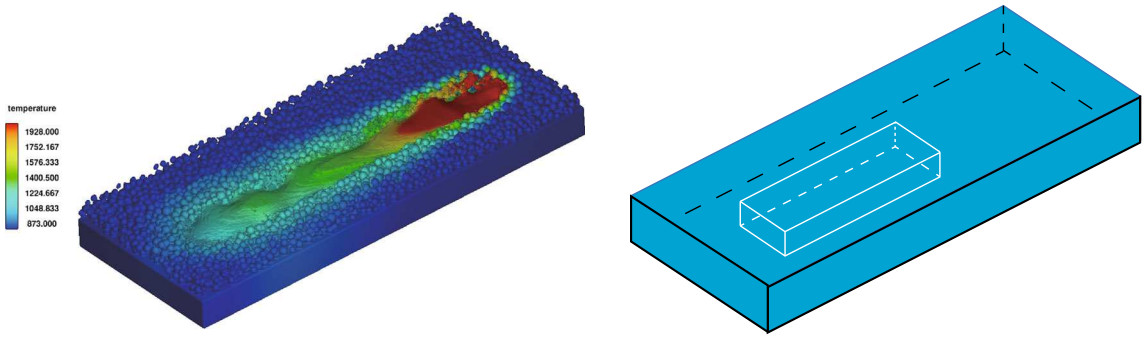

Fig. 12 **a** Temperature solution and track geometry at the end of a single beam pass through the powder bed; **b** outline of the CA region inside the outline of the temperature solution domain shown in (**a**)

4.2 Grain growth during additive manufacturing

In the Selective Electron Beam Melting (SEBM) AM process, melting of material is controlled by multiple processing parameters, including beam power, scanning velocity, tool path, build temperature, powder size distribution and so on, which can lead to different solidification conditions. Therefore, the resulting material microstructure morphology can be heterogeneous and may be different from sample to sample, but share common features under reasonable operating parameters. For AM of Ti–6Al–4V, common features include columnar grain structures and epitaxial growth. A radial distribution pattern of columnar grains can often be observed along the cross-section normal to the heat source scan direction for a single track in accordance to the heat flux distribution, as demonstrated by the typical SEBM Ti–6Al–4V microstructures obtained under different processing parameters reported in [2,3]. The pattern of the columnar structure is attributed to the steep thermal gradient associated with the moving melt pool, where the base plate and consolidated material act as a heat sink and the surrounding loose/lightly sintered titanium powder acts as insulation. Columnar grains grow quickly under the high thermal gradient, without time for nucleation ahead of the solidification front occur, leading to epitaxial re-growth within each melted layer. For each single track, this epitaxial growth can take place on material already consolidated in the previous layer or substrate, or from partially-melted powder particles in the surrounding powder bed, as demonstrated by Fig. 5 in the work of Antonysamy et al. [2]. As a consequence, the grain size of each new layer is significantly dependent on the existing material; for this reason, the grain size shown in Fig. 2 of Al-Bermani et al. [3] is different from that of Fig. 5 in Ref. [2].

In this section, a single track of the SEBM AM process with a Ti–6Al–4V alloy powder bed is simulated to demonstrate the application of the 3D CA model. In this example, the diameters of spherical powder particles range between 40 and 80 μm, the layer thickness is 0.05 mm, and the electron beam power is 60 W with a scan speed of 0.5 m/s. The temperature solution and track geometry at the end of a single beam pass through the powder bed are shown in Fig. 12a. Details of this thermal/fluid simulation can be found in [24]; here, we couple the results of this computation to the 3D CA model and focus only on the grain structure prediction during the solidification process.

Table 6 Growth kinetics and nucleation parameters for Ti–6Al–4V

$v(\Delta T) = \eta_1 \cdot \Delta T + \eta_2 \cdot \Delta T^2$	
$\eta_1 = 0.544 \times 10^{-3}\,\mathrm{m/(s\,K)}$	$\eta_2 = 2.03 \times 10^{-3}\,\mathrm{m/(s\,K^2)}$

The region size of interest is $0.7 \times 0.225 \times 0.055\ \mathrm{mm}^3$, as illustrated by the inner outline shown in Fig. 12b. The parameters for the 3D CA model, taken from Ref. [7], are summarized in Table 6. For this case, nucleation of grains within the melt pool is ignored, and grains grow from the predefined structure of the existing material, which is assumed to have a fine grain structure. In this simulation, we use a cell size of $d_{\mathrm{cell}} = 1\,\mu\mathrm{m}$ and a time step factor of $\xi = 0.2$ for the CA model.

Figure 13 shows four snapshots of the solidification microstructure in the selected domain. The emergence of columnar grain growth from an existing substrate is captured, showing reasonable agreement with experimental results, e.g., Fig. 3 in [2] showing the primary β-grain structure that forms during initial solidification, and Fig. 12 in [3]. Furthermore, the XZ cross-sections of the final grain morphology at three different Y positions are given in Fig. 14a, where the Y direction is across the melt track. All three subfigures show similar grain patterns and demonstrate columnar grains growing upward and tilted slightly along the direction of the beam scan, aligning with the temperature gradient. In Fig. 14b, the grain morphologies in the YZ plane at three dif-

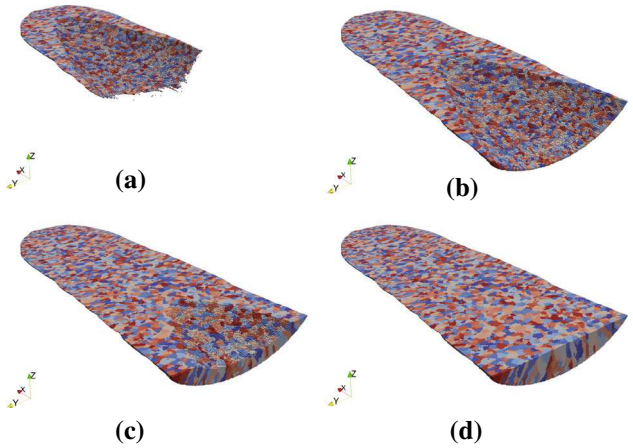

Fig. 13 Evolution of the molten pool microstructure represented by four snapshots at increasing times: **a** 0.3492 ms, **b** 1.1342 ms, **c** 1.4291 ms, and **d** 1.8427 ms

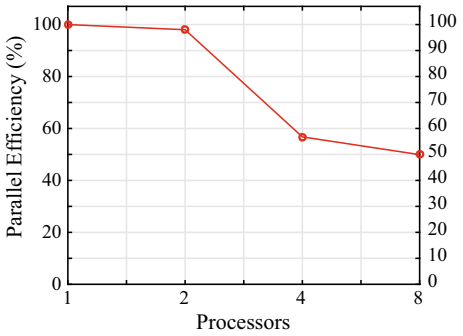

Fig. 15 Parallel efficiency of the proposed parallelized 3D CA model for the Selective Electron Beam Melting (SEBM) simulation

ferent X locations, where X is the direction along the melt track, show the radial, epitaxial grain growth from underlying material. These results show good qualitative agreement with results observed in experiments [2,3].

The parallel efficiency of the proposed method for this problem is presented in Fig. 15. In this scaling study, the domain is decomposed only along the X direction. This domain decomposition strategy leads to a good load balance for the 2 processor case, but becomes increasingly worse with larger numbers of the processors due to the moving molten pool as shown in Fig. 13. As a result, a static domain decomposition strategy may not be desirable for this problem due to load imbalances from the transient evolution of extremely localized events. Other parallelization strategies, including reallocation of domains to rebalance workload, will be explored in future work.

5 Conclusion

In this work, a parallelized 3D cellular automaton (CA) model was presented to simulate the grain structure during solidification of an alloy in both casting and additive manufacturing. In this model, a static domain decomposition strategy is applied in a distributed memory scheme implemented using MPI. For interprocess communications, a Cartesian virtual topology is used together with ghost layers of cells surrounding each processor domains. In CA, growth of individual grains is modeled by capturing cells surrounding it, which requires simple and efficient interprocess communications associated with the captured ghost cells and ghost cells' phase state update. From the numeri-

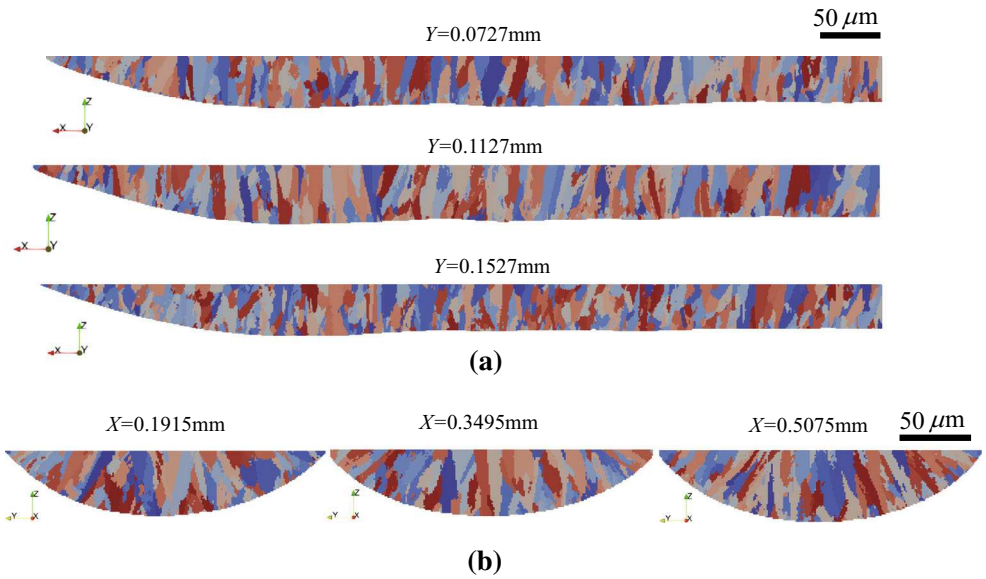

Fig. 14 Cross-sectional views of grain morphology: **a** XZ plane at three different locations, and **b** YZ plane at three different locations

cal casting examples, the parallel efficiency of the proposed model can retain more than 80% with eight processors, and reasonable efficiencies on 64 processors, although load imbalance can lead to loss of efficiency depending on the problem geometry. For the problems tested, it is demonstrated that load imbalances rise from the distribution of the surface and bulk nucleation with different critical undercooling. In addition, keeping the cell size less than the dendrite arm spacing and time step size factor around 0.2 are recommended for accurate CA predictions. Furthermore, the grain growth of Ti–6Al–4V alloy powder-bed by Selective Electron Beam Melting was predicted, where the temperature field with large undercooling rate and a high gradient is solved through a finite volume method based on a high fidelity discrete model. The microstructure results are in qualitative agreement with those of experimental observation. In our upcoming work, the solidification microstructure under different process parameters, such as heat source power, scanning speed, layer thickness, and scan direction, all of which can influence the temperature field, will be presented.

The static domain decomposition strategy may suffer from a significant load imbalance in AM simulations because the actively growing grains are located on only a subset of the computational processors. For cases like this, other parallelization strategies, including reallocation of domains to rebalance workload, may be preferred in order to maintain good parallel efficiency on large numbers of processors. This is a direction for future exploration.

Acknowledgements Gregory J. Wagner and Wing Kam Liu acknowledge the support by the National Science Foundation (NSF) Cyber-Physical Systems (CPS) under Grant No. 359 CPS/CMMI-1646592. Yanping Lian, Wentao Yan, and Wing Kam Liu acknowledge the support by Center for Hierarchical Materials Design (CHiMaD) under Grant No. 70NANB14H012. Stephen Lin acknowledges the support by the NSF Graduate Research Fellowship under Grant No. DGE-1324585.

References

1. Acharya R, Sharon JA, Staroselsky A (2017) Prediction of microstructure in laser powder bed fusion process. Acta Mater 124:360–371
2. Antonysamy AA, Meyer J, Prangnell PB (2013) Effect of build geometry on the β-grain structure and texture in additive manufacture of Ti–6Al–4V by selective electron beam melting. Mater Charact 84:153–168
3. Al-Bermani SS, Blackmore ML, Zhang W, Todd I (2010) The origin of microstructural diversity, texture, and mechanical properties in electron beam melted Ti–6Al–4V. Metall Mater Trans A 41(13):3422–3434
4. Boettinger WJ, Warren JA, Beckermann C, Karma A (2002) Phase-field simulation of solidification. Ann Rev Mater Res 32(1):163–194
5. Carozzani T, Gandin C-A, Digonnet H, Bellet M, Zaidat K, Fautrelle Y (2013) Direct simulation of a solidification benchmark experiment. Metall Mater Trans A Phys Metall Mater Sci 44(2):873–887
6. Dantzig JA, Rappaz M (2016) Solidification. EPFL Press, Lausanne
7. Dezfoli ARA, Hwang W-S, Huang W-C, Tsai T-W (2017) Determination and controlling of grain structure of metals after laser incidence: theoretical approach. Sci Rep 7(41527):1–11
8. Ferreira AF, da Silva AJ, de Castro JA (2006) Simulation of the solidification of pure nickel via the phase-field method. Mater Res 9(4):349–356
9. Gandin C-A, Rappaz M (1994) A coupled finite element cellular automaton model for the prediction of dentritic grain structures in solidification processes. Acta Metall Mater 42(7):2233–2246
10. Gandin C-A, Desbiolles J-L, Rappaz M, Thevoz P (1999) A three-dimensional cellular automation-finite element model for the prediction of solidification grain structures. Metall Mater Trans A 30(12):3153–3165
11. Gandin C-A, Rappaz M (1997) A 3D cellular automaton algorithm for the prediction of dendritic grain growth. Acta Mater 45(5):2187–2195
12. Kim Y-T, Goldenfeld N, Dantzig J (2000) Computation of dendritic microstructures using a level set method. Phys Rev E 62(2):2471–2474
13. Kurz W, Giovanola B, Trivedi R (1986) Theory of microstructural development during rapid solidification. Acta Metall 34(5):823–830
14. Lipton J, Glicksman ME, Kurz W (1984) Dendritic growth into undercooled alloy melts. Mater Sci Eng 65:57–63
15. Liu DR, Reinhart G, Mangelinck-Noel N, Gandin C-A, Nguyen-Thi H, Billia B (2014) Coupled cellular automaton (CA)–finite element (FE) modeling of directional solidification of Al-3.5 wt% Ni alloy: a comparison with X-ray synchrotron observations. ISIJ Int 54(2):392–400
16. (2015) MPI: a message-passing interface standard. Version 3.1. http://mpi-forum.org/docs/mpi-3.1/mpi31-report.pdf
17. Panwisawas C, Qiu C, Anderson MJ, Sovani Y, Turner RP, Attallah MM, Brooks JW, Basoalto HC (2017) Mesoscale modelling of selective laser melting: thermal fluid dynamics and microstructural evolution. Comput Mater Sci 126:479–490
18. Rappaz M, Gandin C-A (1993) Probabilistic modelling of microstructure formation in solidification processes. Acta Metall Mater 41(2):345–360
19. Rodgers TM, Madison JD, Tikare V, Maguire MC (2016) Predicting mesoscale microstructural evolution in electron beam welding. JOM 68(5):1419–1426
20. Rodgers TM, Madison JD, Tikare V (2017) Simulation of metal additive manufacturing microstructures using kinetic Monte Carlo. Comput Mater Sci 135:78–89
21. Scott TJ, Beaulieu TJ, Rothrock GD, O'Connor AC (2016) Economic analysis of technology infrastructure needs for advanced manufacturing: additive manufacturing. In: NIST GCR-16-006, NIST, Gaithersburg, MD
22. Tan L, Zabaras N (2007) A level set simulation of dendritic solidification of multi-component alloys. J Comput Phys 221(1):9–40
23. Wheeler AA, McFadden GB, Boettinger WJ (1996) Phase-field model for solidification of a eutectic alloy. In: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, vol 452, no 1946. The Royal Society, pp 495–525
24. Yan W, Ge W, Qian Y, Lin S, Zhou B, Liu WK, Lin F, Wagner GJ (2017) Multi-physics modeling of single/multiple-track defect mechanisms in electron beam selective melting. Acta Mater 134:324–333
25. Yin H, Felicelli SD (2010) Dendrite growth simulation during solidification in the LENS process. Acta Mater 58(4):1455–1465
26. Zhang J, Liou F, Seufzer W, Taminger K (2016) A coupled finite element cellular automaton model to predict thermal history and grain morphology of Ti–6Al–4V during direct metal deposition (DMD). Addit Manuf 11:32–39

27. Zhao C, Fezzaa K, Cunningham RW, Wen H, Carlo FD, Chen L, Rollett AD, Sun T (2017) Real-time monitoring of laser powder bed fusion process using high-speed X-ray imaging and diffraction. Sci Rep 7(3601):1–11

28. Zinoviev A, Zinovieva O, Ploshikhin V, Romanova V, Balokhonov R (2016) Evolution of grain structure during laser additive manufacturing. Simulation by a cellular automata method. Mater Des 106:321–329