



Cellular Automata-based computational library for development of digital material representation models of heterogenous microstructures

Konrad Boguń¹ · Mateusz Sitko¹ · Mateusz Mojżeszko¹ · Łukasz Madej¹

Received: 24 February 2020 / Revised: 13 January 2021 / Accepted: 9 March 2021 / Published online: 29 March 2021
© The Author(s) 2021

Abstract

The development of an efficient numerical approach for the generation of a wide range of heterogeneous microstructures models with the application of the lean workflow concept is presented in the paper. First, the idea and implementation details of the developed cellular automata-based computational library allowing the development of digital material representation models within a workflow are presented in the paper. Such an approach provides the desired flexibility in the generation of various digital models of heterogenous microstructures. Therefore, the proposed library is mostly implemented within the object-oriented C++ programming language with the assumption of modularity. In this case, the main part of the application consists of classes and methods, which can be treated like base elements to be inherited and extended in other libraries. Each additional dynamic link library implements particular algorithms for the generation of specific microstructure features in the digital model within the unified data structures that allow the application of the workflow concept. The set of developed libraries and their assumptions are described as case studies to show the capabilities of the presented solution. Finally, examples of practical applications of the developed library in the full-field numerical simulations of complex material deformation are presented at the end of the paper.

Keyword Digital material representation; cellular automata; Monte Carlo

1 Motivation

The finite element method is among the most commonly used approaches in modern computational engineering. It has also been intensively applied to a numerical investigation of different metal forming operations as an efficient tool for computer-aided development of new processing solutions [1]. In this case, the accuracy of the obtained results is directly related to the applied finite element mesh density, initial and boundary conditions or material hardening law. The latter is particularly important as it directly captures the physical mechanisms responsible for metal deformation.

Plastometric tests at different deformation conditions are usually used to deliver information on material response

during loading in the form of flow stress data for the investigated range of process conditions. Then, by means of approximation of obtained information, it is possible to define a single flow stress equation, which represents material behavior with increasing deformation. Such a class of equations takes the influence of different phases, inclusions or precipitations in the microstructure on its global behavior in an implicit manner [2, 3]. This is computationally efficient, unfortunately, it is not fully sufficient for modern metallic materials like complex multiphase steels, aluminum, magnesium or copper alloys. These materials can have very specific in-use properties, which are directly related to complex interactions between different phases, inclusions or precipitations at the micro-scale level [4–7]. Therefore, neglecting these local interactions in a single homogenized flow stress model may lead to inaccurate numerical results. Thus, these conventional numerical approaches referred in the literature as mean-field models do not meet the requirements of modern material science, which is trying to develop new materials based on close relations between sophisticated microstructure morphologies and elevated in-use properties.

✉ Mateusz Sitko
msitko@agh.edu.pl

¹ Faculty of Metals Engineering and Industrial Computer Science, AGH University of Science and Technology, Al. Mickiewicza 30, 30-059 Krakow, Poland

That is the reason why it is so important to develop numerical models that can take into account the underlying microstructure's morphology and its evolution in an explicit manner during simulation of, e.g., deformation conditions. This class of models is called full-field approaches and is often based on the digital material representation (DMR) concept [8–10], where each important microstructural component or feature is directly visualized during subsequent calculations, often based on the finite element approach.

A large number of programs and algorithms have been created to generate digital microstructures using, e.g. image analysis techniques [11], Voronoi tessellation [12, 13], cellular automata (CA) [14, 15], Monte Carlo (MC) [16, 17], Molecular Dynamics (MD) [18], numerical homogenization [19, 20] or a combination of these approaches. A detailed review of various mentioned numerical solutions was presented in the author's earlier work [21]. These approaches provide crucial input data for a further full-field micro-scale analysis of material behavior under processing or exploitation conditions.

The main disadvantage of these solutions, limiting their faster development, is the lack of unification (e.g., the data structures), which makes interoperability of these approaches difficult. In addition, each subsequent extension of the available functionality requires detailed knowledge of the structure and concept of the software, which hinders the further development of the mentioned algorithms.

One of the few available solutions for this issue is the *DREAM.3D* software [22]. It is an open-source package focused on providing a high-level programming environment for processing, segmenting, quantifying, representing and manipulating the data related to digital microstructures. The main goal of the *DREAM.3D* is to enable the quantification of the microstructure and transfer the outcome to a digital database using easy-to-use software tools. The *DREAM.3D* core implements a standardized approach to operate and store digital microstructure data. Additionally, there are protocols enabling the independent creation of filters and plugins for integration with other solutions. The *DREAM.3D* environment is structured in such a way that small research groups, government laboratories, start-ups and even the largest industrial corporations can interact and use each other's solutions. However, due to the imposed data structure, the practical use of the *DREAM.3D* software is often associated with the need to re-implement the developed algorithms to adapt them to the required format.

Another solution, with significantly smaller functionality, is the *VGRAIN* [23]. It enables the generation of digital microstructure with specific crystallographic grain orientation only by the Voronoi tessellation and probability theory. Therefore, it is not a universal tool that can be easily expanded.

A similar situation is with the *Digiμ* program, which is again primarily based on a single algorithm to provide digital microstructures—the Voronoi tessellation [24]. The *Digiμ* is an integral element of the commercial Forge software. Thus, in its present form, it is not a framework for creating digital models of microstructures with different approaches.

As presented above, available solutions do not meet the requirements of a general computer environment for the generation of digital microstructures with a wide range of algorithmic solutions. The only exception is the *DREAM.3D*, with a very broad range of capabilities, often not only related to the generation of digital microstructures, which may cause difficulties in easy use by not experienced researchers.

That is why in the present work, a universal *DigiCore* library for fast and efficient implementation of different algorithmic solutions for the generation of digital models of heterogenous microstructures within the workflow was proposed. To highlight the capabilities of joining various models within a workflow, as well as to present the possibilities for further extension of the library, a set of case studies is presented in the following chapters. Finally, examples of practical applications of such digital models in full-field finite element simulations of deformation conditions are presented.

2 *DigiCore* library

The main assumptions of the developed *DigiCore* library were defined based on the information collected from the existing DMR algorithmic solutions [21]. In principle, the computational domain for all present and future developments is based on a discrete regular space of rectangular cells with defined variables. In that sense, the *DigiCore* library is based on the major assumptions of discrete modeling methods like the cellular automata or Monte Carlo.

The most important function of the created framework, in the sense of generation of heterogenous microstructures models, is its modularity allowing the use of a workflow concept in joining various DMR algorithms. To obtain this functionality, the main part of the application consists of classes and methods with a unified data structure, which can be treated like base elements to be inherited and extended in other libraries.

The *DigiCore* microstructure generation elements are implemented in C++ programming language. The undoubtedly advantage of this language is performance, compilation ability in different environments, and popularity [25]. The Microsoft Visual Studio IDE was selected for implementation purposes.

The core element that was developed to allow required flexibility is the main library *MSMcoreLib*, containing all

the definitions, which are necessary to generate the digital microstructure within the cellular automata space. All the definitions of classes contained in the core library and its extensions operate based on the following elements:

- Cell—stores only necessary information to characterize the material: grain id, phase, border and neighbors ids.
- Space—supports 2D and 3D square lattice, based on rectangular or cuboid cells and handle a selection of neighbors from defined Neighbourhoods.
- Neighborhood—standard types of neighborhoods, such as von Neumann, Moore, pentagonal, hexagonal, etc., are available.
- Boundary conditions (BC)—three types of BC are considered: periodic, absorbing and reflective.
- Simulation—virtual-based class with the declaration of functions required to perform the simulation.
- IOStreamer handler—read/write data from/to text files.
- Random point/number generator.

With such a definition, the *MSMcoreLib* can then be extended with other libraries related to a particular DMR model generation algorithm that can also be rearranged within a workflow to provide the digital model of heterogeneous microstructure. Developed and implemented solutions in the form of additional libraries are:

- *MSMgrainGrowthCALib*—classic grain growth algorithm based on the cellular automata method.
- *MSMgrainGrowthMCLib*—grain growth and smoothing algorithm based on the Monte Carlo method.
- *MSMgrainBordersLib*—determination and extraction of grain boundaries from the original microstructure algorithm.
- *MSMinclusionsLib*—generation of inclusions across the original microstructure algorithm.
- *MSMgradientLib*—generation of gradient type microstructure algorithm.

Including all mentioned definitions in one framework library allows for independent and parallel development of the application modules. With that concept, there is a possibility to fill the available DMR software gap and create a comprehensive solution for generating and analyzing digital material microstructures of heterogeneous materials.

The graphical user interface is written in C# and the Windows Presentation Foundation (WPF) technology included in the .NET environment. This enabled the use of native libraries written in C++ in a manageable C# required during the creation of a wrapper project in a managed version of C++ (C++/CLI). A simplified diagram of the proposed solution with the included projects is shown in Fig. 1.

As mentioned, *MSMcoreLib* is the main project in the solution. Like all others created as part of the *DigiCore* application, this library is in the form of a dynamic link

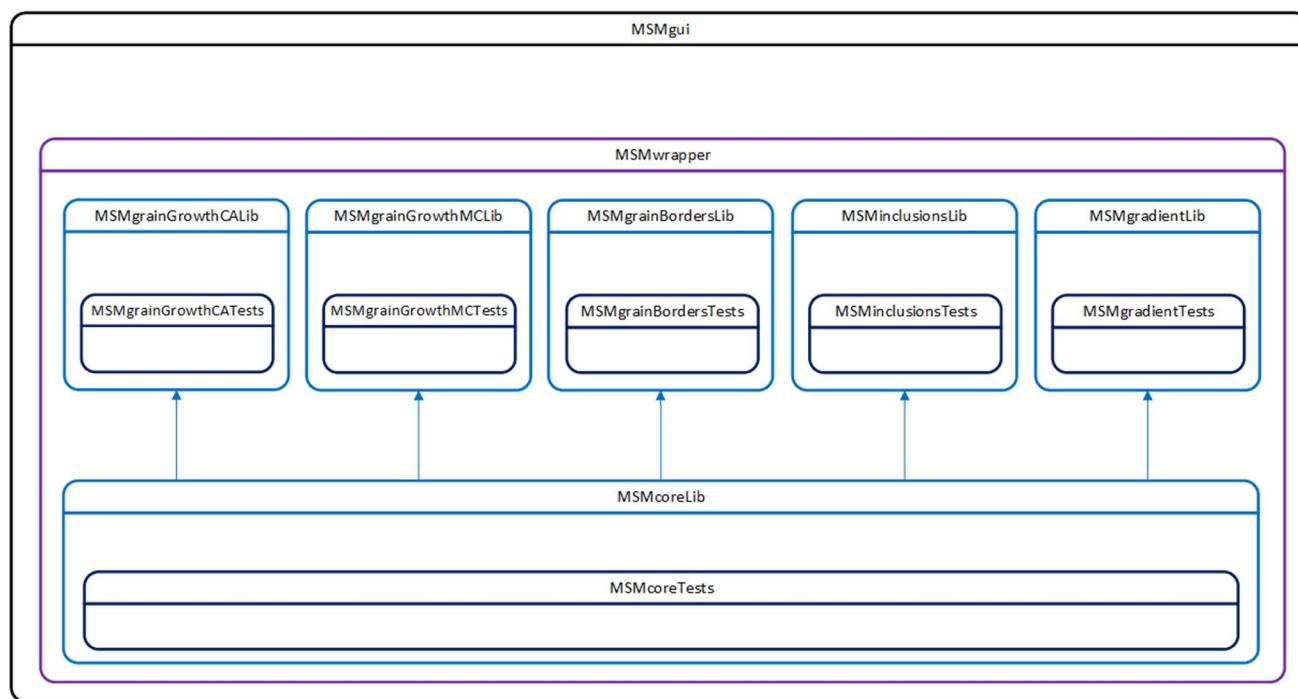


Fig. 1 Simplified diagram of the project's structure in the developed *DigiCore* library

library (DLL) to facilitate their use within the workflow. DLLs are loaded into the computer's operating memory only after their first call, which directly fits the requirements of this work, as unused application modules will not use the available memory.

The definitions of, e.g., microstructure features, simulation type, boundary conditions and neighborhood types are in the *MSMcoreLib* project. Two basic von Neumann's and Moore's neighborhoods are implemented along with more advanced modifications: pentagonal, hexagonal and radius type neighborhoods [26]. The choice of neighborhood is one of the most important factors influencing the calculation results in the discrete cellular automata method.

As mentioned, the *DigiCore* concept is based on the discrete simulations method. Thus, the first addition to the proposed system is the *MSMgrainGrowthCALib* based on the standard CA grain growth model [1], which was implemented and is available for all other models in the form of a library.

The CA grain growth is composed of two main stages: nucleation and grain growth. The method of random generation of nuclei was implemented as a standard solution that can be further modified. When the cell is selected to become a nuclei, it changes its state and is also described by a unique identifier to evaluate which grain it belongs to. With this model, each cell can belong only to one grain. At each step of the simulation, randomly generated nuclei

grow according to the simple evolution law. The simulation of growth continues until the modeled space is completely filled with grains. Details of the algorithm can be found in [21].

Such basic functionality already provides a possibility to obtain a representation of single-phase polycrystalline microstructure in 2D and 3D computational domains, as seen in Figs. 2 and 3, respectively.

As mentioned, the approach's key developed capability is modularity and unified data structure, allowing the use of the workflow concept for the generation of digital models of heterogeneous microstructures. Therefore, multiple subsequent grain growth algorithms within the digital microstructure model can be realized and combined what leads to a representation of highly heterogeneous materials like multiphase or porous ones. With that, the digital microstructure with the precise average grain size and volume fraction can be generated in both 2D and 3D spaces, as seen in Fig. 4.

The *MSMcoreLib*, along with the *MSMgrainGrowthCALib*, can be the basis of other algorithms used to generate or modify available data in a workflow. A description of developed and implemented algorithms within the *DigiCore* library that allow the generation of heterogeneous microstructure models is presented in the following chapters.

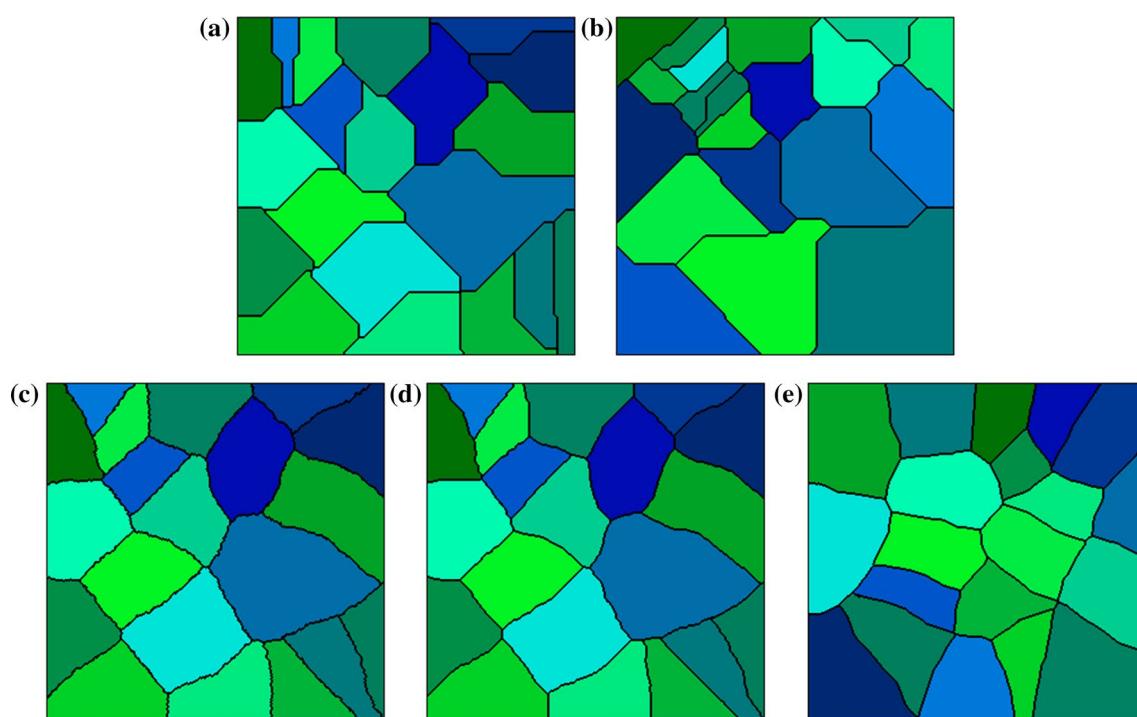


Fig. 2 Single-phase polycrystalline 2D digital microstructures with 20 grains obtained with different neighborhood definitions, **a** von Neumann, Moore, **c** pentagonal, **d** hexagonal and **e** radius type ($r=5$)

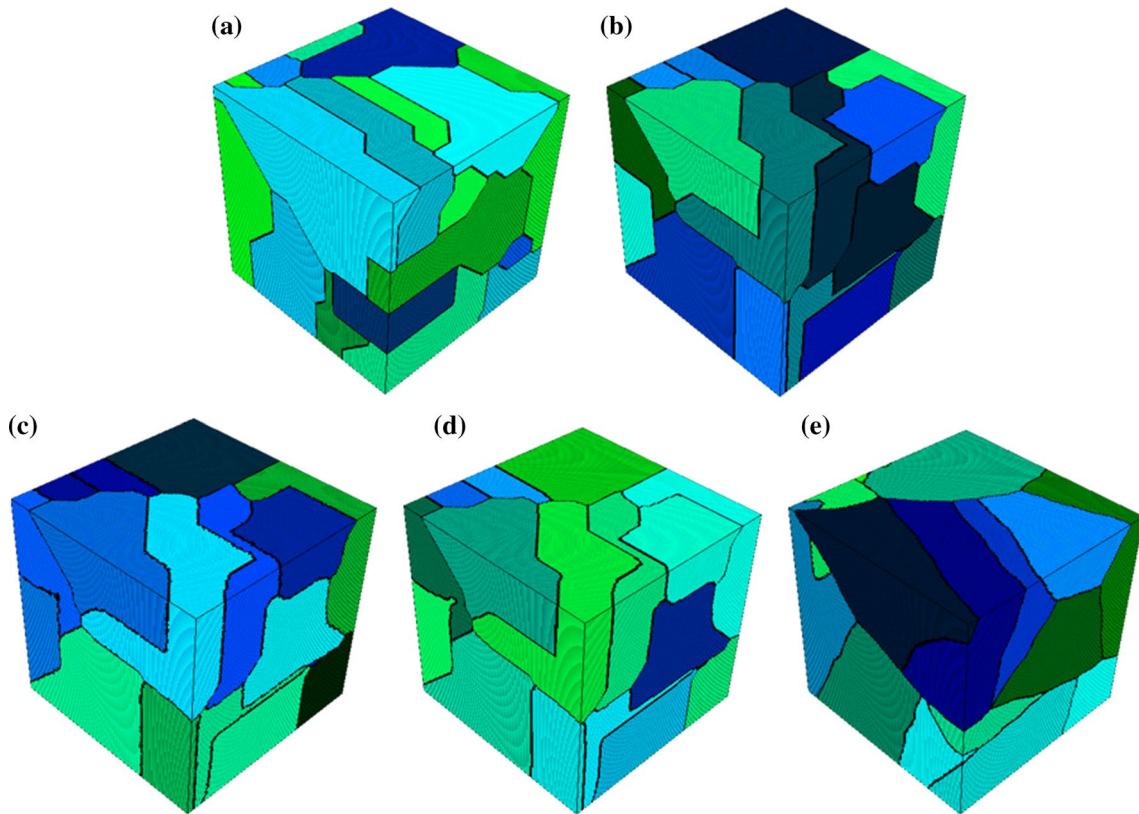


Fig. 3 Single-phase polycrystalline 3D digital microstructures with 20 grains obtained with different neighborhood definitions **a** von Neumann, **b** Moore, **c** pentagonal, **d** hexagonal and **e** radius type

($r=5$) Example of two-phase polycrystalline microstructure **a** grains ids and corresponding **b** visualization of the second phase

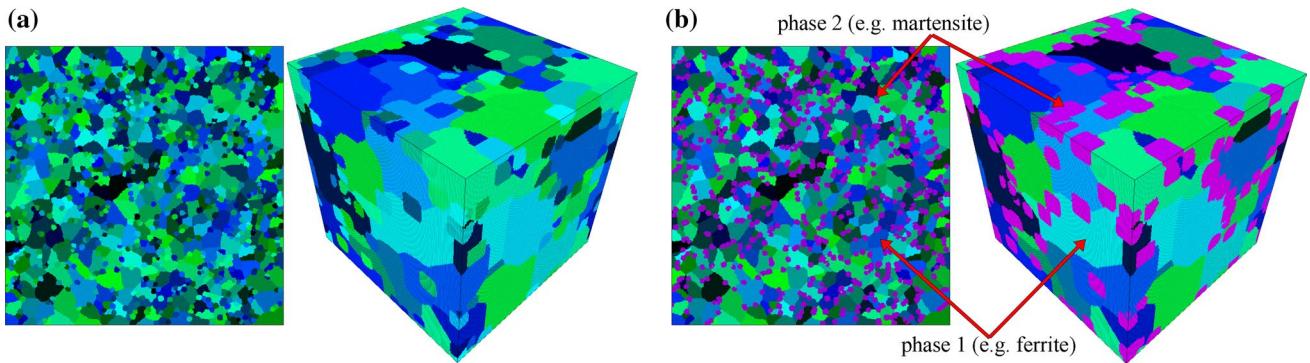


Fig. 4 Example of two phase polycrystalline microstructure **a** grains ids and corresponding **b** visualization of the second phase

3 Algorithms for flexible generation of heterogenous microstructure digital models

A good example of an algorithmic solution often used in combination with the CA grain growth model is the growth based on the Monte Carlo method [27]. The MC approach

can be used as a standalone algorithm to generate digital microstructure, with the use of some available functionality of *MSMcoreLib* or to change existing microstructure generated via different algorithms during a grain boundary smoothing procedure. The MC algorithm is implemented in the library *MSMgrainGrowthMCLib*.

In contrast to the CA grain growth model, where the computational domain is analyzed in a systematic manner,

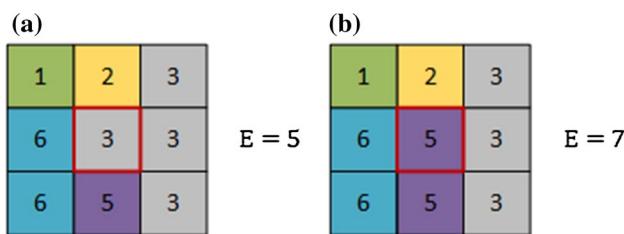


Fig. 5 An example of determining the energy of the system before and after the change of the state of the analyzed cell

the MC selects the investigated cell randomly. The transition rule for the cell state change is based on the energy minimization criterion. The energy of the cell is calculated before and after the random state change according to the formula:

$$E = J_{gb} \sum_{\langle i,j \rangle} (1 - \delta_{S_i S_j}) \quad (1)$$

where: E —energy of the system, J_{gb} —energy grain boundary, $\delta_{S_i S_j}$ —Kronecker delta,

$\langle i,j \rangle$ —analyzed pair of neighboring elements.

Acceptance of the new cell state is related to the probabilistic term:

$$p(\Delta E) = \begin{cases} 1, & \Delta E \leq 0 \\ \exp\left(-\frac{\Delta E}{kT}\right), & \Delta E > 0 \end{cases} \quad (2)$$

where ΔE —change of system energy, k —Boltzmann constant, T —system temperature (coefficient in the MC model).

An example of determining the energy of the system before and after the state change is presented in Fig. 5.

An example of results obtained with the MC algorithm used as a standalone grain growth model as well as operating on already available data from the *MSMgrainGrowthCALib* is shown in Fig. 6.

Solutions that were presented so far usually used to obtain uniform type microstructure with the grain size following the Gaussian distribution Fig. 7.

However, that is not always the case in metallic materials, where, e.g. multimodal distributions are observed [28]. As was pointed out in [29], nonuniform grain size distribution often plays a major role in a material response during deformation. That is why the *DigiCore* software was extended by another library *MSMgradientLib* to generate gradient type microstructures. More sophisticated nucleation mechanisms were developed and implemented into the conventional grain growth model to provide such functionality. Four versions of this algorithm were implemented and are called A-D in the following part of the paper.

The A model takes as arguments:

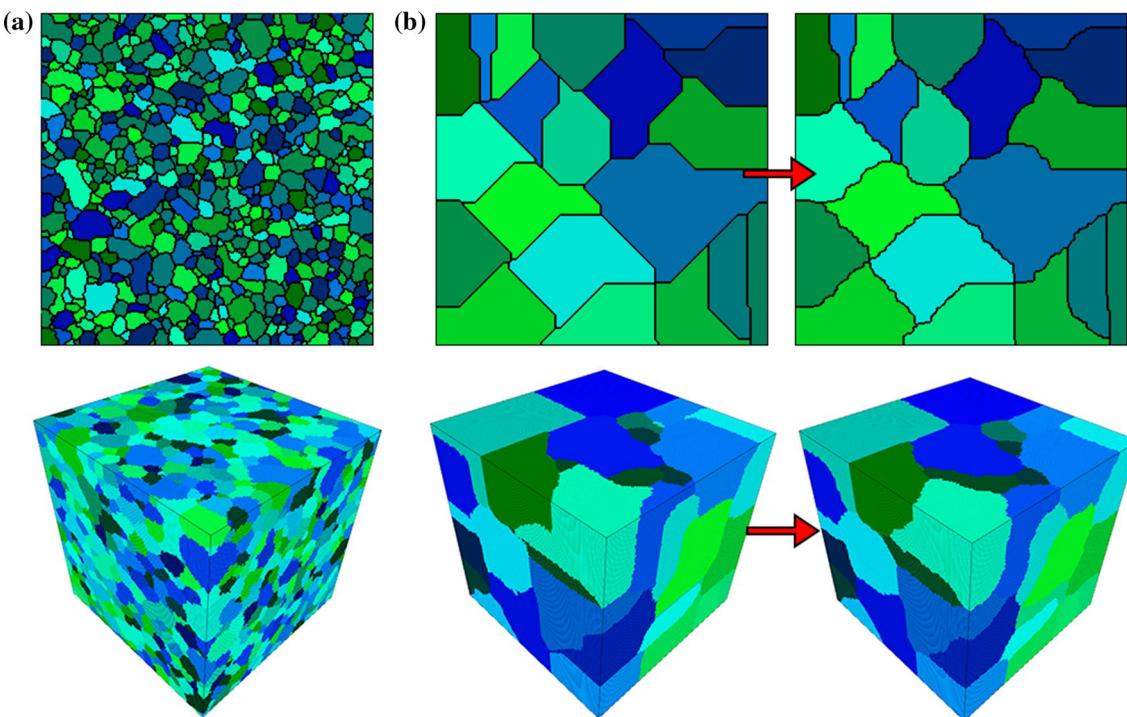


Fig. 6 Single-phase polycrystalline digital microstructures obtained with **a** Monte Carlo method, **b** Cellular Automata grain growth with smoothing procedure using the Monte Carlo

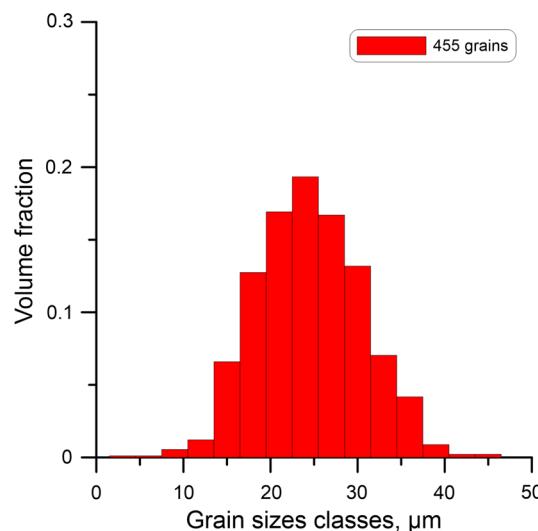


Fig. 7 Grain size distribution after generation of microstructure with random nucleation model with CA grain growth algorithm

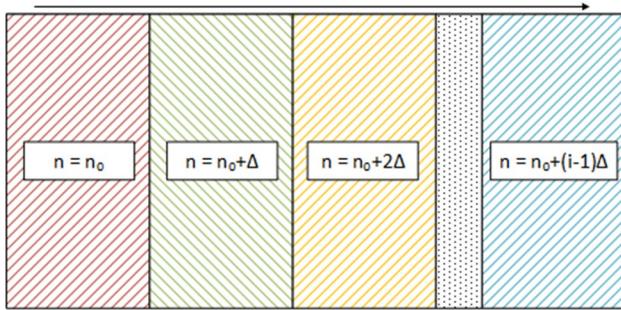


Fig. 8 Diagram of the division of computational space into layers in the A model

- The number of layers that can be generated into the modeled CA space,
- The grain growth rate,
- The initial number of grains (n_0),
- The direction of growth.

In the first layer, the algorithm generates the initial number of nuclei set as an input argument (n_0). In each subsequent layer, this value is increased by the value corresponding to the rate of increase in the number of generated nuclei from the previous layer (Δ). Figure 8 illustrates the implemented concept of the A model.

The B model generates nuclei positions for the gradient microstructure using spheres with certain initial size distribution. First, the initial number of spheres is generated into the CA space according to the sphere growth model [30] (Fig. 9a). Input data for the B model are:

- The minimal and maximum radii of generated spheres,
- The number of algorithm iterations,
- The value by which the size of the spheres is reduced between sets,
- The direction of growth,
- The initial number of spheres in the series.

The C model is similar to the B one, with the exception of the method of inserting spheres into the computing space. A random position on one of the axes is replaced by the generation of a sphere at the beginning of the axis. The following spheres can be created in the range $a + 2r, a + 5r$, where a is the current position on the given axis, while r is the current radius of the sphere (Fig. 9b). Finally, the D model is a modification of the above-described C model (Fig. 9c). In this case, the grain size generation mechanism is changed and the size of the sphere is selected based on a variable that

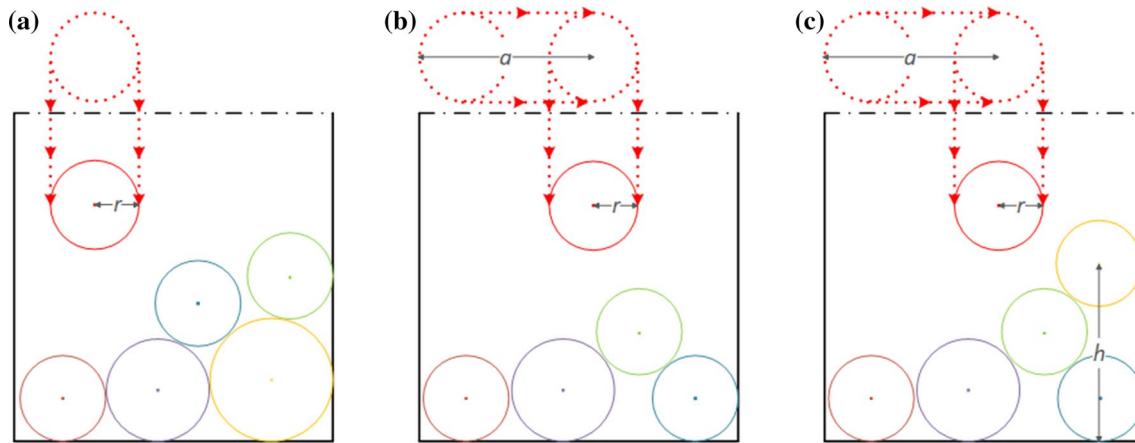


Fig. 9 Scheme of generating the initial grain nuclei by various sphere growth models

specifies the distance from the highest located sphere in the computational domain to its base (h).

Results from all presented approaches are gathered in Fig. 10. For each scenario, following parameters were used: number of layers in microstructure = 4; grain growth rate = 5; initial nucleon size = 1 for the A model (Fig. 10a), initial radius = 4 for the B–D models (Fig. 10b–d). All gradients algorithms were generated in the right direction with a maximum number of iteration set to 100,000. Additionally, for the B–D models, the minimum radius was set to 4, maximum radius to 20, delta to 4 and a parameter to 6.

Other important microstructural features often influencing material behavior during deformation are inclusions, second phase particles or voids. Therefore, a model (*MSMinclusionsLib*) for the generation of such features was also

added as a case study to the *DigiCore* software. In this case, a set of different types of features can be generated: circular and rectangular in 2D spaces, wedge, cuboid, cylindrical, and spherical in 3D spaces (Fig. 11).

Finally, despite the grain morphology information, users very often require an explicit representation of grain boundaries (Fig. 12). This task is realized by another developed algorithm in the form of an additional library, the *MSMgrainBordersLib*.

The most important aspect of the *DigiCore* software is that all the developed libraries can be easily connected in a single workflow to generate the required final microstructure morphology of heterogenous materials, as described in the following part of the paper.

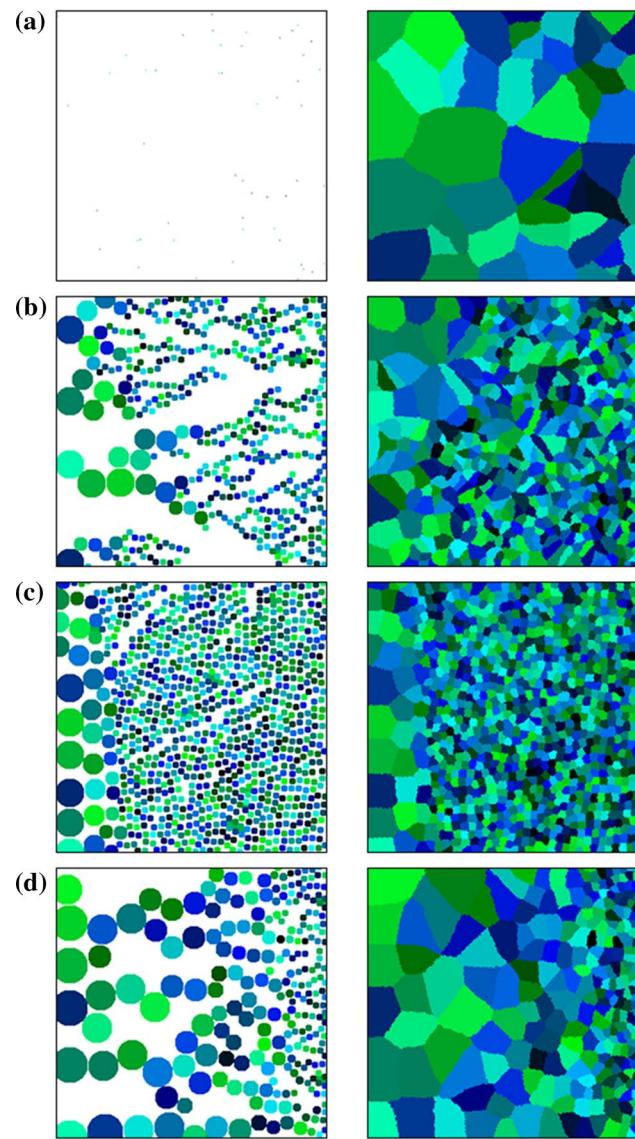
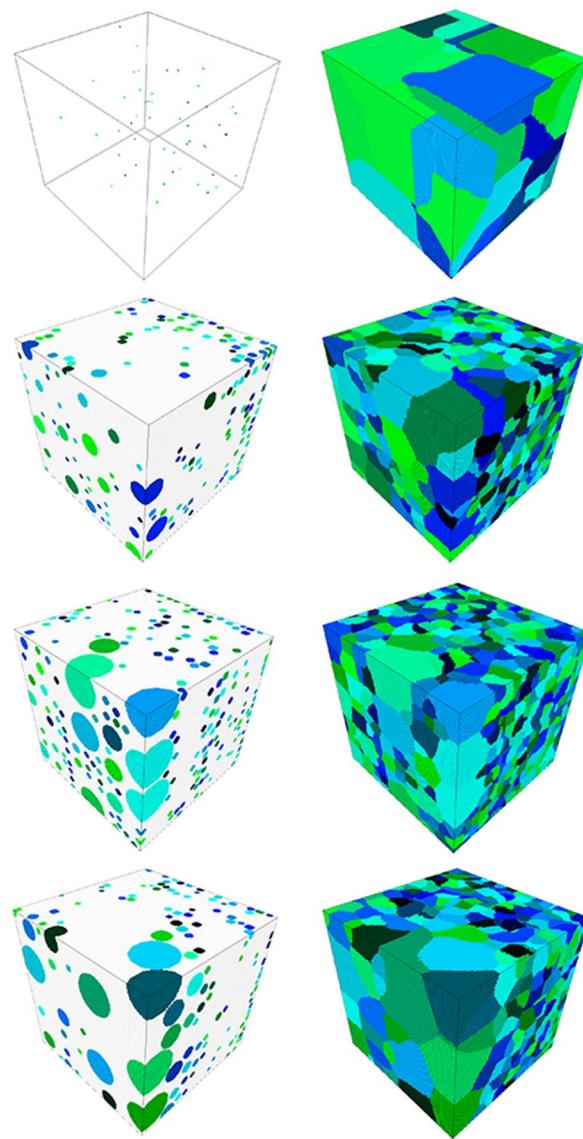


Fig. 10 Different types of nucleation in **a** A, **b** B, **c** C and **d** D models



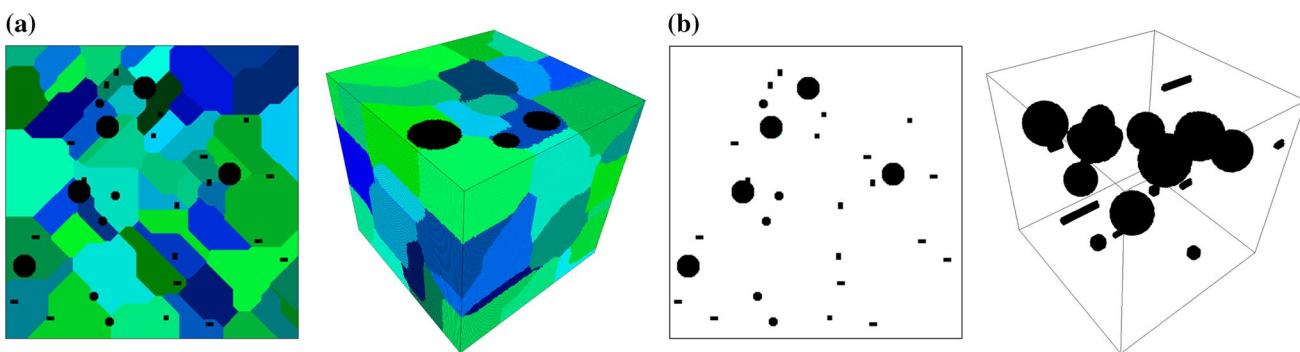


Fig. 11 **a** digital microstructure with additional inclusions, **b** location of inclusions in the computational domain

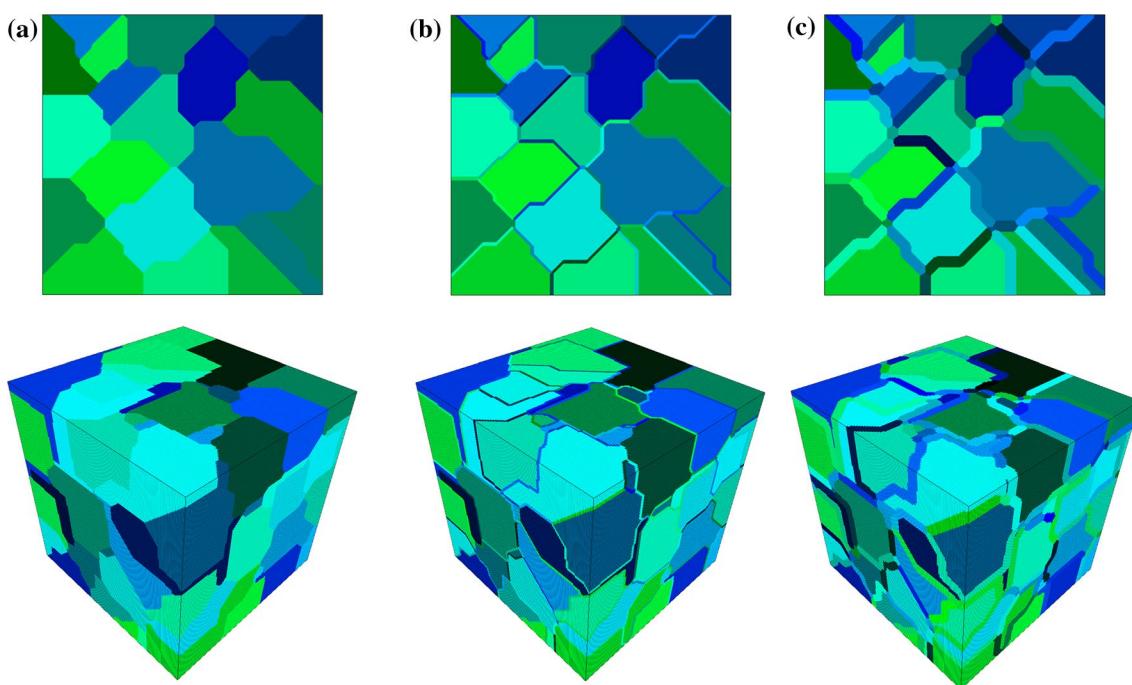


Fig. 12 **a** material morphology generated with CA method, **b, c** selected grain boundaries with different thicknesses

4 Example of the *DigiCore* capabilities

The modularity and universality of the application allow generating microstructures using various algorithms in any order. Moreover, all libraries can be used in external software as an entry point to generate an initial material morphology for further full-field numerical simulation of material behaviour under processing or service conditions with e.g. finite element method. All libraries also can be used in a simulation mode without C# GUI, which is why they can be used at different operating systems and also are designed for further parallelization on a CPU and GPGPU.

It is worth mentioning that at each step of the generation, the microstructure can be saved and loaded when

needed. This allows investigating intermediate growth stages, as seen in Fig. 13.

To highlight the predictive capabilities of the developed approach, two practical case studies of heterogeneous microstructures generations were selected, namely porous sinters and ferrite-pearlite two-phase steel. The 3D experimental data presenting the required digital morphologies were obtained during time-consuming metallography with the serial sectioning approach. The iterative procedure was realized with the use of light (LM) and scanning electron (SE) microscopy, respectively, as seen in Fig. 14. Details on laborious serial sectioning procedure along with the 3D reconstruction algorithms are presented in earlier authors' research [15, 31].

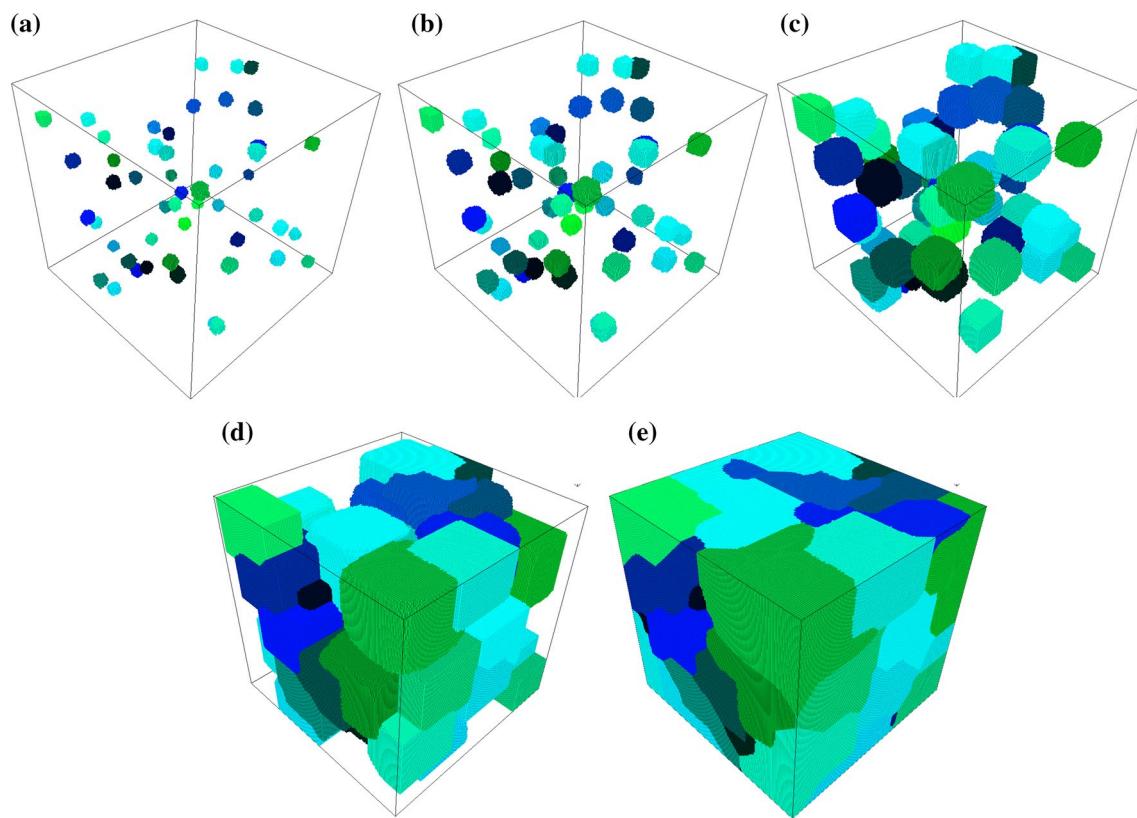


Fig. 13 Example of microstructure evolution in 3D, **a–d** intermediate states, **e** final microstructure

The use of the developed CA library and the workflow concept can provide statistically similar digital models characterized by, e.g. the same volume fractions of particular phases or porosity but within seconds, minimizing the experimental effort and speeding up further investigations. As seen in Fig. 15a, c, the workflow for generation of porous microstructure was composed of: generation of initial material morphology with the Monte Carlo grain growth algorithm (limited number of initial states) → selection of one particular grain *id* and agglomeration of remaining ones → erosion algorithm based on the grain boundary selection model.

In the case of ferrite-perlite microstructure (Fig. 16a, c), the workflow was additionally extended with the two steps: elongation of the CA space to recreate lamellar morphology of perlite islands and finally application of the cellular automata grain growth algorithm for the generation of ferrite grains between perlite islands.

Such digital material representation models can be further subjected to finite element discretization procedure to create a full-field FE model for subsequent simulation of, e.g., loading conditions, as seen in Fig. 15b, d and Fig. 16b, d. A set of hardening models for subsequent microstructural phases have to be identified in this case based on, e.g., nano-indentation or micropillar

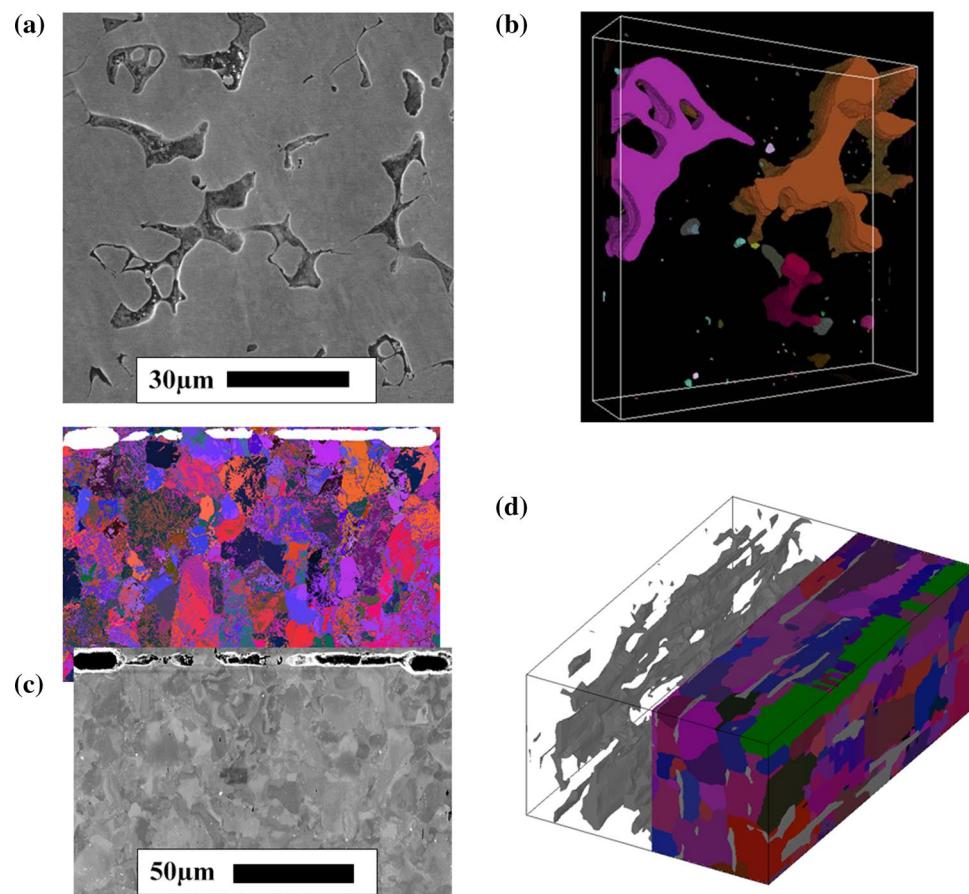
compression tests. As a result, a full-field finite element simulation provides detailed information on local heterogeneities in strain distribution that can be further used during, e.g., fracture analysis or become an input for numerical simulation of subsequent processing operations during heating, indicating stored energy fields for the grain growth.

Finally, as mentioned, the workflow concept can be easily extended by the implementation of other libraries that can be incorporated into the *DigiCore* and use all already available developments. That way, more sophisticated algorithms can be proposed that are focused on a particular model solution rather than the implementation of standard algorithms from the beginning. As a result, digital models of heterogeneous microstructures including metallic and non-metallic ones, can be easily recreated and become an input for further investigations.

5 Conclusion

Based on the presented work, it can be concluded that:

Fig. 14 Microstructure morphology obtained during metallographic investigations **a** LM image of porous microstructure, **b** corresponding 3D reconstructed microstructure after serial sectioning (pores are marked in colors), **c** SE image of ferrite-perlite microstructure (EBSD and SE2 images), **d** corresponding 3D reconstructed microstructure after serial sectioning (perlite marked in gray, ferrite grains marked in colors).



- The modularity and unified data structures of the developed library allow the use of the workflow concept.
- The *DigiCore* can also be easily extended by adding new libraries implementing other microstructure processing algorithms to obtain models of various heterogeneous microstructures.
- All microstructure generation algorithms are fully independent and fully compatible with each other.
- The development of a new algorithm for the generation of the DMR can be realized in parallel because there is no overlapping of existing libraries.
- All implemented libraries are C++-based; therefore, they can be easily used in different applications as an initial digital microstructure morphology generator.
- Developed libraries are operating system independent, they can be used in both Windows and Linux systems.
- The developed solution can provide statistically similar digital microstructures of heterogeneous materials for a further numerical investigation quickly and efficiently.

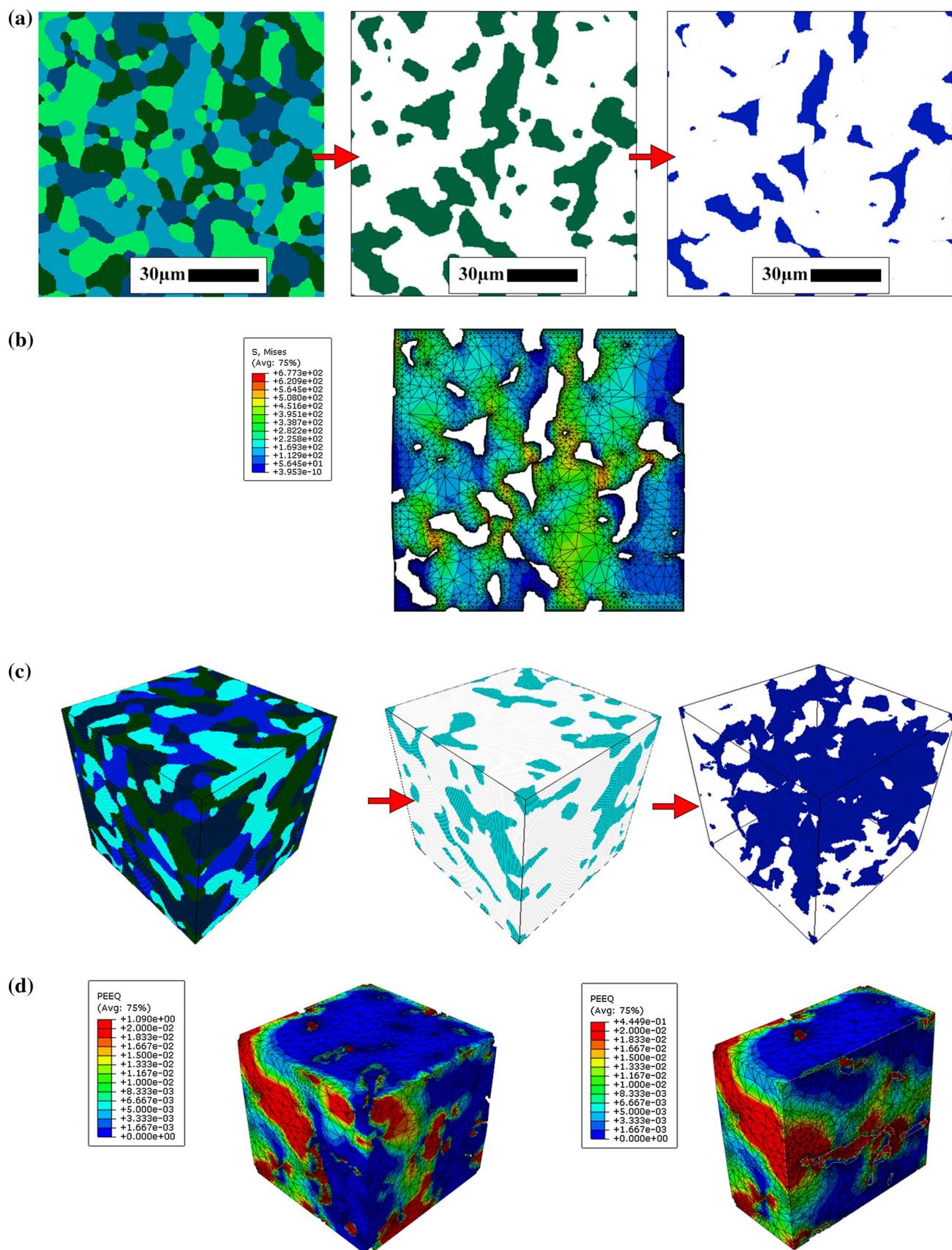


Fig. 15 Workflow for digital material morphology of porous sinter in **a** 2D and **c** 3D generated with presented DigiCore software and **b**, **d** corresponding FE results of loading conditions

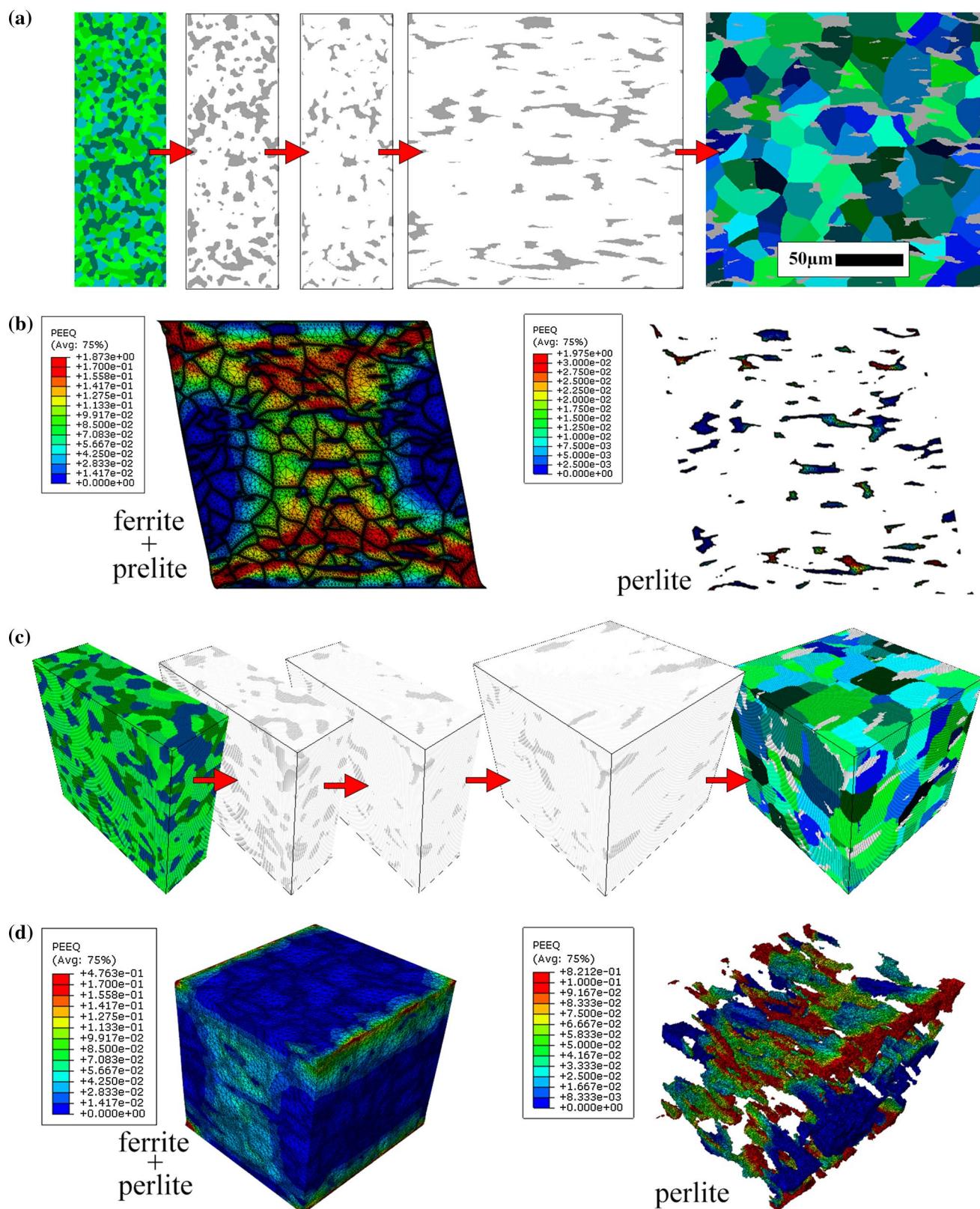


Fig. 16 Workflow for digital material morphology of ferrite-pearlite steel in **a** 2D and **c** 3D generated with presented DigiCore software and **b**, **d** corresponding FE results of loading conditions

Acknowledgment Financial assistance of the National Science Centre project No. 2017/27/B/ST8/00373 is acknowledged.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Pietrzyk M, Madej L, Rauch L, Szeliga D. Computational materials engineering achieving high accuracy and efficiency in metals. 2015.
- Gawad J, Kuziak R, Madej L, Szeliga D, Pietrzyk M. Identification of rheological parameters on the basis of various types of compression and tension tests. *Steel Res Int*. 2005;76:131–7.
- Szeliga D, Gawad J, Pietrzyk M. Inverse analysis for identification of rheological and friction models in metal forming. *Comput Methods Appl Mech Eng*. 2006;195:6778–98. <https://doi.org/10.1016/j.cma.2005.03.015>.
- Lacroix G, Furnemont Q, Jacques PJ, Pardoen T. Mechanisms of damage and fracture in trip assisted multiphase steels. In: Fracture of nano and engineering materials and structures. Dordrecht: Springer; 2006. pp. 819–820.
- Estrin Y, Vinogradov A. Extreme grain refinement by severe plastic deformation: a wealth of challenging science. *Acta Mater*. 2013;61:782–817. <https://doi.org/10.1016/j.actamat.2012.10.038>.
- Pereloma E, Beladi H, Zhang L, Timokhina I. Understanding the behavior of advanced high-strength steels using atom probe tomography. *Metall Mater Trans A*. 2012;43:3958–71. <https://doi.org/10.1007/s11661-011-0782-0>.
- Muszka K, Lopez-Pedrosa M, Raszka K, Thomas M, Rainforth WM, Wynne BP. The impact of strain reversal on microstructure evolution and orientation relationships in Ti-6Al-4V with an initial alpha colony microstructure. *Metall Mater Trans A*. 2014;45:5997–6007. <https://doi.org/10.1007/s11661-014-2590-9>.
- Madej L, Sitko M, Radwanski K, Kuziak R. Validation and predictions of coupled finite element and cellular automata model: Influence of the degree of deformation on static recrystallization kinetics case study. *Mater Chem Phys*. 2016;179:282–94. <https://doi.org/10.1016/j.matchemphys.2016.05.040>.
- Scholtes B, Shakoob M, Settefrati A, Bouchard PO, Bozzolo N, Bernacki M. New finite element developments for the full field modeling of microstructural evolutions using the level-set method. *Comput Mater Sci*. 2015;109:388–98. <https://doi.org/10.1016/j.commatsci.2015.07.042>.
- Vondrouš A, Bienger P, Schreijäg S, Selzer M, Schneider D, Nestler B, Helm D, Möning R. Combined crystal plasticity and phase-field method for recrystallization in a process chain of sheet metal production. *Comput Mech*. 2015;55:439–52. <https://doi.org/10.1007/s00466-014-1115-0>.
- Liu J, Dai Q, Chen J, Chen S, Ji H, Dua W, Deng X, Wang Z, Guo G, Luo H. The two dimensional microstructure characterization of cemented carbides with an automatic image analysis process. *Ceram Int*. 2017;43:14865–72. <https://doi.org/10.1016/j.ceramint.2017.08.002>.
- Falco S, Jiang J, De Cola F, Petrinic N. Generation of 3D polycrystalline microstructures with a conditioned Laguerre-Voronoi tessellation technique. *Comput Mater Sci*. 2017;136:20–8. <https://doi.org/10.1016/j.commatsci.2017.04.018>.
- Lewandowska M, Wejrzanowski T, Kurzydłowski KJ. Grain growth in ultrafine grained aluminium processed by hydrostatic extrusion. *J Mater Sci*. 2008;43:7495–500. <https://doi.org/10.1007/s10853-008-2808-6>.
- Bakhtiari M, Seyed Salehi M. Reconstruction of deformed microstructure using cellular automata method. *Comput Mater Sci*. 2018;149:1–13. <https://doi.org/10.1016/j.commatsci.2018.02.053>.
- Madej L, Legwand A, Mojzeszko M, Chraponski J, Roskosz S, Cwajna J. Experimental and numerical two- and three-dimensional investigation of porosity morphology of the sintered metallic material. *Arch Civil Mech Eng*. 2018;18:1520–34. <https://doi.org/10.1016/j.acme.2018.06.007>.
- Wang P, He W, Mauer G, Mücke R, Vaßen R. Monte Carlo simulation of column growth in plasma spray physical vapor deposition process. *Surf Coat Technol*. 2018;335:188–97. <https://doi.org/10.1016/j.surfcoat.2017.12.023>.
- Villaret F, Harry B, de Carlan Y, Baudin T, Loge R, Maire L, Bernacki M. Probabilistic and deterministic full field approaches to simulate recrystallization in ODS steels. *Comput Mater Sci*. 2020;179:109646.
- Wejrzanowski T, Lewandowska M, Sikorski K, Kurzydłowski KJ. Effect of grain size on the melting point of confined thin aluminum films. *J Appl Phys*. 2014. <https://doi.org/10.1063/1.4899240>.
- Burczyński T, Kuś W, Brodacka A. Multiscale modeling of osseous tissues. *J Theor Appl Mech*. 2010;48:855–70.
- Makowski P, Kuś W. Optimization of bone scaffold structures using experimental and numerical data. *Acta Mech*. 2016;227:139–49. <https://doi.org/10.1007/s00070-015-1421-4>.
- Madej L. Digital/virtual microstructures in application to metals engineering—a review. *Arch Civil Mech Eng*. 2017;17:839–54. <https://doi.org/10.1016/j.acme.2017.03.002>.
- Groober MA, Jackson MA. DREAM 3D: a digital representation environment for the analysis of microstructure in 3D. *Integr Mater Manuf Innov*. 2014;3:1–17. <https://doi.org/10.1186/2193-9772-3-5>.
- Cao J, Zhuang W, Wang S, Lin J. Development of a VGRAIN system for CPFE analysis in micro-forming applications. *Int J Adv Manuf Technol*. 2010;47:981–91. <https://doi.org/10.1007/s00170-009-2135-3>.
- Bernacki M, Digonnet H, Resk H, Coupez T, Logé R. Development of numerical tools for the multiscale modelling of recrystallization in metals, based on a digital material framework. *Comput Methods Mater Sci*. 2007;7:142–9. <https://doi.org/10.1063/1.2740840>.
- TIOBE Index (2019). <https://www.tiobe.com/tiobe-index/>. Accessed May 7, 2019.
- Madej L, Sitko M, Pietrzyk M. Perceptive comparison of mean and full field dynamic recrystallization models. *Arch Civil Mech Eng*. 2016;16:569–89. <https://doi.org/10.1016/j.acme.2016.03.010>.
- Sitko M, Dybich D, Szyndler J, Madej L. Parallelization of the Monte Carlo grain growth algorithm. *Mater Sci Technol*. 2013;3:1657–67.
- Hajder L, Madej L. Sphere packing algorithm for the generation of digital models of polycrystalline microstructures with heterogeneous grain sizes. *Comput Methods Mater Sci*. 2020;20:22–30.
- Lyu H, Hamid M, Ruimi A, Zbib HM. Stress/strain gradient plasticity model for size effects in heterogeneous nano-microstructures. *Int J Plast*. 2017;97:46–63. <https://doi.org/10.1016/j.ijplas.2017.05.009>.
- Madej L, Pasternak K, Szyndler J, Wajda W. Development of the modified cellular automata sphere growth model for creation of the

- digital material representations. Key Eng Mater. 2014;611:489–96. <https://doi.org/10.4028/www.scientific.net/KEM.611-612.489>.
31. Sitko M, Mojzeszko M, Rychlowski L, Cios G, Bala P, Muszka K, Madej L. Numerical procedure of three-dimensional reconstruction of ferrite-pearlite microstructure data from SEM/EBSD serial sectioning. Proc Manuf. 2020;47:1217–22. <https://doi.org/10.1016/j.promfg.2020.04.183>.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.