Materials Research. 2025; 28:e20250204
DOI: https://doi.org/10.1590/1980-5373-MR-2025-0204

Study of Recrystallization in Metals through Equiaxial Growth of Nuclei by Computational Simulation with Three-Dimensional Hybrid Cellular Automata

***Henrique Costa Braga**[a]**, ***Sidney Nicodemos da Silva**[a]*

[a]*Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG), Programa de Pós-Graduação em Engenharia de Materiais, Belo Horizonte, MG, Brasil.*

Received: January 29, 2025; Revised: March 20, 2025; Accepted: April 21, 2025

Cellular Automata (CA) are powerful simulation tools that operate through discrete elements and associations. CA undirected neighborhood searches may produce specific grain shapes (e.g., octahedral, cuboctahedral, cubic), but not spherical ones. However, in recrystallization modeling, the spherical shape plays a crucial role. Thus, Hybrid Cellular Automata (HCA) share several properties with CA but operate in both continuous and discrete modes simultaneously. Unlike deterministic CA, HCA enable equiaxial grain growth without the need for additional correction algorithms, generating grains that remain initially spherical until they encounter boundaries. This behavior has been addressed by other simulation techniques, but HCA provide a simple and highly effective alternative. In this work, HCA are described in detail, including their foundational principles, algorithm performance, calibration, and potential results. The findings highlight the capability of HCA for straightforward and accurate recrystallization simulations with equiaxial growth.

Keywords: *Recrystallization, equiaxied structure, hybrid cellular automata, cellular automata.*

1. Introduction

The study of phenomena such as recrystallization in metals holds broad industrial relevance due to its significant influence on production processes and its direct impact on the mechanical, thermal, and physical properties of metals[1]. The microstructure of metallic materials typically consists of various phases and defects, forming an intricate network of distinct grains with complex geometries.

These grains can undergo recrystallization, often adopting different shapes, a topic of extensive scientific debate and research[2-4]. Among these shapes, equiaxed grain growth plays a crucial role in numerous scenarios, making its simulation and analysis highly pertinent in the field of Materials Engineering.

Various computational modeling techniques are available for simulating recrystallization in metals, such as cellular automata (CA), the Monte Carlo method, and the causal-cone approach, among others[5-8]. No single modeling approach is universally applicable, as each technique offers unique features tailored to specific applications. Furthermore, this is a rapidly evolving field, with ongoing advancements that continuously refine these methods.

Many modeling approaches share a common framework: the metallic material to be simulated is represented as a matrix, with each cell (or site) corresponding to a small representative portion of the material[9,10]. The crystalline state of each cell defines its properties, while seeds are distinguished by states differing from their surroundings.

Among these approaches, CA modeling remains one of the most established techniques[11-13] and continues to be widely employed due to its computational efficiency and robust results[14,15]. Cellular automata operate on a discrete model, comprising a finite, regular matrix of cells. Each cell occupies a finite set of possible states that evolve according to pre-defined rules[16-18]. The evolution of each cell depends on the states of its neighbors and, in some cases, its own state.

By applying transition rules iteratively to the matrix, a new state is generated for the system based on boundary conditions and input parameters[9,10]. The transition rules, often encoded in a truth table, govern the automaton's behavior, defining outcomes for all possible neighborhood configurations.

In the matrix, some cells, due to their value and spatial position, possess the potential for self-replication. This is achieved through a process of neighborhood search and exploration governed by a generating algorithm. The effectiveness of this process depends critically on the search strategy employed within the cell neighborhoods.

In two-dimensional CA modeling, non-oriented first neighborhoods like the Von Neumann and Moore neighborhoods are most commonly utilized[19,20]. In three-dimensional modeling, however, the complexity increases, allowing for various configurations of non-oriented first neighborhoods[21,22].

Three primary configurations of first-neighborhood searches have emerged in three-dimensional CA studies of recrystallization[22]:

1. Face Neighboring (FN): Search limited to directly adjacent faces, resulting in grains with octahedral shapes.
2. Cuboctahedral Neighboring (CN): Search extending to adjacent faces and edges, resulting resulting in grains with cuboctahedral shapes.

*e-mail: bragaseg@yahoo.com.br

3. Total Neighboring (TN): Search encompassing adjacent faces, edges, and vertices, resulting in grains with cubic shapes.

The FN search demonstrates the highest efficiency (lowest computational cost), being conceptually the simplest and considering the ratio between the number of searched and transformed cells. However, the CN search is the one that most closely approximates a spherical shape, although it still deviates significantly[22].

It is important to note that these grain shapes arise during recrystallization from a single seed before being constrained by neighboring grains, barriers, or boundaries. Additionally, these shapes are described within the context of a discrete three-dimensional environment, reflecting the inherent geometry of the computational matrix.

Despite the effectiveness of pure deterministic CA methods in modeling seed growth within specific geometries, they cannot directly generate spherical grains. This limitation makes it challenging to simulate equiaxed grain growth, which requires a spherical morphology.

To address this limitation, this study extends the traditional CA framework by incorporating hybrid elements, resulting in a Hybrid Cellular Automaton (HCA). There are various approaches to achieving this, depending on the type of automaton and the parameter chosen for hybridization[23-25]. In the specific case of this work, hybridization occurs through the use of a continuous mathematical equation as a conditional criterion for the evolution transition rule, where discrete evolution is governed by a continuous mathematical function.

This hybrid approach relaxes some of the strict rules of traditional CA, such as uniformity in state evolution, allowing for greater flexibility and accuracy. As a result, HCA retains many advantages of CA while offering enhanced modeling capabilities.

The integration of HCA facilitates the direct simulation of spherical grains with a higher level of realism, eliminating the need for post-processing corrections or control procedures to mitigate errors.

In this study, HCA is applied to computationally model equiaxed grain growth in metals. The quality of the results, as well as the performance and calibration of the generating algorithm, are analyzed and discussed.

This study is organized as follows: Section 1 presents the introduction, outlining the motivation and scope of the work. Section 2 addresses methodological aspects. Section 3 presents the results and discussion on the generation of spherical grains and the analysis of their performance. Finally, Section 4 presents the main conclusions.

2. Methodology

All experiments were conducted through computational simulations. The programming language used for the entire implementation was Python 3.12.1, supported by its native libraries such as NumPy and Matplotlib. The program was developed following a procedural approach.

The algorithm presented here was computationally implemented on a personal computer (operating system: Microsoft Windows® 10, 64-bit; processor: Intel® Core™ i7-2600 CPU @ 3.40 GHz; RAM: 12.0 GB).

The technique used in the simulation was based on three-dimensional, deterministic hybrid cellular automata modeling within a rectangular matrix of cubic cells. The matrix is non-periodic, meaning that its boundaries do not wrap around, and no interaction occurs with the external environment[16,26]. The searches (and when applicable, replications) are performed only on the first neighborhoods through the faces (FN) of the latest automata generated.

This neighborhood criterion was chosen because, in addition to being the simplest form of search, it is the most computationally efficient[22]. It is true that among the non-oriented first neighborhoods, FN generates significant errors due to the natural octahedral matrix shape of its search. However, due to the hybridization of the modeling, this negative characteristic will not be relevant and will not affect the dimensional results of the formed grains.

The global simulation program (SimCres v. 1.0) consists of three modules: the recrystallization module (growth of the new grain(s)), the parameterization module, and the visualization module. These modules are completely independent of each other. Thus, during the recrystallization simulation, no form of control or direction is applied to the algorithm's performance, its results, or the transforming ambient matrix. Figure 1 illustrates the basic structure of the program and the interrelation between the modules.

In the recrystallization module, a regular three-dimensional matrix with cubic cells is initially created to represent the hypothetical metal under study. This metal is initially considered entirely homogeneous, meaning all matrix cells will have the same state, represented by a single value. The nucleation process adopted is site saturation[10].

A selected number of seeds is generated within the material prior to the start of the simulation. Each seed occupies a single cell and is assigned a distinct state (or value) to differentiate it from the cells of the original structure. This creates the simulation environment matrix, consisting of cells representing the original structure and those identified as seeds.

During the simulation conducted by the growth module, the only actions performed are saving the evolution of the environment matrix in separate files, each corresponding to a specific preselected simulation time step, and displaying the simulation progress on the screen. Once the simulation is complete, a file with the general data of the process is saved.

The parameterization module reads these previously saved files and performs various calculations to better

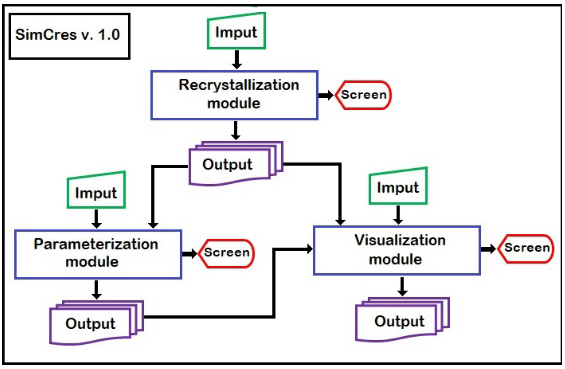


Figure 1. Basic structure of the SimCres program.

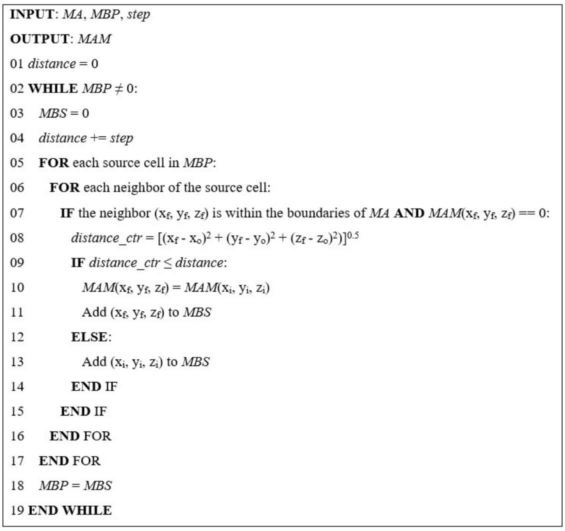
understand the results, obtaining multiple parameters related to both the program's evolution and dimensional aspects. Some images from these calculations can also be generated and saved if desired.

Finally, the visualization module simply provides 3D visualizations and/or views of selected orthogonal planes of the entire environment or specific regions by reading the saved files. No additional external programs are used to process the generated images or assist in their visualization.

The main core of the computationally implemented algorithm for search, exploration, and replication is summarized in the pseudocode shown in Figure 2.

Where:

- (x_o, y_o, z_o) are the coordinates of the seed's origin cell;
- (x_i, y_i, z_i) are the coordinates of the current source cell being processed in the loop;
- (x_f, y_f, z_f) are the coordinates of the neighboring cell to the source cell currently being analyzed;
- *j* is the number of direct neighbors considered (since the search is through faces, there are six possible first neighbors);
- *MA* is the initial 3D matrix representing the metal before the simulation;
- *MAM* is the 3D matrix representing the metal during recrystallization;
- *MAM*(x_i, y_i, z_i) contains the seed number of the source cell being considered;
- *MBP* is the primary search matrix containing the coordinates of all source cells in a given loop;
- *MBS* is the secondary search matrix containing the coordinates of all cells in *MAM* whose values were modified during a given loop;
- step is the incremental evolution of the reference distance in each loop (in this study, *step* = 0.5 is used for generating spherical grains);
- *distance* is the total number of cells evolved from the origin cell along one axis; and
- *distance_ctr* is the linear distance between the center of the cell being checked and the center of the original seed cell (a non-discrete quantity).



```
INPUT: MA, MBP, step
OUTPUT: MAM
distance = 0
WHILE $MBP \neq 0$:
  MBS = 0
  distance += step
  FOR each source cell in MBP:
    FOR each neighbor of the source cell:
      IF the neighbor ($x_f$, $y_f$, $z_f$) is within the boundaries of MA AND MAM($x_f$, $y_f$, $z_f$) == 0:
        $distance\_ctr = [(x_f - x_o)^2 + (y_f - y_o)^2 + (z_f - z_o)^2)]^{0.5}$
        IF $distance\_ctr \leq distance$:
          MAM($x_f$, $y_f$, $z_f$) = MAM($x_i$, $y_i$, $z_i$)
          Add ($x_f$, $y_f$, $z_f$) to MBS
        ELSE:
          Add ($x_i$, $y_i$, $z_i$) to MBS
        END IF
      END IF
    END FOR
  END FOR
  MBP = MBS
END WHILE
```

Figure 2. Pseudocode of the core routine for search, exploration, and replication of the implemented generator algorithm.

This implemented generator algorithm is then applied to the fundamental environment matrix containing the seeds. Initially, the algorithm performs a search on the cells in the first neighborhood by the faces of the seeds. If the state of the directly explored neighboring cells matches the characteristic state of the parent structure, the state of these neighboring cells is changed to the characteristic state of the originating seed of the grain.

After this first loop, the implemented generator algorithm is applied again, but this time not to the seed cells, but to those cells whose values were altered (the new boundaries of the recrystallized grains in evolution). The algorithm then searches the cells in the first neighborhood by the faces of these new grain boundary cells. Just as in the first loop, the state of these neighboring cells belonging to the initial microstructure will be altered.

This process is then repeated successively until no cells in the first neighborhood of the formed grains remain to be searched (i.e., the entire environment has been transformed).

Finally, the 3D images generated from the simulated microstructures, along with their forming planes (XY, YZ, and XZ), used in this study will always follow the Cartesian orientation depicted in Figure 3.

3. Results and Discussions

3.1. Generation of spherical-shaped grains

To evaluate the methodology's suitability for obtaining the matrix-based spherical shape of the grains during formation, simulations of the growth of a single grain were conducted in environments of 21x21x21 cells and 301x301x301 cells. In

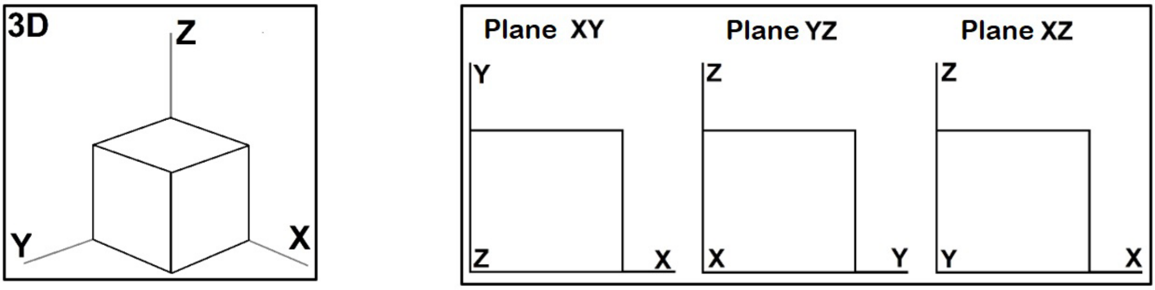


Figure 3. Visual Cartesian orientation patterns used to represent the 3D microstructure matrix and the forming planes XY, YZ, and XZ.

these simulations, the seed was initially located at the center of the environment. Figure 4 presents some of the results.

As visually verified, the results were excellent. Unlike what occurs with the first-neighbor search in a non-oriented CA model, using a HCA approach makes it entirely possible to directly achieve an exact spherical shape - of course, within the precision limits of a discrete environment (matrix-based form).

To better visualize the effects of the shape of multiple grains growing simultaneously during recrystallization, simulations were performed in an environment consisting of 200x200x200 cells (a total of 8 million cells), with 10, 100, and 1000 seeds randomly distributed.

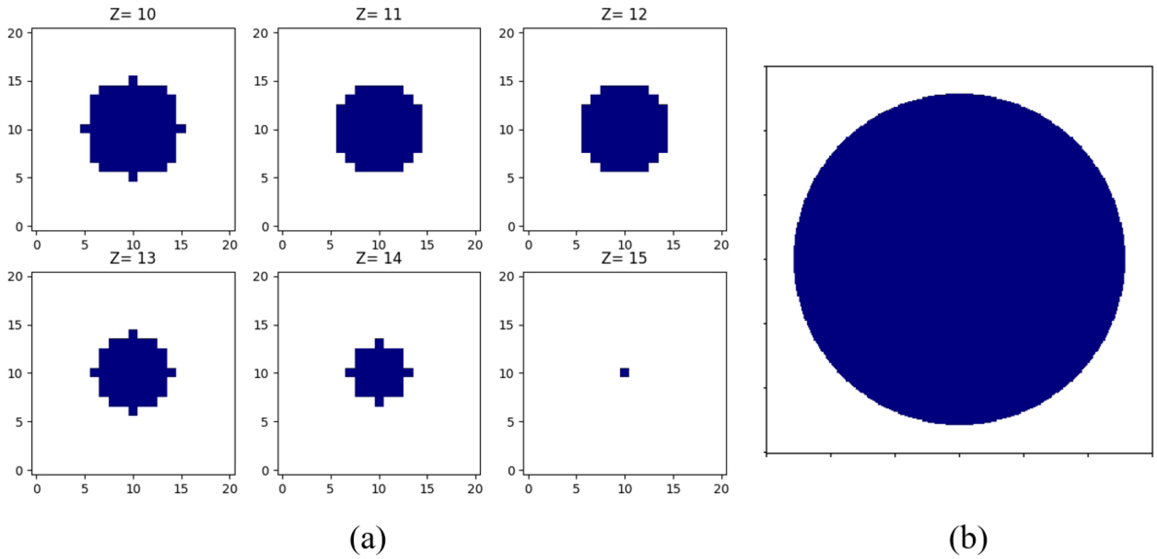
Figures 5 and 6 present the results for simulations with 10 seeds, Figures 7 and 8 show results for simulations with 100 seeds, and Figures 9 and 10 display results for simulations with 1000 seeds.

Finally, Figure 11 presents the final results of the central planes from the simulations with 10, 100, and 1000 seeds (Figures 5, 7 and 9, respectively).

When visually comparing the images of equiaxial seed recrystallization simulations with randomly distributed seeds in Figure 11 to some illustrative images of single-phase polycrystalline microstructures observed under optical microscopy[27], a qualitatively reasonable similarity can be observed.

3.2. Generation of regular microstructural patterns

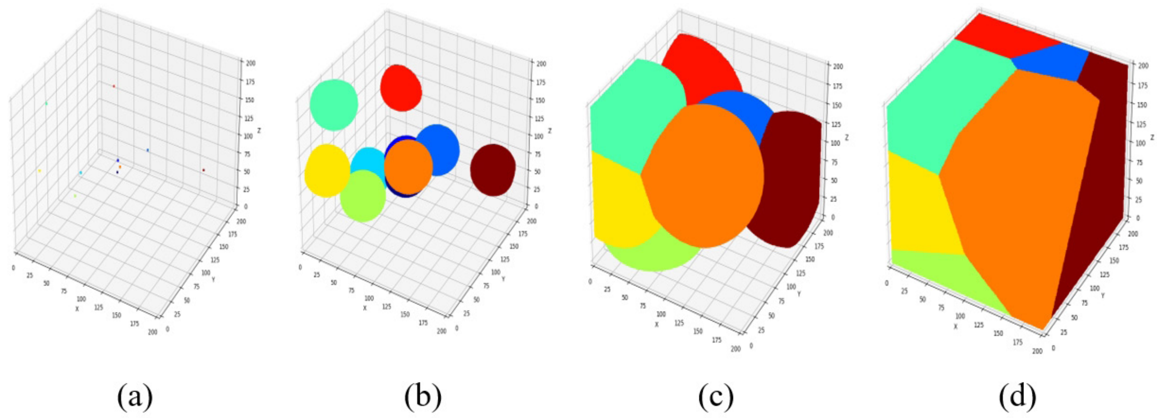
When the grain distribution is not random, it is possible to achieve specific and regular microstructural grain formation patterns. While this behavior is expected—since cubic grains have been observed in various forms of CA modeling[22] - a few illustrative simulations were conducted to confirm and



(a)

(b)

Figure 4. Images of simulation snapshots in an environment initially containing a single seed at its center. Legend: (a) Image of the xy plane with z varying between 10 and 15, in a 21x21x21 cell environment, after loop number 17. (b) Image of the central plane, in a 301x301x301 cell environment, with 33% of cells transformed.



(a)

(b)

(c)

(d)

Figure 5. Simulation in a 200x200x200-cell environment with 10 randomly distributed seeds, shown at different stages: (a) initial state, (b) 10% of the volume transformed, (c) 70% of the volume transformed, and (d) final state.

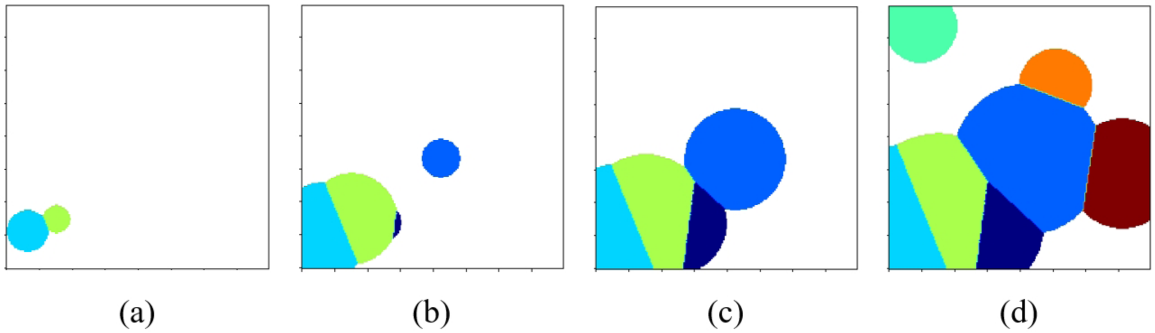


Figure 6. Images of the xz plane at y=100 for various moments during the simulation from Figure 5 (10 seeds), shown at different stages: (a) 10% of the volume transformed, (b) 30% of the volume transformed, (c) 50% of the volume transformed, and (d) 70% of the volume transformed.

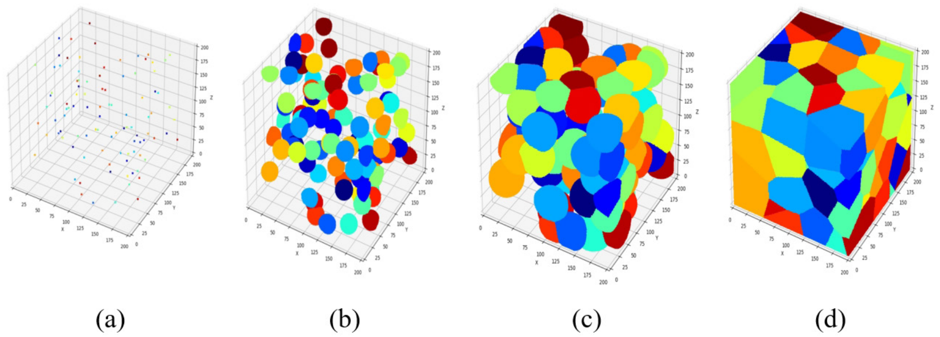


Figure 7. Simulation in a 200x200x200-cell environment with 100 randomly distributed seeds at different stages of the simulation, shown at different stages: (a) initial state, (b) 10% of the volume transformed, (c) 50% of the volume transformed, and (d) final state.

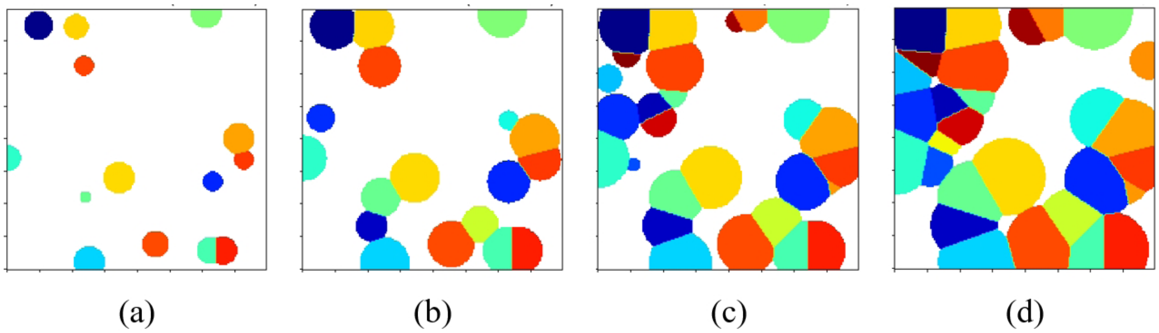


Figure 8. Images of the xz plane at y=100 for various moments during the simulation from Figure 7 (100 seeds), shown at different stages: (a) 10% of the volume transformed, (b) 30% of the volume transformed, (c) 50% of the volume transformed, and (d) 70% of the volume transformed.

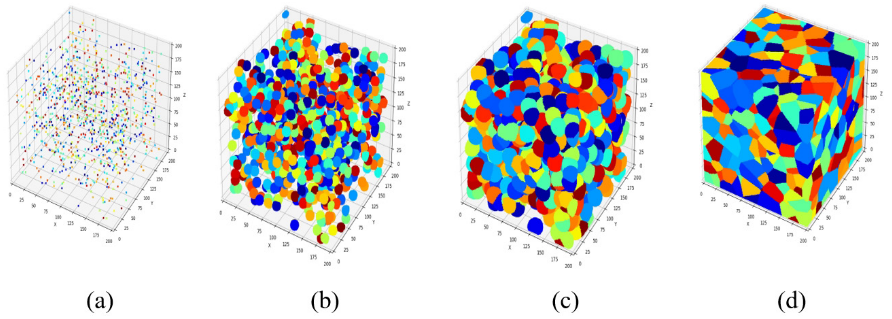


Figure 9. Simulation in a 200x200x200-cell environment with 1000 randomly distributed seeds at different stages of the simulation, shown at different stages: (a) initial state, (b) 10% of the volume transformed, (c) 50% of the volume transformed, and (d) final state.

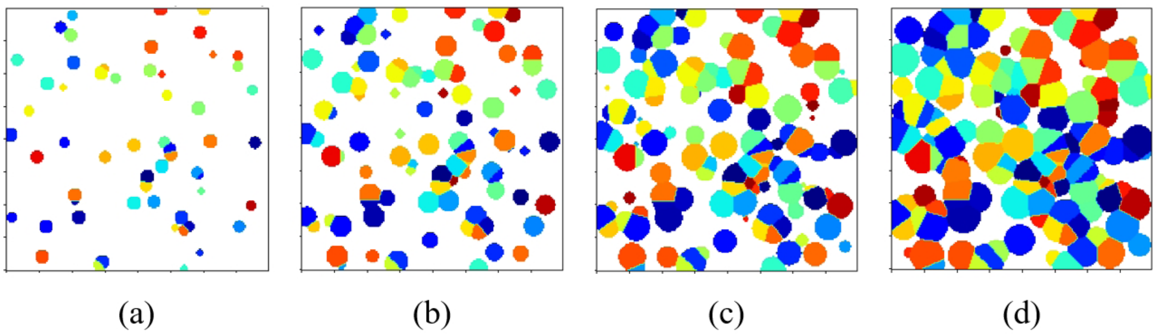


Figure 10. Images of the xz plane at y=100 for various moments during the simulation from Figure 9 (1000 seeds), shown at different stages: (a) 10% of the volume transformed, (b) 30% of the volume transformed, (c) 50% of the volume transformed, and (d) 70% of the volume transformed.

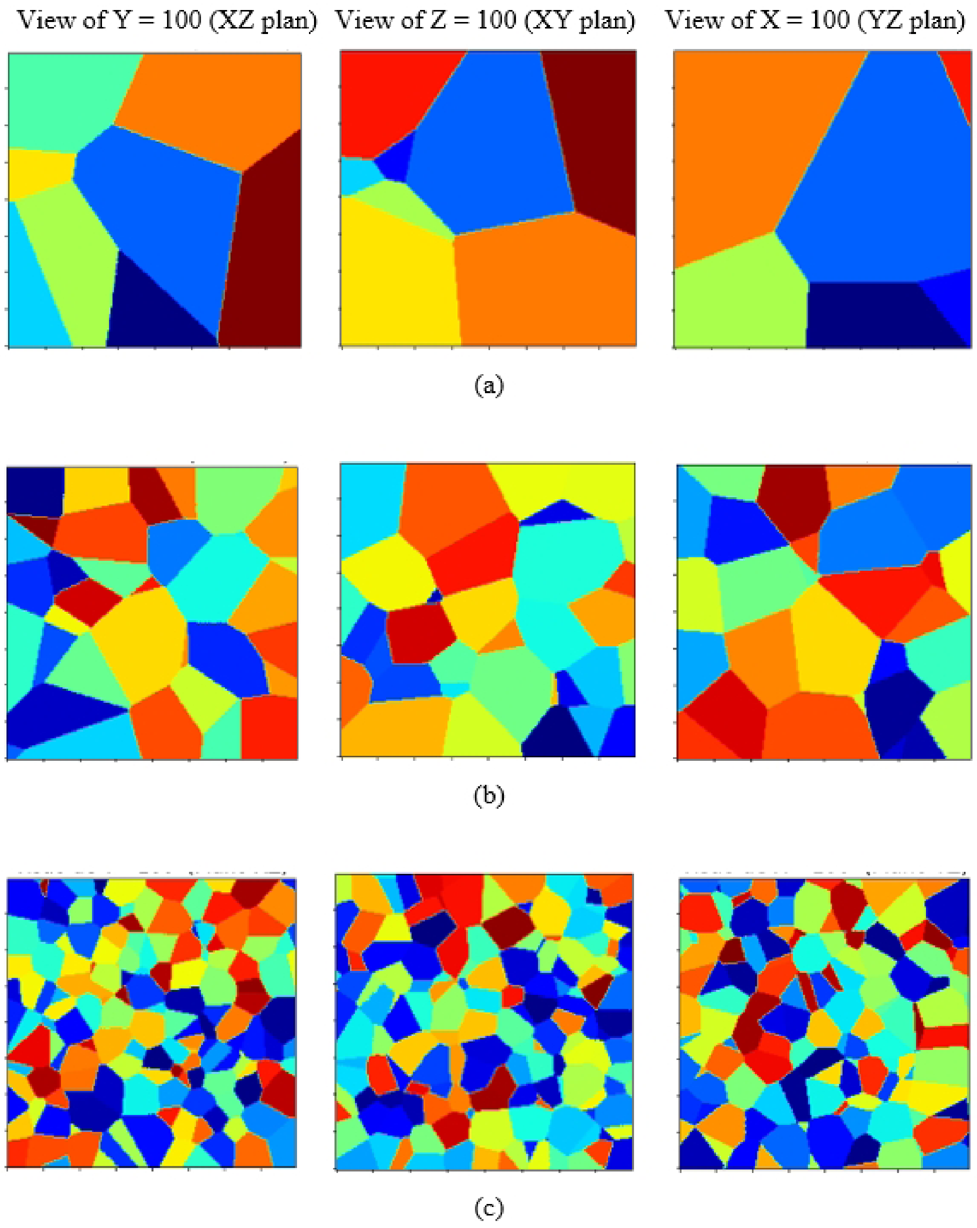

View of Y = 100 (XZ plan)
View of Z = 100 (XY plan)
View of X = 100 (YZ plan)
(a)
(b)
(c)

Figure 11. Images of the central planes from the 200x200x200 environment at the end of the simulations in Figures 5, 7 and 9, with respectively 10 (a), 100 (b), and 1000 seeds (c).

demonstrate the occurrence of such patterns in the presented hybrid automaton (HCA) modeling.

To generate cubic grains, the seeds were positioned in a repeating pattern analogous to a simple cubic (SC) structure. The distance between seeds was set equal to the edge length of the cubic grains formed. Since the HCA modeling results in truly spherical grains at the matrix level - unlike CA modeling - at the moment of collision between growing grains, the appearance of the recrystallized microstructure will closely resemble an SC structure.

To generate parallelepipeds with a square face and their other dimension extending across the full matrix length, it suffices to use only one layer of seeds from the arrangement created for cubic grains.

For the generation of orthotetradecahedrons (OTD) within a Kelvin lattice, the seeds were arranged in a repeating pattern resembling a body-centered cubic (BCC) structure. Similar to the cubic grain microstructure, just before collisions between growing grains occur, the recrystallized microstructure reflects its initial arrangement—in this case, a BCC structure.

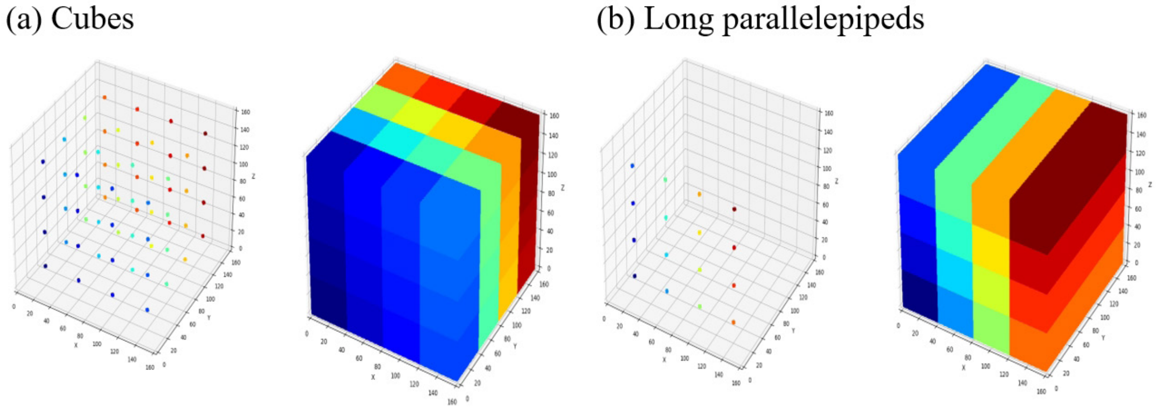
Figures 12 and 13 illustrate the results of these simulations.

3.3. *Verification of performance and calibration of the generator algorithm*

Although the quality of the HCA method for simulating spherical matrix growth has been verified, it is still important to gather information about the performance and behavior of the generator algorithm. For this, simulations were conducted in two different environments: one with 21x21x21 cells, and another with 201x201x201 cells, but now for various step values (0.25, 0.50, 0.60, 0.75, and 1.00). In both simulations, only a single seed was placed at the center of the environment, with no internal obstructions.

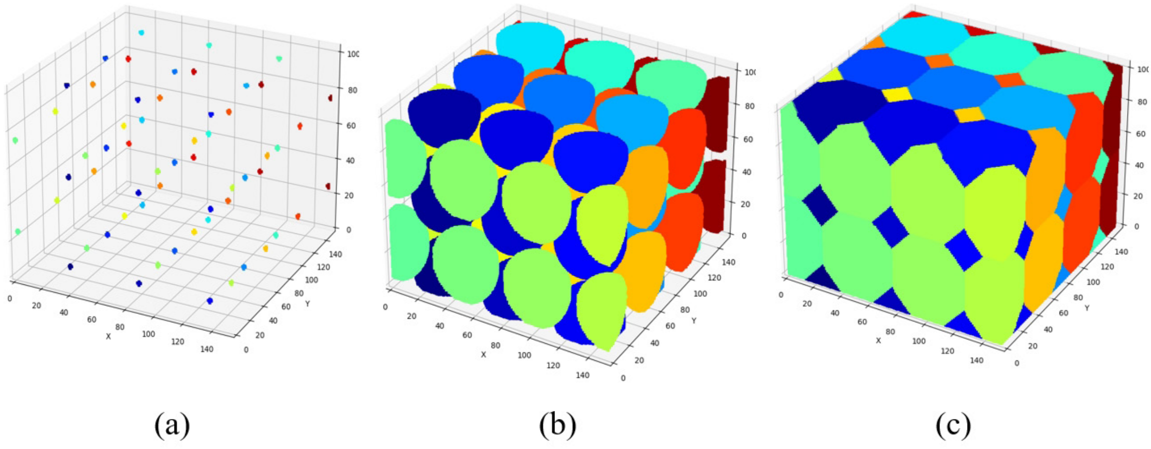
Table 1 presents the results found, and Figures 14 to 17 illustrate some performance data from the simulation for these trials in the 201x201x201 cell environment.

As presented in the Methodology, except for the simulations listed in Table 1, all other simulations in this work used a step size of 0.50. By examining the data from Table 1 and Figures 14 to 17, it is clear that this choice was well-founded for several reasons.


(a) Cubes

(b) Long parallelepipeds

Figure 12. Simulations in environments with 160x160x160 cells, with seeds distributed according to specific profiles. In (a), 64 seeds form cubes, and in (b), 16 seeds form elongated parallelepipeds.


(a)

(b)

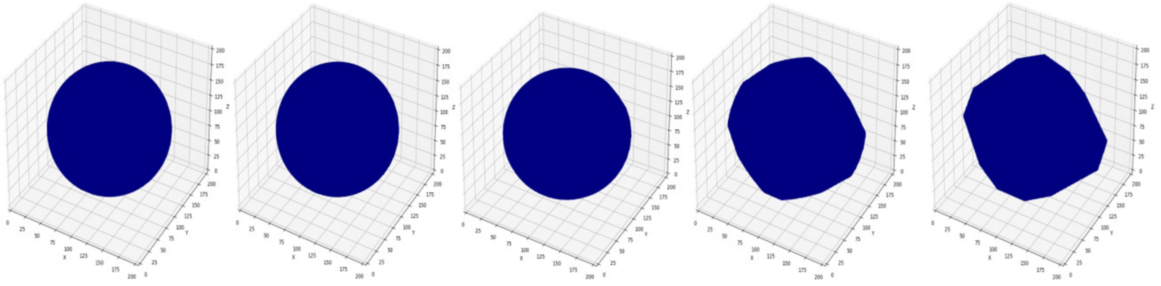
(c)

Figure 13. Simulation in an environment with 152x152x102 cells, with 59 seeds distributed according to the specific BCC profile, resulting in a microstructure of OTDs), shown at different stages: (a) initial state, (b) Intermediate illustrative moment (prior to impingement), and (c) final state (OTD's).

Table 1. Complete simulations in environments with 21x21x21 cells and 201x201x201 cells, starting with a single cell located at the center, for various step values.

Environment (Number total of cells)	*Step*	Total Number of Loops	Total Number of Transformed Cells			
			Transformeds		Searched	
			Number	(%)	Number	(%)
21x21x21 (9,261)	0.25	67	9,261	100	161,490	1744
	0.50	34			82,140	887
	0.60	31			67,374	728
	0.75	30			56,958	615
	1.00	30			55,344	598
201x201x201 (8,120,601)	0.25	691	8,120,601	100	153,353,928	1888
	0.50	347			76,809,120	946
	0.60	301			62,451,360	769
	0.75	301			49,924,848	615
	1.00	301			48,721,806	600

Note: percentages relative to the total number of cells in the environment.



step 0.25 *step* 0.50 *step* 0.60 *step* 0.75 *step* 1.0

Figure 14. Image of the matrix sphere, or the object, generated for the simulation in the 201x201x201 cell environment, with a single central seed, for various step values, at the moments with 33% transformation of the environment.

Looking at Figure 14, for the step size of 0.75, the object formed was an ellipsoid, only minimally resembling a sphere (in fact, the object is an irregular mixture of the sphere generated by hybridization with the octahedron formed by the search for the FN). For the step size of 0.60, the object was very close to a sphere, but still not perfect. For step sizes of 0.50 and 0.25, the objects visually formed were perfect matrix spheres.

Figure 15 provides information about the number of cells searched per loop. Initially, the curve shows continuous growth. This occurs because the searched cells are along the outer surface of the formed object, and as the object grows in size, the external area also increases with each loop of the simulation, causing the number of cells searched to follow this growth. However, at some point, the number of cells searched reaches a maximum, after which it begins to decrease.

This phenomenon is similar to what is observed in CA for FN evolution[22]. Since the object formed by the replication of cells does not coincide with the shape of the simulation matrix, at some point, the object being formed will touch the surfaces, causing the growth rate to slow down until a maximum growth is reached, after which the growth rate starts to decline.

One of the most important points in Figure 15 is the possibility of generating Figure 16, the first-order derivative of the number of cells searched per loop by the loop number. As previously noted[22], when modeling by CA using FN, and even by CN and TN, this graph consists of distinct straight-line segments, whose slope represents the growth (or decay) rate of the curve of the number of cells searched per loop. However, in Figure 16, for both the step size of 0.25 and 0.50, the points do not show any apparent linearity, but rather significant variation in their values.

This is caused by hybridization. In Figure 16, for the step size of 0.60, some variability is observed, but not throughout the entire range, indicating that there was partial hybridization, not total hybridization. Finally, in Figure 16 for the step size of 0.75, little hybridization is seen. So far, several positive points for the choice of step size 0.50 or 0.25 have been found. However, Figure 17 will clarify this point.

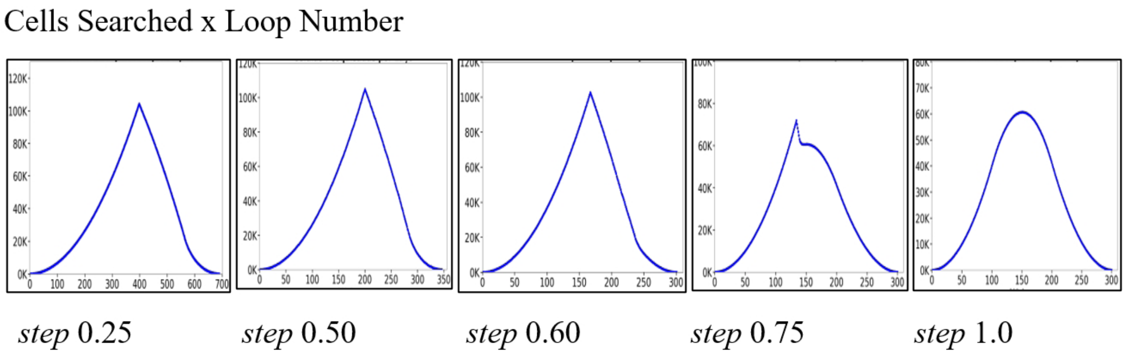


Figure 15. Graph showing the number of cells searched per loop obtained for the simulation in the 201x201x201 cell environment, with a single central seed, for various step values.

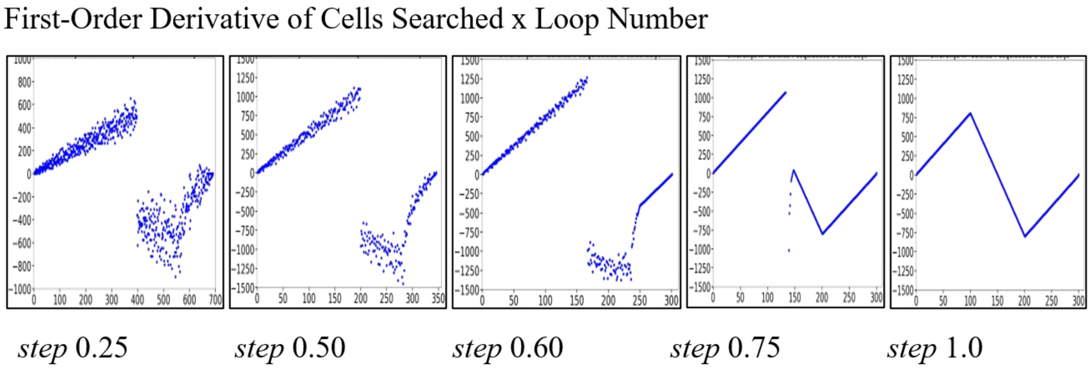


Figure 16. Graph showing the difference in the number of cells searched per loop for the simulation in the 201x201x201 cell environment, with a single central seed, for various step values (first derivative of the graphs in Figure 15).

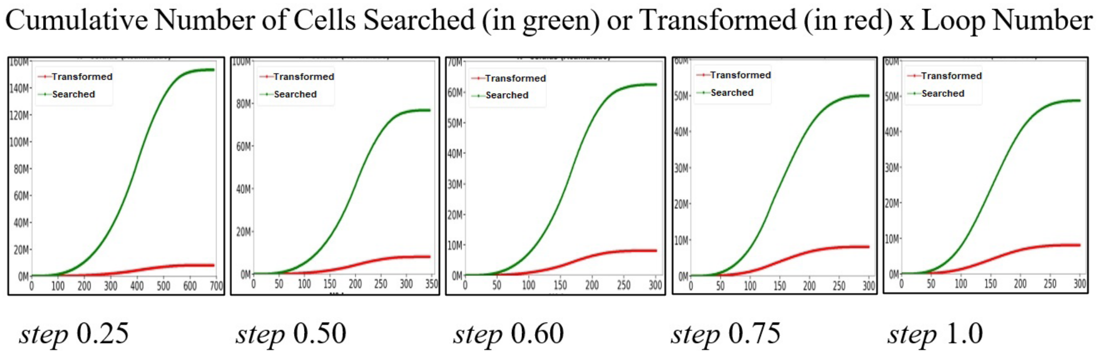


Figure 17. Graph showing the accumulated number of cells searched and transformed per loop for the simulation in the 201x201x201 cell environment, with a single central seed, for various step values.

The curves in Figure 17 look relatively similar in appearance, but the smaller the step size, the greater the number of cells searched to fully transform the environment. Therefore, a reduced step size has a much higher computational cost than using a larger step. Thus, although both step sizes produced excellent results, the step size of 0.50 is preferable compared to 0.25, not because of the quality of the result, as both were similar, but because of the lower computational cost.

Increasing the step size decreases the number of cells searched per loop. However, this decrease in intensity diminishes as the step size increases, until it stabilizes when the number of cells searched equals the value that would occur in the corresponding pure CA (the limiting situation where hybridization no longer occurs). For example, observe the values in Table 1 for the step size of 1.00.

These values are the same as those found when performing this simulation with CA techniques for FN search[22]. This is because all curves obtained with HCA using a step size of 1.00 were identical to those obtained with CA, as at this step size, all hybridization was prevented. In this situation, HCA generated results identical to those of CA with VF search.

Thus, empirically, a step size of 0.50 was found to be appropriate. Figure 18, obtained from data in Table 1, illustrates many of these observed points.

Thus, there is a point to be highlighted. As already discussed, HCAs are fundamentally based on CA, but they cannot, or should not, be classified as a type of CA, as HCAs do not meet all the principles of CA (although there are works that refer to them as modified CAs or similar).

However, at least for the methodology and conditions of the experiment conducted here, it was shown that, in fact, it is more likely that CA is a subclass of HCA. If simply a sufficiently high step size is used, hybridization will not occur even partially.

It is not that HCA will behave similarly to CA; it will be the respective CA according to its neighborhood search evolution. Moreover, by changing the search method in the HCA implemented here, all the corresponding CAs generated by the 3 canonical neighborhoods and 4 combined neighborhoods[22] can be directly obtained by this new HCA generator algorithm. Figure 19 illustrates this feature through an Euler diagram.

Thus, it was found that the HCA modeling can successfully simulate the growth of perfect matrix spheres, making it a technique with great potential for use in simulating recrystallization of equiaxed structures in metals.

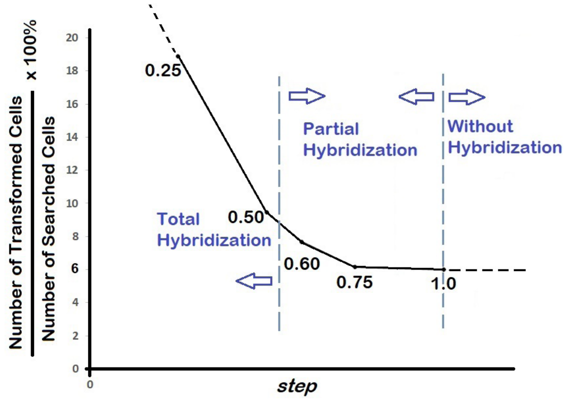


Figure 18. Evolution of the Number of Cells Searched by the Number of Cells Transformed as a function of the step size (environment 201x201x201 cells, data from Table 1, HCA evolving through FN).

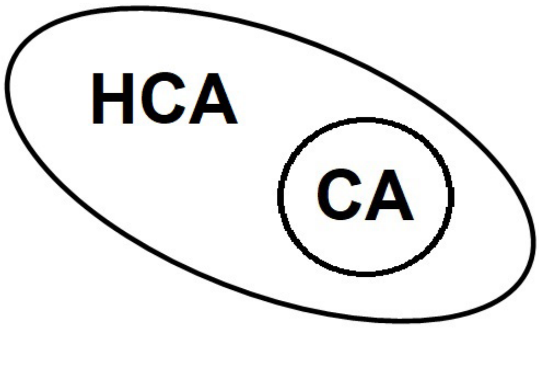


Figure 19. Euler diagram of the proposed relationship between HCA and CA of the type and under the conditions verified in this work.

Finally, up to this point, the simulations have primarily focused on cells and iterations rather than actual physical parameters. However, as future work, the dimensional parameterization of newly formed grains after complete recrystallization is a necessary next step. This will enable the explicit application of the simulation results to real materials and processes, as well as comparisons with other studies. Additionally, various parameters associated with the classical site-saturation Johnson-Mehl-Avrami-Kolmogorov (JMAK) model will be evaluated, such as growth velocity of transformed regions, among others.

4. Conclusions

Cellular Automata (CA) are a powerful simulation tool that operates through discrete elements and associations. On the other hand, Hybrid Cellular Automata (HCA), although their fundamentals are theoretically based on CA, operate in a hybrid mode, both continuous and discrete, simultaneously.

However, with HCA, the distortions in the three-dimensional search space caused by the search through the first non-oriented neighborhoods in pure deterministic CA, which can naturally generate objects in various forms such as octahedra, cuboctahedra, and cubes, do not necessarily interfere with the grain shapes generated by themselves.

Thus, with HCA, it was successfully achieved, without any need for additional correction adjustments or algorithms, while maintaining all the practicality and advantages of using CA, the simulation of the recrystallization of metals through the growth of equiaxed grains, which are perfectly spherical in matrix form during the grain growth stage before contact with any barriers. It is true that this problem has already been solved by various simulation techniques, but HCA proves to be a potentially interesting alternative due to its simplicity combined with excellent results, justifying greater attention to this technique.

Finally, it was found that HCA, depending on the calibration of its parameters, is equivalent to CA, where the search and exploration form is equivalent.

5. References

1. Padilha AF, Siciliano F Jr. Encruamento, recristalização, crescimento de grão e textura. 3. ed. São Paulo: ABM; 2005. 232 p.
2. Hull R, Keblinski P, Lewis D, Maniatty A, Meunier V, Oberai AA, et al. Stochasticity in materials structure, properties, and processing: a review. Appl Phys Rev. 2018;5(1)
3. Glicksman ME, Rios PR, Lewis DJ. Linear measures for polyhedral networks. Int J Mater Res. 2009;100(4):536-42.
4. Romanova VA, Balokhonov RR, Borodina A, Shakhidzhanov VS, Lychagin DV, Emelianova ES, et al. Crystal plasticity finite-element simulations for quasistatic deformation of polucrystals in terms of explicit dynamics. PNRPU Mech Bull. 2023;(5):57-73.
5. Hallberg H. Approaches to modeling of recrystallization. Metals. 2011;1(1):16-48.
6. Zhi Y, Jiang Y, Ke D, Hu X, Liu X. Review on cellular automata for microstructure simulation of metallic materials. Materials. 2024;17(6):1370.
7. Liu X, Zhu J, He Y, Jia H, Li B, Fang G. State-of-the-art review of the simulation of dynamic recrystallization. Metals. 2024;14(11):1230.
8. Payton EJ, Coutinho YA, Gerlt ARC, Simmons JP, Gonzales M, Semiatin SL. Isotropic growth on cartesian voxel grids with

von neumann neighborhoods for rapid generation of synthetic microstructures. Metallogr Microstruct Anal. 2024;13(5):954-65.

9. de Oliveira VT, de Oliveira LP, Rios PR, de Castro JA. Desenvolvimento de um Código em 3D para simular a recristalização pelo método do autômato celular. Tecnol Met Mater. 2006;2(4):34-9.
10. Salazar TC, Assis WL S, Rios PR. Simulation of recrystallization in iron single crystals. Mater Res. 2008;11(1):109-15.
11. Raabe D. Cellular automata in materials science with particular reference to recrystallization simulation. Annu Rev Mater Res. 2002;32(1):53-76.
12. Rios PR, de Oliveira JCPT, de Oliveira VT, de Castro JA. Comparison of analytical models with cellular automata simulation of recrystallization in two dimensions. Mater Res. 2005;8(3):341-5.
13. Srolovitz DJ, Grest GS, Anderson MP, Rollett AD. Computer simulation of recrystallization: II. Heterogeneous nucleation and growth. Acta Metall. 1988;36(8):2115-28.
14. Svyetlichnyy D. Frontal cellular automata for modelling microstructure evolution: computational complexity analysis. Comput Mater Sci. 2023;230:112478.
15. Zhou R, Feng X, Zheng C, Huang Q, Li Y, Yang Y. Numerical simulation of microstructure evolution of directionally annealed pure iron by cellular automata. Metals. 2023;13(2):368.
16. Boccara N. Modeling complex systems. New York: Springer; 2010.
17. Wolfram S. Statistical mechanics of cellular automata. Rev Mod Phys. 1983;55(3):601-44. http://doi.org/10.1103/RevModPhys.55.601.
18. Burguillo JC. Cellular Automata. In: Burguillo JC, Burguillo D, editors. Self-organizing coalitions for managing complexity. London: Springer; 2018. p. 57-67. http://doi.org/10.1007/978-3-319-69898-4_4.
19. Porzycki J, Wąs J. Modeling spatial patterns in a moving crowd of people using data-driven approach: a concept of Interplay Floor Field. Saf Sci. 2023;167:106266.
20. Braga HC, Moita GF, de Almeida PEM. Comparação entre os algoritmos de busca pela vizinhança de von neumann ou de moore para geração do mapa de distâncias em um ambiente construído. Abakós. 2016;4(2):20.
21. Aleksandrov M, Zlatanova S, Heslop DJ. Voxelisation algorithms and data structures: a review. Sensors. 2021;21(24):8241.
22. Braga HC, Silva SN. Computational crystallographic investigation by non-oriented first neighborhoods for the simulation of recrystallization in metals through three-dimensional cellular automata. 2025. In press.
23. Tovar A, Patel NM, Niebur GL, Sen M, Renaud JE. Topology optimization using a hybrid cellular automaton method with local control rules. J Mech Des. 2006;128(6):1205-16.
24. Deng X, Chen H, Xu Q, Feng F, Chen X, Lv X, et al. Topology optimization design of three-dimensional multi-material and multi-body structure based on irregular cellular hybrid cellular automata method. Sci Rep. 2022;12(1):5602.
25. Lynch N, Segala R, Vaandrager F. O automata. Inf Comput. 2003;185(1):105-57.
26. Wolfram S. A new kind of science. Champaign: Wolfram Media; 2002. 871 p.
27. Rios PR, Padilha AF. Transformações de fase. São Paulo: Artliber; 2007. 216 p.