Ultrafast analysis of individual grain behavior during grain growth by parallel computing

M Kühbach, L A Barrales-Mora, C Mießen, G Gottstein
Institute of Physical Metallurgy and Metal Physics, RWTH Aachen University, Kopernikusstrae 14, D-52056 Aachen, GERMANY
E-mail: kuehbach@imm.rwth-aachen.de

Abstract. The possibility to characterize in an automated way the spatial-temporal evolution of individual grains and their properties is essential to the understanding of annealing phenomena. The development of advanced experimental techniques, computational models and tools helps the acquisition of real time and real space-resolved datasets. Whereas the reconstruction of 3D grain representatives from serial-sectioning or tomography datasets becomes more common and microstructure simulations on parallel computers become ever larger and longer lasting, few efforts have materialized in the development of tools that allow the continuous tracking of properties at the grain scale. In fact, such analyses are often left neglected in practice due to the large size of the datasets that exceed the available physical memory of a computer or the shared-memory cluster. We identified the key tasks that have to be solved in order to define suitable and lean data structures and computational methods to evaluate spatio-temporal grain property datasets by working with parallel computer architectures. This is exemplified with data from grain growth simulations.

1. Introduction
Understanding quantitatively annealing microstructures as they evolve during primary recrystallization (PRX), normal and abnormal grain growth (NGG, AGG) requires a characterization of the mechanisms and the impact of structural heterogeneities by in-situ experiments and/or numerical simulations [1, 2]. In that sense, much progress has been made in the past decade. In particular, this has been made possible by experimental techniques such as electron [3–8] and X-ray diffraction microscopy [9–13]. Additionally, progress in computational algorithms and methods for 2D and 3D microstructural evolution models in terms of representative volume elements (RVE) have allowed to study larger systems. Recent examples include sequential and parallel Monte Carlo [14–16], cellular automata [17], phase field [18], vertex models [19–22], Poisson-Voronoi networks [23], and level set approaches [24–26]. Owing to the success of parallel microstructural modeling, it is expected that even larger simulation studies will become feasible.

The main task in transforming numerical results into a quantitative physical understanding requires extensive data processing, filtering, and ultimately a reduction of the information. Successful examples of this procedure [27] are, for instance, the DREAM.3D initiative [5, 28], diffraction analysis tools [29, 30], parallelized 3D microstructure reconstruction from synchrotron experiments [31], the MTex texture toolbox [32], and defect identification from molecular
dynamics data [33, 34]. Clearly, such tools have become vital to evaluate and to analyze data from experiments and simulations.

The key problem of this process during the monitoring of grain boundary migration phenomena is the difficulty to determine a priori the grains or microstructural features, whose evolution becomes dominant for the system, for instance for the initiation of abnormal grain growth or the identification of critical stages during nucleation prior to recrystallization [35, 36]. A possible solution is the tracking of the evolution of the properties for all grains in an ensemble. This poses a due to the high density of data required but also a great potential as hinted by results in 3D grain growth [37–39]. This motivated the present contribution, in which we suggest and provide first examples of a procedure that can be utilized to track in parallel single grain evolution.

For this task parallel computing architectures offer an excellent opportunity to analyze efficiently large datasets. However, the key limitation of these systems is the fact that each data transfer is directed to the central processing unit via a memory hierarchy which is much slower than the capacity of the central processing unit. Consequently, it is necessary to reduce these transfers to a minimum in order to keep the pieces of information as long and as close as possible to the physical computing processors [40–42]. Hence, making parallel codes scalable requires two steps. Namely, to utilize optimal sequential algorithms that are cache locality-aware [41] and to partition each analysis into as many as possible spatial and temporal independent tasks as possible. The term cache locality summarizes both the spatial and temporal proximity with which the processing unit can access data in its fastest memory (L1) during the execution flow. This concept was the key idea behind the algorithm that will be introduced in the following.

2. Analysis setup

The typical outcome of a sampled in-situ experiment or simulation is a collection $M$ (dataset) that was sampled at time steps $t_i$ with $i = 1, 2, \ldots, N_m$ into sets $M_i$. Each of them comprises individual data, the most intuitive of which are the grains $g_j$ with $j = 1, 2, \ldots, N_0^i$ and their characteristics. For these grains, we can distinguish scalar properties, such as a unique nametag $I_j$, the size $V_j$, and the orientation $O_j$, the latter of which is commonly parameterized by the three Euler angles. Depending on the dimensionality of the analysis we distinguish further topological properties. Namely, the number of faces, referring to the linear segments in 2D as well as surfaces in 3D which enclose the grain $g_j$. Each face requires further a nametag that identifies the neighboring grain, and, in 2D, the length of the segment. In three dimensions it is necessary to track the area of each face to identify the bounding triple lines, their length, as well as nametags which identify the grains that meet at the triple lines, respectively. Moreover, quadruple junctions can be monitored in terms of their position and three nametags in order to specify the adjoining triple lines [23, 37].

![Figure 1. Association of data to the computing nodes.](image-url)
Although several data analyses are of interest, in this contribution we focus on utilizing the data collection for answering the following questions:

(i) How can physical mechanisms be identified from property changes over time?
(ii) Which evolutionary differences allow certain grains to grow larger than others?
(iii) How is the size evolution of particular grains correlated with their environment?

Realizing that in the case of grain growth phenomena the number of grains decreases from an initial $N_0^0$ to $N_{fin}^m$ (see figure 2), it is possible to approximate the size of $M$ comprising $N_m$ time steps. By making use of Euler’s polyhedron formula [43], an average information size $S_j^i$ that is necessary to store pieces of information e.g. from a 6-sided grain (in 2D), and a 3D grain with 14 faces on average can be taken as a reasonable average polyhedron to project the dataset size provided the underlying distribution is known. Storing into standard datatypes in 64-bit double precision and integer values would require (in 2D) to store for each grain its nametag, size, orientation, the number of adjoining faces, their neighbors, and line lengths. Hence, a total size of 112 Bytes per grain in 2D and 1106 Bytes in 3D is required.

Consequently, tracking of a dataset $M$ from NGG simulations published in [24] would require $\approx 14 \text{ GB}$ in 2D and only $\approx 5 \text{ GB}$ in 3D as the latter was reported executed in fewer time steps than the 2D simulation. Larger population studies ($N_0^0 = 133110$, $N_{fin}^m = 14150$, $N_m = 300$, in 3D) [26] would range at $\approx 23 \text{ GB}$. Contrary to common perception this reveals that spatio-temporal analyses are not computationally prohibitive on contemporary workstations. However, for even larger simulations, longer lasting or finer resolved in time simulations, such as a hypothetical setup starting from $N_0^0 = 10^6$ grains that is simulated to coarsen to 1% of the population in $N_m = 10000$ steps would require $\approx 0.5 \text{ TB}$ in 2D, and $\approx 5 \text{ TB}$ in 3D. Clearly, this substantiates the upcoming necessity to rely on parallel analyses. Moreover, keeping all pieces of information that describe a grain and its neighbors would result in rather large data blocks per grain compared to contemporary processor data cache sizes (L1: 32 KB). In essence, a restructuring of information into e.g. topology-related and other pieces of information is unavoidable. Based on these facts, we propose the following analysis strategy, as depicted in figure 1. In the post-processing the sets $M_i$ are per se mutually independent. However, the analysis requires pieces of information to be first filtered within the sets $M_i$, and then re-organized into a temporal sequence and output into summary files. This renders the classical MapReduce algorithm inadequate [44]. In the following we detail further our analysis for a distributed memory computer system comprising $N_{nodes}$ computing nodes. Each node is identified as $N_k$ with $0 \leq k < N_{nodes}$ defining the set $N_{nodes}$ and for simplicity of similar performance characteristics, i.e. same local memory $S_{crit}$ capacity and a connection to a mass storage device whose capacity $S_{HDD}$ is significantly larger than $S_{crit}$. Within this framework, it is reasonable to assume that the nodes read and write this device at a rate of $\chi_r^k$, and $\chi_w^k$, respectively. The analysis of the data, as detailed schematically in figure 1, requires the following steps:

(i) Determination of the individual set sizes $S_{M_i}$.
(ii) Grouping sets in time $t_i, t_{i+1}, \ldots, t_{i+n}, n \in \mathbb{N}$ under the constraint $\sum S_{M_i} \leq S_{crit}$.
(iii) Loading and filtering in parallel by maximizing and balancing the total data transmission rates $\sum \chi_r^k$ and $\sum \chi_w^k$, respectively.
(iv) Collecting data along the time axis by explicit communication among the computing nodes.

For the partitioning of the set $M_i$ on the computing nodes a solution to a knapsack-type optimization problem needs to be solved. Our strategy is to assign each computing node a collection of sets $M_i$ as a consecutive interval along the time axis, resulting in a set $M_{local}^k$ that is local to the $k$-th computing node. As the number of grains either reduces during NGG and AGG
or increases during PRX it is further beneficial to keep the number of grains per node balanced [40]. For the loading and filtering of gigabyte-sized datasets in parallel, efficient strategies have been developed that can make use of parallel access to storage devices via plain parallel MPI I/O or similar schemes such as HDF5 or NetCDF. We utilized MPI I/O as it allows reading and writing in parallel.

In what follows, the tasks of filtering and collecting selected grains and properties are essentially trivial-parallel until the local results need to be aggregated into their final form. Specifically, we detail how to efficiently keep track of the evolving neighborhood for selected grains. Here, the key problem is that for writing collectively the characteristics of all neighbors into a file of fixed size, their total number and nametags have to be known before the output can be written. We solve this problem by identifying first, and in parallel, all disjoint nametags. Subsequently, these are sent one after another to a single computing node which eliminates all duplicates before finally redistributing a list of unique nametags to all. This strategy can further be improved by defining a communication hierarchy that permits the systematic elimination of duplicates.

3. Simulation setup
In order to utilize the proposed workflow, a first version of the algorithm was implemented as an MPI-parallel Visual C/C++ version that reads plain text files carrying the properties of all grains and their nearest neighbors by sampling typical 2D grain growth simulation studies. The analyzed dataset was extracted by sampling each integration step along a 2D grain growth simulation, which was conducted with the vertex model [19] as detailed and parameterized in [22]. Specifically, the simulation was initiated by a modified Voronoi tessellation mapping the orientation and grain size distribution from simulation results [22]. The grain boundary energies were assumed to depend on misorientation according to the Read-Shockley model [45]. The energy of high-angle grain boundaries \( \gamma_{HAGB} \) was considered constant \( 0.9 \text{J/m}^2 \). Their mobility was defined as \( m = 10^{-13} \text{m}^4/\text{Js} \) whereas the mobility of low-angle grain boundaries was set to 10% of this value.

4. Results and Discussion

![Figure 2. Population evolution.](image)

![Figure 3. Total topological activity.](image)

Figures 2 and 3 summarize the temporal evolution of the population size, and the topological activity in the network, respectively. As expected, the reduction of the population size was
observed associated with a decreasing number of topological transformations per time step. The fact that the absolute number of individual topological changes was observed to be two orders of magnitude lower than the total number of faces, i.e. linear grain boundary segments, suggests a further potential in reducing the amount of topological data that have to be tracked during data acquisition.

Figure 4. Evolution history of surviving grains that grew to at least $3\bar{V}$.

Application of the analysis scheme to the data allowed extracting the growth-topology history for selected grains which survived the coarsening process and grew to at least three times the average final grain size. This result, as shown in figure 4, suggests that the network evolves through a two stage transient. In the first stage, in particular some grains gain selectively faces whereas in the second stage this process becomes interrupted and followed by a transient phase along which the largest grains lose faces. This suggest an aborted onset of potential AGG.

A closer inspection of particular grains, namely one that became the largest of the population (figures 5, 6) and a significantly smaller grain experiencing stronger transient characteristics (figures 7, 8) were compared. For these grains, the development of their neighbors was tracked and correlated with their own evolution. The fact that with increasing annealing time more and more neighbors became not only topologically similar but as well comparably larger, resulted in the growth competition of grains that initially had a size advantage. As a consequence, the network evolution followed a transient back to normal grain growth. These quantitative results agree well with inspecting qualitatively selected snapshots of the network, such as those which are depicted in figures 9, 10, and 11.

To sum up, the approach of following individual grain behavior was shown to be a task that is computationally intense but not prohibitive when parallel computer architectures are utilized efficiently. This motivates future applications of the method, such as screening in an automatized manner simulation data in order to characterize the chain of events that have caused microstructure change. Particularly for characterizing rare events, such as the initiation of abnormal grain growth or inspecting the birth of recrystallization nuclei, this approach is beneficial.
Figure 5. Largest grain in its competition with nearest neighbors.

Figure 6. Topology of largest grain compared to nearest neighbors.

Figure 7. Size evolution.

Figure 8. Topological evolution.

Figure 9. $t = 0s$

Figure 10. $t = 676s$

Figure 11. $t = 2754s$.

5. Conclusions
A scalable tool for the post-processing and the analysis of large datasets of microstructural evolution on a parallel computer was presented. The tool allows the tracking of local features...
of a microstructure in time and space. In particular, it was applied to a detailed investigation of a 2D grain growth vertex simulation in Ni-5at.%W. Contrasting in details the growth curves of selected grains against their immediate neighbors revealed that the progressive competition of similarly sized grains restricted the potential for triggering abnormal grain growth.

Acknowledgements
The authors gratefully acknowledge the financial support from the Deutsche Forschungsgemeinschaft (DFG) within the Reinhart Koselleck-Projekt (GO 335/44-1) as well as the FZ Jülich and the RWTH Aachen University for granting computing time within the frame of the JARAHPC project no 6687.

References
[1] Bay B, Hansen H, Hughes D A and Kuhlmann-Wilsdorf D 1992 Acta. Met. et Mat. 40 205–219
[2] Gottstein G and Shvidlerman L S 2010 Grain Boundary Migration in Metals 2nd ed (CRC Press, Taylor & Francis, Boca Raton)
[3] Hansen N, Jeffers T and Kjems J K 1981 Acta Met. 29 1523–1533
[4] Zhang C, A S, Ishimaru T and Enomoto M 2004 Metall. Mat Trans. A 35 1927–1933
[5] Groeber M A, Ghosh S, Uchic M D and Dimiduk D M 2008 Acta Met. 56 1257–1273
[6] Rowenhorst D J, Lewis D C and Spanos G 2010 Acta Met. 58 5511–5519
[7] Tucker J C, Chan L H, Rohrer G S, Groeber M A and Rollett A D 2012 Scripta Mat. 66 554–557
[8] Yu T B, Hansen N and Huang X X 2012 Phil. Mag. 92 4056–4074
[9] Lauridsen E M, Poulsen H F, Nielsen S F and Jensen D J 2003 Acta Met. 51 4423–4435
[10] Gundlach C, Panteleon W, Lauridsen E M, Margulies L, Doherty R D and Poulsen H F 2004 Scripta Mat. 50 477–481
[11] Toda H, Ohkawa Y, Kamiko T, Naganuma T, Uesugi K, Takeuchi A, Suzuki Y and Kobayashi M 2013 Acta Met. 61 5535–5548
[12] McKenna I M, Poulsen S O, Lauridsen E M, Ludwig W and Voorhees P W 2014 Acta Met. 78 125–134
[13] Pokharel R, Lind J, Li S F, Kenesei P, Lebensohn R A, Suter R M and Rollett A D 2015 Int. J. Plast. 57 217–234
[14] Holm E A, Miodownik M and Rollett A D 2003 Acta Met. 51 2701–2716
[15] Madison J D, Tikare V and Holm E A 2012 J. Nuc. Mat. 425 173–180
[16] Park C S, Na T W, Kang J K, Lee B J, Han C H and Hwang N M 2013 Phil. Mag. 93 4198–4212
[17] Kühbach M, Barrales-Mora L A and Gottstein G 2014 Modell. Sim. Mat. Sc. Eng. 22
[18] Suwa Y, Saito Y and Onodera H 2008 Mat. Trans. 49 704–709
[19] Barrales-Mora L A, Gottstein G and Shvidlerman L 2008 Acta Mat. 56 5915–5926
[20] Sprague A P, Patterson B R and Gandhi S 2010 Met. Trans. A 41 592–602
[21] Barrales-Mora L A, Gottstein G and Shvidlerman L S 2012 Acta Mat. 60 546–555
[22] Witte M, Barrales-Mora L A, Boer N D, Gilges S, Klöwer J and Gottstein G 2012 Phil. Mag. 92 4398–4407
[23] Lazar E A, Mason J K, MacPherson R D and Srolovitz D J 2013 Phys. Rev. E 88 063309
[24] Elsey M, Esedoğlu S and Smerka P 2009 J. Comp. Phys. 228 8015–8033
[25] Elsey M, Esedoğlu S and Smerka P 2011 Phil. Mag. 91 1607–1642
[26] Elsey M, Esedoğlu S and Smerka P 2011 Proc. Roy. Soc. A 467 381–401
[27] Simmons J, Bouman C, Fahroo F, Kalidindi S R, Knopp J and Voorhees P 2014 Session at the 2014 TMS
[28] Groeber M A and Jackson M A 2014 Integrating Mat. and Manu. Inno. 3
[29] Bergbäck K E, Srensen H O, Wright J P, Goret G and Kieffer J 2013 J. Appl. Cryst. 46 537–539
[30] Schmidt S 2014 J. Appl. Cryst. 47 276–284
[31] Wozniak J M, Sharma H, Armstrong T G, Wilde M, Almer J D and Foster I 2014 Big Data Staging with MPI-IO for Interactive X-ray Science Tech. rep. University of Chicago, Argonne National Laboratory
[32] Bachmann F, Hiedscher R and Schaeben H 2010 Solid. Stat. Pheno. 160 63–68
[33] Panzarino J F and Rupert T J 2014 JOM 66 417–428
[34] Stukowski A 2014 JOM 66 399–406
[35] Humphreys F J 1992 Scripta Met. et Mat. 27 1557–1562
[36] Weygand D, Bréchet Y and Lépineoux J 2000 Phil. Mag. B 80 1987–1996
[37] Patterson B R, Rule D J, DeHoff R T and Tikare V 2013 Acta Mat. 61 3986–4000
[38] Patterson B R, Rowenhorst D J, Tikare V, DeHoff R T and Kaub T M 2014 Acta Mat. 79 411–420
[39] Keller T, Cutler B, Lazar E A, Yauney G and Lewis D J 2014 Acta Mat. 66 414–423
[40] Rabenseifner R, Hager G and Jost G 2009 17th Euromicro Int. Conf. on Parallel, Distributed and Network-Based Processing, Weimar, Germany, Febr 18th - 20th, 2009 p 427–430
[41] Williams S, Waterman A and Patterson D 2009 Communications of the ACM 52 65–76
[42] Hennessy J L and Patterson D A 2012 *Computer Architectures* 5th ed (Morgan Kaufmann, Amsterdam)
[43] Euler L 1752 Elementa Doc. Soli., Novi Comm. Acad. Sc. Imp. Petro. Rep. in Opera, Vol. 26, pp. 71-92
[44] Dean J and Ghemawat S 2004 *OSDI’04: 6th Symp. on Op. System Design and Impl., San Francisco, CA*
[45] Read W T and Shockley W 1950 *Phys. Rev.* 78 275–289