

## 4. THE MONTE CARLO METHOD

### 4.1 Introduction

This chapter is aimed at describing the Monte Carlo method for the simulation of grain growth and recrystallization. It has also been extended to phase transformations and hybrid versions (Monte Carlo coupled with Cellular Automaton) of the model can also accommodate diffusion. If reading this chapter inspires you to program your own version of the algorithm and try to solve some problems, then we will have succeeded! The method is simple to implement and it is fairly straightforward to apply variable material properties such as anisotropic grain boundary energy and mobility. There are, however, some important limitations of the method that must be kept in mind. These limitations include an inherent lattice anisotropy that manifests itself in various ways. For many purposes, however, if you pay attention to what has been found to previous work, the model is robust and highly efficient from a computational perspective. In many circumstances, it is best to use the model to gain insight into a physical system and then obtain a new theoretical understanding, in preference to interpreting the results as being directly representative of a particular material. Please also keep in mind that the “Monte Carlo Method” described herein is a small subset of the broader use of Monte Carlo methods for which an excellent overview can be found in the book by Landau and Binder (2000).

### 4.2 History of the Monte Carlo Method

This section describes the history and development of the MC method. The basic features of the model are described as needed.

The Monte Carlo method as known in the materials community is an adaptation of a method used primarily to study the statistical physics of phase equilibria (Landau and Binder 2000). The name “Monte Carlo” was coined by Metropolis (inspired by Ulam’s interest in poker) during the Manhattan Project of World War II, because of the similarity of statistical simulation to games of chance, and because Monte Carlo, the capital of Monaco was a center for gambling (<http://csepl.phy.ornl.gov/mc/node1.html>). Monte Carlo now refers to any method that utilizes sequences of random numbers to perform statistical simulation. The main requirement to use Monte Carlo method for simulation of a physical system is that it must be possible to describe the system in terms of probability density function (PDF), also called partition function ( $Z$ ). Once the PDF or  $Z$  for a system is known, then the simulation begins by random “sampling” from the PDF, and subsequently determining the desired properties of the sample by conducting some kind of a “trial”. There must be a rule available, based on some reasonable mathematical and/or physical theory, to decide the outcome of such a trial. Many trials are conducted and outcomes of all of these trials are recorded. The final step in the MC method is that the behaviour of the overall system is obtained by computing the average of outcomes of the trails conducted.

MC methods are used in many different ways e.g. as a technique of integration of a function, as a way to model stochastic (random) processes, as tool to calculate properties of state such as  $E$ ,  $T$ ,  $P$ , and  $V$ , and as a model to simulate a system of interacting particles e.g. ferromagnetic materials. In materials science, however, the MC method has been primarily applied to simulate microstructural evolution where equilibrium occurs on a local basis at best (e.g. at triple junctions). In general, the system is far from equilibrium and we attempt to

study the kinetics of the processes that lead to equilibrium as a function of time e.g. grain growth or recrystallization. Although the model has proven to be useful for many different problems, it is important to understand that its ability to simulate physical behavior at the continuum (or mesoscopic) level is heuristic. One notable exception to this remark is recent work that has demonstrated rigorously that interfacial velocity is linearly related to the curvature of the interface (Holm 2002).

#### 4.2.1 Ising and Potts Models

The genesis of the method lies in solid state physics community and the development of models for ferromagnetic materials. The Ising model (1925) represents a magnetized material as a collection of spins where only two states are possible, namely up or down. Potts (Potts 1952) later generalized the Ising model and allowed for  $Q$  states for each particle in the system, hence the term “ $Q$ -state Ising model.” It is the Potts model that has been used most extensively to simulate mesoscopic (where the length scale is of the order of the grain size) behavior of materials such as recrystallization, grain growth and texture evolution. While we will describe the algorithms for solving the Ising model, as the Ising model is the simpler of the two, it should be noted that the same algorithms are equally applicable for solving the Potts model with some modifications to accommodate the  $Q$  states.

In both models, neighboring spins interact with each other through a contribution to the system energy: if the spins are the same, no interaction energy is contributed (ground state) whereas a difference in spin leads to a (positive) contribution to the system energy. The system is symmetric in the sense that the minimum energy condition is reached when either all spins point up or all point down. If there are  $N$  interacting particles and each particle has two states (“up” or “down” in Ising model) then the system as a whole has  $2^N$  possible states in which it can exist. The problem is to be able to determine the behavior of such a system at a temperature  $T$  and predict the properties of the system at equilibrium configuration. Consider a system that is in contact with a thermal reservoir with infinite heat capacity so that energy exchange between the system and the reservoir occurs at constant  $T$ . The precise nature of the reservoir is not very relevant and it can be viewed as consisting of an infinitely large number of copies of the system that we set out to study. For the reservoir as well as for the system,  $V$ ,  $N$  and  $T$  are fixed and  $E$  can vary between 0 and  $\infty$ . The assembly of systems within the reservoir is referred to as a (*macro*)*canonical ensemble* (Pathria 1972). The question now is: what is the probability  $P_i$  that the system is in state  $i$ , with energy  $E_i$  at any time  $t$ ? The Monte Carlo approach consists of generating a series of possible macrostates  $i, j, \dots$  such that the probability  $P_i$  that the system is in state  $i$  within the state space, is given by an appropriate PDF. Such a distribution is called a canonical distribution given according to:

$$P_i = \frac{1}{Z} e^{-\frac{E_i}{k_B T}}, \quad (1)$$

where  $k_B$  is Boltzmann’s constant,  $T$  is the absolute temperature and  $Z$  is the partition function given as follows:

$$Z = \sum_i e^{-\frac{E_i}{k_B T}} \quad (2)$$

The next step is to determine equilibrium properties of the canonical ensemble such as energy and magnetization. The energy associated with each state depends on the exchange energy of the particles and interaction of the particles with the external magnetic field. However, in the absence of an external field, system energy is dependent only on the spin exchange energy,  $E$ :

$$E = \frac{J}{2} \left( \sum_{i=1}^N \sum_{j=1}^z \delta_{S_i, S_j} \right) \quad (3)$$

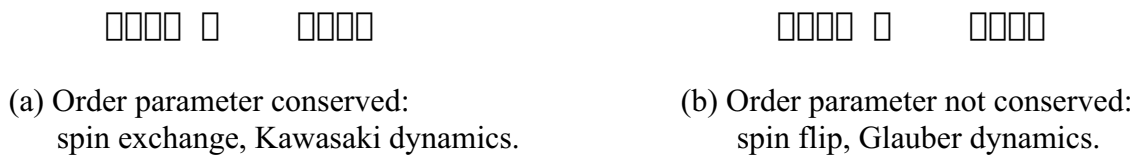
where  $J$  is the energy associated with a dissimilar pair of spins,  $\delta$  is the Kronecker delta,  $S_i$  is the spin on the  $i^{\text{th}}$  site ( $S = \{0, 1\}$ ),  $z$  is the coordination number of each grid point,  $N$  is the number of points in the grid and the factor of one-half compensates for counting each pair of spins twice.

For a given temperature  $T$  and spin exchange energy  $J$ , the system will approach an equilibrium configuration with a consequent system energy  $E$  and an order parameter  $m$  ( $m$  is the expectation value of the magnetization of the system at temperature  $T$ ) In the Ising model,  $m$  is determined from the following implicit equation:

$$m = \tanh [\beta(zJm + H)], \quad (4)$$

where  $\beta = (1/k_B T)$ ,  $z$  is the number of nearest neighbour spins and the quantity  $H$  is proportional to the external magnetic field, but has units of energy. Equation 4 must be solved numerically to obtain the actual value of  $m$ . It is clear from Equations (1) – (4) that the behavior and properties of a system in the Ising model depend critically on the values of  $J$  and  $T$ .

It is noted here that the *Ising spin model does not determine the dynamics* of evolution i.e. there is no reference to how long it takes for the system to approach the equilibrium canonical distribution. The dynamics of the Ising model were subsequently determined by Kawasaki (1972) for a conserved order parameter spin transition and by Glauber (1963) for a non-conserved order parameter approach. The two approaches vary in the way the system's structure evolution is handled as shown schematically in Figure 1.



**Figure 1: Schematic representation of (a) order parameter conserved, and (b) order parameter not conserved spin transitions.**

In Fig. 1 above a simple Ising one-dimensional model is shown schematically. In Fig. 1a, the second and third spins simply exchange their spin orientations so that the total number of *Up* and *Down* spins is not changed during the transition. On the other hand, in Fig. 1b, the second site reverses its spin to line up in the direction of spins of its neighbors on either side. In this case, the order parameter is not conserved, as there are now 3 *Up* spins and 1 *Down* spin in the system. Essentially all applications of the Monte Carlo method to microstructural

problems have used the latter, non-conserved Glauber dynamics: one crystal in a polycrystal can obviously grow at the expense of its neighbor such that the volume of a particular orientation is not conserved. Conserved Kawasaki dynamics are useful in magnetism problems where extensions of the Ising model continue to be studied.

For a system with large numbers of particles, it is extremely complex to analytically solve Equations (1) – (4) above. To deal with this problem, Metropolis *et al.* (1953) proposed an algorithm to solve these equations using Monte Carlo method. The Metropolis algorithm for the solution of the Ising model using the Monte Carlo method is described in the following section.

#### 4.2.2 Metropolis Algorithm

At this point it is useful to introduce the procedure for changing the state of a MC model. The simulation begins by initializing an array of lattice sites, with nearest neighbor interactions that depends upon the state (or spin) of the neighboring sites. The Metropolis algorithm is then enforced on such an array. The key steps of the Metropolis algorithms are as given below (Landau and Binder 2000):

1. Choose a site  $i$  at random
2. Calculate the energy change  $\Delta E$  (using Eq. 3) associated with changing the spin at the  $i^{\text{th}}$  site
3. Generate a random number  $r$  such that  $0 < r < 1$
4. If  $r < \exp(-\Delta E/k_B T)$ , flip the spin
5. Increment time regardless of whether a site changes its spin or not
6. Go to 2 until sufficient data is gathered.

There are several important points to be noted here to elucidate the procedure. First of all, the changes are made on an individual site basis where the choice of site is random in time (Step 1) in contrast to other models where simultaneous updating of all sites is the norm (finite difference, finite element, phase field, some types of cellular automata). Secondly, the  $\Delta E$  mentioned in Step 2 depends upon the states of the neighbors and the number of neighbors depends on the assumed lattice structure as discussed in more detail in Section 4.3.4. In Step 4, the transition from one state to another at any given site is calculated by a *transition probability* given as  $\exp(-\Delta E/k_B T)$ . This is in accordance to Equation 1. There are different schemes to determine the transition probability as discussed below. The transition probability is then compared to a random number that is generated in Step 3. This is a necessary rule to simulate the roll of a dice or the “chance” involved in making the decision to change the state. Finally, the *time* mentioned in Step 5 is *not* the wall-clock time of a program run, but the *simulation time* in some arbitrary units. One iteration of the Metropolis algorithm represents  $1/N$  time increments. On average,  $N$  iterations are required for each site in the lattice to have a chance to change its state and therefore, 1 unit of *system simulation time* elapses after  $N$  iterations. This unit of simulation time is called Monte Carlo Step (MCS) and represents an *integer* time increment. An alternative time accounting method is discussed in section 4.2.3 that uses a *continuous time increment*.

The particular expression given here is the Metropolis method that is the one most commonly used in materials simulations; see (Landau and Binder 2000).

$$p(\Delta E) = \begin{cases} 1 & \text{if } \Delta E \leq 0 \\ \exp(-\Delta E / k_B T) & \text{if } \Delta E > 0 \end{cases} \quad (5)$$

There is also the symmetric method that uses a hyperbolic tangent function. In simulations of domain coarsening, this method yields similar results to the Metropolis method.

$$p(\Delta E) = \tanh(\Delta E / kT) \quad (6)$$

To gain an intuitive understanding of the transition kinetics, the influence of  $\Delta E$  and  $T$  on  $p(\Delta E)$  is shown in Table 1.

Table 1: Influence of  $\Delta E$  and  $T$  on kinetics and equilibrium state of a system.

$T = 0$ <i>Zero Temperature</i>	$\Delta E \leq 0$ $\Delta E > 0$	$P(\Delta E) = 0$ $P(\Delta E) = 1$	System is in equilibrium in ground state, all spins point in the direction of $H$
$T = \infty$ <i>Infinite Temperature</i>	Any $\Delta E$	$P(\Delta E) = 1$	System is in completely disordered state as entropy dominates and spins are randomly oriented w.r.t. $H$
$T < T_c$ $T_c$ is a critical temperature below which system orders spontaneously, $T_c = 2.269185 J/k_B$ for 2D, square, Ising model <i>Finite Temperature</i>	$\Delta E \leq 0$ $\Delta E > 0$	$P(\Delta E) = 1$ $P(\Delta E) < 1$ *	$T$ is expressed in terms of the ratio $J/k_B$ . Variation in $T$ leads to first order (discontinuous) or second order (continuous) transitions
$T > T_c$ <i>Finite Temperature</i>	$\Delta E \leq 0$ $\Delta E > 0$	$P(\Delta E) = 1$ , $P(\Delta E) < 1$ , but greater than * above	- as above -

The major inefficiency associated with the Metropolis algorithm is that during the late stages of evolution (sparse systems), or for low temperatures, the transition probability approaches 0 at most sites so that the system evolves very slowly and many reorientation attempts are wasted. This characteristic increases computation times. This weakness is substantially mitigated at low temperatures by the  $n$ -fold way algorithm described in the following section.

#### 4.2.3 $n$ -fold Way Algorithm

The Monte Carlo method is very inefficient when applied in its basic form to large data sets because of the sparseness of the problem to be worked. Once much coarsening has occurred

in grain growth, or recrystallization is nearly complete, most sites in the lattice are surrounded by sites of the same orientation, i.e. they are in the bulk of a grain. Therefore the probability of their changing orientation is very low and expending computational effort there is wasted. The crucial contribution of Bortz *et al.* (1975) was to propose a method for eliminating the need to compute unsuccessful changes in orientation. The  $n$ -fold way algorithm speeds up simulations by eliminating Steps 3 and 4 in the Metropolis algorithm given above. The essential concept is that for a given state of the system, the *spin-flip transition probability* for each of the lattice sites can be calculated before choosing a site to flip. The spin-flip transition probabilities in a system will have certain specific values depending on the configuration of surroundings and choice of lattice type. The energy associated with site  $i$  is given according to the following modification of Eq. 3:

$$E_i = \frac{J}{2} \left( \sum_{j=1}^z s_i s_j \right) - H s_i \quad (7)$$

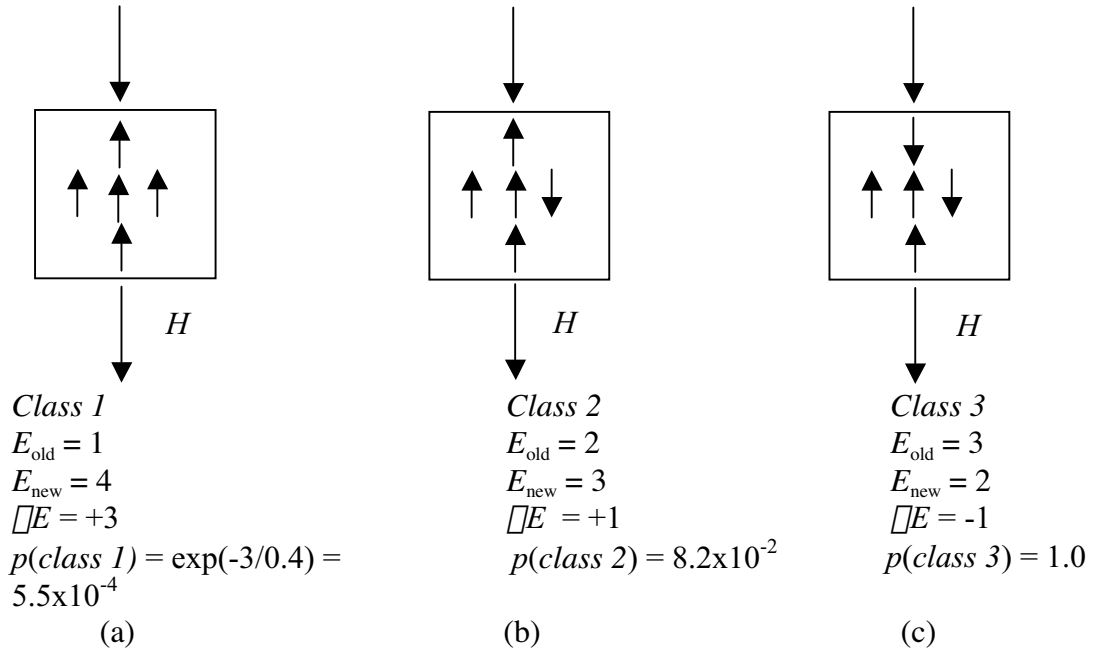


Figure 2: Three examples of different classes in the  $n$ -fold way algorithm for a two-dimensional, square lattice Ising model. Each *class* represents a different possibility for a change in configuration of spins (or orientations when grain growth is considered). The spins are shown in their initial configuration and the transition probability is evaluated for changing (flipping) the central spin (from up to down).

Consider a square lattice Ising model with four nearest neighbors ( $z = 4$ ) and  $J = |H| = 1$ , and  $k_B T = 0.4$ . Based on the orientation of the reference spin and those of its neighbors and the direction of the external field there are 10 distinct values of transition probabilities. The transition probabilities are calculated using the Metropolis scheme, Eqs. 5 & 7. The different possible values are termed *classes* in the  $n$ -fold way terminology. Three such classes and their corresponding transition probabilities are shown in Figure 2. Three examples of different classes and their initial orientation of the external field and spins are shown in Figure 2. For Class 1, the external field is in the opposite direction to the spins while the four neighboring spins are in the same direction as the central reference spin leading to  $E_{\text{old}} = 1$ . If the central reference spin were to flip its spin direction, then it would align itself in the direction of the external field, but would be anti-parallel to all four of its neighbors so the

new  $E_{\text{new}}$  would be 4. The net change for this transition in energy ( $\Delta E$ ) would be  $4 - 1 = +3$ . The probability of this transition is then determined according to Eqs. 5 and 7 as shown in Fig. 2, together with two additional examples. A similar analysis is required for all ten classes of spin configurations.

The next step in the  $n$ -fold way algorithm involves sorting of the lattice sites into these spin-flip transition probability classes. The sum of spin-flip probabilities for the entire system ( $Q_n$ ) is calculated according to

$$Q_n = \sum_{j=1}^n n_j p_j \quad (8)$$

where  $n = 1, 2, 3, \dots, 10$  are the number of classes of spin-flip transition probabilities,  $n_j$  is the number of sites in  $j^{\text{th}}$  class and  $p_j$  is the spin-flip probability for that class. For a 2D isotropic system with  $z = 4$ , there are 10 classes. If anisotropy is introduced between the X- and Y-axes, then there are 18 classes (Novotny 1995). To decide which spin to flip in the  $n$ -fold way algorithm, a random number  $r(0,1)$  is generated and the  $k^{\text{th}}$  class is chosen based on satisfying the following condition:

$$Q_{k-1} + rQ_n < Q_k \quad (9)$$

This procedure thus chooses classes according to their weight – classes with high  $p$  are chosen more often, classes with low  $p$  are chosen less often, while those classes with probability=0 will not be chosen at all. Thus sites internal to a domain or a grain that will not flip (transition probability=0), are simply never chosen to flip in the  $n$ -fold way, thus avoiding the rejections involved (Metropolis Step 4) in the conventional MC method. Within the chosen class  $k$ , the particular site to be flipped is selected randomly from the  $n_k$  sites that belong to that class. Finally, the spin at the chosen site is flipped with probability 1 and this eliminates Steps 3 and 4 in the Metropolis algorithm. The class of the chosen spin and those of its nearest neighbors is then updated. The  $n$ -fold way algorithm may be described in the following manner (Novotny 1995):

1. Generate a random number and increment the time by an appropriate amount.
2. Choose a class  $k$  that satisfies the condition given in Eq. 9
3. Generate a random number to choose one of the sites from class  $k$
4. Flip the spin at the chosen site with probability 1
5. Update the class of the chosen spin and all of its nearest neighbors
6. Determine  $Q_n$
7. Go to 1 until sufficient data is gathered

The time increment in the  $n$ -fold way algorithm is explained in detail by Novotny based on the concept of absorbing Markov chains. In effect, the time increment,  $\Delta t$ , in the  $n$ -fold way algorithm is correlated to the probability that the given system configuration *will change* to a different configuration during the time increment:

$$\Delta t = \frac{-\ln r}{Q_n} \quad (10)$$

Eq. 10 is based on the assumption that the successful re-orientation of a site is described by an exponential probability distribution, so that successive evolution steps are Poisson events.

It is useful to see exactly how the  $n$ -fold way speeds up the computation time. Consider *class 1* as shown in Fig. 2. The transition probability for this class is calculated as  $6.7 \times 10^{-3}$ . Hence, on average, about  $1/p(1) = 148$  spin flip trials are required for a spin in *class 1* to flip successfully using the Metropolis algorithm. For sparse systems, a large proportion of sites have a low transition probability and therefore Metropolis trials require long times in these cases for successful flips whereas the  $n$ -fold way flips those sites in one attempt. One drawback of the  $n$ -fold way algorithm is that it requires more memory than the standard MC algorithm, because it needs to store the class tables and the list of sites that belong to each class. Also, some computation time is spent in updating the class tables after every iteration. A practical consideration is that, at high lattice temperatures, the advantage of the  $n$ -fold way disappears because all sites approach similar activity levels. However, these restrictions are now almost not relevant since the memory size of desktop computers is commonly 2 GB and parallel computing is making inroads into computational materials science. For most grain growth and recrystallization problems, the overall performance of the  $n$ -fold way algorithm is much better compared to the conventional MC algorithm because the density of boundary sites is low and typically decreases during a simulation.

This approach was adapted by Sahni *et al.* (1983) to the  $Q$ -state Potts model to take account of variable *activity* in the system where  $Q$  is the number of possible spin numbers that an individual grid point can take. The system activity,  $A$ , is defined as the sum of the separate probabilities for each site for each possible re-orientation over all distinct spin numbers in the system.

$$A = \sum_{i=1}^N \sum_{j=1}^{Q-1} p_j(S_i \rightarrow S_{ij}) \quad (11)$$

Here the subscript  $j$  indicates the counter for each possible new spin value, of which there are  $(Q-1)$  possibilities at each site (since changing to the same spin is excluded). Based on this definition, the time increment associated with each successful re-orientation is as follows:

$$\Delta t = \frac{\ln(Q-1)}{A} \ln r. \quad (12)$$

This particular approach was, however, limited to zero temperature and uniform grain boundary energy (i.e. a single value of  $J$  in Eq. 3). An important generalization of the  $n$ -fold way was made by Hassold and Holm (1993) who pointed out that, although finite temperature introduces a finite probability that a given site will switch to any of the  $(Q-1)$  other possible orientations, most of the switching probabilities are known analytically. Thus the additional work of computing the activities at each site due to finite temperature is minimized and a practicable algorithm was generated. For 2D grain growth, for example, Holm found that the conventional Monte Carlo method was more efficient only when the mean grain radius was less than 3 and computation efficiency increased monotonically as the simulation progressed (Holm, Glazier *et al.* 1991). Even at high fractions of the critical temperature, the  $n$ -fold way was more efficient for long simulation times.

It has been noted that the time scale depends on the number of distinct orientations or spins in the system (Eq. 12). In reality, however, the rate of grain growth does not depend on the



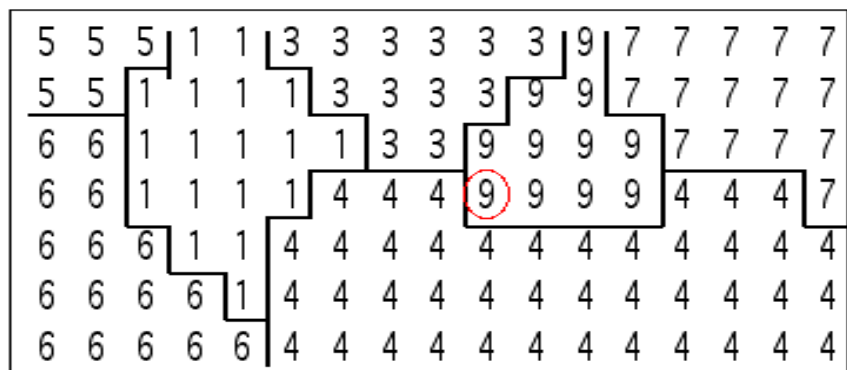
number of different orientations present initially. Therefore it has been suggested that the time increment should be adjusted so as to remove the dependence on  $Q$ :

$$\Delta t \propto \frac{\Delta \ln r}{A}. \quad (13)$$

### 4.3 Description of the Monte Carlo Method for Grain Growth & Recrystallization

#### 4.3.1 Discretization of Microstructure

The Monte Carlo also uses a discretized representation of microstructure; however, site interactions are energetically controlled. A continuum microstructure is mapped onto a two-dimensional (2D) or three-dimensional (3D) lattice. Each lattice site is assigned a number,  $S_i$ , which corresponds to the orientation of the grain in which it is embedded. Lattice sites that are adjacent to sites having different grain orientations are regarded as being separated by a grain boundary, whereas a site surrounded by sites with the same orientation is in the grain interior.



*Figure 3.* Diagram of the square 2D lattice, showing orientation numbers at each site and grain boundaries drawn between sites with unlike orientations. The circled site with spin value 9 has three like neighbors when 1<sup>st</sup> and 2<sup>nd</sup> nearest neighbors are counted, and five unlike neighbors. For isotropic grain boundary energy, flipping this site from  $spin=9$  to  $spin=4$  would leave the number of unlike nearest neighbors unchanged; thus  $\Delta E=0$  and the flip probability is one.

Each site contributes bulk energy,  $H(S_i)$ , to the system; in recrystallization modeling,  $H(S_i)$  is the energy stored at site  $i$  during deformation, analogous to dislocation densities in real crystals. In addition, each unlike pair of nearest neighbors contributes a unit of grain boundary free energy  $J$  to the system as already described. Summing bulk and surface energy contributions, the total energy of the system is calculated via the Hamiltonian specified in Eq. 7. In static recrystallization simulations, the stored energy per site is assumed to be positive for initially unrecrystallized material and zero for recrystallized material. When dynamic recrystallization is simulated, however, the stored energy at each point is a function of both time and position.

#### 4.3.2 Evolution of the Microstructure

The evolution of the structure is modeled by picking a site and a new orientation at random from the set of allowable values. The change in total system energy  $\Delta E$  for reorienting the

site is computed, and the reorientation is implemented with the transition probability,  $p$  as discussed in Section 4.2.2. It is important to note the difference between the meaning of temperature in the context of the Monte Carlo model and the physical parameter relevant to recrystallization. In the simulation, temperature governs the degree of disorder in the lattice, and below some critical temperature  $T_c$ , dependent on the lattice type, the system orders spontaneously. Only second order effects are observed for variations in simulation temperature (Hassold, Holm et al. 1990) on the kinetics of grain growth. Because of this lack of sensitivity to lattice temperature, much simulation work with model has been performed at zero lattice temperature. The consequence of this is to simplify the transition probabilities as follows.

$$p(\Delta E) = \begin{cases} 1 & \text{if } \Delta E \leq 0 \\ 0 & \text{if } \Delta E > 0 \end{cases} \quad (15)$$

The usual procedure in Monte Carlo simulations is to start with a completely random structure that corresponds to a structure obtained at infinite temperature. The structure may then be evolved based on the manner in which it is cooled. One approach is to *quench* the structure to a temperature below  $T_c$  and observe the evolution as the system approaches the equilibrium ordered state. This is the so-called *Quenched Potts* model. When the quenching temperature is low, i.e.  $T$  approaches 0, then the evolution rate is low and it would take long time for the system to approach equilibrium. Therefore, an alternative approach termed *simulated annealing* is used to evolve the structure. In this case, the temperature of the system is lowered slowly and the structure is allowed to attain equilibrium at each temperature step. The time steps needed to ensure slow cooling are proportional to  $\exp(1/T)$  such that, as the temperature decreases, the number of time steps needed increases exponentially. In such cases, a *fast simulated annealing* procedure may be used, see Plischke and Bergensen (1994) for more details.

One Monte Carlo time step (1 *MCS*) is typically defined as  $N$  reorientation attempts, i.e. each site is given an opportunity to change orientation. The number of Monte Carlo Steps is assumed to be proportional to physical time. Recently, it has been pointed out that this definition results in a dependence on the  $Q$  value, i.e. the number of orientations in the system. The higher the  $Q$  value, the slower the rate at which boundaries will move. This is clearer in the definition of the time step for  $n$ -fold way in section 4.2.3 (Eq. 12).

### 4.3.3 Inert Particles

Particles are introduced into the simulation as sites which have an orientation different from any of the grains and which cannot be reoriented during the course of the simulation. An individual particle may consist of a cluster of contiguous sites. In 2D, single site particles are effective but in 3D it is important to use a particle size that is comparable (or bigger than) the interaction distance implicit in Eq. 3. The particles do not react, dissolve or grow themselves and hence are called *inert* rather than *second phase particles* or *precipitates*. This assumption results in an equality of the particle-matrix interfacial energy and the grain boundary energy, which is reasonable for particles that are incoherent with respect to the matrix. Just as the grain boundary energy can be made a function of the boundary type, so the particle-matrix energy can be varied. Also, the particles cannot move through the lattice, which means that grain boundary drag of particles is not permitted (Ashby and Centamore 1968). Results for particle pinning effects are discussed in sections 4.9.3, 4.9.4.

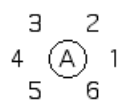
## 4.3.4 Lattices

Implicit so far in the discussion has been the connectivity of the points that represent the discretized microstructure. It turns out that the lattice can have a strong effect on the results of the simulation. A survey of lattice types for both two and three dimensions is available in the thesis by Holm (1992). The grain boundary energy per unit length is anisotropic with respect to the boundary orientation in the lattice. This anisotropy can be characterized by a Wulff shape, which is directly related to the coordination number and symmetry of the lattice. Tables 2 and 3 list the lattice types with their geometries and the anisotropy of the Wulff plot. The number in parentheses after the lattice type denotes the number of shells of neighbors, such that square (1,2) means a square lattice with first and second nearest neighbors. The lattice type cubic (2\*) denotes a simple cubic lattice with first, second, third nearest neighbors and neighboring points located at [222].

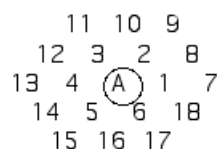
*Table 2. Listing of 2D lattice types with geometries and anisotropies.*

Lattice Type	Wulff Shape	Coordination Number	Anisotropy	Grain Growth
Square (1)	Square	4	1.414	Inhibited
Triangular	Hexagon	6	1.154	Normal
Square (1,2)	Octagon	8	1.116	Normal
Triangular (1,2)	Dodecagon	18	1.057	Normal

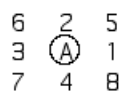
(a) tri(1)



(b) tri(1,2)



(c) sq(1,2)



*Figure 4:* Diagram of the nearest neighbor relationships around a central point labeled 'A' in: (a) triangular lattice with first nearest neighbors; (b) triangular lattice with 1<sup>st</sup> and 2<sup>nd</sup> nearest neighbors; (c) square lattice with 1<sup>st</sup> and 2<sup>nd</sup> neighbors.

*Table 3. Listing of 3D lattice types with geometries.*

Lattice Type	Wulff Shape	Coordination Number	Grain Growth
Cubic (1)	Cube	6	Inhibited
Cubic (1,2)	18-hedron	18	Inhibited
Cubic (1,2,3)	26-hedron	26	Normal
Cubic (2*)	98-hedron	124	Normal

fcc (1)	Rhomboid Dodecahedron	12	Inhibited
fcc (1,2)	18-hedron	18	Inhibited
hcp (1)	Trapezoidal Dodecahedron	12	Inhibited

The characteristic of many lattices that has been ignored by several authors is the tendency towards self-pinning for grain growth (explained later) in simulations performed at zero temperature. The last column in each table shows which lattices can sustain grain growth without self-pinning and are therefore suitable for studies of microstructural evolution. For example, in three dimensions, both of the close packed lattices, face centered cubic and hexagonal, cannot sustain coarsening to long times. The reason for this is a combination of high lattice anisotropy tending to favor grain facets that lie on high symmetry planes, and a tendency for kinks or steps in boundaries to anneal out with time. If the microstructural features that allow boundaries to move are lost, then it is not surprising that self-pinning occurs. Finite temperature can be used, however, to maintain a population of kinks and steps thereby allowing grain growth to proceed. In general it is advisable to use a finite temperature in order to avoid faceting and loss of the kinks that allow interface motion to occur, regardless of the particular lattice used.

If the grain boundary energy  $J$  is uniform for all boundary segments in a recrystallization simulation, the triangular lattice is suitable for the simulation of recrystallization in two dimensions (Srolovitz, Grest *et al.* 1986), and that the cubic (1,2,3) lattice is suitable in three dimensions (Anderson, Grest *et al.* 1989). For other types of simulations, careful examination for lattice effects must be made even for these lattices. Figure 3 shows a 2D triangular lattice with boundaries drawn between regions of uniform orientation. Despite the availability of 3D simulation methods, no substantial results have been reported for 3D Monte Carlo simulation of recrystallization.

#### 4.3.5 Boundary Conditions

It is common to use periodic boundary conditions on the spatial domain. This has been likened to modeling a circle or Möbius strip of points in 1D, or a set of points on a torus in 2D. The advantage of using periodic boundary conditions is that it avoids the singularity of edges in a finite domain. The effect of such boundary conditions is illustrated in Figure 5. A point on the edge of the domain is connected to points on the opposite edge. The equivalent effect can be obtained by copying the domain along all the edges. Care is needed if a triangular lattice with an oblique two index-addressing scheme is used in order to avoid an implicit shear from connecting the top and bottom edges of the lattice.

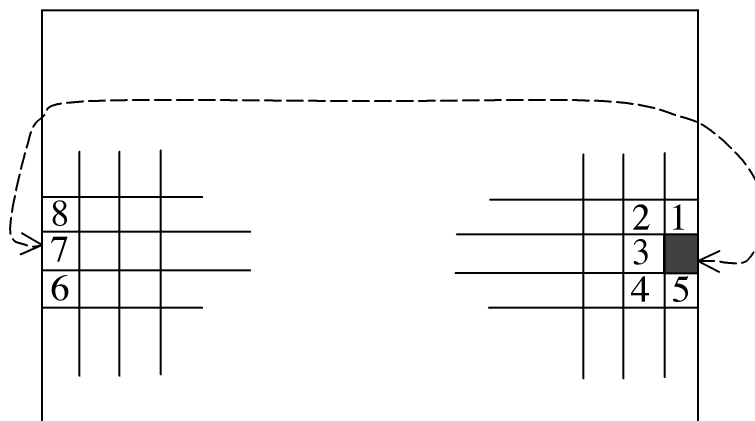


Figure 5: Periodic boundary conditions illustrated for a 2D lattice with 1<sup>st</sup> and 2<sup>nd</sup> nearest neighbors. The site in solid shading has 5 of its neighbors with similar index values and 3 of its neighbors on the left hand side of the lattice. The arrow indicates the connection from one side of the lattice to the other. The concept is easily extended into 3D.

Periodic boundary conditions are simple to implement and a natural choice when the initial state of the lattice is artificial. If, however, a microstructure has been measured experimentally, e.g. by orientation imaging microscopy, and is used to initialize the lattice then periodic boundary conditions are obviously inappropriate (Baudin, Paillard *et al.* 1999; Cheong, Hilinski *et al.* 2003). The typical choice is to make each edge a mirror, which has the effect of allowing boundaries to terminate at an edge but be able to slide along the edge as dictated by the driving forces that act on them.

#### 4.3.6 Parallelization of the Monte Carlo Algorithm

Since parallelization of the algorithm is largely concerned with boundary conditions, it is appropriate to review the efforts that have been made in connection with the Monte Carlo model. Two main concepts are discussed here: one is based on dividing the simulation domain into subdomains, and the second is based on the checkerboard approach (Holm, 1992). Lubachevsky (1987, 1988) proposed parallelization of the Ising model based on an asynchronous scheme. Korniss *et al.* (1999) developed a parallel MC model for studying magnetic domains using Lubachevsky approach, described in more detail below. In complete contrast to the subdomain approach, Miodownik (2000) used the checkerboard approach in which a subset of the pixels is operated on, chosen such that in each time-step, each active pixel (or voxel) has no nearest neighbors that are operated on simultaneously. Given that each active pixel is independent of all others, the work of deciding what flip to perform can be distributed across processors without any need for communication between them: only the results of each individual flip must be recorded in memory. At the next time step, the origin of the checkerboard is translated by one pixel and the next subset of (spatially independent) pixels is operated on. Implementation of the n-fold way method was not possible with this approach. Also, the particular simulations involved required high lattice temperatures in order to avoid problems associated with faceting of grain boundaries in configurations where they are pinned by second-phase particles. At high enough lattice temperatures, as noted elsewhere, the high rate of flips associated with positive changes in energy is high enough to nullify the advantages of the n-fold way and the standard MC method is more efficient.

The main ideas involved in the asynchronous parallelization of the  $n$ -fold way algorithm for Ising-like systems are as follows. The first step in parallelization involves dividing the simulation domain into suitably sized pieces for each processor to work with. One such example of domain decomposition is shown in Figure 6 below.

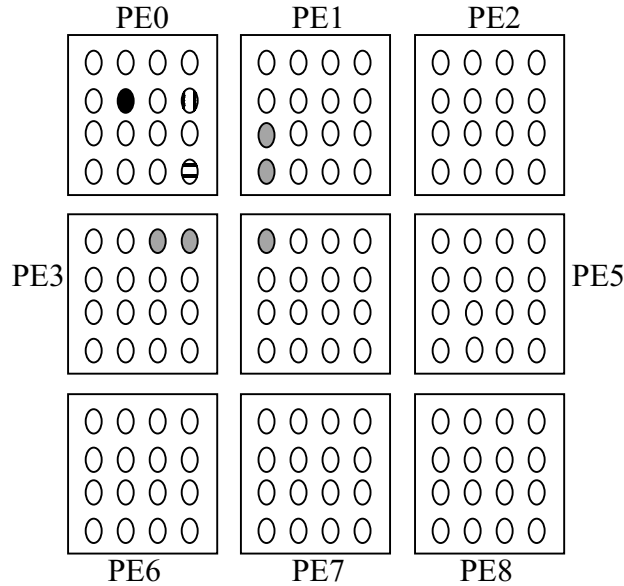


Figure 6: Domain decomposition in parallel processing. In the example shown in the figure, a *global* simulation domain size of 12 x 12 sites is divided up into 9 processors, each carrying a *local* simulation domain size of 4 x 4 sites. Note that the sites situated at a processor corner (e.g. horizontal lines) have some of their neighbors located on three other processors (gray), sites at a processor boundary (vertical lines) have some of their neighbors residing in one other processor while sites within the kernel of the local domain (solid) have all of their neighbors within the same processor. The numbers shown are the processing element (PE) or processor ID numbers.

Each processor works on its piece of simulation domain with its local simulation time. When a processor chooses a site that is situated on a processor boundary or a corner, special care must be taken to ensure that the simulation trajectory is not corrupted. This is achieved by allowing a border or corner site update only if its local simulation time is less than or equal to the local times of all the corresponding processors that carry its neighboring sites. For example, in Figure 6, if the processor 0 has chosen the diamond-patterned site to update, then this update will be evaluated only when the local simulation time of processor 0 becomes less than or equal to the local simulation time of processors 1, 3 and 4 that carry the neighboring sites marked with circles. Until this happen, processor 0 must wait. Lubachevsky and Korniss *et al.* defined a special *class* for boundary and corner sites to handle PE boundaries. The activity of all of the PE boundary and corner sites was always set to 1. Assigning a high weight to PE boundary sites leads to PE boundary sites being picked more often and this compensate for the *waiting* at the PE boundaries. This procedure ensures comparable evolution kinetics of the kernel and boundary sites. The time increment in the parallel  $n$ -fold way is as follows:

$$\Delta t = \frac{1}{N_b \sum_{j=1}^n n_j p_j} \ln r. \quad (16)$$

Where  $N_b$  is the total number of sites on PE boundaries.

The parallel  $n$ -fold way algorithm based on the asynchronous approach is then stated as follows:

1. Select a class based on Eq. 9. Note that in parallel, asynchronous  $n$ -fold way, PE boundary sites have a class of their own as explained above.
2. (a) If the chosen site is in kernel, flip it with probability 1 and go to Step 3  
 (b) If the chosen site is in the boundary class, then wait until the local simulation time of this updates becomes less than or equal to the local simulation times of the neighboring PEs. When this condition is satisfied, evaluate the environment, compute the transition probability and flip with *Metropolis probability*, Eq. 5. Go to Step 3.
3. Update tabulation of spin classes in kernel
4. Determine time of next update based on Eq. 16
5. Go to 1 until sufficient data is gathered.

#### 4.4 Nucleation in Recrystallization

Nucleation presents several challenges at the level of mesoscopic simulation of microstructural evolution. It has long been known (Martin *et al.* 1997) that homogeneous nucleation is impossible and that for all practical purposes new grains arise from the existing deformation microstructure. The subgrain structure and other relevant features of the deformation microstructure have, however, a characteristic length scale on the order of 1  $\mu\text{m}$  or less (Miodownik *et al.*, 1999). This is more than an order of magnitude smaller than the typical grain size after recrystallization (10 to 100  $\mu\text{m}$ ) and so it is impracticable to model the nucleation process in detail. New grains must be introduced *deus ex machina* according to empirical rules for their location and orientation, as described below. There are two exceptions to these remarks. One is work by Radhakrishnan *et al.* who have studied combined models of plastic deformation and recrystallization (Radhakrishnan *et al.* 1998; Sarma *et al.* 2002). The second is work by Holm *et al.* (2003) on modeling the process subgrain coarsening in detail which has led to a theory of nucleation of new grains, discussed in more detail below in section 4.9.1 on abnormal grain growth.

From an algorithmic point of view, the fact that recrystallization introduces new grains means that it is not practicable to work with a fixed list of orientation values as can be done in grain growth, wherein the maximum numbers of grains occurs at time zero. Therefore it is convenient to introduce each new grain with a unique orientation value. Alternatively, the range of orientation values can be partitioned into recrystallized and non-recrystallized sets. Nucleation is accomplished by introducing new (recrystallized) grains into the simulation with orientation numbers chosen from the appropriate set. The computational efficiency of the continuous time method is unaffected by this characteristic of recrystallization simulations.

Nucleation of recrystallized grains is modeled by adding small embryos to the material at random positions at the beginning of the simulation (i.e. site saturated nucleation) (Srolovitz, Grest *et al.* 1986). The stored energy is set to zero at each site belonging to the embryo. Adding embryos at regular intervals during the simulation simulates continuous nucleation. In both cases the effective nucleation rate decreases with time because, at long times, most of the available space has been recrystallized and has zero stored energy; embryos placed in recrystallized material will shrink and vanish. Dynamic recrystallization is modeled by

adding stored energy to each lattice point continuously (Rollett, Luton *et al.* 1992). This means that it is not possible to distinguish between recrystallized and unrecrystallized material after the first cycle of recrystallization is complete. The process of work hardening starts anew with each new grain, which means that the stored energy at any given point is related to the length of time that has elapsed since the nucleation event associated with that grain.

Embryos have orientations that differ from those of all other grains and particles as discussed above. If the bulk stored energy  $H$  is too small, the embryos are sub-critical and shrink away. The value of  $H$  required for embryo growth depends on its surroundings and on the lattice being used. Above some critical  $H/J$ , an isolated embryo is super-critical and can grow as a new grain nucleus. Below that  $H/J$ , the embryo must be adjacent to an existing grain boundary in order to become a nucleus; its growth then occurs preferentially along the prior grain boundaries. In the 2D triangular lattice, homogeneous nucleation cannot occur for  $H/J < 2$ . Embryos of two lattice sites can grow when  $2 \leq H/J < 4$ . Single-site embryos can grow when  $H/J \geq 4$ .

*Table 4. Critical Embryo Sizes in the MC model for Recrystallization (Holm 1996)*

Stored Energy: Boundary Energy	Critical Size (lattice sites)
2D, Triangular Lattice:	
$H/J < 2$	(Very large)
$2 < H/J < 4$	2
$4 < H/J < 6$	1
$H/J > 6$	(Any embryo grows)
2D, Square Lattice, with 2 <sup>nd</sup> nearest neighbors:	
$H/J < 1$	(Very large)
$1 < H/J < 2$	3
$2 < H/J < 8$	1
$H/J > 8$	(Any embryo grows)
3D, Simple Cubic lattice, with 2 <sup>nd</sup> and 3 <sup>rd</sup> nearest neighbors	
$H/J < 3$	(Very large)
$3 < H/J < 5$	5
$5 < H/J < 8$	3
$8 < H/J < 26$	1
$H/J > 26$	(Any embryo grows)

#### 4.5 Initialization of MC Simulations

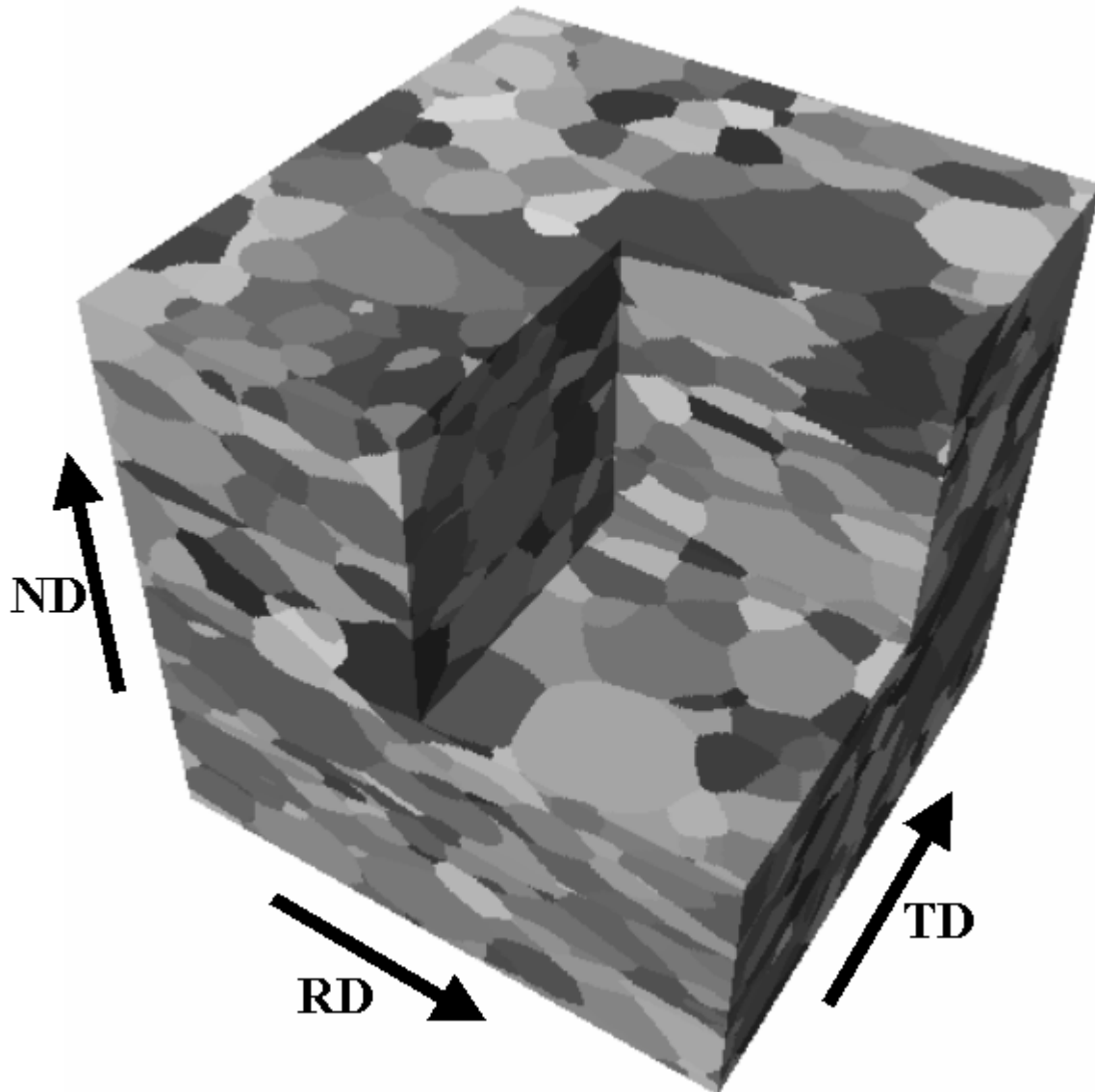
The 2D MC recrystallization simulations reviewed here were typically initialized with a microstructure obtained from grain growth simulations (Anderson *et al.* 1984; Srolovitz *et al.* 1984). Grain growth for a period of  $10^3$  MCS in a domain size of  $200 \times 200$  sites yields a microstructure with approximately 1000 grains and a mean grain area of approximately 40 sites. Whether new grains placed in the structure survive and grow will depend on their spatial location because prior boundaries act as heterogeneous nucleation sites (Srolovitz *et al.* 1988). If second phase (inert) particles are required, single-site "particles" are randomly placed within the microstructure to obtain a certain area fraction. The typical method is to



assign particle sites an orientation number that is not permitted to change during a simulation. The stored energy is initialized to the desired value for each site. It is also reasonable to start with no prior grain structure, which is equivalent to having a single crystal. The survival and growth of new grains is then independent of spatial location in the lattice.

Another interesting aspect of initialization is how to represent experimentally measured microstructures. Several authors have recently reported using experimental orientation maps to specify the initial state of the model (Baudin *et al.* 1999; Cheong, *et al.* 2003). These maps are currently obtained from automated electron back-scatter diffraction (EBSD) characterization of metallographic specimens in a scanning electron microscope (SEM). The transfer is exceptionally simple because the size of the maps is well within the capacity of modern 2D MC codes. It is also noteworthy that Demirel *et al.* (2003) made a direct comparison of experimentally characterized grain growth with simulations albeit using a finite element code rather than an MC model. They were able to demonstrate that it was essential to include the anisotropy of the grain boundary properties in order to obtain a good agreement between simulation and experiment.

In three dimensions, the challenges of devising accurate representations of microstructure are considerable. Miodownik *et al.* (1999) have used a simulated annealing procedure (see Section 4.2.3) to match a misorientation distribution to a measured distribution. More recently Saylor *et al.* have extended this concept to the specification of 3D initial structures that reproduce measured grain shape, orientation and misorientation distributions, again based on EBSD measurements (Saylor *et al.* 2003). Figure 7 shows an example of a 3D microstructure generated by matching EBSD orientation maps on orthogonal sections of a recrystallized aluminum 1050 alloy. This statistical approach opens up the possibility of being able to perform simulations of microstructural evolution that can be compared directly with experimental measurements.



*Figure 7: 3D model microstructure generated by fitting ellipsoids to measured grain shapes in a sample of aluminum 1050 alloy and then associating orientations with each grain such that both the average texture and the average grain boundary character corresponds to the experimental measurements (Saylor *et al.*, 2003).*

#### 4.6 Verification of the Monte Carlo Model

An important aspect of any model is to verify that it behaves as expected. This is not a trivial issue for this model because most of the basic features governing grain growth are not imposed on the model. For example, there is nothing in the formulation given above to guarantee that the migration rate of boundaries is proportional to the driving force, or that local equilibrium is maintained at triple junctions between the surface tensions.

The most basic verification that the model works as desired is to simulate a single isolated grain shrinking under the action of the curvature of its boundary. Mullins' (1956) analysis shows that, for velocity proportional to mean curvature, the rate of change of area,  $dA/dt$ , of any grain is constant and equal to the integral of the turning angle around the perimeter of the grain. In the case of a grain with  $n$  vertices, this leads to the expression:

$$\frac{dA}{dt} = -2\gamma M \langle n \rangle \quad (17)$$

where  $M$  is the mobility, and  $\gamma$  is the grain boundary energy. For an isolated grain with one side, the rate of area change is constant. The reason for 6-sided grains being neutral is that, for isotropic grain boundaries, the turning angle at a three-fold vertex (triple junction) must be  $60^\circ$ .

To test this relationship for the case of  $n=1$  is simple because the multi-state model collapses to the Ising model with one grain isolated within another grain. In either 2D or 3D, the result is the same, i.e. that  $dA/dt$  is constant (and negative). Recently, Holm has verified (Holm 2002) that the rate of collapse of the isolated grain is exactly equivalent to motion by curvature for the case of a 2D square lattice with first nearest neighbors. This is also important because it verifies that the model satisfies the relationship

$$V = M P, \quad (18)$$

where  $V$  is the boundary velocity,  $M$  is the mobility,  $P$  is the driving pressure, taken here to be the product of boundary energy and curvature,  $P = \gamma \langle \kappa \rangle$ .

A second test of the model is to examine grain growth kinetics in a polycrystal. Classical theory states that the average grain area,  $\langle A \rangle$ , should vary linearly with time:

$$\langle A \rangle - \langle A \rangle_{t=0} = kt. \quad (19)$$

This is justified on the basis of the observed self-similarity of the microstructures, which simply means that snapshots taken at different times can only be distinguished if either a time or a length scale is provided. More specifically, self-similarity means that the average curvature in the system scales with the inverse grain size. This permits a simple differential equation to be solved as follows:

$$d\langle R \rangle / dt = k \langle \kappa \rangle = k \langle R \rangle^{-1}$$

$$d\langle R \rangle \langle R \rangle = k dt$$

Therefore,

$$\langle R \rangle^2 = kt + C. \quad (20)$$

Thus the predicted exponent on area is one (or two for grain diameter). Provided that self-pinning does not occur (through poor choice of lattice) the exponent obtained from MC simulations is very close to one. There has been considerable attention paid to this point because the earliest results suggested an exponent slightly less than one. Various authors have ascribed this discrepancy to either a too small simulation domain, incorrect choice of switching algorithm (Radhakrishnan and Zacharia 1995), or to improper regression analysis. Nevertheless, the MC model is sufficiently robust that, provided a reasonable choice of lattice, domain size, and method of regression analysis is made, then the expected theoretical grain growth kinetics will be obtained. A more complex issue in grain growth is that of grain size distributions which space does not permit us to review in detail here. Suffice it to say that the distributions observed in 2D MC simulations correspond closely to experimental

observations (Srolovitz *et al.* 1984) and that, as of the time of writing, the theoretical basis for size distributions is still an active topic of discussion in the literature.

Another way of looking at the connection between polycrystal coarsening and motion by mean curvature is to examine the link between topology and size. There is a strong correlation between topological class and size that is known in both the experimental literature (Feltham 1957; Aboav and Langdon 1969) and for the MC model (Srolovitz *et al.* 1984). The nearly linear relationship between size and number of sides of a grain provides a direct link between the  $(n-6)$  rule, Eq. 17, and the mean field equation for growth rate of an individual grain, below. It is this connection that Hillert exploited to derive his seminal theory of grain growth based on the Lifshitz-Slyozov-Wagner theory of coarsening.

$$\frac{dR}{dt} = k \left( \frac{1}{R_{critical}} - \frac{1}{R} \right) \quad (21)$$

In recrystallization, there are two basic verifications of the model. The first is that the driving forces of curvature and stored energy can be set in opposition to one another. In the context of the isolated grain, this means that the Gibbs-Thomson effect can be verified: in other words, for a given grain boundary energy (set by  $J$ ) and stored energy (set by  $H$ ) there should be a grain size that neither shrinks nor grows. This is indeed the case for the MC model as has been verified (Rollett and Raabe 2001). Figure 8 shows the result of simulating the behavior of a single grain with an initial radius of eight sites and a range of stored energies associated with the sites surrounding the single grain. For a large enough stored energy, the grain radius grows with constant velocity as expected, Eq. 18. For small stored energies, the grain shrinks under capillary pressure. At some intermediate stored energy the size is metastable: fluctuations on either side will lead to growth or shrinkage.

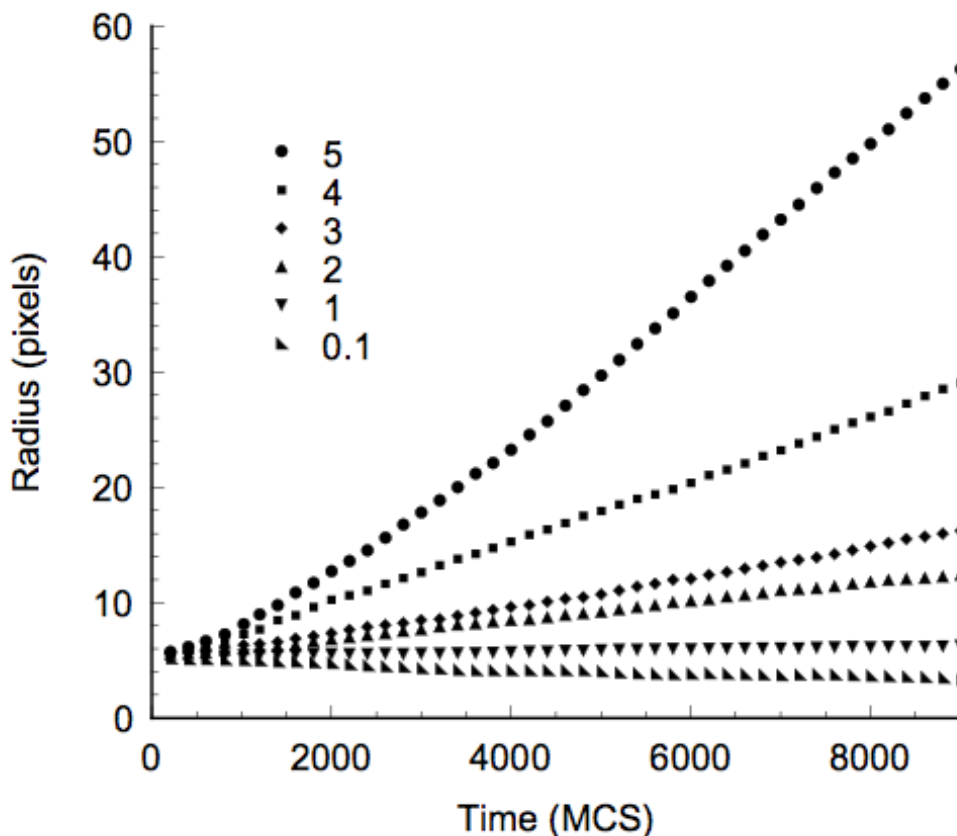


Figure 8: Plot of radius of single grain versus time for various bulk energies showing the Gibbs-Thomson effect. For large stored energies, the grain grows and vice versa; for an initial size of 5 sites and a stored energy of about one, the grain neither grows nor shrinks although the metastability of the equilibrium is reflected in the fact that any deviation in size will lead to either growth or shrinkage (Okuda, 2002).

The second verification is to perform simulations of recrystallization and compare the kinetics against the Kolmogorov-Johnson-Mehl-Avrami (KJMA) theory. This has been done by several investigators (Srolovitz *et al.* 1986) and extensions of the model have been made in order to explore experimentally observed deviations from the expected kinetics (Rollett *et al.* 1989). The slopes observed in KJMA plots correspond to the expected theoretical values for a variety of nucleation and growth morphologies.

#### 4.7 Scaling of Simulated Grain Size to Physical Grain Size

Most computer simulations require significant amounts of computer time so it common practice to minimize the size of the lattice that is used. Therefore it is useful to analyze the relationship between grain size in the Monte Carlo recrystallization model, and physical grain sizes (Rollett *et al.* 1992). As of the time of writing, increases in computer power have made it feasible to perform simulations in a few hours on desktop computers with more than a million grid points over simulation times of many millions of MCS.

We will assume that the ratio of stored energy to grain boundary energy per unit volume can be equated in the model and in real, physical recrystallization process. The physical process has two characteristic driving pressures for boundary migration,  $P^{\text{store}}$  and  $P^{\text{grgr}}$ . The stored dislocation content typically yields a  $P^{\text{store}}$  of 10 MPa. For grain growth driven by grain

boundary curvature,  $P^{\text{grgr}} = \gamma \langle R \rangle$ , where  $\gamma$  is the grain boundary energy per unit area, and  $\langle R \rangle$  is the mean grain radius. In the 2D triangular lattice, the stored energy per unit area is given by  $P^{\text{store}} = H/[3s^2 \sin(60^\circ)]$  where  $s$  is the unit boundary length on the lattice. The stored energy due to boundary curvature is

$$P^{\text{grgr}} = \frac{\gamma^{\text{model}}}{R^{\text{model}}} = \frac{J}{s R^{\text{model}}}. \quad (22)$$

The initial grain size in the Monte Carlo simulations is typically about  $6s$ . Thus, for the model

$$\frac{P^{\text{store}}}{P^{\text{grgr}}} = \frac{H R^{\text{model}}}{3sJ \sin 60^\circ} = 2.5 \frac{H}{J}. \quad (23)$$

Using a typical value for the grain boundary energy,  $\gamma = 0.5 \text{ J.m}^{-2}$ , for physical systems

$$\frac{P^{\text{store}}}{P^{\text{grgr}}} = 20 R \gamma^{\text{model}}. \quad (24)$$

Equating the energy densities for the model and physical systems and rearranging gives

$$\langle R \rangle = 0.125 H/J \text{ } \mu\text{m}. \quad (25)$$

Then for a typical simulation with  $H/J = 2$ , we can estimate  $\langle R \rangle = 0.25 \text{ } \mu\text{m}$ , which is a small but not unphysical grain size. Clearly, however, it would be preferable to simulate recrystallization with lattices with linear dimensions an order of magnitude larger than is currently typical.

#### 4.8 Recrystallization Kinetics in the Monte Carlo model

A characteristic of the Monte Carlo model of recrystallization is that a finite recrystallized volume is introduced at the beginning of simulations when site saturated nucleation conditions apply. This becomes apparent in a KJMA plot as a curvature at early times. A simple correction may be made for the finite initial fraction transformed by adding a constant to the measure of time. The correction is of course heuristic because it depends on the results themselves.

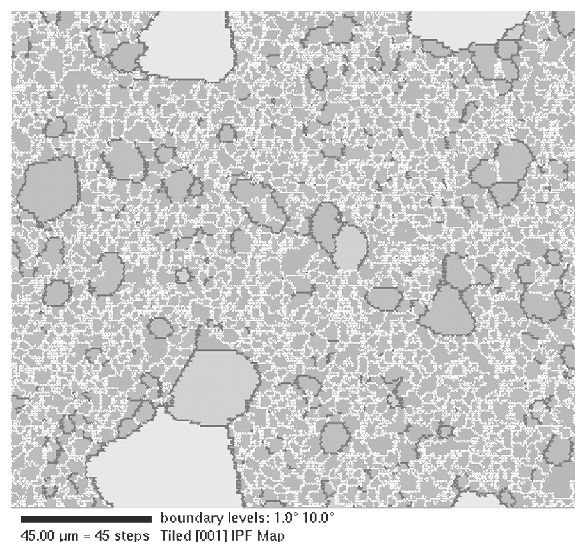
A limitation of the Monte Carlo model is that the growth rate of recrystallized grains is not linearly related to the stored energy density. As with nucleation, each integer increment of  $H/J$  leads to a discrete change in the number of sites that can change orientation with a neutral or negative change in system energy.

#### 4.9 Results of Simulation of Recrystallization by Monte Carlo method

The section provides a review of a number of areas in which useful results have been obtained from recrystallization simulation using the Monte Carlo method. Note that all the results discussed here are for two-dimensional simulations only.

#### 4.9.1 Abnormal Grain Growth

The early stages of recrystallization are labeled as nucleation even though no new phase appears in the material. At the level of the dislocation structure, however, the existing heterogeneities of the deformed structure coarsen in the process known as polygonization. The heterogeneities exist at several length scales from cells to shear bands to prior grain boundaries. Many observations have been made which suggest that individual subgrains acquire a growth rate advantage over their neighbors and become identifiable as new grains. This growth advantage can result from a difference in mobility between the boundary of the new grain and the boundaries in the surrounding material. This process of competitive growth has been observed in the Monte Carlo model in both 2D (Rollett *et al.* 1989) and 3D (Grest *et al.* 1990) simulations. By altering the rate at which sites are sampled for reorientation, the mobility of specific grain boundaries can be varied. Small ratios in mobility between one grain and another lead to marked abnormal grain growth behavior (Rollett and Mullins 1996), which may correspond to the early growth of new grains in recrystallization. Holm *et al.* (2003) have recently derived a theory for the frequency of abnormal grains as a function of orientation spread and the characteristic angle at which the grain boundary mobility transitions from low to high values. The key feature of the simulations carried out in this work was that coarsening in microstructures based on a single texture component with a realistic spread in orientation of the subgrain structure will occasionally exhibit abnormal grain growth. It turns out that if a particular grain has the topology required for growth (more than six sides, for example) and happens to be at the edge of the orientation distribution such that its perimeter possesses a high misorientation, it will grow much faster than the average size of the matrix. Figure 9 shows a snapshot of the microstructure from such a simulation after MCS. High angle grain boundaries are drawn in solid black whereas low angle boundaries are white. Grains that have grown to sizes significantly larger than the average size clearly tend to be surrounded by high angle, mobile boundaries.



*Figure 9:* Microstructure showing abnormal grain growth in a subgrain structure whose spread about the cube orientation was  $8^\circ$ . High angle grain boundaries ( $>10^\circ$ ) are drawn as black lines and low angle boundaries ( $>1^\circ$ ) are white. Large grains are associated with highly misoriented boundaries.

#### 4.9.2 Static Recrystallization

A key observation in grain growth is that an isolated grain will shrink and eventually vanish in response to tendency to minimize grain boundary area. When an isolated grain can eliminate stored energy by growing, however, it does so, provided that the grain is not smaller than a critical size. In a polycrystalline structure with  $0 < H/J < 1$ , any recrystallized grain will grow at the expense of its neighbors because of the bias imposed on the unit step of the grain growth process. That is to say, the motion of kinks along a boundary is reversible when the change in energy associated with a step is zero, as is often the case for the 2D triangular lattice; however, such energy neutral steps become irreversible when biased by the elimination of stored energy at each step. Clearly, the Monte Carlo simulation of the motion of recrystallization fronts is much closer to a deterministic model (e.g. cellular automaton) than for grain growth.

The kinetics of recrystallization have been found to reproduce those anticipated from theoretical analysis very closely. For example, in 2D simulations (Srolovitz et al. 1986) site saturated nucleation conditions with a high enough stored energy density ( $H/J > 2$ ), a KJMA plot of the fraction recrystallized versus time show a slope of 2 at long times; continuous nucleation gives a slope of 3. Both results are as predicted from classical KJMA analysis. For low stored energy densities ( $H/J < 2$ ), nucleation is heterogeneous in the sense that embryos must be adjacent to existing boundaries in order to survive and grow. In this case the kinetics show significant deviations from the classical KJMA pattern (Srolovitz and Hassold 1987). The key observation here is that the growth of new grains at low stored energies is highly dependent on the prior structure. For  $H/J \approx 1$ , the recrystallization front grows out from a triple point and is concave with respect to the unrecrystallized side. The net effect is that growth is very slow in early stages of recrystallization.

Work by Martin (1994) with the Monte Carlo method has examined the effect of spatially non-random distributions of nuclei under site-saturated conditions. This work used a variant of the  $n$ -fold way in which only flips to nearest neighbor orientations were considered. Also, a stored energy (as a scalar contribution to the system energy associated with each unrecrystallized site) was not explicitly considered. Instead, once a recrystallized grain was inserted, flips are allowed from unrecrystallized to recrystallized, but not vice versa, thereby guaranteeing growth of the recrystallized regions. The main result was to demonstrate the importance of the distribution of nuclei. Nucleation on a square lattice gave narrow size distributions whereas (heterogeneous) nucleation on a coarse prior grain structure gave wide distributions and non-compact grain shapes. Subsequent grain growth broadens the size distribution and, as expected, grain shapes become compact.

#### 4.9.3 Grain Growth in the Presence of Particles

Miodownik *et al.* (2000) applied a parallel version of the 3D model with the primary aim of being able to perform simulations on particle pinning with large domain sizes. For the problem of particle pinning that they studied, this was of crucial importance because they needed to use particles larger than one voxel (a unit volume element) in order to avoid thermally activated unpinning. Their result was instructive: in contrast to earlier 2D studies (Srolovitz *et al.* 1984; Doherty *et al.* 1990) that suggested that particles pinned grain boundaries more effectively than predicted by the Zener-Smith theory (Zener 1948), the new 3D results showed that the classical theory was indeed applicable. The difference between 2D and 3D appears to be that in 2D, particles can remove curvature because a boundary can effectively pivot about a particle. Thus the limiting grain size is similar to the nearest



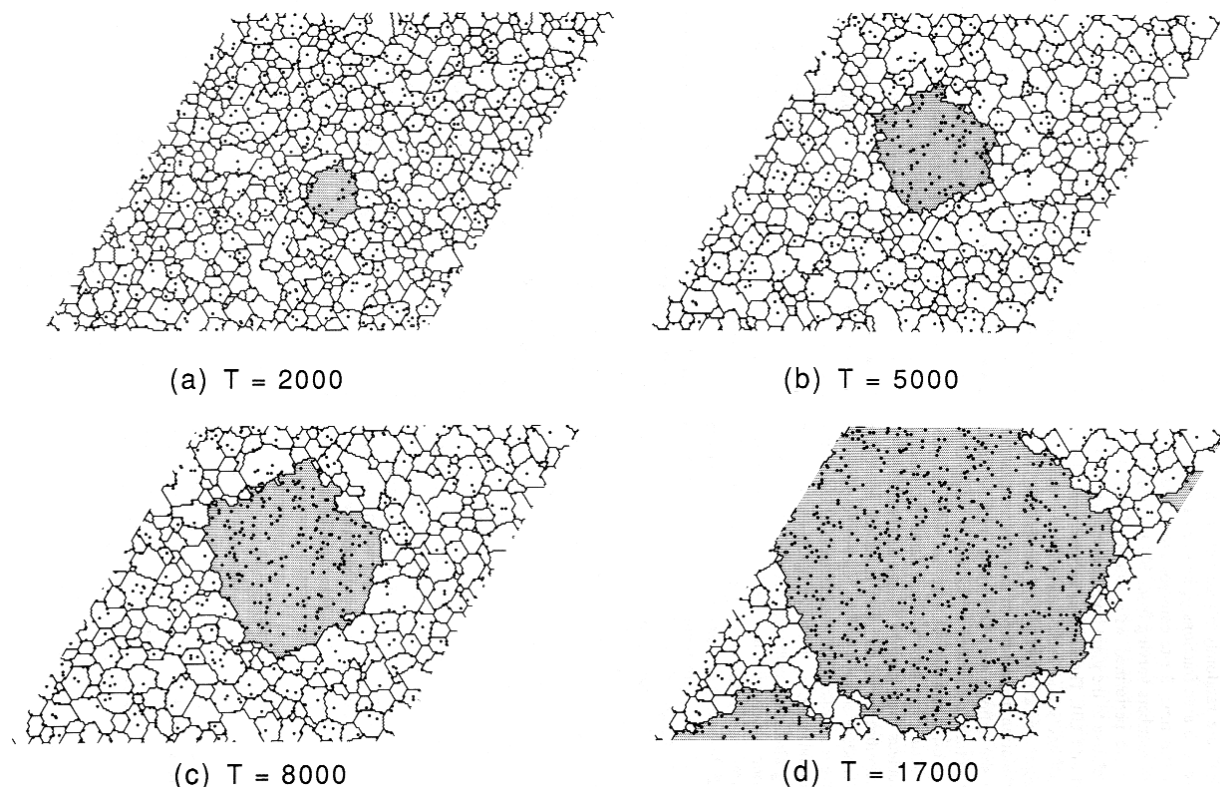
neighbor spacing,  $\Delta_2$ . In 3D, however, the effect of a particle on a boundary is more local and so the limiting grain size is similar to the mean free path between particles.

#### 4.9.4 Recrystallization in the Presence of Particles

Recrystallization with inert particles present is easily modeled by assigning sites orientation values that cannot be changed. Although only the effects of single site particles have been studied (Rollett *et al.* 1992) in 2D simulations (whereas particle shape and size has been examined for grain growth (Hassold *et al.* 1990)), the results appear to be general, at least for small particles. Large particles in physical systems have the effect of stimulating nucleation, an effect that has not been addressed by microstructural simulations. Another worthwhile extension of computer modeling of recrystallization would be three-dimensional simulations because the interaction of boundaries with particles is quite different than in two dimensions.

During recrystallization simulation with sufficient stored energy ( $H/J \geq 3$  in the triangular lattice) the recrystallization front can readily bypass particles regardless of particle size or area fraction. Under these circumstances the recrystallization growth kinetics are unaffected by particles. The overall kinetics is accelerated slightly, however, by heterogeneous nucleation on particles.

At intermediate stored energies ( $1 \leq H/J \leq 3$ ) nearly all recrystallization boundaries can move past single-site particles. Boundaries with a very high particle density will stop moving. However, the irreversible propagation of grain boundary kinks allows most boundaries to achieve a configuration from which two kinks can join to move past single-site particles. The presence of prior grain boundaries further enhances recrystallized boundary motion. In these systems, boundaries intersect particles at random, and recrystallization kinetics are substantially unaffected by particles, as shown in Figure 10, where a single recrystallized grain shows unrestricted growth for  $H/J = 1$ . Larger particles may inhibit recrystallization in this stored energy regime, however, since two boundary kinks cannot join directly.



*Figure 10.* Microstructures of a single recrystallizing grain growing into a matrix containing inert second phase particles. The last picture illustrates the effect of periodic boundary conditions as the new grain wraps around the edges of the simulation.

At low stored energies ( $H/J < 1$ ) grain boundary energy governs boundary motion, and the recrystallization front is strongly pinned by particles which leads to a much higher than random density of particles on the recrystallization front. In these circumstances recrystallization is strongly inhibited which usually results in incomplete recrystallization, as shown in Fig. 13. When only small particle fractions are present, however, recrystallization may go to completion because pinning does not occur until after the transformation is complete.

In addition, prior grain boundaries may still enhance motion of recrystallized grain boundaries, so that the recrystallized grains can grow (at low particle fractions) much larger than the deformed matrix grains. This is sufficient to drive the recrystallizing grains past some particles, but only if the matrix grain size is much smaller than the interparticle spacing. In other words, recrystallization at low stored energy and at very low particle fractions is similar to abnormal grain growth.

Grain boundaries undergoing curvature driven growth (both in the deformed state and after complete recrystallization) rapidly acquire a higher than random density of particles which then inhibits grain growth. If the recrystallized grain size is smaller than the critical grain size, grain growth continues until pinning occurs and the microstructure is a normal grain growth microstructure. However, when the recrystallized grain size is large compared to the critical grain size, particle pinning occurs almost immediately following the completion of recrystallization thus preserving the non-compact grain shapes and sharply peaked grain size distribution that are characteristic of randomly distributed nuclei.

Analysis of recrystallization in particle containing materials suggests that there are two limiting values of particle drag; a low (Zener) value with a random density of particles and a much higher value if particles have become highly correlated with the recrystallization front (as in grain growth). The simulation results show both these behaviors, depending on the H/J ratio. Figure 11 plots the density of particles on boundaries for two types of boundaries; recrystallization fronts have a lower, near random spatial distribution of particles; general boundaries exhibit a higher than random density, suggesting that they are more strongly pinned. Experimental studies appear to show only the lower particle drag (Zener) as studied by Ashby et al. (1969). A note of caution about the comparison is that boundary-particle interaction is more complex in three dimensions than in two.

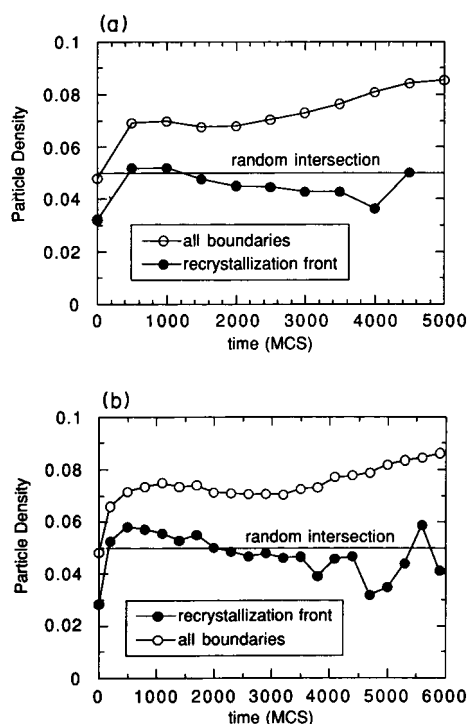


Figure 11. Plot of the fraction of particles on either recrystallization fronts (solid symbols) or on all boundaries (open symbols) for two different densities of nuclei –(a) 100 and (b) 1000 site saturated nuclei.

#### 4.9.5 Texture Development

Tavernier and Szpunar extended the 2D Monte Carlo model of Anderson *et al.* (1984) to account for the different boundary energies and mobilities expected for different texture components. They chose low carbon steel sheet as their model material and developed parameters to account for eight different texture components. They found that the  $\langle 111 \rangle$  class of texture component tended to become dominant after a period of grain growth when the simulations were run with high boundary energies between the  $\langle 111 \rangle$  component grains and other texture components. They also simulated the recrystallization process by making provision for the variation of stored energy and boundary mobility, depending on whether a given volume of material is recrystallized or not. Unfortunately many details of the key parameters were omitted from their paper so that it is difficult to judge the success of their efforts. They also point out, reasonably enough, that there is very little experimental data available for the variation of grain boundary energy and mobility over the general range of

misorientations. This lack is currently being addressed by both simulation of grain boundary properties (Upmanyu *et al.* 1998) and by experimental measurement (Yang *et al.* 2001; Saylor *et al.* 2002). More recently, Hinz & Szpunar have used the MC model to investigate the role of special boundary types (coincident site lattice boundaries) on texture development in electrical steels, especially with respect to the Goss texture,  $\{110\}\langle 001\rangle$ .

Holm *et al.* (2001) have investigated the development of grain boundary character as quantified by the misorientation distribution during grain growth. They find that low energy boundaries become strongly preferred during coarsening in 2D structures where the range of orientation is limited to only one degree of freedom. In 3D coarsening with 3-parameter orientations, however, even if the special boundaries based on CSL relationships are assigned low energies, the grain boundary character does not reflect the energy distribution because of the geometric constraints.

Recently Rollett (2003 JOM) has investigated the development of the cube texture component in fcc metals. The investigation was confined to grain growth with no stored energy driving forces. The initial microstructure was based on a microstructure from simulations of isotropic grain growth. A rolling texture was imposed on the structure except for a small fraction of the grains that were assigned orientations close to the cube component. A combination of a Read-Shockley model of grain boundary energy for low angle boundaries and a broad maximum in mobility around a misorientation of  $40^\circ\langle 111\rangle$  with very low mobility at low misorientation angles was used to describe the grain boundary properties. For a sufficiently large ratio of maximum to average mobility, the cube component was observed to increase markedly during grain growth.

Incorporating texture into the Monte Carlo model is straightforward although, as always, symmetry (both crystal and sample-based) must be dealt with carefully. The most efficient way to proceed is to calculate all the grain boundary properties that could be required during a simulation at the beginning and store them in a look-up table. Each spin number of the  $Q$  possible spins represents an individual orientation. The required lookup table is then  $Q(Q-1)/2$  in then size for each property in order to account for all possible boundary types in the system.

#### 4.9.6 Texture

Although texture is a substantial topic in itself, it is perhaps helpful to the reader to provide a basic introduction while providing references to the standard works. Texture has to do with how one quantifies the relationship between the crystal axes in a particular grain of a polycrystal and a set of axes associated with the external shape of the polycrystalline body. Most often the term texture is used in conjunction with x-ray pole figures which are the most efficient experimental means of quantifying the average texture of a (polycrystalline) material. Most fundamental is the understanding that crystallographic orientation requires specification of a rotation. This rotation is most often used as an axis transformation in order to express properties known in crystal axes into properties in the frame of the material. In the context of Monte Carlo simulations, orientations are used to determine the properties of grain boundaries. The second vital piece of information is that rotations can be expressed in a wide variety of mathematical and not-so-mathematical parameterizations, all of which have three independent parameters. The standard symbol for orientation is “ $g$ ” but other symbols are used, especially for the less well-known *Rodrigues-Frank vectors* and *quaternions*. The table below provides a summary of commonly used approaches. The entries are ordered by

familiarity to materials scientists. Specification of a plane and direction by *Miller indices* is most intuitive but numerically awkward! *Euler angles* are common because of the convenience series expansion methods based on generalized spherical harmonics. Serious computational work, however, uses quaternions for speed and simplicity. *Axis-angle* descriptions are the most intuitive description of grain boundaries because of the intimate connection to crystal geometry and Rodrigues-Frank vectors have some very attractive features for both representation and for certain types of computation. A few conversion formulae are given in the table in order to provide some clarification of the meaning of the parameters. There are formulae available in the standard texts to convert between any pair of representations (Bunge 1982; Kocks *et al.* 1998).

Name	Parameters	Conversion Formulae
<b>Texture Component:</b> Specifies alignment of a plane normal with a sample direction (e.g. ND) and a crystal direction with, e.g. RD. Each normal and direction is, in effect, a unit vector and they must be perpendicular, yielding three independent parameters.	(hkl)[uvw]	
<b>Euler Angles:</b> The Euler angles specify a triple of rotations (transformations) about the Z, X and Z directions. Many variants of Euler angles are known.	$g = g(\alpha, \beta, \gamma)$	$a_{ij} = \begin{pmatrix} \cos\alpha_1 \cos\alpha_2 & \sin\alpha_1 \cos\alpha_2 & \sin\alpha_2 \sin\alpha_1 \\ \sin\alpha_1 \cos\alpha_2 & \cos\alpha_1 \cos\alpha_2 & \cos\alpha_2 \sin\alpha_1 \\ \cos\alpha_1 \sin\alpha_2 & \sin\alpha_1 \sin\alpha_2 & \cos\alpha_2 \sin\alpha_1 \\ \sin\alpha_1 \cos\alpha_2 \cos\alpha_3 & \cos\alpha_1 \cos\alpha_2 \cos\alpha_3 & \cos\alpha_3 \\ \sin\alpha_1 \sin\alpha_2 & \cos\alpha_1 \sin\alpha_2 & \cos\alpha_3 \end{pmatrix}$
<b>(Orthogonal) Matrix:</b> The coefficients of an axis transformation are defined by: $a_{ij} = \hat{e}_i \cdot \hat{e}'_j$ , where the $e$ are the unit basis vectors in the primed and unprimed coordinate systems. All columns and rows are unit vectors which reduces the number of independent coefficients to 3.	$g = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} =$	$a_{ij} = \begin{pmatrix} t_1 & n_1 \\ t_2 & n_2 \\ t_3 & n_3 \end{pmatrix} \text{ Crystal} \begin{pmatrix} b_1 & t_1 & n_1 \\ b_2 & t_2 & n_2 \\ b_3 & t_3 & n_3 \end{pmatrix} \text{ Sample}$
<b>Axis-Angle:</b> The rotation axis is specified by a unit vector, $\mathbf{n}$ , and the rotation angle by $\alpha$ . For	$g = g(\alpha, \mathbf{n})$	$a_{ij} = \cos\alpha \delta_{ij} + \sin\alpha \epsilon_{ijk} n_k + (1 - \cos\alpha) n_i n_j$

<p>grain boundaries, the rotation axis is often specified in crystallographic terms with a set of Miller indices.</p>		
<p><b>Rodriguez-Frank Vector:</b> The RF-vector is the rotation axis but scaled by the tangent of the semi-angle. In this space, <i>all</i> rotations that share a common rotation axis lie on a straight line.</p>	$\mathbf{r} = \{r_1, r_2, r_3\} = \tan\left(\frac{\theta}{2}\right) \mathbf{n}$	$r_1 = \frac{\tan\left(\frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right) \cos\left(\frac{\theta}{2}\right)}{\cos^2\left(\frac{\theta}{2}\right)}$ $r_2 = \frac{\tan\left(\frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right)}{\cos^2\left(\frac{\theta}{2}\right)}$ $r_3 = \tan\left(\frac{\theta}{2}\right) \cos\left(\frac{\theta}{2}\right)$
<p><b>Quaternion:</b> Closely related to the Rodriguez-Frank vector. The <math>n_i</math> are the components. For rotations, the quaternion is always of unit length. Be aware that the quaternions representing <math>\theta</math> and <math>2\pi - \theta</math> are the negative of each other but represent the same orientation (2-to-1 mapping).</p>	$q = \{q_1, q_2, q_3, q_4\} = \left\{ \sin\left(\frac{\theta}{2}\right) n_1, \sin\left(\frac{\theta}{2}\right) n_2, \sin\left(\frac{\theta}{2}\right) n_3, \cos\left(\frac{\theta}{2}\right) \right\}$	$q_1 = \sin\left(\frac{\theta}{2}\right) \cos\left(\frac{\theta}{2}\right) n_1$ $q_2 = \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right) n_2$ $q_3 = \cos\left(\frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right) n_3$ $q_4 = \cos\left(\frac{\theta}{2}\right) \cos\left(\frac{\theta}{2}\right)$

The next step in understanding and using texture is to become familiar with the characteristic preferred orientations of the particular material and processing history of interest. This is far too broad a subject for treatment here and the reader is referred to Kocks, Tomé and Wenk (1998) and Randle and Engler (2002) for detailed information and analysis.

From a mathematical point of view, the next step is to realize that rotations can be combined together. The simplest method is matrix multiplication: any pair of orthogonal matrices can be (matrix) multiplied together to yield another rotation. Note that group theory is very useful in this context. Less intuitive but just as useful are the methods of combining (or ‘composing’) two Rodrigues vectors or quaternions. Two Rodrigues vectors combine to form a third as follows where  $\mathbf{r}_2$  follows after  $\mathbf{r}_1$ :

$$(\mathbf{r}_1, \mathbf{r}_2) = \{\mathbf{r}_1 + \mathbf{r}_2 - \mathbf{r}_1 \times \mathbf{r}_2\} / \{1 - \mathbf{r}_1 \cdot \mathbf{r}_2\}. \tag{26}$$

The algebraic form for combining quaternions is given as, where  $\mathbf{q}_B$  follows  $\mathbf{q}_A$ :

$$\mathbf{q}_C = \mathbf{q}_A \cdot \mathbf{q}_B$$

$$\begin{aligned} q_{C1} &= q_{A1} q_{B4} + q_{A4} q_{B1} - q_{A2} q_{B3} + q_{A3} q_{B2} \\ q_{C2} &= q_{A2} q_{B4} + q_{A4} q_{B2} - q_{A3} q_{B1} + q_{A1} q_{B3} \\ q_{C3} &= q_{A3} q_{B4} + q_{A4} q_{B3} - q_{A1} q_{B2} + q_{A2} q_{B1} \\ q_{C4} &= q_{A4} q_{B4} - q_{A1} q_{B1} - q_{A2} q_{B2} - q_{A3} q_{B3} \end{aligned} \quad (27)$$

Now we can introduce the concept of misorientation, or the difference in orientation between two crystals, to which the properties of boundaries can be related. Mathematically, this is just the combination of one orientation with the inverse of the other. One very important caveat must be noted here which is that the order in which the orientation and the inverse orientation are combined is important: recall that matrix multiplication is not commutative. One combination produces a rotation whose axis is in crystal coordinates whereas the inverse order produces a rotation whose axis is in sample coordinates. This is also important with respect to the application of symmetry operators which will be discussed next. The most common approach, by far, is to write misorientations in crystal coordinates because that is the physically more meaningful approach in almost all cases. If the orientations are expressed as matrices representing axis transformations, the misorientation is given by the following, where superscript  $T$  denotes *transpose*:

$$\Delta g = g_2 \cdot g_1^T \quad (28)$$

Similarly for quaternions, the expression is:

Also, to obtain a misorientation in crystal axes,

$$\mathbf{q}_C = \mathbf{q}_A \cdot \mathbf{q}_B^{-1}$$

$$\begin{aligned} q_{C1} &= q_{A1} q_{B4} - q_{A4} q_{B1} + q_{A2} q_{B3} - q_{A3} q_{B2} \\ q_{C2} &= q_{A2} q_{B4} - q_{A4} q_{B2} + q_{A3} q_{B1} - q_{A1} q_{B3} \\ q_{C3} &= q_{A3} q_{B4} - q_{A4} q_{B3} + q_{A1} q_{B2} - q_{A2} q_{B1} \\ q_{C4} &= q_{A4} q_{B4} + q_{A1} q_{B1} + q_{A2} q_{B2} + q_{A3} q_{B3} \end{aligned} \quad (29)$$

Finally we must summarize the effect of crystal symmetry because there are many physically equivalent descriptions of any misorientation because of the multiplicity of ways in which crystal axes can be labeled. The following expression summarizes the way in which smallest possible rotation angle can be identified. The formula for determining the smallest misorientation angle,  $\square^*$ , is as follows, where the symmetry operators,  $O$ , are drawn from the set of  $n$  members of the (proper rotation) point group appropriate to the crystal symmetry:

$$\square^* = \min \left\{ \cos^{-1} \left[ \frac{1}{2} \text{tr} \left( O^{(i)} g_B g_A^{-1} O^{(j)} \right) \right], \cos^{-1} \left[ \frac{1}{2} \text{tr} \left( O^{(k)} g_A g_B^{-1} O^{(l)} \right) \right] \mid i, j, k, l \in \{1 \dots n\} \right\} \quad (30)$$

Inspection of this expression will reveal that it can also be used to specify the misorientation in a unique way. In fact, one can choose specific symmetry operators in such a way as to *always* locate the misorientation axis in a particular asymmetric unit such as the *standard stereographic triangle (SST)* for cubic materials. Finally, we note that the properties of grain

boundaries (and other interfaces) very often depend also on the boundary normal in addition to the misorientation. This adds another two parameters to the three already required to describe misorientation and boundary normals require additional computational effort in the MC model (and which has not yet been implemented as of the time of writing).

#### 4.9.7 Dynamic Recrystallization

Dynamic recrystallization has been successfully modeled with the 2D Monte Carlo model (Rollett *et al.* 1992). In its simplest form, stored energy at each point of the lattice is increased at a fixed rate and embryonic new grains are continually added at a constant rate. The basic result is that temporal oscillations are observed both in the grain size and in the stored energy, which is analogous to the oscillations in the flow curve, as shown in Figure 12. These oscillations are observed over almost the entire range of recrystallization parameters examined (energy storage rate, nucleation rate, initial grain size) and damp out over time periods that decrease with increasing storage rate and increasing nucleation rate. The oscillations in both grain size and stored energy have the same period but are out of phase by approximately one quarter of a period. Examination of the simulated dynamic recrystallization microstructures which were formed under the same conditions but with different initial grain sizes shows that the evolution of the microstructure may be divided into three distinct stages: an initial microstructure dependent transient stage, an initial microstructure independent transient stage, and a steady state stage. While necklace nucleation was observed in these simulations under some circumstances, it is apparent that it is not a necessary condition for grain refinement in dynamic recrystallization. This phenomenon is associated with refinement of the relatively coarse initial microstructure and over-damped oscillations in the flow stress. Therefore, even when there are no obvious oscillations in the flow curve and no necklace nucleation is observed, dynamic recrystallization cannot be precluded.

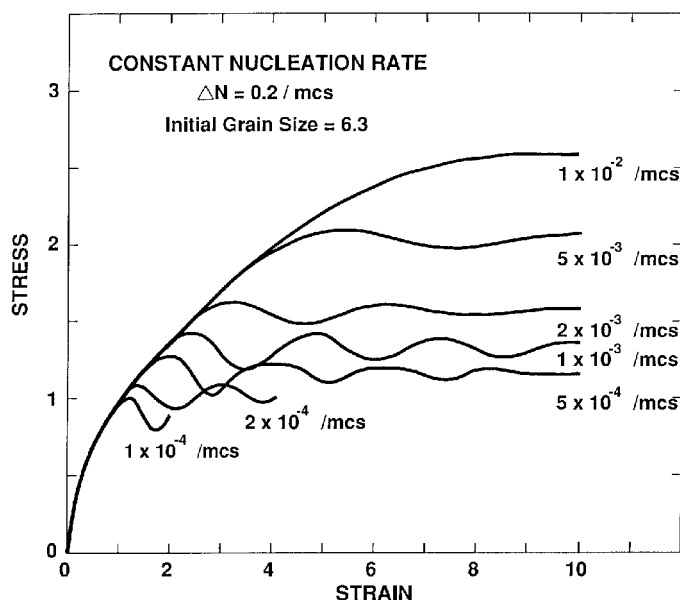


Figure 12. Stress plotted versus strain for a range of rates of addition of stored energy in a Monte Carlo model, Rollett (1992). Note the oscillations observed in the simulated flow curves whose period varies with the rate of addition of stored energy.



It should be noted that the geological community had applied the Monte Carlo model to dynamic recrystallization but had only examined the microstructural aspects. Jessel (1988) described an adaptation of the Monte Carlo model for the simulation of deformation of quartzite. Although the model is similar to the combined grain growth and recrystallization model, the Jessel work used only differences in stored energy between sites to evaluate transition probabilities for orientation changes. In this form, the model is essentially a cellular automaton (CA) model. Also, the only results given were for microstructural evolution, with no attempt to investigate stress-strain relationships. The simulations were used to investigate the development of texture (*fabric* in geological terms), based on the Taylor model (strain compatibility enforced on all grains) with some degree of success.

Peczak and coworkers have investigated several aspects of the correspondence between this type of simulation and have added several refinements to the model, see for example (Peczak 1995), in order to allow detailed comparisons with experimental data. They have modified the nucleation process such that the probability of an embryo appearing in a region of high stored energy is higher than for low stored energies. This has the consequence that the effective nucleation rate increases with time as stored energy is added to the system and is also position dependent during the simulation. They have also modified the rate of stored energy addition from the original constant addition rate to correspond to a Voce-type equation (i.e. an exponential work hardening curve) with a saturation (asymptotic) flow stress expressed in terms of temperature-compensated strain rate (Zener-Holloman parameter). By making these modifications they have been able to reproduce many of the characteristics of the phenomenon of dynamic recrystallization. For example, the transition from multiple peaks to a single peak in the flow curve occurs when the ratio of the initial to the steady state grain size goes over 2.3. This is in agreement with Sakai's experimental observations on the effect of initial grain size on dynamic recrystallization in OHFC copper (Sakai 1995). The relationship between the steady state grain size and flow stress in the model shows grain size decreasing with increasing stress as expected; the slope is  $\sim 1$ , Figure 13, which is close to the experimental value of 0.7 (Derby 1991).

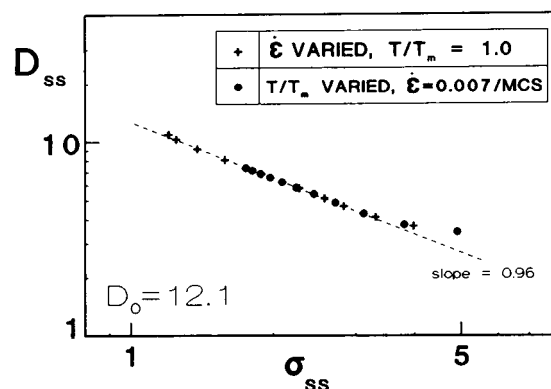


Figure 13. Plot of asymptotic grain size against steady state stress derived from Monte Carlo simulations, after Peczak (1995).

#### 4.10 Summary

The principal methods for modeling the phenomenon of recrystallization have been reviewed. In addition to the geometrical and analytical models, several methods of modeling microstructural evolution during recrystallization are briefly described. Many of the latter examples illustrate the need to be able to model the local environment of grains. In view of the frequency with which the Monte Carlo model has been used in the literature, that method

has been reviewed in more depth. The characteristics of the various lattice types, the Hamiltonians for the system energy, the transition probabilities, and the continuous time method were briefly treated. Key results from the simulation literature have been reviewed and summarized.

#### 4.11 *Acknowledgements*

Numerous discussions over the years with D.J. Srolovitz, Elizabeth Holm, A. Karma, Y. Brechet, B. Radhakrishnan, S. Cheong, K. Okuda and others are gratefully acknowledged. This work was supported primarily by the MRSEC program of the National Science Foundation under Award Number DMR-0079996. Additional support was provided by the Computational Materials Science Network, a program of the Office of Science, US Department of Energy and by the Programming, Environment and Training Program under the auspices of the High Performance Computing Modernization Office of the US Department of Defense.

## 4.11 References

- Aboav, D. and T. Langdon (1969). "The shape of grains in a polycrystal." Metallography **2**: 171-178.
- Anderson, M.P., G. Grest and D.J. Srolovitz (1989). "Computer simulation of normal grain growth in three dimensions." Phil. Mag. B **59**(3): 293-329.
- Anderson, M. P., D. J. Srolovitz, G. S. Grest and P. S. Sahni (1984). "Computer simulation of grain growth- I. Kinetics." Acta metallurgica **32**(5): 783-791.
- Ashby, M. F. and R. M. A. Centamore (1968). Acta metall. **16**: 1081.
- Ashby, M. F., J. Harper, et al. (1969). "The interaction of crystal boundaries with second-phase particles." Trans. Am. Inst. Min. Engrs. **245**: 413.
- Baudin, T., P. Paillard, et al. (1999). "Simulation of the anisotropic growth of Goss grains in Fe3%Si sheets (grade HiB)." Scripta Materiala **40**(10): 1111-1116.
- Bortz, A. B., Kalos, M. H., Lebowitz, J. L. (1975). "A new algorithm for Monte Carlo simulations of Ising spin systems." Journal of Computational Physics **17**: 10-18.
- Bunge, H., Texture Analysis in Materials Science (1982). Butterworths, London.
- Cheong, S. W., E. J. Hilinski, et al. (2003). "Grain Growth in a Low-Loss Cold-Rolled Motor-Lamination Steel." Metallurgical and Materials Transactions: accepted for publication.
- Demirel, M. C., A. P. Kuprat, et al. (2003). "Bridging Simulations and Experiments in Microstructure Evolution." Physical Review Letters **90**: 016106.
- Derby, B. (1991). "The dependence of grain size on stress during dynamic recrystallization." Acta metallurgica **39**: 955-962.
- Doherty, R. D., K. Li, et al. (1990). Modelling of recrystallization kinetics and particle limited grain boundary motion. Recrystallization 90, TMS-AIME Warrendale, PA.
- Feltham, P. (1957). Acta metallurgica **5**: 97.
- Glauber R. J. (1963). "Time-dependent statistics of the Ising model." J. Mathematical Physics **4**(2): 294 – 307.
- Grest, G. S., M. P. Anderson, et al. (1990). "Abnormal grain growth in three dimensions." Scripta metall. et mater. **24**: 661-665.
- Hassold, G. N. and E. A. Holm (1993). "A fast serial algorithm for the finite temperature quenched Potts model." Computers in Physics **7**(1): 97-107.
- Hassold, G. N., E. A. Holm, et al. (1990). "Effects of particle size on inhibited grain growth." Scripta metallurgica et materiala **24**: 101-106.
- Hinz, D.C. and J.A. Szpunar (1995.) "Modeling the effect of coincidence site lattice boundaries on grain growth textures." Physical Review, **B52**(14): p. 9900-9909.
- Holm, E. A. (1992). Modeling microstructural evolution in single-phase, composite and two-phase polycrystals, Ph. D. Thesis, Univ. Michigan.
- Holm, E. A. (1996). Critical Nucleus Sizes in the Potts model (*private communication*).
- Holm, E. A. (2002). "Proof of boundary motion by curvature in the quenched ferromagnetic Ising model." Physical Review Letters: submitted.
- Holm, E. A., J. A. Glazier, D.J. Srolovitz and G.S. Grest (1991). "The effects of lattice anisotropy and temperature on domain growth in the 2-D Potts model." Phys. Rev. A **43**(6): 2662-2668.
- Holm, E. A., G. N. Hassold and M. Miodownik (2001). "On misorientation distribution evolution during anisotropic grain growth." Acta materiala **49**: 2981-2991.
- Holm, E. A., M. Miodownik and A.D. Rollett (2003). "On abnormal subgrain growth and the origin of recrystallization nuclei." Acta materiala: **51**, 2701-2716.
- Ising, E. (1925). "Beitrag zue Theorie des Ferromagnetismus." (Contribution to the theory of ferromagnetism) Zeitschrift für Physik, **31**: 253-258.

- Jessel, M. W. (1988). "Simulation of fabric development in recrystallizing aggregates-I. Description of the model." J. Structural Geology **10**(8): 771-778.
- Kawasaki, K. (1972), in Phase Transitions and Critical Phenomena, C. Domb and M.S. Green, Editors, Academic Press: New York.
- Kocks, U.F., C. Tomé, and H.-R. Wenk, eds. (1998). Texture and Anisotropy, Cambridge University Press, Cambridge.
- Korniss, G., Novotny, M. A. and Rikvold, P. A. (1999). "Parallelization of a dynamic Monte Carlo algorithm: a partially rejection-free conservative approach." Journal of Computational Physics **153**: 488-508.
- Landau, D. P. and K. Binder (2000). A Guide to Monte Carlo Simulations in Statistical Physics. Cambridge, England, Cambridge University Press.
- Lubachevsky, B. D. (1988). "Efficient parallel simulations of dynamic Ising spin systems." Journal of Computational Physics, **75**: 103 – 122.
- Lubachevsky, B. D. (1987). "Efficient parallel simulations of dynamic Ising spin systems." Complex Systems, **1**: 1099 – 1123.
- Martin, J., R.D. Doherty, and B. Cantor (1997). Stability of Microstructure in Metallic Systems, 2nd ed., Cambridge University Press, Cambridge, England.
- Martin, D. G. (1994). "Computer simulation of recrystallization and grain growth." Materials Science and Technology **10**: 855-861.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). "Equation of state calculations by fast computing machines." The Journal of Chemical Physics **21**(6): 1087-1092.
- Miodownik, M., A. Godfrey, E.A. Holm, D.A. Hughes (1999). "On boundary misorientation distribution functions and how to incorporate them into three-dimensional models of microstructural evolution." Acta materialia **47**(9): 2661-2668.
- Miodownik, M., E. Holm, et al. (2000). "Highly parallel computer simulations of particle pinning: Zener vindicated." Scripta Materialia **42**: 1173-1177.
- Mullins, W. W. (1956). "Two-dimensional motion of idealized grain boundaries." J. Appl. Phys. **27**: 900-904.
- Novotny M. A. (1995). "A new approach to an old algorithm for the simulation of Ising-like systems", Computers in Physics, **9**(1): 46 – 52.
- Okuda, K. (2002). private communication.
- Pathria R. K. (1972): Statistical Mechanics, Pergamon Press, Oxford.
- Peczak, P. (1995). "A Monte Carlo study of influence of deformation temperature on dynamic recrystallization." Acta Metall. **43**(3): 1279-1291.
- Plischke, M. and Bergensen, B. (1994). Statistical Equilibrium Physics, 2<sup>nd</sup> ed., World Scientific Publishing, Singapore.
- Potts, R. B. (1952). "Some generalized order-disorder transitions", Proc. Cambridge Philosophical Society, **48**: 106 – 109.
- Radhakrishnan, B. and T. Zacharia (1995). "Simulation of curvature-driven grain growth using a modified Monte Carlo algorithm." Metallurgical and Materials Transactions **26A**: 167.
- Radhakrishnan, B., G.B. Sarma, and T. Zacharia (1998). "Modeling the kinetics and microstructural evolution during static recrystallization- Monte Carlo simulation of recrystallization." Acta materialia, **46**(12): p. 4415-4433.
- Randle, V. and O. Engler (2000). Texture Analysis: Macrotecture, Microtexture & Orientation Mapping, Amsterdam, Holland: Gordon & Breach. 388.
- Rollett, A. D., M. J. Luton, et al. (1992). "Computer Simulation of Dynamic Recrystallization." Acta metall. mater. **40**: 43-55.

- Rollett, A. D. and W. W. Mullins (1996). "On the growth of abnormal grains." Scripta metall. et mater. **36**(9): 975-980.
- Rollett, A. D. and D. Raabe (2001). "A Hybrid Model for Mesoscopic Simulation of Recrystallization." Computational Materials Science **21**(1): 69-78.
- Rollett, A. D., D. J. Srolovitz, et al. (1989). "Simulation and Theory of Abnormal Grain Growth- Variable Grain Boundary Energies and Mobilities." Acta Metall. **37**: 2127.
- Rollett, A. D., D. J. Srolovitz, et al. (1992). "Computer Simulation of Recrystallization - III. Influence of a Dispersion of Fine Particles." Acta Metallurgica **40**: 3475-3495.
- Rollett, A. D., D. J. Srolovitz, et al. (1989). "Computer Simulation of Recrystallization in Non-Uniformly Deformed Metals." Acta Metall. **37**: 627.
- Sahni, P., D. Srolovitz, et al. (1983). "Kinetics of ordering in two dimensions. II. Quenched systems." Phys. Rev. B **28**: 2705.
- Sarma, G.B., B. Radhakrishnan, and P.R. Dawson (2002). "Mesoscale modeling of microstructure and texture evolution during deformation processing of metals." Advanced Engineering Materials, **4**(7): p. 509-514.
- Saylor, D. M., J. Fridy, et al. (2003). "Statistically Representative Three-Dimensional Microstructures Based on Orthogonal Observation Sections." Metallurgical and Materials Transactions: in press.
- Saylor, D. M., A. Morawiec, et al. (2002). "The Distribution and Energies of Grain Boundaries as a Function of Five Degrees of Freedom." Journal of The American Ceramic Society: submitted.
- Srolovitz, D. J., M. P. Anderson, et al. (1984). "Computer simulation of grain growth- III. influence of a particle dispersion." Acta metallurgica **32**: 1429-1438.
- Srolovitz, D. J., M. P. Anderson, et al. (1984). "Computer simulation of grain growth - II. grain size distribution, topology and local dynamics." Acta metallurgica **32**: 793.
- Srolovitz, D. J., G. S. Grest, et al. (1986). "Computer simulation of recrystallization- I. Homogeneous nucleation and growth." Acta metallurgica **34**: 1833-1845.
- Srolovitz, D. J., G. S. Grest, et al. (1988). "Computer simulation of recrystallization: II. Heterogeneous Nucleation and Growth." Acta metallurgica **36**(8): 2115-2128.
- Srolovitz, D. J. and G. N. Hassold (1987). Effects of diffusing impurities on domain growth in the Ising model. preprint LA-UR 86-3445.
- Upmanyu, M., D. Srolovitz, et al. (1998). "Atomistic simulation of curvature driven grain boundary migration." Interface Science **6**(1-2): 41-58.
- Yang, C.-C., A. D. Rollett, et al. (2001). "Measuring relative grain boundary energies and mobilities in an aluminum foil from triple junction geometry." Scripta Materiala **44**: 2735-2740.
- Zener, C. (1948). communication to C.S. Smith. Transactions of the Metallurgical Society of AIME. **175**: 15.