SIMULATIONS AND MODELING OF UNEQUAL SIZED PARTICLES SINTERING

by

Vineet Kumar

A dissertation submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Metallurgical Engineering

The University of Utah

May 2011

The University of Utah Graduate School

STATEMENT OF DISSERTATION APPROVAL

The dissertation of **Vineet Kumar**

has been approved by the following supervisory committee members:

Zhigang Z. Fang	, Chair	**10/25/10** Date Approved
Ravi Chandran	, Member	**10/25/10** Date Approved
Hong Y. Sohn	, Member	**10/25/10** Date Approved
Paul C. Fife	, Member	**10/25/10** Date Approved
Stuart I. Wright	, Member	**10/25/10** Date Approved

and by **Jan D. Millar** , Chair of

the Department of **Metallurgical Engineering**

and by Charles A. Wight, Dean of The Graduate School.

ABSTRACT

Numerical simulation methods, Monte Carlo simulation and phase field simulation methods were applied to the solid state sintering of unequal size particles. A geometrical model describing the solid state sintering was also developed. The numerical simulation methods and developed geometrical model were compared against results of the solid state sintering experiments.

Monte Carlo simulations were performed using Kawasaki and Glauber dynamics to accurately simulate the solid state sintering. The simulation results of two unequal particles showed that sintering occurs in three subprocesses: (1) neck growth, (2) coarsening and (3) grain boundary migration. A finite overlap between the three subprocesses was also observed in the simulation results.

The phase field model using conserved and nonconserved fields was applied to the sintering in solid state. The thermodynamics equations describing the energetics of the system were developed for performing the phase field simulations. An application of phased field simulations on two unequal size particle yielded results similar to those obtained by Monte Carlo simulations. The phase field simulation method was also applied to sintering of multiple particles. Realistic microstructures of multiparticle simulations were obtained.

A geometric model based upon two particles simulation results was developed. The geometrical model describes the overlapping three sintering subprocesses of neck growth, coarsening and grain boundary migration. Analytical expressions for the three subprocesses were developed. These expressions were used to calculate microstructural evolution of two unequal particles and a linear array of particles.

The numerical simulations and the developed geometrical model were compared with experimental data. The experimental data were obtained from sintering of nanosized tungsten powders. The geometric model successfully predicted the observed linear grain growth during sintering of tungsten.

Dedicated to,

Ram Chandar ji, Sita bhabhi,

Kanhaiya,

and *Shankar bhagwan.*

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I am heartily thankful to my supervisor, Prof. Z. Z. Fang, for encouragement, guidance and support during the course of my doctoral program. I am also very thankful to Prof. P. C. Fife, Prof. D. Shetty and Prof. H. Y. Sohn for helpful guidance on different materials in the project. I am also very thankful for Ashutosh Pandey for teaching me the art of computing.

CHAPTER 1

INTRODUCTION

This research described in this dissertation is related to sintering process involving bonding between material particles at elevated particles. This chapter provides the motivation, objective and scope of the research performed. This chapter concludes with a description of the organization of the dissertation.

1.1 Motivation

The sintering process is usually understood by geometrical models of sintering. These models are based upon idealized geometries of particles [1, 2]. These idealized geometries are used to describe the sintering process of equal sized particles. However, the consolidation of real powders involves sintering of unequal sized particles. The sintering behavior of real powders can be described using unequal sized particles sintering models.

The existing unequal sized particles sintering models do not provide a complete quantitative description of the sintering process. For example, Coble's model [3] is limited to the initial stage of sintering and does not consider mass transport between particles. On the other hand, Lange's thermodynamical model [4, 5] of unequal sized particles sintering describes the complete process of sintering. The model describes the sintering process to be occurring in three subprocesses: (1) neck growth, (2) coarsening, and (3) rapid grain boundary migration. Although Lange's model provides a complete description of three subprocesses, it does not provide kinetic equations of subprocesses. Furthermore, Lange's model neglects any overlap of sintering subprocesses, whereas a finite overlap may occur near transition of subprocesses.

The models described above suggest that the sintering of unequal sized particles is not well understood. Specifically, neck growth and coarsening kinetics for unequal sized particles are not known. Also, the expression of rapid grain boundary migration is not known. The reasons for not understanding the kinetics could be due to experimental difficulties and analyzing the unequal size particle sintering data. For example, the high temperature of the sintering process makes real time observation challenging. Furthermore, additional sintering variables such as particle size ratio and grain boundary mobility add to the complexity in deriving

expressions of unequal particle sized sintering kinetics. Therefore, the lack of understanding of unequal sized particles sintering is the main motivation behind this work.

1.2 Objective and scope

The primary objective of this study is to understand the mechanism of unequal sized particles sintering. In order to achieve the objective, a geometrical model quantitatively describing the three sintering subprocess will be developed. The geometrical model should also be capable of describing the overlap of the sintering subprocesses.

The objective is achieved using numerical simulations and geometrical modeling. The numerical simulation methods include Monte Carlo simulation and phase field simulation. The numerical simulations can provide a microstructural evolution and kinetics during sintering. This observation of microstructural evolution circumvents the problem of real time observation encountered during sintering experiments. The sintering results obtained from the numerical simulation methods and geometrical model are compared against experimentally obtained sintering data.

The numerical simulations and geometrical models can be used to obtain the sintering kinetics. Both of the methods have their advantages and drawbacks. The Monte Carlo simulations and phase field simulations do not make assumptions regarding the particle shape and size. These models provide accurate sintering kinetics with respect to geometrical assumptions. The Monte Carlo and Phase field simulation methods are applied to the sintering of two particles. The models show the sintering to be occurring in three sequential subprocesses: (1) neck growth, (2) coarsening, and (3) grain boundary migration. The sintering of two particles shows an overlap of the three subprocesses. In the first subprocess of neck growth, a neck and a grain boundary between the particles form and grow. This neck and grain boundary provides the necessary path for interparticle mass transport during the coarsening subprocess. Next, during the grain boundary migration subprocess, the grain boundary sweeps across the small particle until the small particle disappears. Although Monte Carlo and the phase field simulation methods provide sintering kinetics without making any geometrical assumptions, the kinetics are provided on a relative time and length scales. Because real time and length scales would rather be obtained, geometrical modeling is applied in addition to the simulations.

The geometrical models make assumptions regarding particle geometry and mass transport. Although these assumptions regarding geometry and mass transport add to the inaccuracy of the sintering model, geometrical models are very useful. Their usefulness stems

from modeling the sintering in real time and length scale. At the same time, these models provide the dependence of sintering variables on kinetics in closed form expressions. These closed form expressions enhance the usefulness of the geometrical models. The geometrical model developed in this work provides results similar to the Monte Carlo and the phase field simulation method. The geometrical model shows that subprocesses of the neck growth and the grain boundary migration are kinetically fast as compared to coarsening. This indicates that the coarsening is the rate determining subprocesses in the sintering.

The numerical models described above provide the sintering kinetics of two unequal particles. The models are compared against the experiments. Sintering experiments on tungsten powders are used for comparison. A satisfactory comparison between modeling and experiments is found.

1.3 Organization of dissertation

This dissertation is divided in two parts. The first part of the dissertation describes the background of sintering theories and numerical simulations whereas the second part contains the results and discussion of research performed in this dissertation. In the first part of the dissertation, Chapter 2 describes the sintering theory. This chapter also summarizes the analytical models of unequal sized particle sintering. Background, methodology and recent applications of Monte Carlo and phase field simulation methods are described in Chapter 3 and 4, respectively. The second part of the dissertation includes results and discussion of numerical simulations. The second part also contains details and application of geometrical model developed in this dissertation. The application of Monte Carlo simulation to sintering of two unequal particle sintering is discussed in Chapter 5. Phase field simulation method is applied to various initial microstructure. Chapter 6 describes the application and discusses the results of the phase field simulations. The geometrical model is described in Chapter 7. Application of the geometrical model to sintering of linear array of particles and a comparison with sintering of tungsten nanopowder is also described in Chapter 7. Finally, summary, conclusions and comments on future work are provided in Chapter 8.

PART I

BACKGROUND AND LITERATURE REVIEW

CHAPTER 2

SINTERING THEORY AND MODELS

Sintering can be defined as a material processing method to produce density controlled materials and components from powers by application of thermal energy. Sintering involves the elimination of the pores between the particles. The elimination of the pores requires the movement of the vacancies from the interior of the sample to the exterior of the sample. The movement of the vacancies is equivalent to the movement of the atoms in the opposite direction. The movements of the vacancies and atoms can occur along numerous paths in the sintering microstructure. These paths include surfaces, grain boundary and bulk materials. The mass transport due to the vacancies and the atoms movement along these paths defines the sintering mechanism. The sintering mechanisms are described in detail in Section 2.1. In practice, several sintering mechanisms are simultaneously active resulting in mass transport via various routes. The simultaneously active sintering mechanisms indicate the complex nature of the sintering process. The complexity of the sintering is aggravated by a continuous change in particle size and shape.

A comprehensive sintering theory describing the complete sintering process has not been developed. Rahman suggested that the development of such comprehensive theory is unlikely [6]. This suggestion is based upon the complex nature of the sintering process. The conventional approach to describe the sintering process is to separate the sintering process into subprocess or stages. The separation of the stages is based upon the mass transport mechanisms and the sintering geometry. The sintering kinetics for simple geometry can be developed. The conventional sintering models provide kinetics for equal size spherical particles. The influence of sintering parameters such as particle size and temperature can be inferred from these models. A description of these models is given in Section 2.2

The densification of the real powders involves sintering of unequal particles. The sintering models for unequal particles have drawn attention recently. Researchers have adopted several approaches to establish geometrical changes and sintering kinetics during the sintering of unequal sized particles. These approaches include numerical simulations [7,8] and geometrical modeling [5, 9]. Numerical simulations provide microstructural evolution during sintering

using computational methods. The sintering kinetics in numerical simulations are obtained from microstructural evolution. Therefore, numerical simulations are similar to sintering experiments. On the other hand, The geometrical models are based upon idealized geometries of particles during sintering. The geometrical models are generally used to describe the dependence of sintering parameters and materials properties on sintering kinetics in terms of analytical expressions. Sec. 2.3 summarizes the geometrical models of two unequal sized particles sintering reported in the literature.

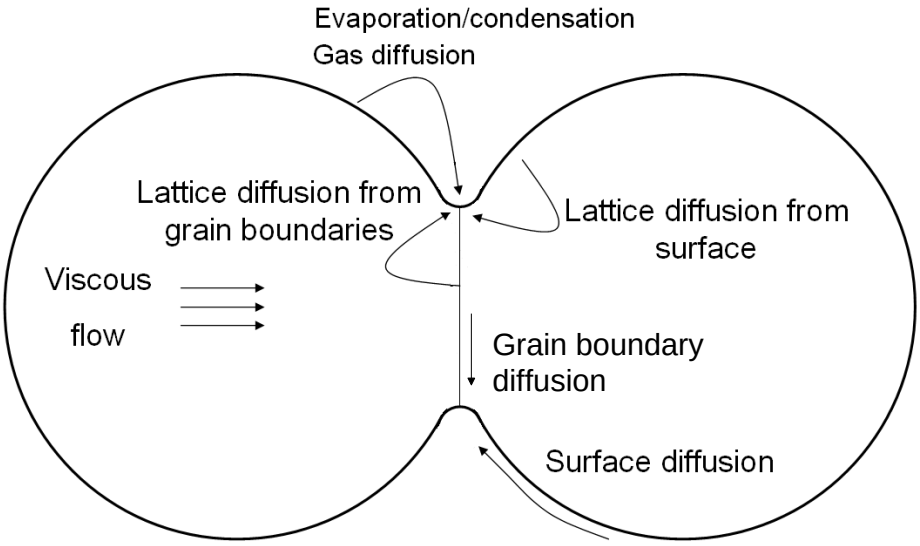
2.1 Sintering mechanisms

The densification during the sintering occurs due to the mass transport. The mass transport may take place by various mass transport mechanisms along different paths. A sintering mechanism refers to a combination of mass transport mechanism and mass transport paths. The sintering mechanisms depend upon the source and sink for the mass transport. A chemical potential exists between the source and sink region. The mass transport reduces the chemical potential between the source and sink regions. The sintering mechanisms also depend upon the mass transport paths such as surfaces and grain boundaries. The six sintering mechanisms are possible based upon the sink and source of mass transport. These mechanisms are listed in Table 2.1 and graphically shown in Fig. 2.1

The driving force for the above mentioned sintering mechanisms originates from the chemical potential difference among different points in the microstructure. The regions of high chemical potential act as the source of the material for transport. The sink regions in the microstructure have lower chemical potential. The source and the sink regions are given in Table 2.1. The chemical potential difference primarily results from difference in surface curvature. The difference in the surface curvature may cause difference in bulk pressure, vapor pressure and vacancy concentration among regions in the microstructure. These differences

2.1: Sintering mechanisms

S. no.	Sintering mechanism	Material source	Material sink
1	Lattice diffusion from surface	Particle surface	Neck
2	Lattice diffusion from grain boundaries	Grain boundary	Neck
3	grain boundary diffusion	Grain boundary	Neck
4	surface diffusion	Particle surface	Neck
5	viscous flow	Bulk grain	Neck
6	gas phase transport via evaporation/condensation or gas diffusion	Particle surface	Neck



2.1: Various sintering mechanisms in sintering

in bulk pressure, vapor pressure and vacancy concentration can be identified with two particle geometry as follows.

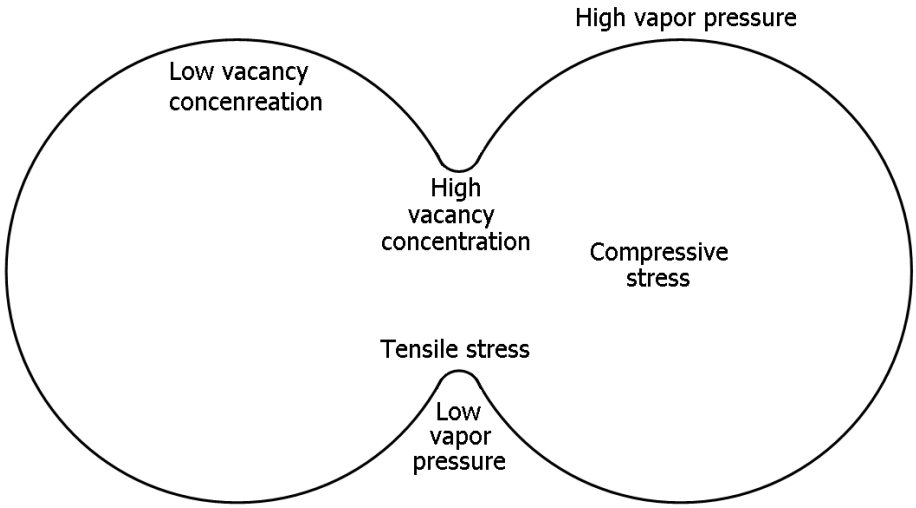
1. bulk pressure difference or stress between particle interior and neck region
2. vapor pressure difference over the neck and the particle surfaces
3. vacancy concentration difference between particle interior, surface and neck region

Fig. 2.2 shows the bulk pressure, vapor pressure and vacancy concentration at different locations in the microstructure. The bulk pressure, vapor pressure and vacancy concentration can be expressed as a function of surface curvature.

The stress σ under a curved surface can be given by Young-Laplace equation [6]:

$$\sigma = \gamma_s \left(\frac{1}{R_1} + \frac{1}{R_2} \right) \tag{2.1}$$

where γ_s is the specific surface energy, and R_1 and R_2 are the principle radii of curvature. The radius of curvature at the particle surface has a positive curvature resulting in a compressive



2.2: Vacancy concentration, vapor pressure and bulk pressure at various locations in the microstructure

stress under the particle surface. The neck has a concave curvature with a negative radius of curvature. The negative radius of curvature results in a tensile stress at the neck.

The vapor pressure p over a curved surface can be given by the Young-Laplace equation [10]:

$$\log_e\left(\frac{p}{p_0}\right) = \frac{\gamma_s \Omega}{kT}\left(\frac{1}{R_1} + \frac{1}{R_2}\right) \tag{2.2}$$

where p_0 is the equilibrium vapor pressure over a flat surface, Ω is the atomic volume, k is the Boltzmann constant and T is the absolute temperature. The radii of curvature at the particle surface and the neck lead to a higher and lower vapor pressure than a flat surface respectively.

The vacancy concentration c_v can beneath a curved surface can be given as [11]:

$$c_v = c_0 \exp\left[-\frac{\gamma_s \Omega}{kT}\left(\frac{1}{R_1} + \frac{1}{R_2}\right)\right] \tag{2.3}$$

where c_0 is equilibrium vacancy concentration beneath a flat surface. The concave radius of curvature at neck results in an excess vacancy concentration. The vacancy concentration is less than equilibrium vacancy concentration beneath the particle surface.

The mass transport for the sintering occurs due to the difference in vacancy concentration, vapor pressure and bulk pressure. The mass transport due to these differences takes place along paths given in Table 2.1. The sintering by surface diffusion and gas phase brings shape change to the particles. These sintering mechanisms do not bring particle centers closer. Therefore, these two mechanisms are considered as nondensifying mechanisms. The other sintering mechanisms result in a shrinkage and are referred to as densifying mechanisms. The densifying and nondensifying mechanisms play roles in the sintering.

2.2 Geometrical models

The geometrical models of sintering provide sintering kinetics. The relationship between sintering time and sintering parameters such as particle size, temperature and diffusion coefficient can be obtained with the geometrical model. The geometrical models assume idealized geometries to describe the particle shapes in the sintering powder. The geometry of the particles and pores changes during sintering and cannot be idealized by one particle geometry. The sintering process is conceptually divided into three stages to avoid the problems of idealized geometries [6]. These three stages are: (1) initial stage sintering, (2) intermediate stage sintering and (3) final stage sintering.

The model for initial stage of sintering was proposed by Kingery and Berg [12]. They used an idealized microstructure of two touching particles to describe neck formation during the initial stage of sintering. The neck between the particles forms as soon as the sintering begins. The neck growth continues until it about 40-50% of the partice size [6]. Kang assumed the maximum neck size to be about 20% of the particle size [13]. The densifying and nondensifying mechanisms contribute in neck growth. The initial stage of sintering results in linear shrinkage of about 3-5%. The microstructure at the end of the initial stage consists of a skeleton formed by the particles. The pores remain open at the end of the initial stage of sintering.

The intermediate stage of sintering begins when pores acquire their equilibrium shapes [14]. The pores form a continuous structure in the microstructure. The particle shapes are idealized as space filling tetrakaidecahedron. The pores are assumed to have cylindrical shape and located along the particle edges. The lattice diffusion and grain boundary diffusion result in a reduction of pore radius. The pore radius is reduced until the pores become unstable and pinch off. The intermediate stage of sintering usually constitutes a major part of overall sintering leading to a relative density becomes about 0.09. The microstructure at the end of the intermediate stage sintering consists of grains having pores at the corners.

The final stage sintering begins when pores are isolated [14]. The spherical pores are assumed to located on tetrakaidecahedron grains. The pores shrink due to densifying mechanisms until the sintering microstructure becomes fully dense. The relative density increases from ~ 0.9 to 1.0 during final stage of sintering.

The sintering kinetics can be derived for the idealized geometries in each stage of the sintering. The kinetics are obtained by setting up a mass flux equation for the idealized geometry. The mass flux is solved using appropriate boundary conditions.

2.2.1 Initial stage sintering

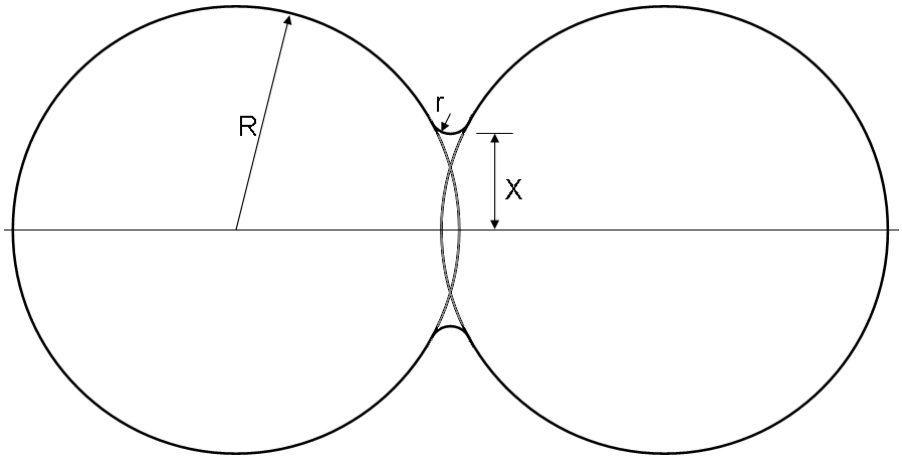
The initial stage of sintering involves a formation and growth of neck between the particles [12]. The neck growth driven by a reduction in surface energy at the expense of the grain boundary energy. The neck growth can be accomplished by densifying and nondensifying mechanisms. The densifying and nondensifying mechanisms require different geometries to model the sintering. These geometries consist of two spherical particles. The geometries for the densifying and nondensifying mechanisms, referred to as sintering geometry "with shrinkage" and "without shrinkage", respectively, are shown in Fig. 2.3.

In case of sintering with shrinkage, the two spherical particles intersect. The two particles touch each other for sintering without shrinkage. The neck between the particles has a circular profile. The radius of curvature r at neck, area A and volume V of the neck can be given as:

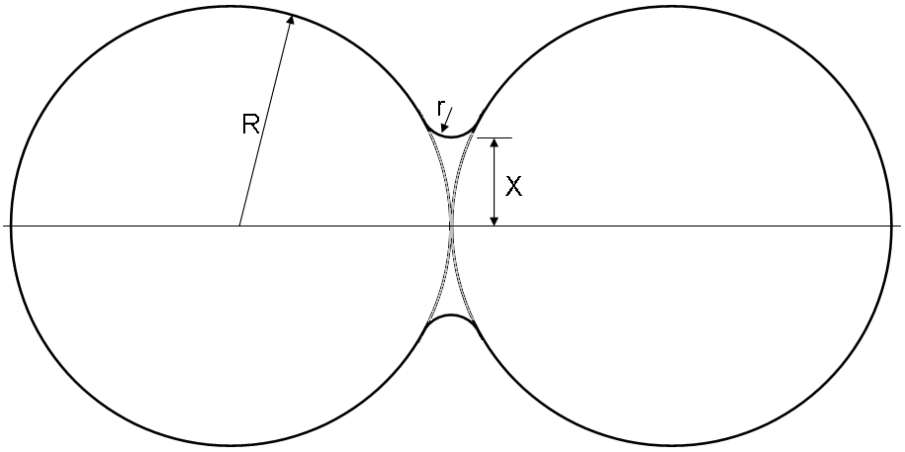
$$\{r, A, V\} \approx \begin{cases} \left\{\dfrac{x^2}{2R}, \dfrac{2\pi x^3}{R}, \dfrac{\pi x^4}{2R}\right\} & \text{without shrinkage} \\ \left\{\dfrac{x^2}{4R}, \dfrac{\pi x^3}{R}, \dfrac{\pi x^4}{4R}\right\} & \text{with shrinkage} \end{cases} \tag{2.4}$$

The kinetics of the initial stage of sintering can be obtained by equating the mass transported from the materials source and change in the neck volume. For the neck growth with lattice diffusion from grain boundaries

$$\begin{aligned} \frac{\mathrm{d}V}{\mathrm{d}t} &= JAV_m \\ \frac{\pi x^3}{R}\frac{\mathrm{d}x}{\mathrm{d}t} &= \frac{D_l}{kT}\nabla\sigma AV_m \\ &= \frac{D_l}{kT}\frac{\gamma_s}{rx}\frac{\pi x^3}{R}V_m \\ \Rightarrow x^4 &= \frac{16 D_l \gamma_s R V_m}{kT}t \end{aligned} \tag{2.5}$$



(a) with shrinkage



(b) without shrinkage

2.3: Initial stage sintering geometries

where J is the atom flux, V_m is the molar volume, k is the Boltzmann constant and T is the absolute temperature. In eq. 2.5, D_l is the volume diffusion coefficient and t is the time. The time required for neck growth by other sintering mechanisms can be calculated in a similar fashion. The kinetics for the initial stage sintering can be expressed as:

$$\left(\frac{x}{R}\right)^n = \frac{B(T)}{R^m}t \tag{2.6}$$

In eq. 2.6, $B(T)$ is a temperature dependent term. The term $B(T)$ also includes materials properties such as diffusion coefficients and geometrical constants. In eq. 2.6, the exponent n is termed as mechanism characteristic exponent whereas the exponent m defines the particle size dependence. The value of the term $B(T)$ and coefficients m, n depends upon the sintering mechanisms. The initial stage sintering kinetics for different mechanisms are given in Table 2.2.

The exponents n, m in eq. 2.6 can obtained by plotting relative neck size x/R with sintering time t and particle size R on logarithmic scale. The sintering mechanism can be obtained from the exponents given in Table 2.2. The determination of mechanism from the exponent is based on the assumption that only one sintering mechanism is active. When several sintering mechanisms are simultaneous active, the exponent may correspond to an entirely different mechanism. For example, Kingrey and Breg [12] showed that the exponents n, m indicate the lattice diffusion was the dominant neck growth mechanism. Later analysis showed the surface diffusion was the dominant mechanism with a significant contribution from lattice diffusion [6].

2.2.2 Intermediate stage sintering

The microstructure at the end of the initial stage contains an interconnected open pore structure [14]. The pores are located on the grain edges. The intermediate stage of the sintering involves the removal of the open pores. The removal of pores is obtained by densifying sintering mechanisms, namely, grain boundary diffusion and lattice diffusion.

The geometry to describe the intermediate stage sintering assumes cylindrical pores on the edges of tetrakaidecahedron grains. The driving force for this stage of sintering is the reduction in pore surface area. The pores shrink as sintering progresses until the Rayleigh's surface instability criterion is met [15]. When Rayleigh's criterion is met, the pores become isolated. The change in the pore structure indicate the end of the intermediate stage of sintering.

The intermediate stage of sintering is modeled with tetrakaidecahedron grains. The assumed microstructure consists of grains arranged in bcc arrangement with pores along the grain edges. A tetrakaidecahedron has 36 edges, 24 corners and 14 faces as shown in Fig. 2.4. The volume of a tetrakaidecahedron can be given as:

$$V_t = 8\sqrt{2}{l_p}^3 \tag{2.7}$$

2.2: Initial stage sintering kinetics. Adapted from ref. [6]

Sintering mechanism	Kinetics expression	n	m
Lattice diffusion from particle surfaces	$x^5 = \frac{20 D_l \gamma_s V_m R^2}{kT} t$	5	3
Lattice diffusion from grain boundary	$x^4 = \frac{16 D_l \gamma_s V_m R}{kT} t$	4	3
Grain boundary diffusion from grain boundary	$x^6 = \frac{48 D_{gb} \delta_{gb} \gamma_s V_m R^2}{kT} t$	6	4
Surface diffusion from particle surfaces	$x^7 = \frac{56 D_s \delta_s \gamma_s V_m R^3}{kT} t$	7	4
Viscous flow	$x^2 = \frac{4 \gamma_s R}{\eta} t$	2	1
Gas phase transport via evaporation/condensation	$x^3 = \sqrt{\frac{18}{\pi}} \frac{p_0 \gamma_s}{d^2} \left(\frac{M}{kT}\right)^{(3/2)} Rt$	3	2
Gas phase transport via gas diffusion	$x^5 = 20 p_0 D_g \gamma_s \left(\frac{V_m}{kT}\right)^2 R^2 t$	5	3

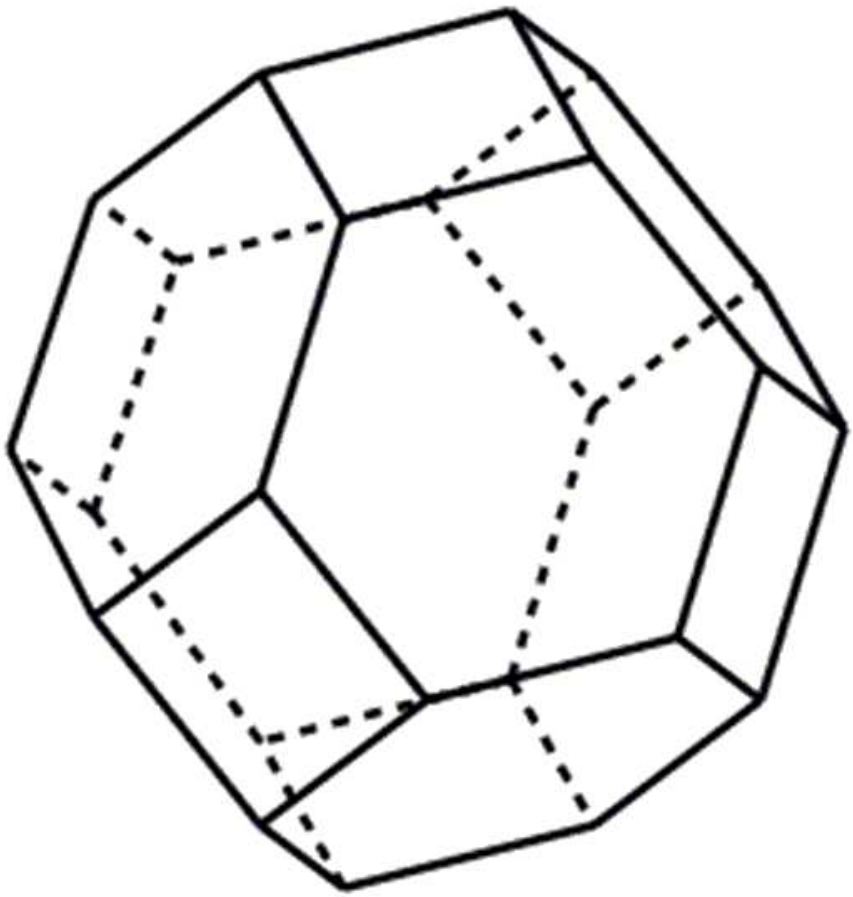
x = neck size; R = particle radius; t = time; D_l = lattice diffusion coefficient; D_{gb} = grain boundary diffusion coefficient; D_s = surface diffusion coefficient; γ_s = specific surface energy; δ_s = diffusion thickness of the surface diffusion; δ_{gb} = grain boundary width; V_m = molar volume; k = Boltzmann constant; T = absolute sintering temperature; d = material density; D_g = diffusivity of gas atoms; p_0 = equilibrium vapor pressure over a flat surface; η = viscosity; M = molecular weight.

where l_p is the edge length of a tetrakaidecahedron. A edges in the assumed closed packing is shared by three tetrakaidecahedron. Therefore, the volume of the porosity can be given as:

$$V_p = \frac{1}{3} 36\pi r^2 l_p \tag{2.8}$$

where r is the pore radius.

The kinetics of the intermediate stage sintering is obtained using a vacancy flux from the pores to the grain boundary. Diffusion of vacancies from circular source to the center of grain boundary faces is assumed. The edge effects of the pores are neglected for deriving sintering kinetics. The diffusion flux equation for sintering kinetics is set up by analogy of sintering geometry with steady state heat dissipation electrically heated cylindrical conductor with



2.4: Tetrakaidecahedron

concentric surface cooling.

The flux per unit length of the cylinder J/l can be given as:

$$J/l = 4\pi D_v \Delta c \tag{2.9}$$

where D_v is the vacancy diffusion coefficient and Δc is the difference in vacancy concentration between the pore (source) and the grain boundary (sink). If the length l in eq. 2.9 is taken as $2r$, the number of vacancies arriving at the grain boundary per unit time $\dot{N}_v$ can be given as:

$$\dot{N}_v = 4\pi D_v \Delta c.2r \tag{2.10}$$

The vacancy flux of eq. 2.10 is distributed over two grains separated by the grain boundary. Since there are 14 faces in a tetrakaidecahedron, the volume flux per unit time per polyhedron can be given as:

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \frac{14}{2}\dot{N}_v\Omega = 56\pi D_v \Delta c r \Omega \tag{2.11}$$

The principal radii of curvature of cylindrical pore are r and infinity. The vacancy concentration difference can be approximated from radii of curvature using eq. 2.3 as:

$$\Delta c = \frac{c_0 \gamma_s \Omega}{kTr} \tag{2.12}$$

with substitution of 2.12 in eq. 2.11 and using $D_l = D_v c_0 \Omega$, a change in volume of the vacancies can be given as:

$$\mathrm{d}V = \frac{56\pi D_l \gamma_s \Omega}{kT}\mathrm{d}t \tag{2.13}$$

The volume of the vacancies V can be correlated to change in volume the pores V_p as:

$$\mathrm{d}V = -\mathrm{d}V_p = -\frac{36}{3}2\pi lr\mathrm{d}r = \frac{56\pi D_l\gamma_s\Omega}{kT}\mathrm{d}t \tag{2.14}$$

Eq. 2.13 can be integrated from initial pore radius r_0 to pore radius r as:

$$r^2 - {r_0}^2 = -\frac{14}{3}\frac{D_l\gamma_s\Omega}{kTl}t \tag{2.15}$$

Densification rate derived from eq. 2.15 can be given as:

$$\frac{1}{\rho}\frac{\mathrm{d}\rho}{\mathrm{d}t} \approx \frac{14D_l\gamma_s\Omega}{3\rho G^3kT} \tag{2.16}$$

where G is the grain size.

The grain boundary diffusion is also an active sintering mechanism that leads to densification in intermediate stage of sintering. The expression of sintering kinetics can be derived using a similar procedure described as above. The densification rate for grain boundary diffusion can be given as:

$$\frac{1}{\rho}\frac{\mathrm{d}\rho}{\mathrm{d}t} \approx \frac{4}{3}\frac{D_{gb}\delta_{gb}\gamma_s\Omega}{\rho(1-\rho)G^4kT} \tag{2.17}$$

2.2.3 Final stage sintering

The final stage of sintering is idealized with tetrakaidecahedron grains having isolated pores on grain corners [14]. The pores located on the grain corners are assumed to have spherical shapes. A tetrakaidecahedron has 24 corners which are shared by 4 grains in a bcc arrangement. The volume of the pores per grain can be given as:

$$V_p = \frac{24}{4}\frac{4\pi}{3}r^3 = 8\pi r^3 \tag{2.18}$$

The diffusion flux for the spherical geometry has not been solved. The sintering kinetics are obtained by using equations of flux between two concentric spherical shells. The flux equation for concentric spherical shells can be given as:

$$\dot{N}_v = 4\pi D_v \Delta c \frac{r_1 r_2}{r_2 - r_1} \tag{2.19}$$

In eq. 2.19, r_1 and r_2 are inner and outer radii of spherical shells. If $r_1 << r_2$, eq. 2.19 can be simplified as:

$$\dot{N}_v = 4\pi D_v \Delta c r_1 \tag{2.20}$$

The volume flow per tetrakaidecahedron can be given as:

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \frac{24}{4}\dot{N}_v \Omega = 24\pi D_v \Delta c \Omega r \tag{2.21}$$

where r = r_1 is the pore radius. The difference in vacancy concentration gradient can be approximated from eq. 2.3 as:

$$\Delta c = \frac{c_0 \gamma_s \Omega}{kT} \frac{2}{r} \tag{2.22}$$

The volume flow can be obtained by substituting eq. 2.22 and $D_v = D_l c_0 \Omega$ in eq. 2.21 as:

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \frac{48\pi d_l \gamma_s \Omega}{kT} \tag{2.23}$$

The volume flow of the vacancies can be correlated to the pore size as:

$$dV = -dV_p = -d(8\pi r^3) = -24\pi r^2 dr \tag{2.24}$$

The change in the pore size as a function of time can be obtained by substituting eq. 2.24 in eq. 2.23 and integrating as given below:

$$r^3 - {r_0}^3 = -\frac{6D_l\gamma_s\Omega}{kT}t \tag{2.25}$$

2.3 Sintering of unequal size particles

The consolidation of real powders involves the sintering of unequal sized particles. The sintering of unequal particles involves additional sintering mechanisms. These mechanisms include coarsening and grain boundary migration. We define coarsening as change in particle size due to long range diffusion. The change in the particle size is caused by chemical potential difference among particles of different sizes. The mass transport due to this chemical potential occurs by diffusion or gas transport. Similar to coarsening, a migration of grain boundary between particles also results in a particle size change. However, this particle size change occurs due to short range movement of atoms. Atoms located on different sides of the grain boundary may have different chemical potential difference. Atoms of high chemical potential may jump across the grain boundary to the region of lower chemical potential. These jumps involve movement of atoms of the order of lattice parameter. The jump of the atoms results in the migration of the grain boundary. The mass transport due to coarsening and grain boundary migration is during sintering of equal particles due to the assumed symmetry of sintering geometry. The sintering kinetics of unequal sized particles involves the effect of particle size distribution and grain boundary migration on sintering kinetics. These two effects make the sintering of unequal sized particles sintering complex. The sintering of unequal sized particles is not as comprehensively studied as sintering of equal particles described in Sec. 2.2. Coble proposed a theory of sintering of unequal size particles [3]. Coble's model can be used to predict the neck growth kinetics and shrinkage for unequal sized particles. Coble's model does not consider the interparticle mass transport and grain boundary diffusion. Lange and Kellett proposed a thermodynamic model to explain the sintering behavior of unequal particles [5]. Lange's model described the neck growth, interparticle mass transport or

particle coarsening and grain boundary migration. The sintering kinetics are not provided in Lange's model. Another model for sintering of unequal particles is proposed by Colbeck [9]. Colbeck's model focuses on bonding of particles. Colbeck's model was developed to explain the bonding behavior of snow particles. These models provide insight into the sintering of unequal particles. These models are summarized in this section.

2.3.1 Coble's model

Coble proposed a model to describe the sintering of unequal particles [3]. The model was presented to address issues related to unequal particles which could not be explained by equal particles sintering models. Coble pointed out that sintering of unequal sized particles array in two dimensions (2D) may require a more complex analysis. The examples of unequal sized particles array could be a wire wound mandrel and a sintering geometry where small particles just fit into interstices of the large particles.

The sintering of the wire wound mandrel would require the same shrinkage rate along the axis of the mandrel. However, a differential rate of shrinkage would exist as large particles exhibit smaller shrinkage. This difference in the shrinkage of mandrel and wires would result in stress along the neck. The existing models of sintering do not address the issue of stress between the particles.

Similarly, the sintering of small particles just fitting into interstices of the large particles cannot be explained by existing models. The sintering model for particles on plate shows that the sintering between a large and a small particle occurs more rapidly than sintering of the large particles alone [16]. This differential rate of sintering will result in a tensile stress in the small particles and a compressive stress in large particles. The sintering of touching small particles may require a debonding from some contacts or plastic deformation to compensate for the sintering rate between the particles.

Coble suggested that the sintering of a row of unequal sized particles can be analyzed for an accumulative shrinkage. The stress effects due to differences in the particle size can be neglected during the sintering of linear array of particles.

While developing a model for unequal sized particles sintering, Coble assumed a negligible change in morphology by surface diffusion, evaporation-condensation and liquid phase sintering. The developed model included only lattice diffusion and grain boundary mechanism as mass transport mechanisms. The model can be applied to cases when densifying sintering mechanisms are dominant. Coble further suggested that the model was developed with assumed initial particle sizes and would not be applicable when particle growth occurs during

initial stage sintering.

The two particle sintering geometry to derive the sintering kinetics is shown in Fig. 2.5. The particles of radii R_1 and R_2 intersect each other. The radii of curvature at the neck are ρ_1 and ρ_2. The geometrical relations for the geometry are given below.

$$y_1 = \frac{x_1^2}{2R_1} \tag{2.26}$$

$$y_2 = \frac{x_1^2}{2R_2}$$

$$x_2 = \sqrt{2}x_1 \tag{2.27}$$

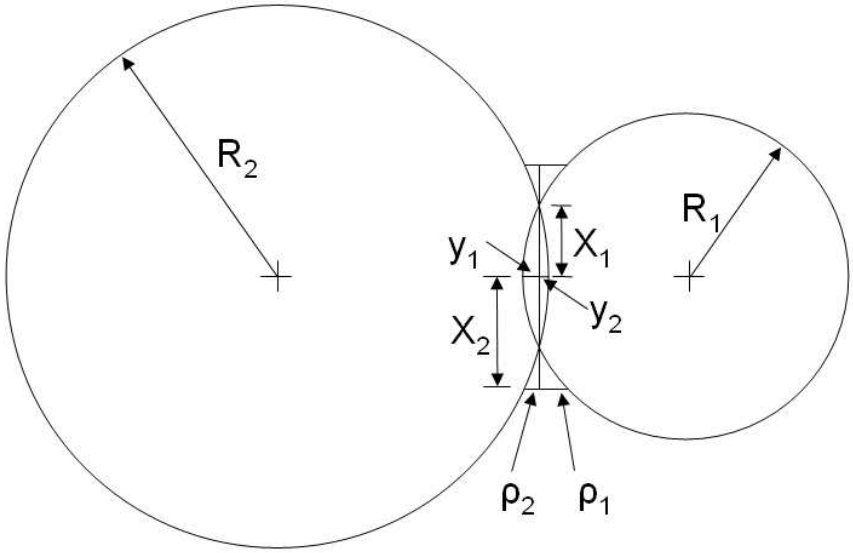
$$\rho_1 = y_1 = \frac{x_2^2}{4R_1} \tag{2.28}$$

$$\rho_2 = y_2 = \frac{x_2^2}{4R_2}$$

$$V_1 = \frac{\pi x_1^4}{4R_1} \tag{2.29}$$

$$V_2 = \frac{\pi x_1^4}{4R_2}$$

where V_1 and V_2 volume of spherical segments in sphere with radii R_1 and R_2, respectively. The radius of curvature K at the neck is defined as:



2.5: Schematic cross section of pair of spherical particles; geometric notation used for the model is shown.

$$K = \frac{2}{\rho_1 + \rho_2} - \frac{1}{x_2} \tag{2.30}$$

The definition of radius of curvature at the neck given in eq. 2.30 is different from that used for equal size particles [1] to avoid a discontinuity at grain boundary. The last term with neck size x_2 is usually neglected as $x_2 >> \rho_1, \rho_2$. The kinetics using the above geometrical parameters can be obtained for sintering by lattice diffusion and grain boundary diffusion.

2.3.1.1 Lattice diffusion

The diffusion flux equation used for equal particles (eq. 2.9) is also used for sintering of unequal particles as given below:

$$J/l = 4\pi D_v \Delta c \tag{2.31}$$

The volume of the flowing vacancies can be given as:

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \frac{J}{l}\Omega(\rho_1 + \rho_2) = 4\pi D_v \Delta c \Omega(\rho_1 + \rho_2) \tag{2.32}$$

The volume of the flowing vacancies should be equal to the volume change in the neck. The volume change can be expressed as:

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \pi x_2^2 \frac{\mathrm{d}(\mathrm{y}_1 + \mathrm{y}_2)}{\mathrm{d}t} \tag{2.33}$$

The value of vacancy concentration difference Δc can be approximated using Gibbs - Thompson relation in eq. 2.3 as:

$$\Delta c = c_0 \frac{2}{\rho_1 + \rho_2} \frac{\gamma_s \Omega}{kT} \tag{2.34}$$

The sintering kinetics can be obtained by solving eq. 2.32 - 2.34 and using $D_v = D_v c_0 \Omega$.

$$\begin{aligned} \pi x_2^2 \frac{\mathrm{d}(y_1 + y_2)}{\mathrm{d}t} &= \frac{16\pi D_l \gamma_s \Omega}{kT} \\ \Rightarrow \frac{\pi}{2}\left(\frac{1}{R_1} + \frac{1}{R_2}\right) x_2^3 \frac{\mathrm{d}x_2}{\mathrm{d}t} &= \frac{16 D_l \gamma_s \Omega}{kT} \end{aligned} \tag{2.35}$$

Integration of eq. 2.35 gives the sintering kinetics as:

$$x_2^4 \left(\frac{1}{R_1} + \frac{1}{R_2}\right) = \frac{128 D_l \gamma_s \Omega}{kT} t \tag{2.36}$$

2.3.1.2 Grain boundary diffusion

For the grain boundary diffusion, the equation used in lattice diffusion (eq. 2.31) can be used to model the sintering. However, the diffusion coefficient should be replaced by grain boundary diffusion coefficient D_{gb} and the length l should be taken as grain boundary width δ_{gb}. The diffusion equation for the grain boundary diffusion can be written as:

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \frac{J}{l} \Omega \delta_{gb} = 4\pi D_{gb} \Delta c \Omega \delta_{gb} \tag{2.37}$$

The values of concentration difference Δc and volume difference $\mathrm{d}V$ in eq. 2.37 can be substituted from Eq. 2.33 and 2.34.

$$(\rho_1 + \rho_2)\pi x_2^2 \frac{\mathrm{d}(y_1 + y_2)}{\mathrm{d}t} = \frac{16\pi D_b \delta_{gb} \gamma_s \Omega}{kT} \tag{2.38}$$

With geometrical relations in eq. 2.26 - 2.29, eq. 2.38 gives:

$$\left(\frac{1}{R_1} + \frac{1}{R_2}\right)^2 \frac{\pi x_2^5}{8} \frac{\mathrm{d}x_2}{\mathrm{d}t} = \frac{16\pi D_b \delta_{gb} \gamma_s \Omega}{kT} \tag{2.39}$$

Eq. 2.39 can be integrated to obtain the sintering kinetics:

$$x_2^6 \left(\frac{1}{R_1} + \frac{1}{R_2} \right)^2 = \frac{768 D_b \delta_{gb} \gamma_s \Omega}{kT} t \tag{2.40}$$

2.3.1.3 Applications of Coble's model

Coble applied his unequal sized particle model to a linear array of particles and two dimensional array of particles described in Section 2.3.1. Coble applied the model to bimodal, trimodal and pentamodal particles size distributions and obtained effective particle size that can be used to describe the linear shrinkage. The effective particle size was found to be intermediate of minimum and maximum particle size. Coble also analyzed the effect of stress produced due to particles size differences in two dimensional arrays. The stress is found to increase the rate of sintering. Coble showed that the driving force increases by $\approx 25\%$ due to inclusion of interstitially contacting small particles.

2.3.2 Lange and Kellett's model

Lange and Kellett developed a thermodynamical model for sintering of unequal particles [5]. Although the model does not provide sintering kinetics, the model provides insight into the effect of grain boundary migration in sintering. The model is based upon three observations from various experiments [17–19]:

1. Grain growth occurs after necks are formed.

2. In partially sintered body, grain growth in appears to be dependent only upon density.

3. Grain growth and densification are closely related.

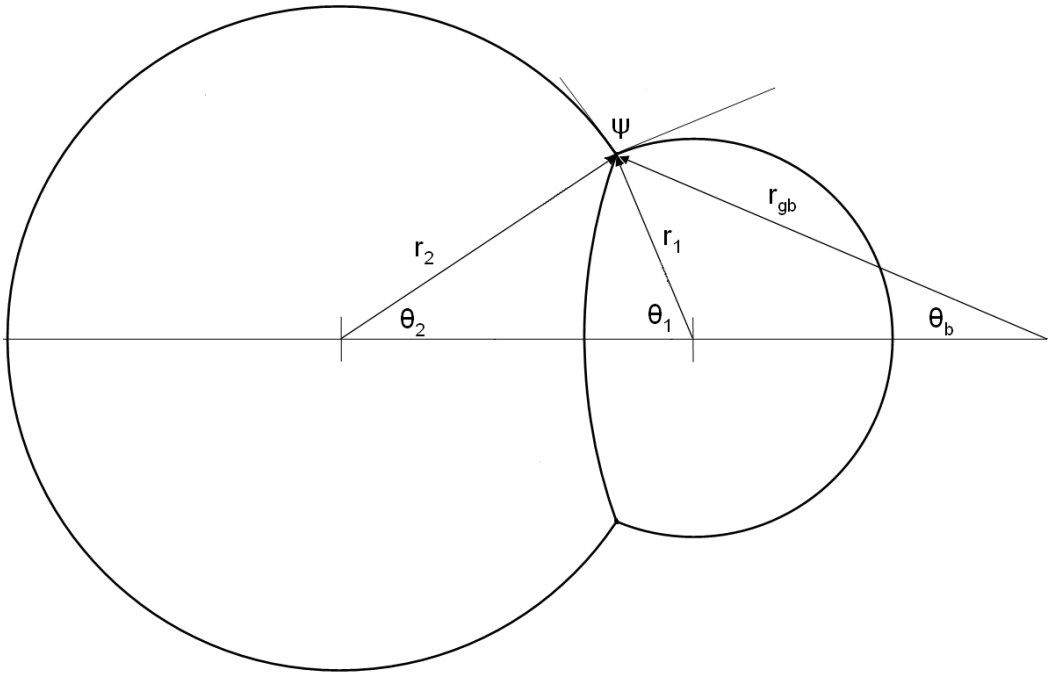
The model utilized some results from Lange and Kellett's equal particle model [4]. These results are: (1) Grain boundary forms until the particle array achieved the minimum energy configuration, (2) the pores with coordination number n exceeding a critical value n_c do not shrink but acquire their equilibrium configurations. Lange and Kellett assumed that the energy calculation that provided these results for equal particles holds for unequal particles as well. The terms coarsening and sintering in their model are defined differently from their usual definitions. The sintering is defined as a phenomenon where mass is transported to form a neck at the contact point of particles. The sintering refers to a phenomenon where interparticle mass transport causes larger particle to grow at the expense of smaller particle.

The geometrical model is based on an interpenetrating particles geometry as shown in Fig. 2.6. It is assumed that the radius of curvature is independent of the position on the surface, i.e., the particles maintain a spherical shape everywhere. As particles penetrate each other during sintering, the mass of penetration is redistributed over particles such that the each particle maintains its mass. This redistribution of mass results in an increased particle radius. The mass of particles change during the coarsening under the condition that the total mass is conserved. This geometry neglects the presence of a neck with a negative radius of curvature. The grain boundary is assumed to be bisecting the contact angle ψ. The sintering of two particles and an array of particles can be described using this geometry.

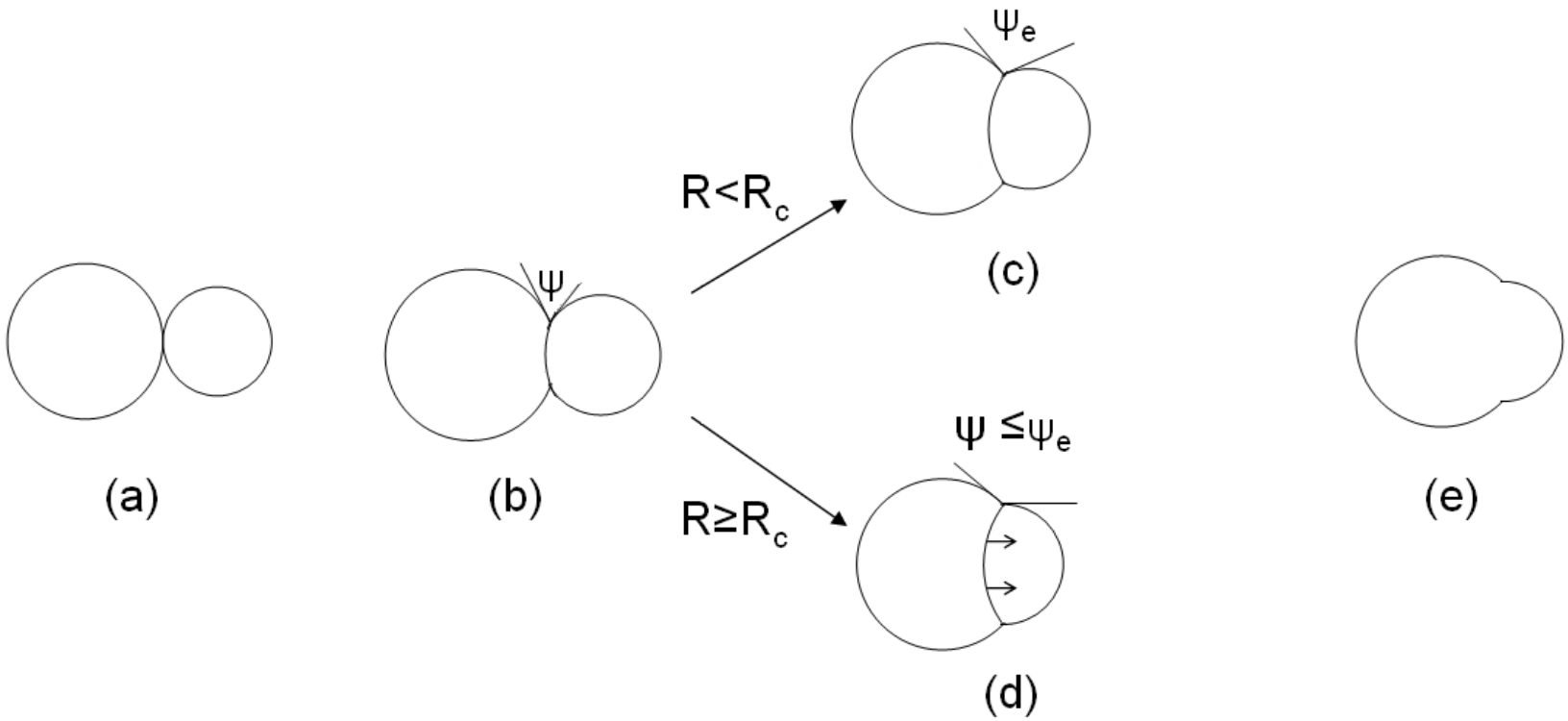
2.3.2.1 Two particles

The sintering of two unequal particles starts with a touching contact between them; $\psi = 0$. As sintering continues, the sintering geometry changes. Lange and Kellett used contact angle ψ to describe the geometrical changes. The geometrical changes in the sintering geometry are shown in Fig. 2.7.

The contact angle between the particles in the beginning of sintering is zero ($\psi = 0$) as shown in Fig. 2.7(a). The system lowers its energy by a particle penetration. As particles



2.6: Schematic cross section of interpenetrating spherical particles for Lange and Kellett's model; geometric notation used for the model is shown.



2.7: Geometrical changes during sintering and coarsening of two particles. Subfigure (D) shows the configuration when grain boundary can migrate through small grain and continuously decrease the free energy of the system.

penetrate into each other, the contact angle ψ increases (Fig. 2.7(b)). Lange and Kellett suggested that the system lowers its energy until contact angle becomes equal to equilibrium dihedral angle ψ_e by particle penetration. The radius of each particle $r_{1,2}$ normalized by initial radius $r_{i\ 1,2}$ in Fig. 2.6 can be determined using following equations.

$$R_1 = \frac{r_1}{r_{i\ 1}} = \Bigg[1 - \frac{1}{4}(1-\cos\theta_1)^2(2+\cos\theta_1) + 2\left(\frac{R}{R-1}\right)^3 \cos^3\left(\frac{\psi}{2}\right)(1-\cos\phi)^2(2+\cos\phi)\Bigg]^{-1/3} \tag{2.41}$$

$$R_2 = \frac{r_2}{r_{i\ 2}} = \Bigg[1 - \frac{1}{4}(1-\cos\theta_2)^2(2+\cos\theta_2) - 2\left(\frac{1}{R-1}\right)^3 \cos^3\left(\frac{\psi}{2}\right)(1-\cos\phi)^2(2+\cos\phi)\Bigg]^{-1/3} \tag{2.42}$$

where

$$\theta_1 = \arctan\left(\frac{R\sin\psi}{R\cos\psi+1}\right) \tag{2.43}$$

$$\theta_2 = \psi - \theta_1$$

$$\phi = \theta_1 - \psi/2$$

$$R = r_2/r_1 \tag{2.44}$$

The radius of curvature of the grain boundary can be given as:

$$r_{gb} = \frac{2r_1r_2}{r_2-r_1}\cos\frac{\psi}{2} = \frac{2r_2}{R-1}\cos\frac{\psi}{2} \tag{2.45}$$

As sintering progresses, the contact angle ψ increases. Lange and Kellett pointed out that angle ψ can be used to find a critical radius ratio R_c with which the grain boundary will migrate. They suggested that the grain will migrate by reducing its energy when center of the small particle lies on an imaginary plane formed by joining contact points of sphere. In this situation, the distance $r_1\cos\theta_1$ will be equal to zero.

$$
\begin{aligned}
r_1 \cos\theta_1 &= 0 \\
\Rightarrow \theta_1 &= \pi/2 \qquad \because r_1 \neq 0 \\
\arctan\left(\frac{R_c \sin\psi_e}{R_c \cos\psi_e + 1}\right) &= \pi/2 \qquad \text{from eq. 2.43} \\
R_c \cos\psi_e + 1 &= 0 \\
\Rightarrow R_c &= -(\cos\psi_e)^{-1}
\end{aligned} \tag{2.46}
$$

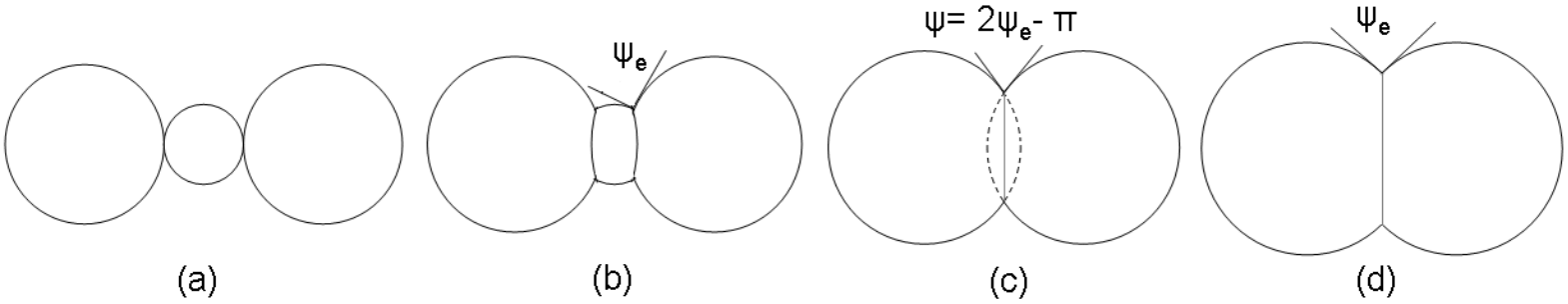
With respect to the critical radius ratio R_c, two geometrical configurations can be achieved as shown in Fig. 2.7(c)-(d). Configuration 2.7(c) occurs when $R < R_c$ and represent the lowest energy configuration during sintering without interparticle mass transport. Lange and Kellett suggested that development of configuration (e) from (c) in Fig. 2.7 would require an interparticle mass transport. The interparticle mass transport would result in increase in ratio R. The configuration in Fig. 2.7(d) would occur when ratio R becomes R_c. The grain boundary migration at this point will lead to configuration (e). The configuration in Fig. 2.7(d) can occur from (b) when $R \geq R_c$. In this situation, the grain boundary can migrate without any coarsening.

The condition at which the grain boundary migration occurs with critical radius ratio R_c in eq. 2.46. Lange and Kellett analyzed the effect of equilibrium dihedral angle on the critical radius ratio R_c. They found that the initial particle ratio required for spontaneous grain boundary migration R_c decreases from 6.6 to 1.1 when dihedral angle ψ_e increases from 100° to 150°. This indicates that the grain boundary migration will occur easily in systems with higher dihedral angle. This will result in a larger grain growth during sintering.

2.3.2.2 Three particles

Lange and Kellett applied a geometrical analysis to a three touching particles geometry. A geometrical analysis similar to performed in Sec. 2.3.2.1 performed on collinear particles can reveal the microstructural evolution as shown in Fig. 2.8.

The system lowers its energy by forming the neck between the particles in transition from Fig. 2.8 (a) to (b). If particle size ratio R is less than R_c, coarsening would be required before the grain boundary migrates. The grain boundary can spontaneously migrate after sintering if $R \geq R_c$. The small particle would disappear as a result of grain boundary migration as shown in FIg. 2.8(c). The contact angle between the large particles after disappearance of small particle can be given as:



2.8: Geometrical changes produced by sintering and coarsening of three collinear particles. The disappearance of small particles in subfigure (c) reinitiates sintering

$$\psi = 2\psi_e - \pi \tag{2.47}$$

The new contact angle ψ is less than the equilibrium dihedral angle. The contact angle given by eq. 2.47 will result in sintering until dihedral angle increases to ψ_e. This indicates that the disappearance of the small particle reinitiates the driving force for mass transport for neck formation that leads to configuration shown in Fig. 2.8(d). Lange and Kellett concluded that the coarsening within an equilibrium configuration can reestablish the driving force for sintering.

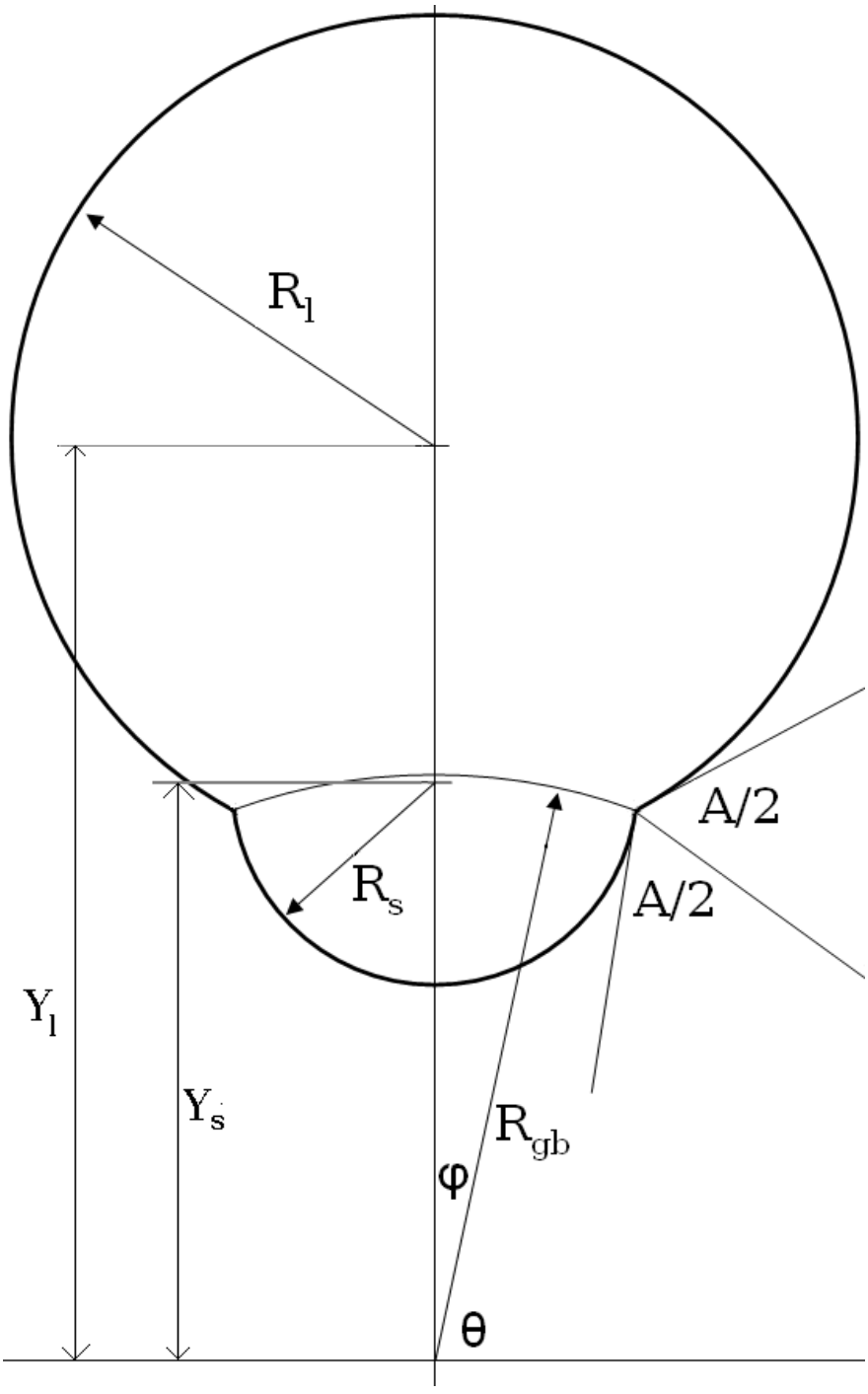
Lange and Kellett also analyzed the shrinkage in collinear particles during microstructural evolution shown in Fig. 2.8. The analysis was performed for initial sintering ((a)-(b)), coarsening during shrinkage of small particle ((b)-(c)) and sintering between two large particles ((c)-(d)). The analysis showed that the sintering causing the neck growth contributed significantly in overall shrinkage. The process of coarsening which causes the small particle disappearance produced a small shrinkage. Lange and Kellett argued that the shrinkage due to reduction in small particle size was compensated by the growth of large particles. Their analysis showed that the shrinkage during coarsening is nil for systems with dihedral angle ψ_e greater than 130°. Lange and Kellett concluded that the grain growth within the linear array resulting from coarsening reinitiates the driving force for coarsening, and thus shrinkage.

2.3.3 Colbeck's model

Colbeck's model provides the neck growth kinetics by grain boundary diffusion on a relative time scale [9]. The geometrical model provides insights into the bonding of particles in layers. The study of the bonding of layers is of great interest in packing of snow to avoid avalanche.

The idealized geometry used in Colbeck's model to derive the sintering kinetics is shown in Fig. 2.9.

The geometry shown in Fig. 2.9 consists of two interpenetrating spherical particles. The grain boundary is assumed to have radius R_{gb}. The radii of the small and the large particles are assumed to be R_s and R_l respectively. The dihedral angle is assumed to be A. Distances in the geometry can be described with respect to the center of the sphere forming the grain boundary. In this geometry, the radius of curvature of the grain boundary is assumed to be constant. The particle radii and center positions calculated in terms of grain boundary curvature are given in eq. 2.48-2.51.



2.9: Geometry of two particle microstructure with a grain boundary of fixed curvature

$$R_s = \frac{R_{gb}\cos\theta_\circ}{\cos(\theta_\circ - A/2)} \tag{2.48}$$

$$R_l = -\frac{R_{gb}\cos\theta_\circ}{\cos(\theta_\circ + A/2)} \tag{2.49}$$

$$Y_s = \frac{R_{gb}\sin\theta_\circ}{\cos(\theta_\circ - A/2)} \tag{2.50}$$

$$Y_l = -\frac{R_{gb}\sin\theta_\circ}{\cos(\theta_\circ + A/2)} \tag{2.51}$$

where subscript $\circ$ in angle $\theta_\circ$ refers to value of angle θ at the cusp of the boundary. Colbeck argued that the grain boundary flux J is driven by the gradient of normal stress σ along the grain boundary. The neck grows by molecules diffusing out of the grain boundary. Coble set up the flux equation describing the molecular flux along the curved boundary as:

$$J = \frac{\delta_{gb} D_{gb}}{kT}\frac{\mathrm{d}\sigma}{\mathrm{d}s} \tag{2.52}$$

where, δ_{gb} is the grain boundary width, D_{gb} is grain boundary diffusion coefficient and s is the distance along the grain boundary from its center. The distance s can be given as:

$$s = R_{gb}(\pi/2 - \theta) \tag{2.53}$$

Colbeck derived sintering kinetics expression using flux equation (eq. 2.52), continuity equation and force balance at the grain boundary as:

$$\frac{\mathrm{d}(Y_l - Y_s)}{\mathrm{d}t} = -\frac{\Omega\delta_{gb}D_{gb}\gamma_s}{2kT{R_{gb}}^2}\left[\frac{\frac{1}{R_{sm}} - \frac{1}{R_s} + \frac{1}{R_{lm}} - \frac{1}{R_l}}{2\ln\left(\frac{2}{1+\sin\theta_\circ}\right) - \left(\frac{1-\sin\theta_\circ}{\cos\theta_\circ}\right)^2}\right] \tag{2.54}$$

where R_{sm} and R_{lm} are the radii of small and large particles respectively when the neck is fully formed. A relationship between R_{sm} and R_{lm} in terms of R_{gb} can be found by analyzing the pressure difference across at the grain boundary.

$$
\begin{aligned}
& P_s - P_l = \frac{2\gamma_s}{R_{sm}} - \frac{2\gamma_s}{R_{lm}} = \frac{2\gamma_{gb}}{R_{gb}} \\
\Rightarrow \quad & \gamma_s \left(\frac{1}{R_{sm}} - \frac{1}{R_{lm}} \right) = \gamma_{gb} \frac{1}{R_{gb}} \\
\Rightarrow \quad & \frac{1}{R_{sm}} - \frac{1}{R_{lm}} = \frac{\gamma_{gb}}{\gamma_s} \frac{1}{R_{gb}} \\
\Rightarrow \quad & \frac{1}{R_{sm}} - \frac{1}{R_{lm}} = 2R_{gb} \cos\left(\frac{A_m}{2} \right)
\end{aligned} \tag{2.55}
$$

where P_s and P_l are the pressures in the small and the large particles respectively. Angle A_m in eq. 2.55 represent the equilibrium dihedral angle. The values of R_{sm} and R_{lm} can be calculated with eq. 2.55 and volume conservation in eq. 2.56.

$$
\begin{aligned}
{R_{s\circ}}^3 + {R_{l\circ}}^3 = & R_{sm}^3 \left(1 - \frac{1}{4} \cos\left(\theta_\circ + \frac{A_m}{2} \right) \left(1 - \cos\left(\theta_\circ + \frac{A_m}{2} \right) \right)^2 \right) \\
& + R_{lm}^3 \left(1 - \frac{1}{4} \cos\left(\theta_\circ - \frac{A_m}{2} \right) \left(1 - \cos\left(\theta_\circ - \frac{A_m}{2} \right) \right)^2 \right)
\end{aligned} \tag{2.56}
$$

where $R_{s\circ}$ and $R_{l\circ}$ are the initial radii of the small and the large particle, respectively. The rate of sintering can be determined from eq. 2.54 using R_{sm} and R_{lm} obtained from eq. 2.55 - 2.56.

Colbeck analyzed the geometrical model with respect to the bonding between the snow layers. Coble observed that the grain boundary becomes highly curved and smaller grain becomes more deeply embedded in the larger grain as differences in the particle size increases. If particle size ratio (R_l/R_s) is large enough, the exposed area of the small grain may not be sufficient for bonding with other grains. This inability of bonding of smaller particle may result in weaker bonding between snow layers. Coble also estimated of effect of particle size ratio on the rate of neck growth. Colbeck found that the stress gradient in the grain boundary increases and the grain boundary becomes more curved as the particle size ratio (R_l/R_s) increases. Both of these effects result in a decrease of the rate of neck growth for two reasons: (1) flux rate in the grain boundary decreases and (2) path length for the flux increases.

CHAPTER 3

MONTE CARLO SIMULATION

Monte Carlo simulation method can be used to obtain sintering kinetics. Monte Carlo simulation method utilizes random numbers and a probability distribution function. The microstructural evolution is determined by generating a random number and comparing it with probability distribution function. The methodology of the simulation method are described in Section 3.1. Although the method was initially used by scientists working on atomic bomb, it has successfully been applied to simulate microstructural evolution. The development and recent application of Monte Carlo simulations in microstructural evolution are reviewed in Section 3.2. In this dissertation, Monte Carlo method is applied to simulation the sintering of two unequal particles. The simulation is performed on a two-dimensional microstructure. The microstructural evolution shows subprocesses of neck growth, coarsening and grain boundary migration. The details of Monte Carlo simulation performed in this dissertation are provided in Section 5. Results of Monte Carlo simulation are also discussed in Section 5.

3.1 Methodology

In Monte Carlo simulation method, the microstructure is determined using a probability distribution function (PDF). The Monte Carlo method utilizes a sequence of random numbers to simulate the sintering phenomenon. The random numbers set up a trial for the change in microstructure. The outcome of the trial is decided by the change in energy of the microstructure and probability distribution function (PDF). The details of the Monte Carlo simulation implemented in this dissertation are described below.

A microstructure in the Monte Carlo simulation method is described as a two-dimensional (2D) lattice. In this work, A square lattice ($n \times n$) with regular spacing between the lattice points is chosen. This choice of the square lattice is very common for the simulation of sintering [20]. The lattice sites on the exterior of the square region have fewer neighboring sites. For avoiding the difference between interior and exterior lattice sites, a periodic boundary condition is applied. The periodic boundary condition is a frequently applied

boundary condition for microstructural simulation. The periodic boundary condition involves assigning a neighbor of each side of the edge to the opposite side of the edge.

Although a simulation of isotropic material is intended, the choice of the square lattice induces anisotropy along [1 0] and [1 1] direction [21]. The anisotropy in surface energy and mobilities along [1 0] and [1 1] are induced differently. The difference of anisotropy in surface energy and mobilities compensates each other and reduces overall anisotropy in square lattice microstructure.

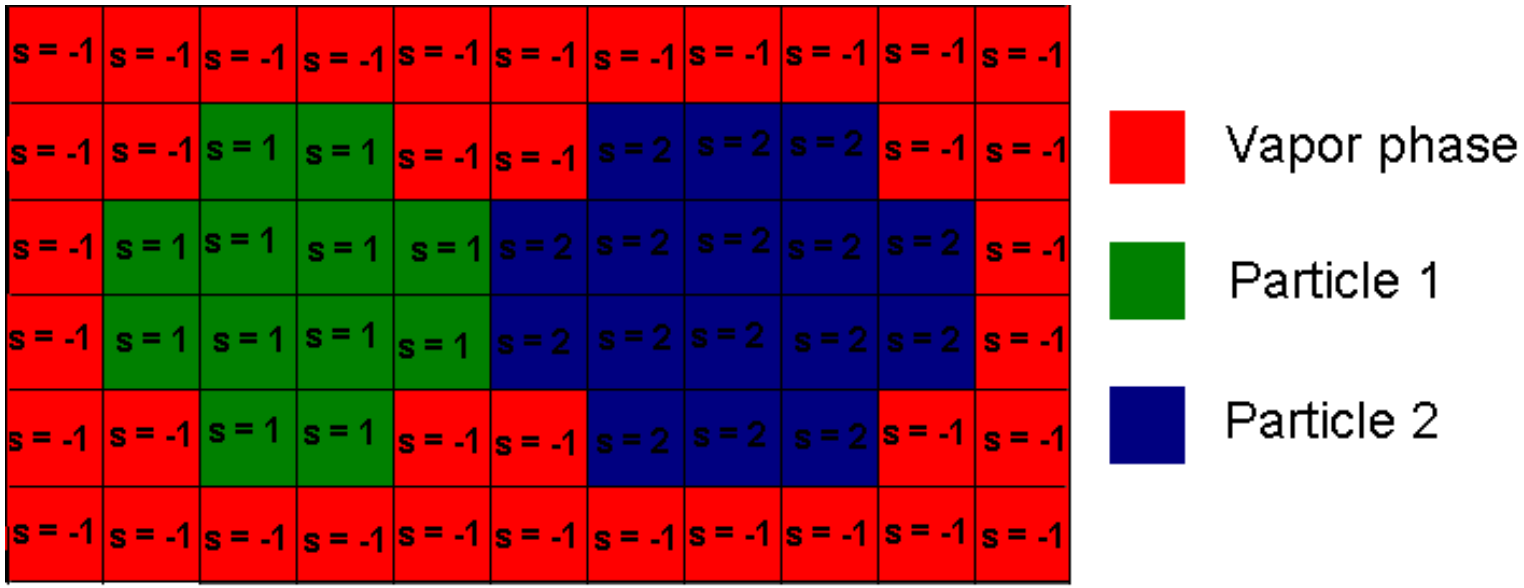
Each lattice site in the microstructure is assigned a spin state s as shown in Fig. 3.1. The spin states of lattice site in solid phase are assigned a positive integer less than Q. Q is the number of possible spin states in the simulation. The spin state s_i of i^{th} lattice site is related to its crystallographic orientation. The spin states of lattice sites in vapor phase are assigned with negative unity (-1). A grain in the microstructure can be defined as a contiguous region of lattice sites with same spin states s. The neighboring lattice sites of different spin states contribute to the energy E of the microstructure. The total energy E of the microstructure can be given as

$$E = \frac{1}{2}\sum_{i=1}^{n^2}\sum_{j=1}^{z}\gamma(s_i, s_j) \tag{3.1}$$

where z refers to the number of neighbors of a site in microstructure. The value of z up to second neared neighbors in the square lattice is 8. In eq. 3.1, contributions from bulk free energy and other sources are assumed to be zero due to their invariable nature during the simulation. The term γ in eq. 3.1 is defined such that

$$\gamma(s_i, s_j) = \begin{cases} 0 & \text{for } s_i = s_j \\ \gamma_s & \text{for } s_i \neq s_j \text{ and } s_i s_j < 0 \\ \gamma_{gb} & \text{for } s_i \neq s_j \text{ and } s_i s_j > 0 \end{cases} \tag{3.2}$$

The distribution of the spin state s in the lattice determines the microstructure. The microstructure is evolved by change in the spin states s. For changing the spin states, r^{th} lattice site and neighboring t^{th} lattice sites are randomly chosen. A new spin state s'_r and s'_t are temporarily assigned to r^{th} and t^{th} lattice sites in the following way:



3.1: Distribution of spin states s in lattice of Monte Carlo simulation.

$$(s'_r,\ s'_t) = \begin{cases} (-1,\ -1) & \text{if } s_r < 0 \text{ and } s_t < 0 \\ (R(Q),\ -1) & \text{if } s_r < 0 \text{ and } s_t > 0 \\ (-1,\ R(Q)) & \text{if } s_r > 0 \text{ and } s_t < 0 \\ (R(Q),\ s_t) & \text{if } s_r > 0 \text{ and } s_t > 0 \end{cases} \tag{3.3}$$

where $R(Q)$ represent a random number in [1 Q]. It should be noted that assignment of eq. 3.3 conserves the spin states of vapor phase ($s = -1$). When one of the (r^{th}, t^{th}) lattice sites has a spin state of -1, the assignment ensures that one of the s'_r or s'_t is also negative to conserve the number of sites having spin states of -1. This scheme of conserving the fraction of spin states is called Kawasaki dynamics. On the other hand, spin states for the solid phase are not conserved. This schemed is called Glauber dynamics.

The microstructural evolution involves calculation of change in energy before and after assigning the temporary spin states to r^{th} and t^{th} lattice sites. The change in total energy of the system ΔE due to the spin states from s_r and s_t to s'_r and s'_t can be calculated using eq. 3.1. The changes in the spin states of s^{th} and t^{th} lattice sites are accepted or rejected based upon the change in energy ΔE and probability distribution function (PDF). The probability $P(\Delta E)$ of the change to spin states can calculated from probability distribution function given in eq. 3.4.

$$P(\Delta E) = \begin{cases} 1 & \text{if } \Delta E \ \leq \ 0 \\ \exp\dfrac{-\Delta E}{kT_s} & \text{if } \Delta E \ > \ 0 \end{cases} \tag{3.4}$$

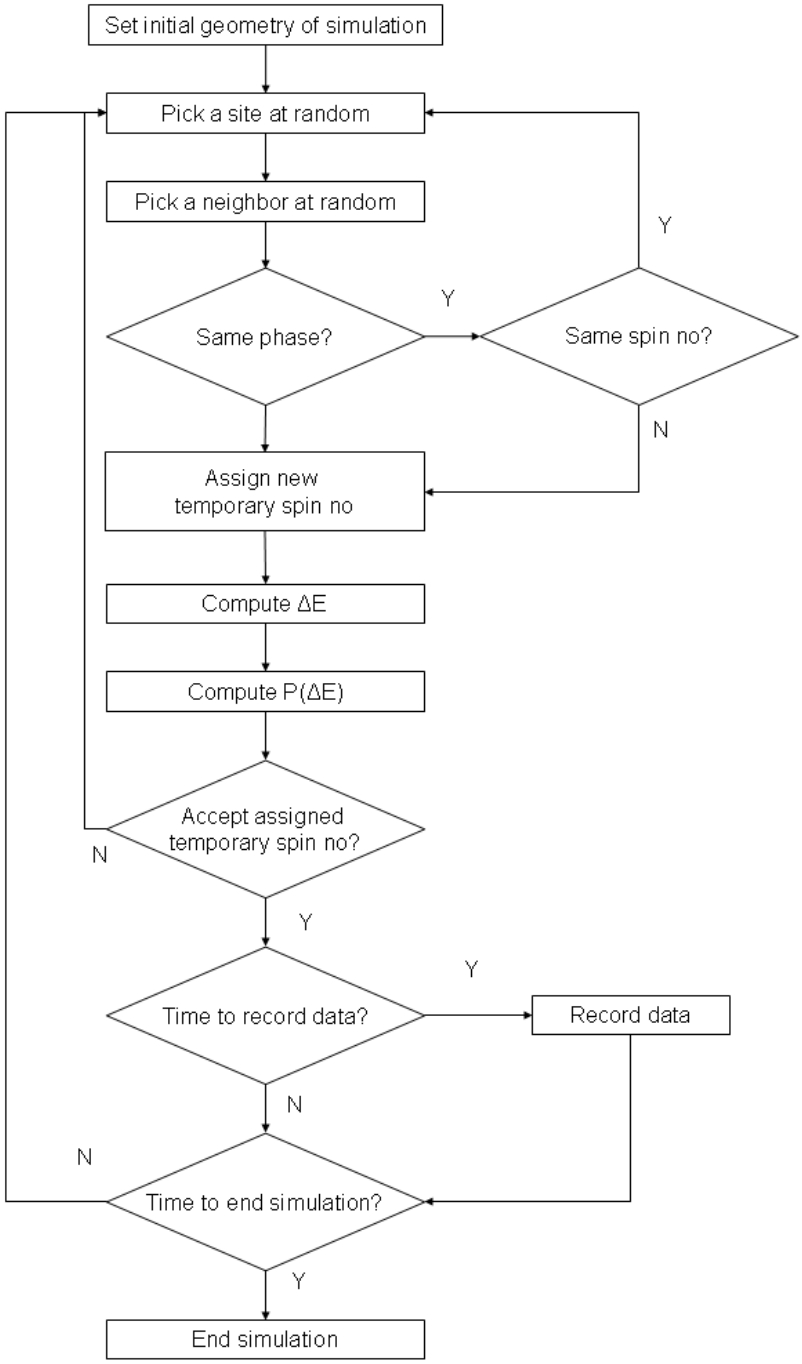
where kT_s defines the thermal energy of the simulation. The term kT_s is analogous to the thermal energy available at the sintering temperature but it is not directly related. The probability distribution function given in eq. 3.4 is known as Metropolis function. The Metropolis function is a common choice of PDF [22].

The acceptance or the rejection of temporarily assigned spin states is finally decided by generating a random number R' in [0 1]. If R is less than or equal to the probability $P(\Delta E)$, the changes in the spin state s_r and s_t are accepted. The microstructure evolution is calculated by sampling the microstructure for change in spin state s.

The scheme for Monte Carlo simulation for the sintering can be described as given in Listing 3.1 as algorithm. The algorithm of Monte Carlo simulation can also be depicted as flow chart in Fig. 3.2.

Listing 3.1: Algorithm for Monte Carlo simulation of sintering

```
Initialize microstructure
r = position of randomly chosen site
t = position of randomly chosen neighbor of r
s_r_old = spin state of site r
s_t_old = spin state of site t
s_r_new = temporarily assigned spin state of site r
s_t_new = temporarily assigned spin state of site t
E_old = total energy with spin states s_r_old and s_t_old
E_new = total energy with spin states s_r_new and s_t_new
if (E_new - E_old) > 0 {
  s_r_new is accepted
  s_t_new is accepted
}else {
  P_dE = exp(-(E_new - E_old)/kT)
  R = a random number between 0 and 1
  if (random < P_dE){
    s_r_new is accepted
    s_t_new is accepted
  }
}
```



3.2: Flow chart for Monte Carlo simulation of sintering.

3.2 Development and applications

The Monte Carlo method was developed under the Manhattan project during World War II [22]. The Monte Carlo method was first applied in solid state physics to model the ferromagnetic materials. Potts simulated the magnetic spins of two possible types: spin up and spin down [23]. The scheme of spin up and spin down is known as the Ising model [24]. Anderson *et al.* were the first to apply the Monte Carlo method to grain growth in single phase materials [25]. They simulated the shrinkage of a circular grain embedded in a larger grain using a triangular lattice. A reduction in the grain area $A(t) - A(t = 0)$ was found to be linear with respect to time t such that:

$$A(t) - A(t = 0) = -\alpha t \tag{3.5}$$

where α is a constant. These kinetics of grain shrinkage were found in agreement in theoretical predictions [26–28]. Anderson *et al.* also applied the Monte Carlo method simulation of polycrystalline microstructure [25]. The grain growth exponent n in eq. 3.6 was found to be independent of choice of orientations Q when $Q > 30$.

$$\bar{R}(t)^n = kt \tag{3.6}$$

where $\bar{R}$ is the mean grain size, k is a prefactor and t is the time. The value of grain growth exponent n was found to be 0.41 ± 0.03. Anderson *et al.* argued that the Monte Carlo method can be used to simulate a truly continuous range of possible grain orientations ($Q = \infty$) with a model in which Q is large but finite. The grain growth exponent n was found to be independent of temperature. The observed dependence of exponent n was in agreement with predictions of theory [29].

Srolovitz *et al.* studied the grain size distribution, topology and local dynamics in Mote Carlo simulations of grain growth [30]. The grain size distribution obtained from the simulations resulted in a self similar log normal curve at various times. The obtained log normal grain size distribution showed a good agreement between simulation results and experimental observations during grain growth of high purity aluminum [31, 32]. The

simulation results were also compared to Beck's data for grain edge distribution. An excellent agreement between simulations and experiments was found.

Srolovitz *et al.* also incorporated second phase particles to the grain growth simulations [33]. The lattice sites of the second phase particles were assigned with orientation Q number different from all other grains. The simulation for the grain growth in presence of second phase particle was set up such that the particles were immobile. The simulation results showed a normal grain growth which was later pinned by the presence of second phase particles. The grain size distribution and growth kinetics were found to be independent of particle concentration. The final average grain size and time required to obtain pinned microstructure was proportional to inverse of the particle concentration. Srolovitz *et al.* proposed a topological theory explaining these results [33].

Grest *et al.* introduced an anisotropy in grain boundary energy during grain growth simulations [34]. The anisotropy in grain boundary energy was assumed to be dependent upon the grain misorientation. Three different functions of grain boundary energy were chosen such that the low angle boundaries had lower energies than the high angle energies. The anisotropic grain boundary energy resulted in faceted grain structure. Grain growth kinetics showed that the grain growth exponent n decreases when grain boundary energy increases. The grain growth exponent n decreased from 0.42 ± 0.02 for isotropic grain boundary energy to 0.25 ± 0.02 for highly anisotropic grain boundary energy. The exponent n of 0.25 ± 0.02 was found to be independent of choice of function describing grain boundary energy. This indicated a universal nature of the exponent.

Srolovitz *et al.* extended the Monte Carlo method to study abnormal grain growth [35]. The abnormal grain growth in bulk sample simulation was achieved in a two-step process. In the first step, the normal grain growth was allowed to take place and in the second step, a large grain was physically introduced in the microstructure as a nucleus. An abnormal growth of the nucleus was observed during the simulation. Abnormal grain growth in the presence of particle dispersions and in thin films was also studied.

Holm *et al.* successfully implemented the Monte Carlo method to simulate grain growth in two phase materials [36]. The model was implemented by assigning different signs of orientations Q to different phases. The signs of orientations were utilized in calculating grain boundary and interphase energy. The ratio of grain boundary and interphase energies were shown to have strong effect of microstructural evolution. The microstructural evolution was analyzed with respect evolution of grain boundary trijunctions. The microstructure was found to be realistic having only thermodynamically stable features. The trijunction angles

acquired values close to their equilibrium values determined by interface energy and grain boundary ratios.

Liu and Lin successfully demonstrated the application of Monte Carlo simulation method to liquid phase sintering [37]. They modeled the grain growth by coalescence in initial stage of sintering. The also assumed an anisotropy in grain boundary energy. The grain boundary energy was considered to be dependent upon the grain misorientation angle. The coincidence site lattice (CSL) boundaries were assigned with lower grain boundary energies. A particle size distribution in the initial microstructure was also considered. The simulation utilized a random microstructure based upon a given particle size distribution. Liu and Lin focused the sintering simulation on agglomeration of particles. They concluded that the size distribution of agglomerates strongly depends upon solid volume fraction and standard deviation of initial particle size distribution. Liu and Lin found that size distribution of agglomerated particles can be broadened by either increasing standard deviation or decreasing volume fraction of liquid. Liu and Lin argue that increased probability of solid solid contact due to larger standard deviation of the size distribution or low volume fraction of liquid results in wider size distribution of agglomerated particles. Monte Carlo simulation also showed the evidences of the particle coalescence. Liu and Lin also compared the simulation results with experimental data on W-Ni-Fe alloy. A satisfactory agreement between the simulation and experimental results was observed.

Zhang *et al.* studied grain growth during intermediate and final stage sintering, and during Ostwald ripening [6] in $BaTiO_3$ based ceramics [38]. The liquid phase was observed to be well distributed on grain boundaries. Zhang *et al.* also observed evidence of grain coalescence. A grain growth exponent of 4.2 obtained from simulation was in good agreement with experiments [39].

Monte Carlo simulations described above utilize surface energies and dihedral angle to determine microstructural evolution. Based upon results in ref. [40], Aldazabal *et al.* suggested that dissolution and precipitation of solute based upon solubility should be incorporated in liquid phase sintering simulations [41]. They incorporated the solute concentration using phase diagrams which affected the dissolution and precipitation of solute. The distribution of the solute was also affected by diffusion. They used Fick's second law to determine the movement of solute. In the scheme devised by Aldazabal *et al.*, different weights were attributed to first and second nearest neighbor lattice sites. A certain number of weighted neighbor was needed for dissolution and precipitation events. They applied their scheme to simulation of tungsten particles in nickel matrix. Aldazabal *et al.* found the volume

fraction of solid to be consistent with predictions of phase diagram. However, the time to obtain the equilibrium volume fractions was affected by the choice of diffusion rates. Aldazabal *et al.* concluded that the diffusion rates have strong influence on morphology of final microstructures.

Monte Carlo simulation was applied to grain growth in cemented tungsten carbides by Kishno *et al.* [42]. They argued that classical theory of Ostwald ripening is insufficient to explain the grain growth mechanism in cemented carbides. The insufficiency of the theory is prominent in case of sintering with low binder content. In case of sintering with low binder content, all of the carbide grains are not surrounded by cobalt. The grain growth of these grains is controlled by grain boundary migration instead of Ostwald ripening. Kishino *et al.* applied Monte Carlo simulation due to its potential to incorporate Ostwald ripening and grain boundary migration simultaneously. The grain boundary migration was implemented following a procedure similar to grain growth; however, a random walk method was employed to simulate dissolution and precipitation to simulate Ostwald ripening. They used the simulation to investigate the continuous and discontinuous grain growth mechanisms and effect of grain growth inhibitors such as VC and Cr_3C_2. For simulation of sintering without grain growth inhibitors, two grain boundary energies γ_{ss1} and γ_{ss2} where chosen such that

$$\gamma_{sl} < 2\gamma_{ss1}; \qquad \gamma_{sl} < 2\gamma_{ss2} \tag{3.7}$$

where γ_{sl} is the interfacial energy between matrix and particles. The choice of two grain boundary energies resulted in two types of interfaces: (1) solid solid interface and (2) solid liquid interface. Kishino *et al.* observed a continuous grain growth with lower binder content (<5%) and high binder content (>20%). A discontinuous grain growth was observed for sintering with 10% binder content. The expermental results of sintering of cemented tungsten carbides confirmed the binder content regime for discontinuous grain growth [42]. Based upon the simulation and experimental results, Kishino *et al.* proposed a mechanism of discontinuous grain growth. They proposed that the grain boundary migration controls the grain growth in sintering with lower binder content. On the other hand, the grain growth is controlled by Ostwald ripening in case of higher binder content. Both of these mechanisms result in continuous grain growth. The discontinuous grain growth at certain binder content is a result of localization of binder phase. Kishno *et al.* also studied the effect of coarse grains on discontinuous grains. A discontinuous grain growth was observed when a large

grain surrounded by binder was placed in the simulation microstructure. Based upon the simulation results, they recommeneded that the contamination by coarse grains must be avoided in the manufacturing process of fine grain carbides.

Kishino *et al.* also studied the effect of grain growth inhibitor doping on grain growth during sintering of cemented carbides [42]. The effect of doping materials was incorporated by changing the grain boundary energies such that eq. 3.7 does not satisfy. The choice of surface energy was justified based upon the contiguity results in ref. [43]. With addition of doping materials, a continuous grain growth was observed irrespective of binder content. The grain growth with doping was observed to be controlled by the grain boundary migration. However, the contribution of grain boundary migration in grain growth diminished with higher binder content. These results were verified by experimental data [43].

The sintering models described above incorporate grain boundary migration and dissolution – reprecipitation as sintering mechanisms. Tikare *et al.* suggested the addition of vacancy annihilation to model densification during solid state sintering [44]. They developed a sintering model involving three processes: (1) grain growth by short range diffusion across the grain boundary, (2) long range diffusion of material to pores by grain boundary diffusion or surface diffusion and (3) annihilation of vacancies. The grain growth by short range diffusion was applied using a scheme of grain growth in single phase materials described in ref. [25]. The pore migration was simulated using conserved dynamics. The pore and grain lattice sites were switched in favor of reduction of energy. This switch of sites ensures that the solid and vapor volume fractions in the simulation domain are conserved. A higher value of temperature term (K_BT) was used to simulate pore migration. The choice of the temperature was justified based upon the results of their previous research [45, 46]. For simulating the annihilation of vacancies, isolated pore sites were considered as vacancies. When a vacancy at the grain boundary was annihilated, the center of neighboring grain was moved towards the grain boundary. This movement resulted in a densification. Tikare *et al.* argued that the rate controlling factor in the annihilation of a vacancy is the diffusion along the entire grain boundary. Therefore, they correlated the frequency of vacancy annihilation to the time required to diffuse along the grain boundary. They applied their model to sintering of three equal particles. They showed that their model accurately predicted the expected topological changes and kinetics of densification. A comparison between of topological changes and kinetics of densification between their model and experimental evidences in literature showed a good agreement. Braginsky *et al.* applied the model to sintering of array of particles and sintering of randomly arranged particles [20] and found a satisfactory performance.

CHAPTER 4

PHASE FIELD SIMULATION

The previous chapter described the application of the Monte Carlo simulation method to sintering. The Monte Carlo method is a stochastic method which relies on random sampling to calculate the microstructural evolution. In Monte Carlo simulation, the microstructure cannot be exactly determined from microstructure at previous time step. On the contrary, the phase field simulation method is a deterministic simulation method in which we can exactly determine the microstructural evolution. It should be noted that Monte Carlo and phase field simulation methods have different methodology; they use minimization of energy to obtain the microstructural evolution. Therefore, both of them can be used to gain insights into mechanism of sintering process. This chapter describes the phase field simulation method and recent developments for its application to sintering process.

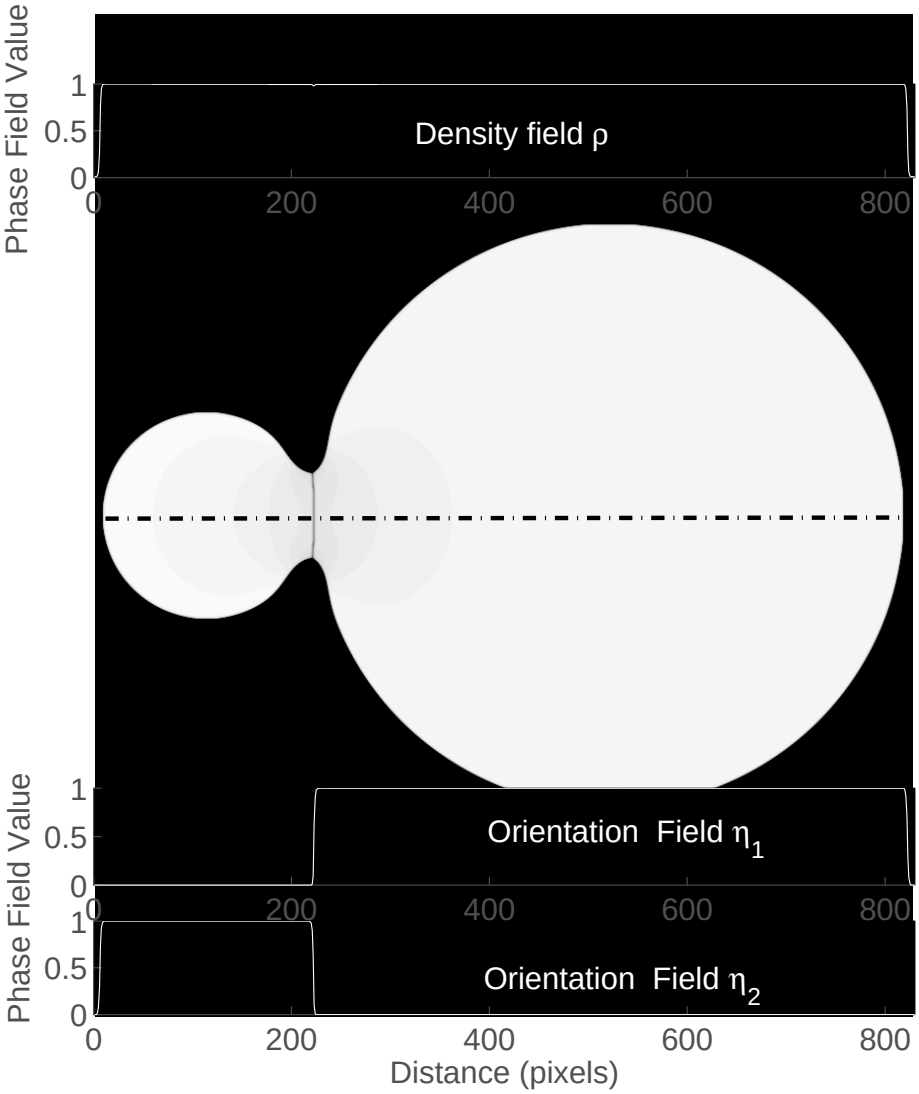
A microstructure in the phase field method is described by spatial distribution of density and crystal orientation. Changes in distribution of density and crystal orientations are derived by minimizing a free energy functional. The free energy functional contains a global minimum with respect to the density and crystal orientation. This minimum corresponds to a solid phase. Changes in distribution of density and crystal orientations are used to obtain the microstructural evolution. A detailed scheme of obtaining microstructural evolution using the phase field method is described in Section 4.1. Although the phase field model can be used to obtaining the microstructural evolution, it was developed for deriving the interface width by Cahn and Hilliard [47]. Cahn later applied the model to simulate spinodal decomposition [48]. Thereafter, the model has been modified to simulate various microstructural phenomena. The development and application of the phase field method are reviewed in Section 4.2. In this dissertation, the phase field method is applied to simulate the sintering of two unequal sized particles, equal sized particles, pores, and randomly arranged particles. Specific details of these simulations are given in Chapter 6. The results of these simulations are also discussed in Chapter 6.

4.1 Methodology

The phase field simulation method is based upon temporal evolution of thermodynamic quantities such as density, crystal orientation and temperature. In this method, spatial distributions of these thermodynamic quantities are referred to as phase fields [49]. The microstructural evolution is determined by a change in these phase fields using a free energy functional. This free energy functional which describes the total free energy of the system can be expressed in terms of thermodynamic quantities. The microstructure evolves by minimization of the total free energy of the system. Kinetic equations of microstructural evolution obtained by the minimization of energy are variational derivatives containing phase fields.

In this dissertation, thermodynamic quantities, density ρ and crystal orientation η are considered as the phase fields. The density field refers to mass per unit volume in the microstructure. The density field acquires the value of solid density ρ_{sol} inside and vapor density ρ_{vap} outside a solid particle, respectively. The orientation field refers to crystal orientation of the particles. Each particle is assigned with one orientation field η_i which is unity inside the associated particle and zero elsewhere. In case of a two particle microstructure, orientation fields η_1 and η_2 can be assigned to the first and second particles. Fig. 4.1 shows the variation of density field ρ and orientation field η_1, η_2 across various interfaces. Fig. 4.1 depicts that value of density field ρ gradually increases from vapor density to sold density across vapor solid interface. Similarly, Fig. 4.1 indicate that value of orientation fields η_1 and η_2 transition gradually between 0 and 1.

The density field ρ and orientation field η can be used to calculate free energy at a position in microstructure. The free energy equation is chosen such that the energy is at minimum when phase fields acquire their equilibrium values. An energy minimum occurs inside the solid phase when the density field is equal to the solid density ρ_{sol} and the crystal orientation field for only one grain is equal to unity. An energy minimum also occurs when the density field is equal to the vapor density ρ_{vap} and the orientation fields for all of the particles are zero. Variational derivatives of the above mentioned free energy equation are used to determine the kinetics of microstructure evolution. The kinetics of microstructure evolution also depend on diffusion coefficients. The diffusion coefficient is generally chosen as unity everywhere in the microstructure [50–52]. However, a variable diffusion coefficient is used in this work [53]. The free energy, diffusion coefficient, and kinetics equations are further described in the following sections.



4.1: Variation of density field ρ and orientation field η in microstructure.

4.1.1 Free energy

The total free energy F over the microstructure volume can be defined as a sum of bulk free energy, interface energy and grain boundary energy as given in eq. 4.1.

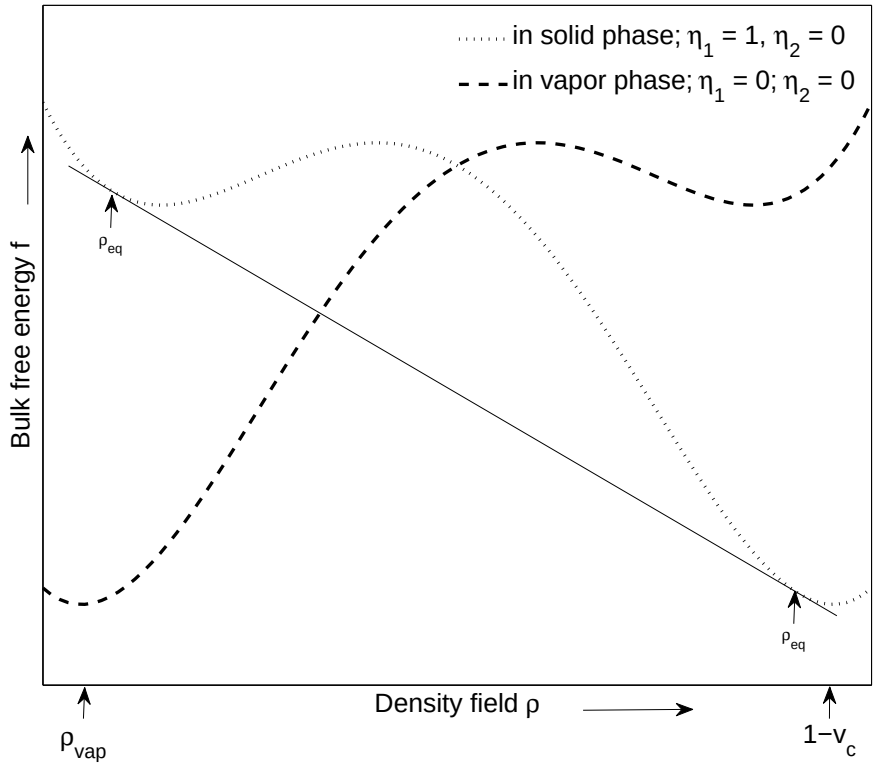
$$F = \int_v \left[\underbrace{f(\rho, \eta_{1..p})}_{\text{Bulk free energy}} + \underbrace{\frac{\kappa_\rho}{2}(\nabla\rho)^2}_{\text{Interphase energy}} + \underbrace{\frac{\kappa_\eta}{2}\sum_{i=1}^{p}(\nabla\eta_i)^2}_{\text{Grain boundary energy}} \right] \mathrm{d}v \tag{4.1}$$

where p is the total number of grains, and κ_ρ and κ_η are gradient energy coefficients. The bulk free energy $f(\rho, \eta_{1..p})$ has a form of the Landau type potential [54] given by eq. 4.2.

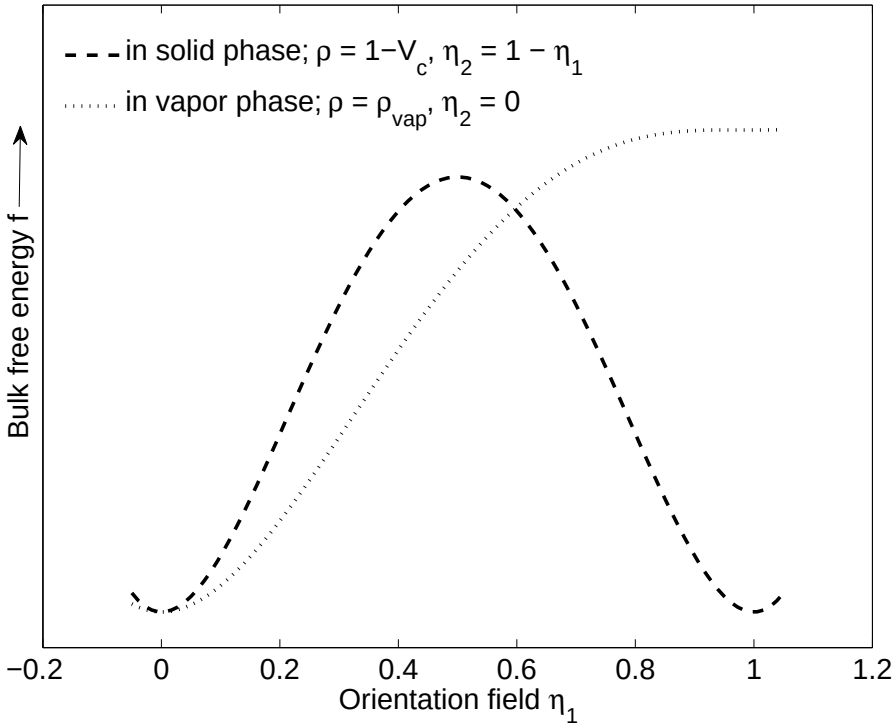
$$\begin{aligned} f(\rho, \eta_{i..p}) =& A(\rho^4 + \frac{-4\rho_{vap} - 4(1-V_c) - 2}{3}\rho^3 \\ &+ \frac{4\rho_{vap}(1-V_c) + 2\rho_{vap} + 2(1-V_c)}{2}\rho^2 \\ &- 2\rho_{vap}(1-V_c)\rho) \\ &+ B(\rho^2 + 6(1-\rho)\sum_i^p \eta_i^2 - 4(2-\rho)\sum_i^p \eta_i^3 \\ &+ 3\sum_i^p \eta_i^4 + C_{gbe}\sum_i^p \sum_{j,j\neq i}^p \eta_i^2\eta_j^2) \end{aligned} \tag{4.2}$$

where A and B are constants. The parameters ρ_{vap} and V_c in eq. 4.2 represent the equilibrium vapor density and equilibrium vacancy concentration, respectively, $v_c = 1 - \rho_{sol}$. The parameter C_{gbe} in eq. 4.2 affects the grain boundary energy. Eq. 4.2 is chosen such that the bulk free energy $f(\rho, \eta_{1..p})$ is minimized when the value of density field ρ approaches solid density $(1 - V_c)$ or vapor density ρ_{vap}. This suggests that the bulk free energy is minimized in solid phase and vapor phase. The minimization of bulk free energy $f(\rho, \eta_{i..p})$ in solid phase requires that only one of the orientation fields η_i approaches unity and rest of the orientation fields approach zero. The minimization of bulk free energy $f(\rho, \eta_{i..p})$ in vapor phase requires that all of the orientation fields η_i approach zero. The positions of free energy minima in the density ρ and orientation η fields for a two particles microstructure are shown in Fig. 4.2 and 4.3.

The positions of free energy minima at desired values of density ρ and orientation fields η are obtained by changing of terms associated with constants A and B in eq. 4.2. The term



4.2: Variation of density field ρ in vapor and solid phases.



4.3: Variation of orientation field η in vapor and solid phases for two particle case

associated with A is chosen such that two minima exist at density field value $\rho = rho_{vap}$ and $\rho = 1 - V_c$. The term associated with B is related to the effect of the grain boundaries. This term leads to a deviation in the values of density field ρ at minima. The values of density field ρ at minima can also be manipulated by changing the values of ρ_{vap} and V_c in eq. 4.2. The equilibrium values of density field ρ are determined by drawing a common tangent to the free energy curve as shown in Fig. 4.2. The term associated with constant B is chosen such that free energy minima occur at orientation field value $\eta_i = 0$ and $\eta_i = 1$ as shown in Fig. 4.3. Fig. 4.3 shows that bulk free energy in the vapor phase ($\rho \approx \rho_{vap}$) is minimized when all of orientation fields $\eta_i, i \in [1\ p]$ approach zero. The free energy inside the solid phase ($\rho \approx 1 - V_c$) is minimized when only one of the orientation fields approaches unity and rest of them approach zero.

4.1.2 Diffusion coefficients

The phase field simulation can be applied with a variable diffusion coefficient in the microstructure. With this approach, diffusion coefficient acquires different value in solid phase, vapor phase, surfaces and grain boundaries. A variable diffusion coefficient can be calculated using the local density field ρ and orientation field η. The following expression for diffusion coefficient was developed in this dissertation.

$$D = D_{vol}\phi + D_{vap}(1-\phi) + D_{surf}\rho^2(1-\rho)^2 + D_{gb}\rho(1-\sum_{i}^{p}\eta_i^2) \tag{4.3}$$

where D_{vol}, D_{vap}, D_{surf}, and D_{gb} are the parameters that can be adjusted to achieve the desired value of diffusion coefficient in solid phase, in vapor phase, along the vapor solid interphase and along the grain boundary, respectively. The function ϕ in eq. 4.3 is defined as $\phi = \rho^4(7\rho^2 - 18\rho + 12)$. The function ϕ acquires maximum in solid phase and minimum in vapor phase. The variable diffusion coefficient approach in eq. 4.3 allows diffusion coefficient D to acquire different values in different regions of microstructure. For example, diffusion coefficient will transition between D_{vap} to D_{vol} across the particle surface. This transition can be inferred from eq. 4.3. For example, the value of ρ, η_i, ϕ approach zero in vapor phase. A substituting these values in eq. 4.3 results in $D \approx D_{vap}$. The values of parameters D_{vol}, D_{vap}, D_{surf}, and D_{gb} should chosen such that diffusion coefficient D represent diffusion in a real microstructure.

4.1.3 Simulation kinetics

The simulation kinetics pertain to temporal evolutions of density field ρ and orientation field η. The temporal evolutions of these fields are obtained by using Cahn Hilliard equation and Ginzburg-Landau equation. Cahn Hilliard equation [47,48] which relates diffusional flux to the chemical potential is used to obtain temporal evolution of density field ρ.

$$\vec{J} = -D\nabla\mu = -D\nabla\frac{\delta F}{\delta\rho} \tag{4.4}$$

Using Cahn Hilliard equation (eq. 4.4) and continuity equation ($\partial\rho/\partial t = -\nabla.\vec{J}$), the temporal evolution of the density field ρ can be calculated as:

$$\begin{aligned}\frac{\partial\rho}{\partial t} &= \nabla.\left(D\nabla\frac{\delta F}{\delta\rho}\right) \\ &= \nabla.\left(D\nabla\left(\frac{\partial f(\rho,\eta_{1..p})}{\partial\rho} - \kappa_\rho\nabla^2\rho\right)\right)\end{aligned} \tag{4.5}$$

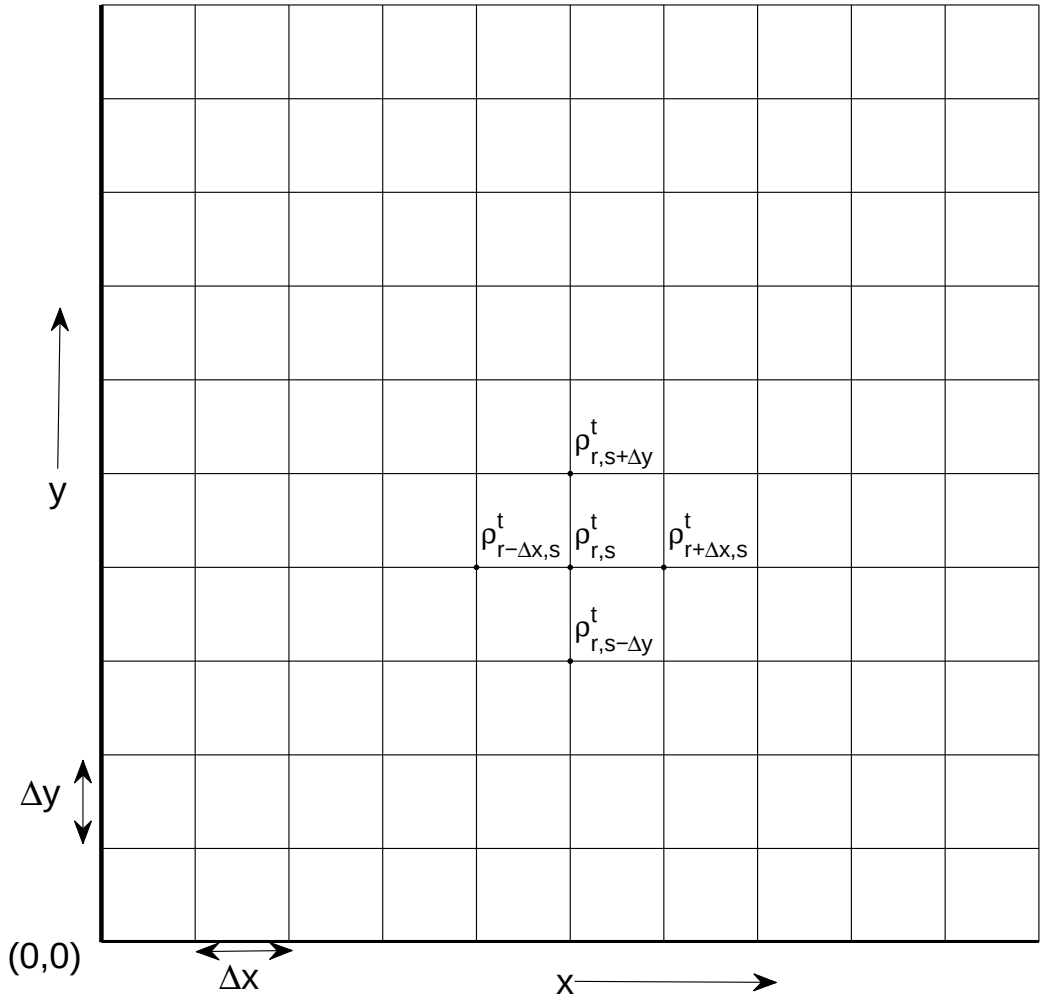
The temporal evolution of the orientation field is calculated using the time dependent Ginzburg-Landau equation given in eq. 4.6 [55].

$$\begin{aligned}\frac{\partial\eta_i}{\partial t} &= -L\frac{\partial F}{\partial\eta_i} \\ &= -L\left(\frac{\partial f(\rho,\eta_{1..p})}{\partial\eta_i} - \kappa_\eta\nabla^2\eta_i\right)\end{aligned} \tag{4.6}$$

where L is a kinetic coefficient. The kinetics equations described above are solved numerically to simulate the microstructure evolution.

4.1.4 Numerical solution of equations

The phase field method for simulating the sintering is implemented in two dimensions in this dissertation. The microstructure in two dimensions can be described by a square lattice with uniform spacing as shown in Fig. 4.4.



4.4: Discretized microstructure for phase field simulations

Each lattice point the square lattice is associated with density field ρ, and orientation fields η_i. The initial values of the phase fields ρ and η_i can be determined from the initial microstructure. The evolution of the microstructure is determined by the changes in the phase fields ρ and η_i with time. The changes in the phase fields ρ and η_i are determined by numerically solving the kinetic equations (eq. 4.5, 4.6) on the square lattice.

The kinetic equations (eq. 4.5, 4.6) can be solved using discrete derivatives obtained from Taylor series. Taylor series in two dimensions can be given as:

$$\begin{aligned} f(x+\Delta x, y+\Delta y) = f(x,y) & \\ + \frac{1}{1!}\left[\Delta x f_x(x,y) + \Delta y f_y(x,y)\right] & \\ + \frac{1}{2!}\left[(\Delta x)^2 f_{xx}(x,y) + 2\Delta x \Delta y f_{xy}(x,y) + (\Delta y)^2 f_{yy}\right] & \\ + \frac{1}{3!}\left[(\Delta x)^3 f_{xxx}(x,y) + 3(\Delta x)^2 \Delta y f_{xxy}(x,y) + 3\Delta x (\Delta y)^2 f_{xyy}(x,y)\right] & \\ + e \quad (4.7) & \end{aligned}$$

where $f_x(x,y)$ is $\partial f(x,y)/\partial x$, $f_y(x,y)$ is $\partial f(x,y)/\partial y$, $f_{xy}(x,y)$ is $\partial f(x,y)/\partial x \partial y$ and so on. In eq. 4.7, e contains higher order terms of Taylor series. For a small Δx and Δy, the higher order terms can be neglected. From Taylor series in eq. 4.7, it follows that:

$$f(x+\Delta x, y) = f(x,y) + \Delta x f_x(x,y) + \frac{1}{2!}(\Delta x)^2 f_{xx}(x,y) + \frac{1}{3!}(\Delta x)^3 f_{xxx}(x,y) + e \quad (4.8)$$

$$f(x-\Delta x, y) = f(x,y) - \Delta x f_x(x,y) + \frac{1}{2!}(\Delta x)^2 f_{xx}(x,y) - \frac{1}{3!}(\Delta x)^3 f_{xxx}(x,y) + e \quad (4.9)$$

The value of second derivative $f_{xx}(x,y)$ can be determined from eq. 4.8 and 4.9 as

$$f_{xx}(x,y) = -2f(x,y) + \frac{f(x+\Delta x, y) + f(x-\Delta x, y)}{(\Delta x)^2} \quad (4.10)$$

Similarly,

$$f_{yy}(x,y) = -2f(x,y) + \frac{f(x,y+\Delta y) + f(x,y-\Delta y)}{(\Delta y)^2} \tag{4.11}$$

The numerical first derivatives can be obtained as:

$$f_x^+ x,y = \frac{f(x+\Delta x,y) - f(x,y)}{\Delta x} \tag{4.12}$$

$$f_x^- x,y = \frac{f(x,y) - f(x-\Delta x,y)}{\Delta x} \tag{4.13}$$

$$f_y^+ x,y = \frac{f(x,y+\Delta y) - f(x,y)}{\Delta y} \tag{4.14}$$

$$f_y^- x,y = \frac{f(x,y-\Delta y) - f(x,y)}{\Delta x} \tag{4.15}$$

The superscripts $+$, $-$ in eq. 4.12-4.15 refer to forward and backward derivatives. The terms higher than the second order in Taylor series were neglected in deriving first derivatives in eq. 4.12-4.15.

The derivatives described above can be used to transform the partial differential equation into algebraic equations. The microstructure evolution can be obtained by solving these algebraic equations. The temporal evolution of the density field ρ is calculated from kinetic equation 4.5. Firstly, the term with the Laplacian operator ∇^2 can be calculated as:

$$\begin{aligned} g_{r,s}^t &= \frac{\partial f_{r,s}^t}{\partial \rho} - \kappa_\rho \nabla^2 \rho_{r,s}^t \\ &= \frac{\partial f_{r,s}^t}{\partial \rho} - \kappa_\rho \left(\frac{\partial^2 \rho_{r,s}^t}{\partial x^2} + \frac{\partial^2 \rho_{r,s}^t}{\partial y^2} \right) \\ &= \frac{\partial f_{r,s}^t}{\partial \rho} - \kappa_\rho \left(-4\rho_{r,s}^t + \frac{\rho_{r+\Delta x,s}^t + \rho_{r-\Delta x,s}^t}{(\Delta x)^2} + \frac{\rho_{r,s+\Delta y}^t + \rho_{r,s-\Delta y}^t}{(\Delta y)^2} \right) \end{aligned} \tag{4.16}$$

The subscripts and superscripts in eq. 4.16 refer to position and time, respectively. The same convention of subscripts and superscripts is used in the following equations.

Secondly, the value of $g_{r,s}$ at all lattice points in the microstructure obtained from eq. 4.16 is used in calculating the temporal evolution of the density field ρ. The kinetic eq. 4.5 can be written as follows:

$$\begin{aligned}\frac{\partial \rho^t_{r,s}}{\partial t} &= \nabla.(D^t_{r,s}\nabla g^t_{r,s}) \\ &= \nabla D^t_{r,s}.\nabla g^t_{r,s} + D^t_{r,s}\nabla^2 g^t_{r,s} \\ &= \frac{\partial D^t_{r,s}}{\partial x}\frac{\partial g^t_{r,s}}{\partial x} + \frac{\partial D^t_{r,s}}{\partial y}\frac{\partial g^t_{r,s}}{\partial y} + D^t_{r,s}\left(\frac{\partial^2 g^t_{r,s}}{\partial x^2} + \frac{\partial^2 g^t_{r,s}}{\partial y^2}\right)\end{aligned} \tag{4.17}$$

Eq. 4.17 can be numerically solved using derivatives in eq. 4.8 - 4.15 as:

$$\begin{aligned}\rho^{t+\Delta t}_{r,s} =& \rho^t_{r,s} + \frac{\Delta t}{2}\left[\frac{D^t_{r+\Delta x,s}(g^t_{r+\Delta x,s} - g^t_{r,s})}{\Delta x^2} + \frac{D^t_{r-\Delta x,s}(g^t_{r-\Delta x,s} - g^t_{r,s})}{\Delta x^2}\right. \\ &+ \frac{D^t_{r,s+\Delta y}(g^t_{r,s+\Delta y} - g^t_{r,s})}{\Delta y^2} + \frac{D^t_{r,s-\Delta y}(g^t_{r,s-\Delta y} - g^t_{r,s})}{\Delta y^2} \\ &\left.+ D^t_{r,s}\left(\frac{-2\rho^t_{r,s} + \rho^t_{r+\Delta x,s} + \rho^t_{r-\Delta x,s}}{\Delta x^2} + \frac{-2\rho^t_{r,s} + \rho^t_{r,s+\Delta y} + \rho^t_{r,s-\Delta y}}{\Delta y^2}\right)\right]\end{aligned} \tag{4.18}$$

Similarly, the temporal evolution of orientation field η can be calculated as:

$$\begin{aligned}&\eta^{t+\Delta t} i\ r,s = \eta^t_{i\ r,s} + \Delta t L\left(\frac{\partial f^t_{r,s}(\rho,\eta_i)}{\partial \eta_i}\right. \\ &\left.+ \kappa_\eta\left(\frac{-2\eta^t_{i\ r,s} + \eta^t_{i\ r+\Delta x,s} + \eta^t_{i\ r-\Delta x,s}}{\Delta x^2} + \frac{-2\eta^t_{i\ r,s} + \eta^t_{i\ r,s+\Delta y} + \eta^t_{i\ r,s-\Delta y}}{\Delta y^2}\right)\right)\end{aligned} \tag{4.19}$$

The temporal evolution of the phase fields ρ and η in eq. 4.18 - 4.19 utilizes Euler's explicit scheme of solving partial differential scheme. The time step (Δt) and distance step (Δx) required for accurate results in the explicit schemes should be small [56]. The time steps in explicit schemes are constrained by distance steps for solving Cahn-Hilliard equation such that:

$$\Delta t \approx (\Delta x)^4 \tag{4.20}$$

An implicit scheme of solving partial differential equations usually allows larger time steps. The implicit scheme also improves the stability of the simulations. The application of the implicit scheme in this study requires long computational time due to use of a variable diffusion coefficient. Therefore, the implicit scheme could not be used in this study.

4.2 Development and applications

Phase field model was primarily developed by Cahn and Hilliard for estimating the interface energy [47]. Chan derived the kinetics equations to extend the model to describe the spinodal decomposition [48]. The phase field method of Cahn and Hilliard has been modified to simulate microstructural evolution including cases of grain growth and sintering [52, 57].

Cahn and Hilliard assumed that intensive properties such as composition and density are spatially distributed in a system [47]. These properties of the system were assumed to be varying smoothly across the interface. They referred such a system as being a nonuniform system and interface as being a diffuse interface. Cahn expressed the total free energy F of the system as:

$$F = N_v \int_v [f_\circ(c) + k(\nabla c)^2 + ...]\mathrm{d}V \tag{4.21}$$

where $f_\circ$ is the free energy per molecule of a solution of with uniform composition c, N_v is the number of molecules per unit volume in a system with volume V. The term κ known as gradient energy coefficient is defined as:

$$\kappa = -\left(\frac{\partial^2 f(c)}{\partial c \partial \nabla^2 c}\right)_\circ + \left(\frac{\partial^2 f}{\partial^2 \mid \nabla c \mid}\right)_\circ \tag{4.22}$$

where f the local free energy per molecule of the system. The subscript $\circ$ refers to quantities at equilibrium. While deriving eq. 4.21, it is assumed that the composition gradient is large compared to intermolecular distance and that concentration c and its derivatives are independent variables. Cahn pointed out from eq. 4.21 that the total free energy of a small

volume of a nonuniform system can be expressed as a sum of two terms: (1) bulk free energy term ($f_\circ$) and (2) gradient energy ($k(\nabla c)^2 + ...$). The first term refers to bulk free energy far away from interfaces. The gradient energy term which refers to interfacial energy is non zero within interfaces. Cahn also derived interface energy σ of an interface between phase A and B in a binary system as:

$$\sigma = N_v \int_{-\infty}^{+\infty} \left[\Delta f(c) + \kappa \left(\frac{\mathrm{d}c}{\mathrm{d}x} \right) \right] \mathrm{d}x \tag{4.23}$$

where $\Delta f(c)$ is given by

$$\begin{aligned} \Delta f(c) &= f_\circ(c) - [c\mu_B^{(e)} + (1-c)\mu_A^{(e)}] \\ &= c[\mu_B^{(c)} - \mu_B^{(e)}] + (1-c)[\mu_A^{(c)} - \mu_A^{(e)}] \end{aligned} \tag{4.24}$$

In eq. 4.24, superscripts c and e refer to chemical potentials at interface and at equilibrium, respectively. If equilibrium compositions of phases A and B are c_α and c_β, the interface energy can be expressed as:

$$\sigma = 2N_v \int_{c_\alpha}^{c_\beta} \sqrt{\kappa \Delta f(c)} \mathrm{d}c \tag{4.25}$$

Cahn and Hilliard expressed the interface profile x and the interface width l as:

$$x = \int_{c_\alpha}^{c_\beta} \sqrt{\frac{\kappa}{\Delta f}} \mathrm{d}c \tag{4.26}$$

$$l \cong \frac{c_\beta - c_\alpha}{(\mathrm{d}c/\mathrm{d}x)_c} = (c_\beta - c_\alpha) \sqrt{\frac{\kappa}{\Delta f_{max}}} \tag{4.27}$$

Cahn and Hilliard derived the dependence of gradient energy coefficient κ and excess free energy Δf_{max} on interface width given by eq. 4.27. The interfaces with higher *kappa* and

small Δf_{max} are expected to be wider. The above analysis is based upon the assumption that the interface width l is much larger than intermolecular distance.

For deriving the kinetics of spinodal decomposition, Cahn expressed the total energy of the system in terms of concentration c as [48]:

$$F = \int_v \left[f_\circ(c) + \frac{\eta^2 E}{1-\nu}(c - c_\circ)^2 + \kappa(\nabla c)^2 \right] \mathrm{d}V \tag{4.28}$$

Cahn derived the flux $\vec{J}$ from total free energy such that:

$$\vec{J} = -M\nabla \left[\frac{\partial f_\circ}{\partial c} - \frac{\partial \kappa}{\partial c}(\nabla c)^2 - 2\kappa\nabla^2 c \right] \tag{4.29}$$

where$M$ is a positive quantity. Cahn obtained the temporal evolution of concentration c with Fick's second law such that [48]:

$$\frac{\partial c}{\partial t} = -\nabla.\vec{J} = -\nabla.\left(M\nabla \left[\frac{\partial f_\circ}{\partial c} - \frac{\partial \kappa}{\partial c}(\nabla c)^2 - 2\kappa\nabla^2 c \right] \right) \tag{4.30}$$

Cahn and Hilliard later showed that the temporal evolution for any differentiable function F of composition c and derivations can be expressed as [58]:

$$\frac{\partial c}{\partial t} = -\nabla.\vec{J} = \nabla.\left[M\nabla \left(\frac{\delta F}{\delta c} \right) \right] \tag{4.31}$$

The temporal evolution described above can be applied to variables which are conserved over the volume of system. Density or concentration are examples of such conserved variables. In certain cases, the total free energy F also depends upon nonconserved variables such as crystal orientations. Allen and Cahn used an order parameter approach to the describe a migration of antiphase boundary between different crystal orientations [28]. They described the free energy per unit volume $f_\circ$ of a homogeneous phase as a function of a long range order parameter η. They assumed a symmetric double well potential for free energy $f_\circ$ below order

disorder transition temperature. Allen and Cahn considered two states that order parameters can acquire η' and $-\eta'$. In equilibrium the system achieves state such that:

$$\frac{\partial f_\circ}{\partial \eta} = 0 \tag{4.32}$$

Allen and Cahn suggested that the equilibrium order parameters will be of equal magnitude and opposite sign ($+\eta_e$ and $-\eta_e$) for the chosen free energy function. The described the antiphase boundaries as regions where order parameter has values between $-\eta_e$ and $+\eta_e$. Allen and Cahn introduce the function $\Delta f_\circ$ to discuss the free energy of the interfaces. They defined $\Delta f_\circ$ as the free energy difference between a homogeneous state of an arbitrary order parameter and that with $\eta = \pm\eta_e$. They described the total energy F of the system in terms of $\Delta f_\circ$ and order parameter η as:

$$\Delta F = \int_v \Delta f_\circ + 2\kappa(\nabla\eta)^2 \mathrm{d}Vs \tag{4.33}$$

where κ is called the gradient energy coefficient. Allen and Cahn derived the expression for the boundary width l and energy σ as:

$$l = 2\eta_e \sqrt{\frac{\kappa}{(\Delta f_\circ)_{max}}} \tag{4.34}$$

$$\sigma = \int_{-\eta_e}^{+\eta_e} 2\sqrt{\kappa \Delta f_\circ} \mathrm{d}\eta \tag{4.35}$$

Allen and Cahn derived the following expression for temporal evolution of order parameter η:

$$\frac{\partial \eta}{\partial t} = -\alpha \frac{\delta F}{\delta \eta} = -\alpha \frac{\partial \Delta f_\circ}{\partial \eta} + 2\kappa\alpha\nabla^2\eta \tag{4.36}$$

where α is a positive kinetic coefficient. The formulation described above has been used to determine the microstructural evolution in several studies. Some of these studies are described below.

Fan *et al.* analyzed the effect of grain boundary width and number of order parameters on grain growth kinetics [59, 60]. They showed that the grain growth kinetics can be accurately simulated with a large but finite number of order parameters or crystal orientations η. They also showed that the grain growth kinetics slows down with insufficient grid points in simulation to resolve the grain boundaries. They found the grain growth kinetics to be independent of grain boundary width when grain boundary contained more than seven grid points. With seven grid points, the motion of the grain boundary was found to be identical to its sharp interface limit. Fan *et al.* compared their phase field simulation results with grain growth kinetics obtained from the Monte Carlo method [60]. A good agreement between the two methods was found.

Ma *et al.* applied the phase field method to simulation of grain growth [61]. They examined the effect of anisotropy in grain boundary mobility and energy. They also studied the effect of initial microstructure on evolution of texture component. They used Read-Shockley formula for small angle boundaries [62] to apply grain boundary energy anisotropy. They followed approached of Huang and Humphreys for anisotropy in grain boundary mobility [63]. The initial microstructure contained a small fraction of grain with cube texture embedded in randomly oriented grains. Ma *et al.* observed that the fraction of texture component increases when grain boundary energy is anisotropic; however, the texture component decreases when mobility is anisotropic with isotropic energy. They also found the texture growth to be decelerating whereas the texture reduction accelerated with time. Ma *et al.* simulated grain growth with different distributions of cube textured grains to study the texture evolution. They found grain boundary energy density γ_{gb}/d to be the parameter controlling the texture evolution. Here, γ_{gb} is the specific grain boundary energy and d is the mean grain size in a cluster of textured grains. They observed that the fraction of texture component increases when γ_{gb}/d of randomly oriented grains is greater than γ_{gb}/d of textured grains and vice-versa.

Ko *et al.* implemented the phase field method in three dimensions (3D) to simulate the grain growth [64]. They studied the effect of grain boundary energy anisotropy on abnormal grain growth. They considered a presence of two types of grain boundaries with energy values of 1.0 and 0.3. They observed a normal grain growth in absence of grain boundaries with energy of 0.3. Ko *et al.* also observed that the grain boundary energy anisotropy results in abnormal grain growth.

Fan *et al.* employed the phase field method to simulate Ostwald ripening in high volume fraction regime [51]. The Ostwald ripening was simulated in two dimensions to study the effect of volume fraction of solid on coarsening kinetics. They used the conserved field approach of Cahn and Hilliard [47] to obtain the temporal evolution of concentration. However, the temporal evolution of noncoserved order parameters η describing the crystal orientation of solid phase was determined using Allen-Cahn kinetics [28]. The free energy functional for the simulation was chosen such that grain boundary energy to interface energy ratio of 2.14 was obtained. This energy ratio ensured the presence of liquid phase sintering (cf. eq. 3.7). The coarsening kinetic coefficient k and exponent m in eq. 4.37 were obtained from simulation results.

$$R_t^m - R_0^m = kt \tag{4.37}$$

where t refers to sintering time and R_t, and R_0 are the mean grain size at time t and at time 0, respectively. Fan *et al.* found a nice fit of power law in eq. 4.37 to coarsening data with exponent m of 3.0. The coarsening exponent m did not vary significantly even for high volume fraction (>90%) of coarsening phase. Fan *et al.* found the kinetic coefficient k to be strongly dependent upon the volume fraction of coarsening phase. The value of kinetic coefficient k increased from 0.833 for volume fraction of 0.25 to 24.45 for volume fraction of 0.90. The rapid increase in the kinetic coefficient k was attributed to dramatic decrease in diffusion distance of atoms in the matrix phase at higher volume fraction of coarsening phase. They also compared the simulation results with coarsening experiments on Fe-Cu, Pb-Sn and Sn-Pb systems [65]. The comparision confirmed the obtained trend in kinetic coefficient k.

Chen and Fan applied the phase field model to coupled grain growth and Ostwald ripening in two phase Al_2O_3-ZrO_2 system [66]. The grain boundary to interface energy ratio for the simulation was chosen to produce experimentally obtained energy ratio. The simulation was performed with varying volume fractions of phases. The simulated microstructures appeared to have striking resemblance to experimentally observed microstructures [67]. The phase field model predicted the main features of coupled grain growth and Ostwald ripening, as observed experimentally in Ref [67]. In a microstructure containing low volume fraction of zirconia, zirconia grains were found to be located at grain boundaries and trijunctions. The coarsening for such a microstructure was controlled by Ostwald ripening process. The motion of alumina grain boundaries was pinned by zirconia grains, and the alumina grain size was

limited by location and distribution of zirconia grains. The migration and readjustment of alumina grain boundaries was observed after the disappearance of zirconia grain. The growth of alumina grains decreased with higher volume fraction of zirconia phase. Chen and Fan argued that the reduction in grain growth was due to increased pinning sites.

Aps and Agren developed a phase field simulation method using a vacancy diffusion approach [68]. They treated solid phase as a region with low vacancy concentration and pore as region of high vacancy concentration. The surface of the solid phase was characterized by a continuous variation in vacancy concentration. Aps and Agren used thermodynamics of vacancies to describe the free energy for the simulation. The temporal evolution of vacancies was calculated using Cahn Hilliard and Allen Cahn kinetics. The kinetic equations were numerically solved using finite element method.

Kumar et al. [69] applied phase field simulation method to sintering of unequal sized particles. They found that sintering of unequal sized particles occur in three subprocesses: (1) neck growth, (2) coarsening, and (3) grain boundary migration. They also found a slow grain boundary migration concurrent with coarsening. This slow migration was neither observed nor proposed earlier.

Wang argued that a rigid body motion is required for simulating the solid state sintering [53]. Wang incorporated the rigid body motion into phase field equations of Chan-Hilliard [47] and Allen-Cahn [28]. Wang defined the total flux $\vec{J}$ for sintering as the sum of diffusion flux $\vec{J}_{diff}$ and advection flux $\vec{J}_{adv}$ from rigid body motion. Wang set up the rigid body motion such that it was only active when density inside the grain boundary deviates from equilibrium density. The advection force of rigid body motion tries to restore the equilibrium density inside the grain boundary. Wang's formulation incorporated Newton's third law of motion to ensure zero force on the sample. Wang also argued that the diffusion coefficient in the microstructure is not constant. He treated diffusion coefficient as a variable that acquires different values at grain boundary, particle surface, inside solid and in vapor phases. The model with rigid body motion and variable diffusion coefficient was applied to sintering of two equal particles. The neck growth kinetics were obtained in terms of exponent n in eq. 4.38.

$$\left(\frac{X}{D}\right)^n = kt \tag{4.38}$$

where X is the neck size, D is the particle diameter, n is the neck growth exponent, k is the neck growth coefficient and t is the time. Wang calculated the neck growth exponent n for sintering by grain boundary diffusion, volume diffusion, surface diffusion and vapor transport. A satisfactory agreement between obtained exponents and predictions of theoretical models was found. Wang also applied the model to the sintering of multiple particles. The model showed several realistic microstructural evolution phenomena such as grain boundary migration and pore migration.

PART II

RESULTS AND DISCUSSION

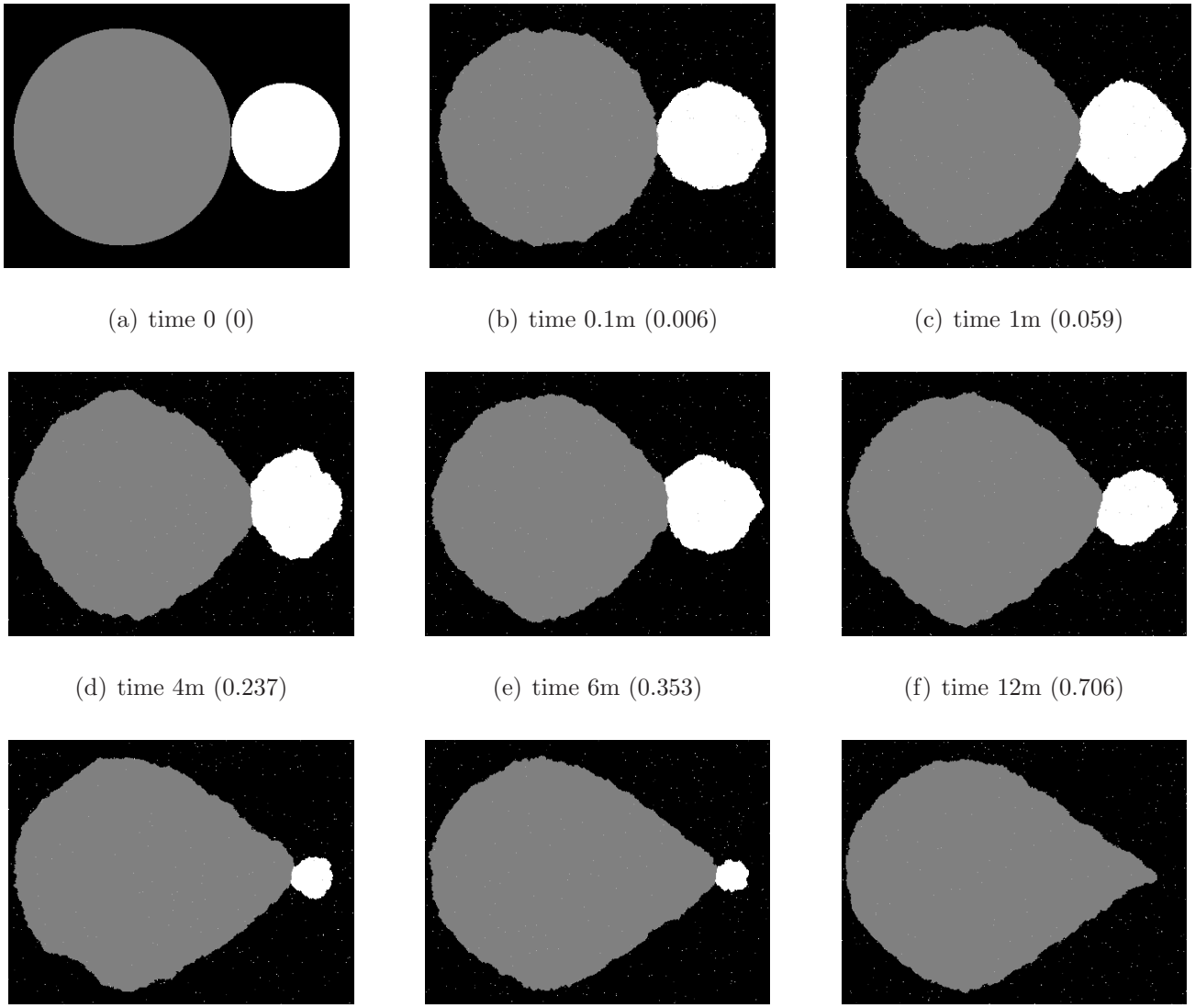
CHAPTER 5

MONTE CARLO SIMULATION

The Monte Carlo simulation method described in Section 3.1 was applied to the sintering of two unequal sized particles. The sintering simulation was performed in two dimensions (2D) on a 512×512 uniform grid of points. Sintering started with two touching circles of radii 128 pixels and 64 pixels. The sintering of circles in two dimensions is equivalent to sintering of long cylindrical particles in three dimensions. The large and small particles were assigned with spin state Q of 1 and 2, respectively. The total number of possible spin states Q was taken as 5. The values of the grain boundary energy and the surface energy per unit area were assumed to be $0.6kT$ and $1.0kT$, respectively. The simulations were performed with a thermal energy of $0.7kT$ until the small particle disappeared.

Figure 5.1 shows the simulation results for two unequal sized particles. The change in the sintering geometry is quantitatively shown in Fig. 5.2 on a relative scale. The geometrical parameters and simulation time are rescaled in [0 1] in Fig. 5.2. The initial configuration of the sintering simulation consisted of two touching particles as shown in Fig. 5.1(a). A neck between the particle form in 0.6% of total sintering time. The neck growth was rapid initially but slowed down with increasing neck size. It can be observed from Fig. 5.1(c) that the neck growth slowed at relative time 0.059. It should be noted that a negligible particle size change occured until the neck was grown significantly. The coarsening was found to be accompanied with a slow grain boundary migration. The coarsening and slow grain boundary migration remained active mass transport mechanisms from relative time 0.059 to 0.9. The grain boundary started migrating rapidly at relative time 0.9. The rapid grain boundary migration resulted in a rapid shrinkage rate of the small particle. The sintering process was completed by coalescence of the small particle.

The results from the Monte Carlo simulation method show that the sintering of the two unequal particles involves three subprocesses. The three subprocesses are: (1) neck growth, (2) coarsening, and (3) grain boundary migration. The mechanisms mass transport in coarsening and grain boundary migration are explained in Section 2.3. The three subprocesses overlap to some extent. In the first subprocesses of neck growth, a neck and a grain boundary

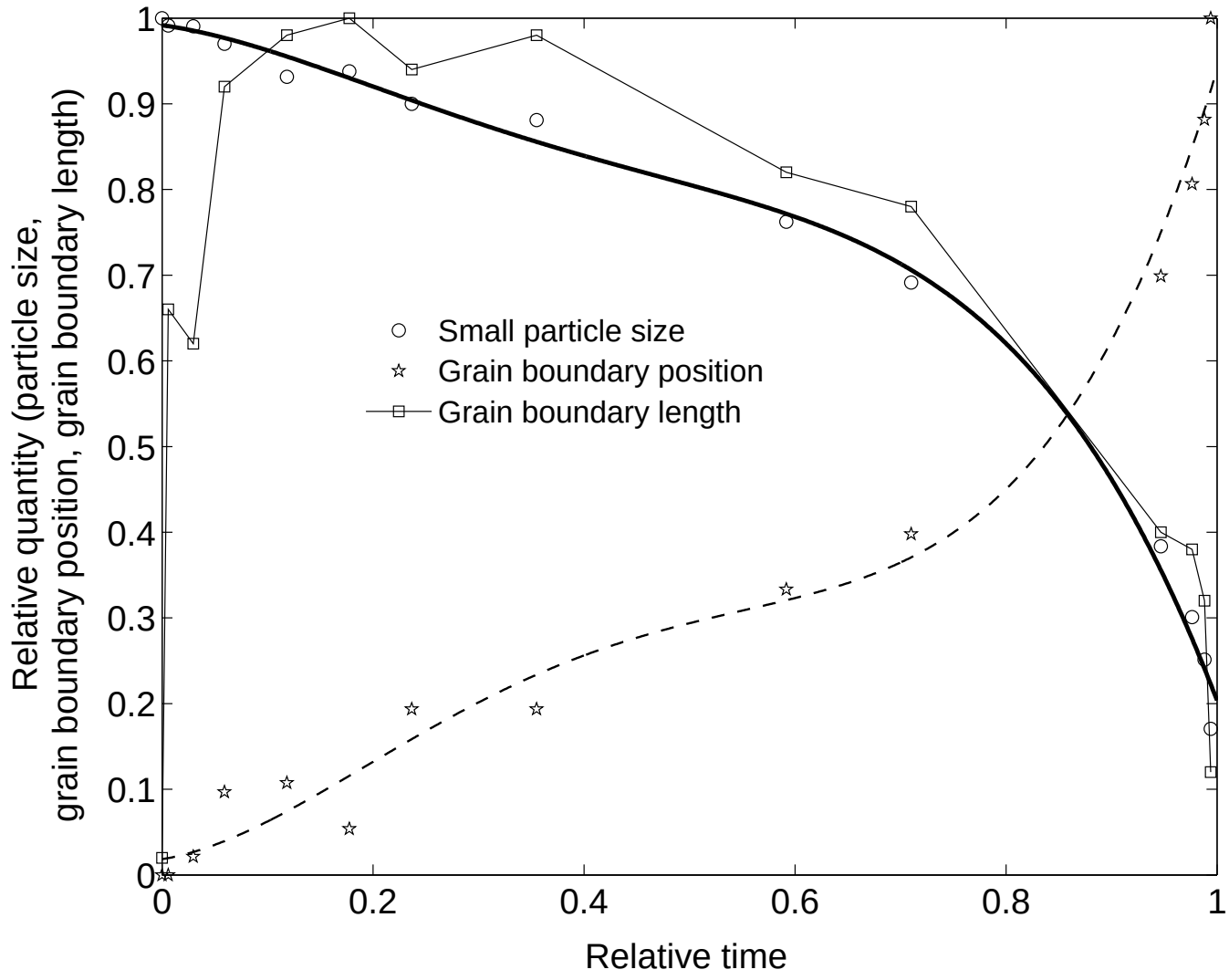


(a) time 0 (0) (b) time 0.1m (0.006) (c) time 1m (0.059)

(d) time 4m (0.237) (e) time 6m (0.353) (f) time 12m (0.706)

(g) time 16m (0.947) (h) time 16.5m (0.976) (i) time 16.9m (1.0)

5.1: Monte Carlo simulations of two particle sintering. The number below images refers to sintering time in Monte Carlo steps (m refers to million). Time in parentheses indicates relative time to sinter.



5.2: The geometrical changes during two unequal sized particles sintering using the Monte Carlo simulation method

forms. The neck formation and growth is rapid and takes a relatively short time as compared to the total sintering. A negligible coarsening occurs while neck grows significantly. A noticeable coarsening occurs when neck is fully formed. The particle size changes during the coarsening subprocess. The coarsening causes the large particle to grow and the small particle to shrink. The coarsening is a relatively slow process as compared to the neck growth. The rate of coarsening increases as coarsening progresses. A slow grain boundary accompanies the coarsening. The grain boundary migrates rapidly towards the end of the coarsening subprocess. The kinetics of the rapid grain boundary migration are faster than the coarsening kinetics. The sintering of two unequal particles was completed when small particle disappeared due to the grain boundary migration. The simulation results show that the coarsening is the slowest among the three subprocesses of sintering. Therefore, the coarsening is the rate determining subprocess during the sintering process.

After performing sintering simulations with different specific surface energy, specific grain boundary energy, and sintering temperature, it was found that these parameters affect the neck growth, coarsening and grain boundary migration. The relative ratios of the three subprocesses depend upon the choice of parameters in Monte Carlo simulations. However, the microstructural evolution described above was not found very sensitive to the choice of parameters.

CHAPTER 6

PHASE FIELD SIMULATION

The phase field model described in Section 4.1 was applied to simulate solid state sintering. The simulations were performed for various initial microstructures. These microstructures include: (1) two unequal sized particles, (2) two equal sized particles, (3) closed packed array of particles, (4) multiparticle random microstructures, and (5) isolated pores. The details of the above mentioned microstructural simulations are described in following sections. The results of the microstructural simulations are also discussed.

6.1 Two unequal sized particles

The phase field simulation method described in Section 4.1 was applied to the sintering of two unequal sized particles. The phase field simulation was performed in two dimensions (2D) on a 256×256 regular grid of points. The sintering started with two circular particles making a point contact. Similar to the Monte Carlo simulation method, the sintering of circular particles in 2D is considered to be equivalent to sintering of cylindrical particles in 3D. The initial diameters of the large and the small particles were 120 pixels and 60 pixels, respectively. Two phase fields variables were used to simulate the sintering: (1) density field ρ and (2) orientation field η. Each of the particles was associated with one orientation field. The total free energy of the microstructure was assumed to be comprising of the bulk free energy and surface energy. The total energy of the microstructure was obtained from the phase fields ρ and η given by eq. 6.1.

$$F = \int_v f(\rho, \eta_{1..p}) + \frac{\kappa_\rho}{2}(\nabla\rho)^2 + \sum_{i=1}^{p} \frac{\kappa_\eta}{2}(\nabla\eta_i)^2 \mathrm{d}v \tag{6.1}$$

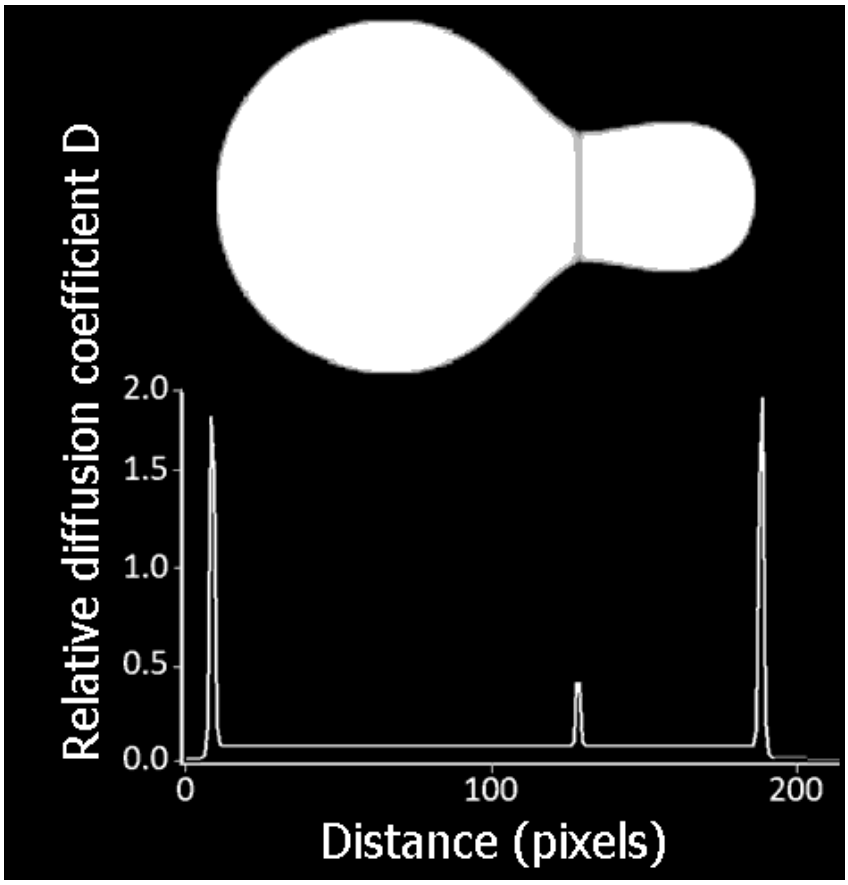
where $f(\rho, \eta_{1..2})$ is the bulk free energy. The values of κ_ρ, κ_η and p were taken as 10, 3.75 and 2, respectively. The expression of bulk free energy of the microstructure is given in eq. 6.2.

$$
\begin{aligned}
f(\rho,\eta_{i..p}) =& A(\rho^4 + \frac{-4\rho_{vap} - 4(1-V_c) - 2}{3}\rho^3 \\
&+ \frac{4\rho_{vap}(1-V_c) + 2\rho_{vap} + 2(1-V_c)}{2}\rho^2 \\
&- 2\rho_{vap}(1-V_c)\rho) \\
&+ B(\rho^2 + 6(1-\rho)\sum_i^p \eta_i^2 - 4(2-\rho)\sum_i^p \eta_i^3 \\
&+ 3\sum_i^p \eta_i^4 + C_{gbe}\sum_i^p \sum_{j,j\neq i}^p \eta_i^2\eta_j^2)
\end{aligned}
\tag{6.2}
$$

For phase field simulation, values of A , B, C_{gbe}, ρ_{vap} and V_c were chosen as 16, 1, 7, 0.009 and 0.001, respectively. The values of these variables was found such that the bulk free energy $f(\rho,\eta_{i..p})$ acquires minima in equilibrium phases. For example, bulk free energy $f(\rho,\eta_{i..p})$ is minimized in solid phase, i.e., when $\rho = 1 - V_c$, $\eta_i = 1$ and $\eta_j = 0$; $j \neq i,\ j \in [1\ p];\ i \leq p$. Similarly, the choice of parameters in eq. 6.2 ensures that the bulk free energy in vapor phase is minimized. This minima of energy occur when $\rho = \rho_{vap}$ and $\eta_i = 0$; $i \in [1\ p]$. The diffusion coefficient D for the simulation was chosen such that it varied with positions in the microstructure. The diffusion coefficient for the simulation is given in eq. 6.3.

$$
D = D_{vol}\phi + D_vap(1-\phi) + D_{surf}(1-\rho)^2 + D_{gb}\rho(1-\sum \eta^2) \tag{6.3}
$$

where $\phi = \rho^4(7\rho^2 - 18\rho + 12)$ and D_{vol}, D_{vap}, D_{surf}, and, D_{bg} are parameters used to adjust the value of diffusion coefficient along bulk, vapor, surface and grain boundary paths, respectively. The values of D_{vol}, D_{vap}, D_{surf}, D_{gb} were chosen as 0.08, 0.012, 45 and 4.1, respectively. The values of parameters in eq. 6.3 is chosen such that diffusivities along various paths follow trends similar to diffusivites of real materials. The diffusion coefficients were calculated using eq. 6.3. The calculated values of D were 2.0 at surface, 0.4 at grain boundary, 0.08 in the particle and 0.016 in vapor phase. The value of diffusivity of vapor transport is an effective coefficient that determines the mass transport via gas phase which contain quantities such as sticking coefficient and vapor pressure. Fig. 6.1 depicts the variation of diffusion coefficient along the a line joining the particle centers in the microstructure. The variation of diffusion coefficient in Fig. 6.1 indicates that diffusion coefficient at surfaces is the highest and lowest in vapor phase. This difference in values of D is expected as we anticipate more mass transport



6.1: Variation of diffusion coefficient D in the microstructure. The values in the graph correspond to diffusion coefficient along a line joining the particle centers. Reprinted with permission from ref. [69]

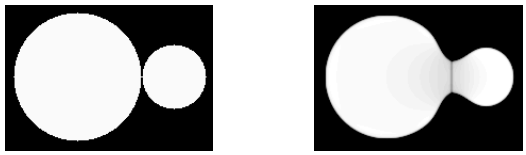
via surface diffusion than vapor transport during sintering of real powders. A similar scheme of variable diffusion coefficient was used by Wang [53].

The phase field simulation for two unequal sized particles was performed with parameters described above. The simulation results are described on a relative time and length scale. The results on simulation time and relative time-length scale are given in Table 6.1. The simulation results shown graphically in Fig. 6.2 depict the geometrical changes during the sintering. The geometrical changes during the sintering are also given in Table 6.1 quantitatively. These geometrical changes are plotted in Fig. 6.3. These geometrical changes (data in Table 6.1) can be used to derive the sintering kinetics in terms of neck growth, particle size change and grain boundary velocity provided in Table 6.2.

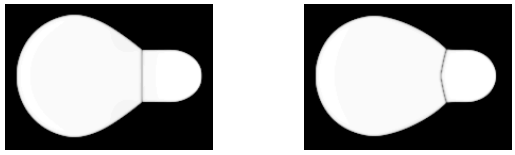
The phase field simulation of two unequal sized particles began with two particles making a point contact as shown in the Fig. 6.2(a). Fig. 6.2(b) shows that the neck and the grain

6.1: Microstructure parameters (in pixels)

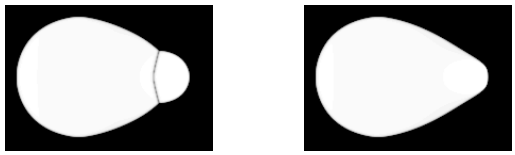
Relative time	Sintering time	Small particle size	Neck size	Grain boundary position
0	0	59.7	0	129
0.01	268	59.7	31	129
0.2	5353	59.0	51	129
0.5	13382	55.1	52	129
0.95	25427	43.0	51	139
1.0	26765	0.0	0	154



(a) time 0 (0) (b) time 268 (0.01)

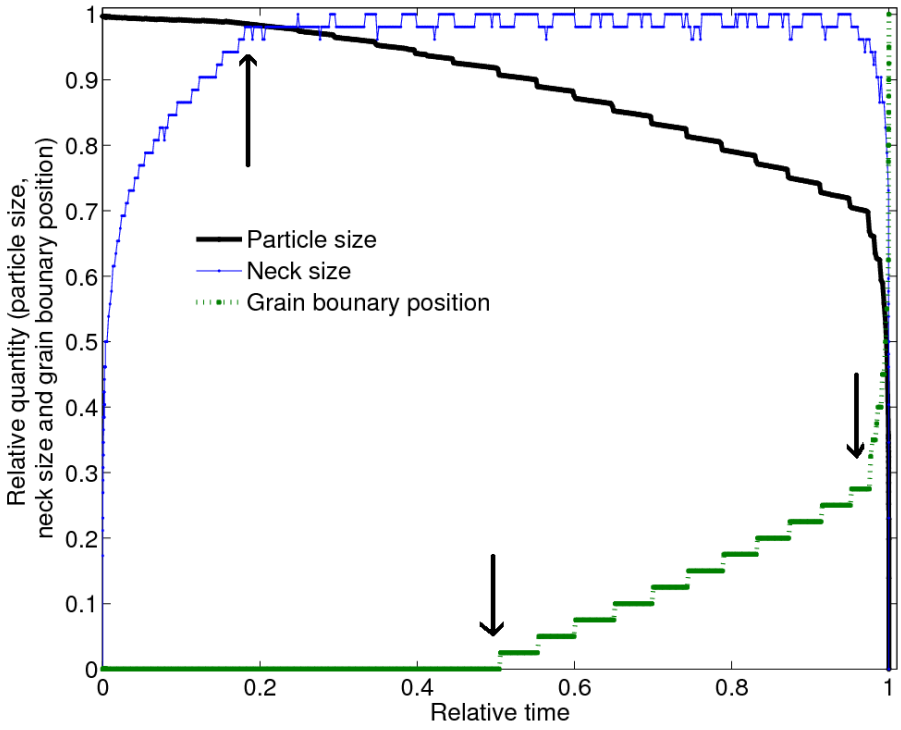


(c) time 5353 (0.2) (d) time 13382 (0.5)



(e) time 25427 (0.95) (f) time 26765 (1.0)

6.2: Microstructure evolution of two unequal size particles during sintering. Time below subfigures indicates sintering time and time in parentheses refers to relative time to sinter. Reprinted with permission from ref. [69].



6.3: Geometrical changes during sintering of two unequal sized particles. The particle size is defined as the radius of particle with equivalent volume. Reprinted with permission from ref. [69]

boundary between the particles formed quickly. The neck formed with a rate of 3100 pixels per unit time in the beginning of the sintering. The rate of neck growth can be observed in Fig. 6.3 which shows that the 60% of the neck grew during 1% of sintering time. The rate of the neck growth slowed down during 0.01-0.20 time to 105 pixels per unit time. The neck growth stopped at time 0.20. The neck size remained the same between 0.20-0.95 time. The neck and the grain boundary shrank rapidly between time 0.95 to 1.00. The rate of neck

6.2: Rate of sintering subprocesses (pixel per unit time)

Time	Stage	Small particle shrinkage rate	Neck growth rate	Grain boundary velocity
0.00-0.01	Rapid neck growth	0.0	3100.0	0
0.01-0.20	Slow neck growth	3.7	105.3	0
0.20-0.50	Coarsening	13.0	3.3	0
0.50-0.95	Coarsening with slow grain boundary migration	26.9	2.2	22.2
0.95-1.0	Rapid grain boundary migration	860.0	-1020.0	300.0

shrinkage between time 0.95-1.00 was 1020 pixels per unit time.

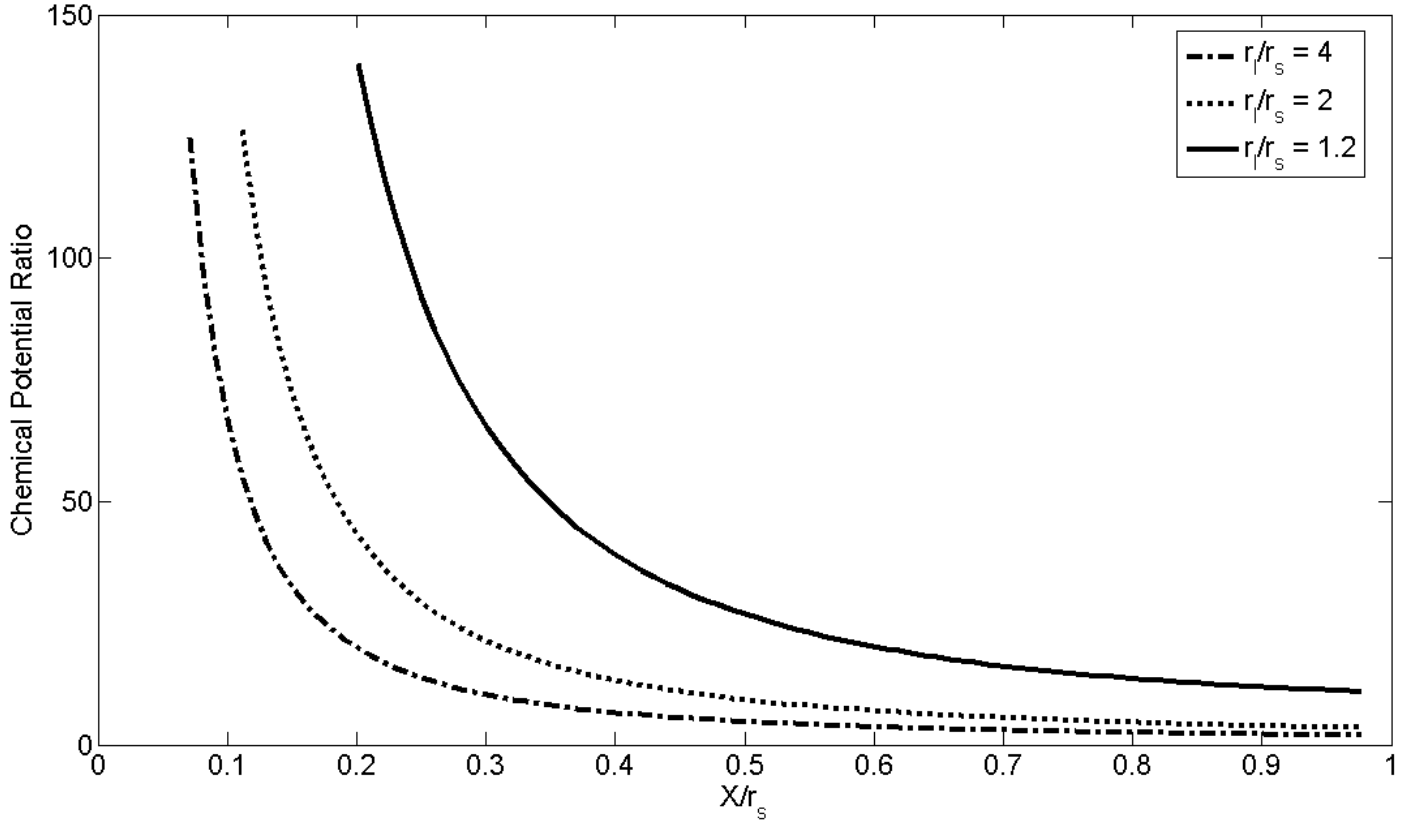
Change in the particles size can be inferred from Fig. 6.3 and Table 6.1. The particle size in Fig. 6.3 and Table 6.1 is defined as radius of circle with equivalent area. It can be inferred that the particle size did not change noticeably until the time 0.2, i.e., until the neck was fully formed. The small particle shrank slowly between time 0.2 to 0.5 with a rate of 13 pixels per unit time. During the next 0.45 time units, the rate of coarsening doubled. The small particle shrank rapidly after 0.95 time units until it disappeared.

A boundary migration can also be observed in the simulation of two unequal particle sintering. Fig. 6.3 shows the position of the grain boundary during sintering. It can be observed that the grain boundary remained stationary until time 0.50. It should be noted that the neck grew to completion and coarsening occurred significantly before time 0.50. The grain boundary migrated slowly between time 0.5-0.95. The mean grain boundary velocity between during this time interval was 22.2 pixels per unit time. The grain boundary migrated rapidly between time 0.95-1.0 with a velocity of 300 pixel per unit time.

Based upon the results of the phase field simulations, the sintering of two unequal particles can be described into three subprocesses: (1) neck growth, (2) coarsening, and (3) grain boundary migration. Approximate transitions between these subprocesses are highlighted by arrow marks in Fig. 6.3. A neck and grain boundary between the particles form in the first subprocess of the neck growth. The driving force $\Delta\mu_n$ for the neck growth is the difference between neck curvature and particle curvature given by eq. 6.4.

$$\Delta\mu_n = \gamma\Omega\left(\frac{1}{r_n} - \frac{1}{r_p}\right) \tag{6.4}$$

where r_n and r_p are radii of curvature at neck and particle surface, respectively. The radius of curvature increases with neck growth. Therefore, the rate of the neck growth reduces as neck grows. The neck and the grain boundary provide a path for the interparticle mass transport for sintering. Therefore, the subprocess of coarsening does not noticeably begins until a significant size of neck is formed. The reason for a negligible coarsening before a significant neck formation could be the difference in chemical potential. The chemical potential difference for coarsening given in eq. 6.5 can be compared with chemical potential difference for neck growth in Fig 6.4.



6.4: Chemical potential ratio for neck growth and coarsening.

$$\Delta\mu_n = \gamma\Omega\left(\frac{1}{r_s} - \frac{1}{r_l}\right) \tag{6.5}$$

In eq. 6.5, r_s and r_l are the radii of the small and the large particle, respectively. Fig. 6.4 shows that the driving force for coarsening is orders of magnitude smaller than driving force for coarsening. Therefore, the coarsening begins slowly after a significant neck growth. There is a small overlap of the neck growth and the coarsening subprocesses. The coarsening subprocess also overlaps with a grain boundary migration. The grain boundary velocity during coarsening is small. The slow grain boundary migration indicates a small driving force for the grain boundary migration. The small driving force can be attributed to drag force due to increase in grain boundary during grain boundary migration. The grain boundary migrates rapidly towards the end of the sintering. A reduction in grain boundary area could result in the rapid migration rate. During this rapid grain boundary migration, a rapid shrinkage of small particle occurs. The sintering of two unequal particle sintering is completed when the small particle disappears.

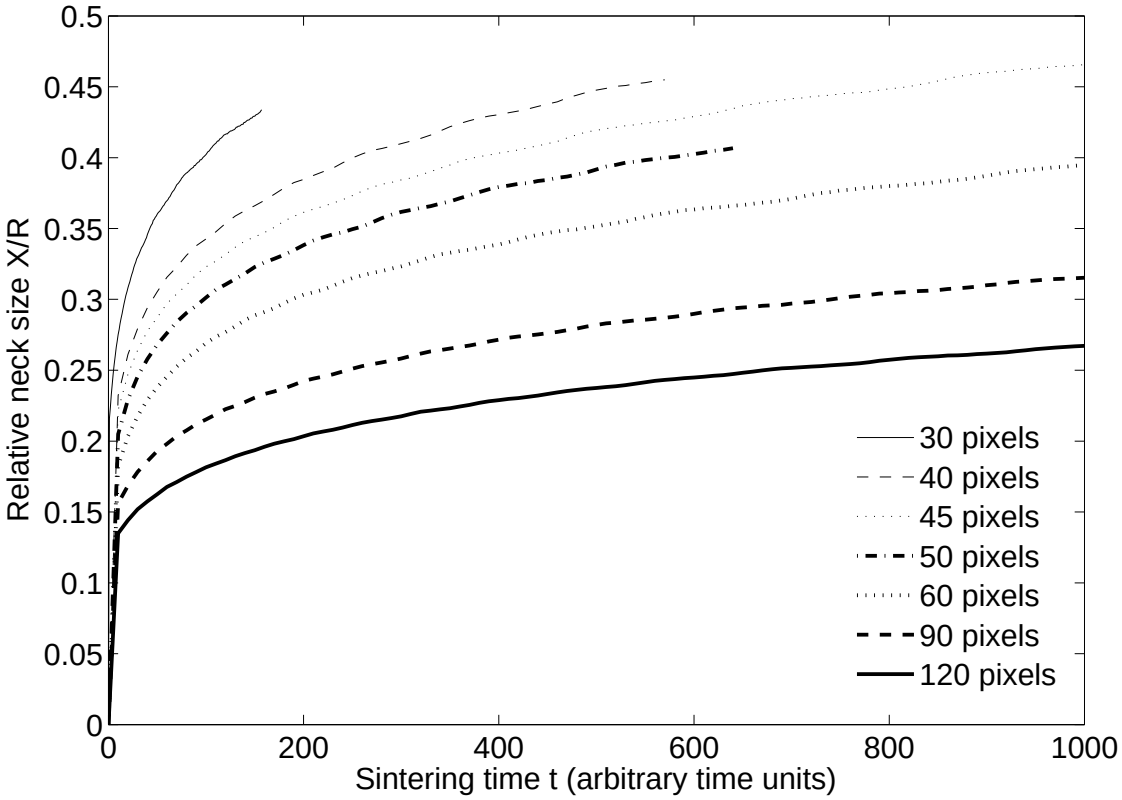
6.2 Two equal sized particles

The phase field simulations on two equal sized particles were performed. The simulations were performed on a uniform square grid of 512×512 pixels. The sintering variables used in simulation of equal sized particles were the same as described in Section 6.1. The simulations were performed for particle sizes given in Table 6.3. Fig. 6.5 shows the neck size during simulations of various particle sizes. It can be observed from Fig. 6.5 that the neck growth slowed down with time.

The dependence of the ratio X/R on particle size and time can be used to extract the mass transport mechanism during neck growth. The kinetics of the neck growth can be expressed as follows [6]:

6.3: Particle size for sintering simulations of equal sized particles

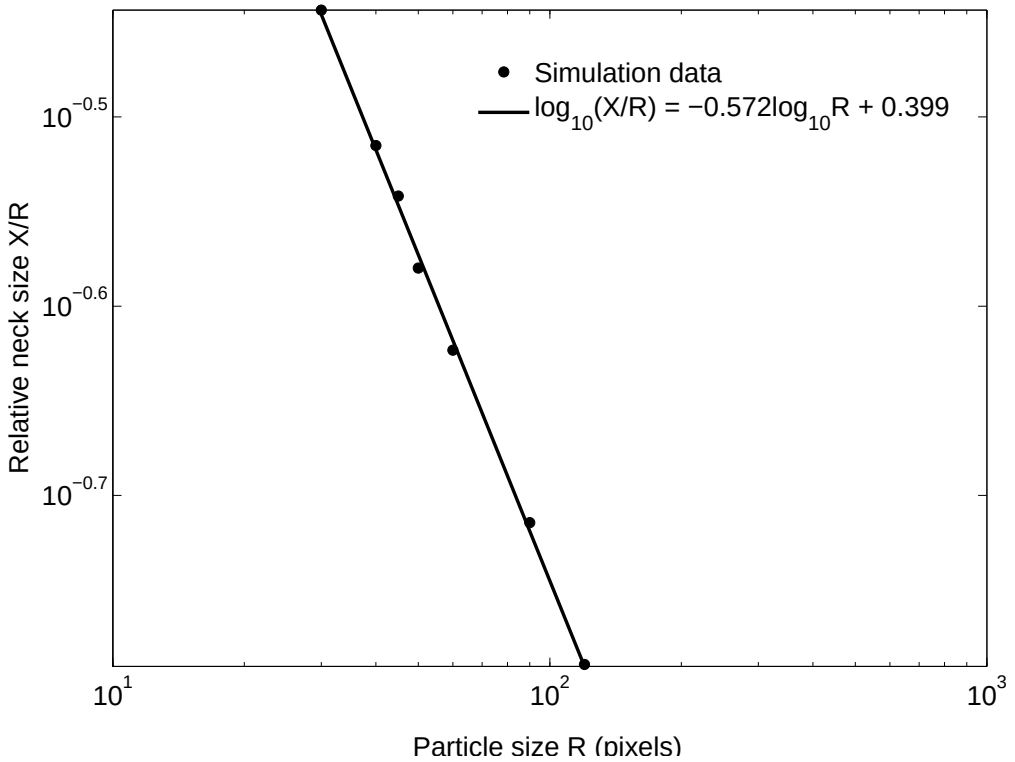
S. no.	Particle radius (pixels)	time required to achieve 0.3 X/R (simulation time)
1	30	18
2	40	45
3	45	70
4	50	100
5	60	190
6	90	720
7	120	2040



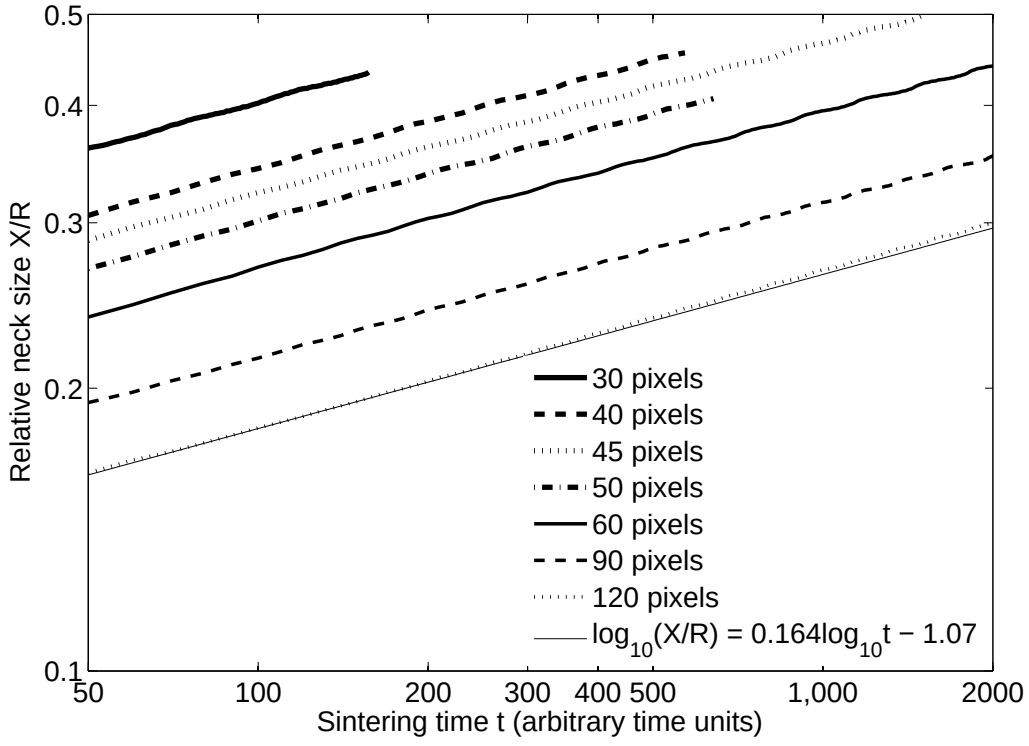
6.5: Neck size during the sintering of equal sized particles

$$\left(\frac{X}{R}\right)^n = \frac{B(T)}{R^m}t \tag{6.6}$$

where X, R and t are the neck size, particle radius and time, respectively. The term $B(T)$ in eq. 6.6 depends upon temperature, materials properties and geometrical constants. The exponent n is termed as the mechanism-characteristic exponent. The exponents m and n depend upon the mass transport mechanism. The values of the exponents m and n can be obtained for phase field simulation results using the neck growth kinetics shown in Fig. 6.5. The neck growth kinetics shown in Fig. 6.5 is shown as a log-log plot in Fig. 6.6. Fig. 6.6 showing a relationship between relative neck size X/R and particle size R at 50 simulation time t provide a value of exponent ratio m/n of 0.572. The relationship between the relative neck size X/R and sintering time t is shown Fig. 6.7 on a log-log scale for various particle sizes. The slopes and intercepts of the curves in Fig. 6.7 are related to exponents m and n. It can be observed from Fig. 6.7 that all of the particle sizes show similar slopes. The similarity of the slopes indicates similar mass transport mechanisms for all of the particle sizes in Fig.



6.6: Effect of particle size on neck growth kinetics for equal sized particle sintering



6.7: Effect of particle size on sintering time for equal sized particle sintering

6.7. A trend line is fit to neck growth kinetics of particle radius 120 pixels in Fig. 6.7. The trend line shows a slope of 0.164. The slope 0.164 of trend line in Fig. 6.7 refers to value of $1/n$. The value of exponents n and m obtained from the neck growth kinetics are given as:

$$n = 6.09; \qquad m = 3.48$$

The value of mechanism-characteristic exponent $n = 6.09$ suggests the grain boundary migration to be the dominent mass transporting mechanism. The value of exponent m is expected to be 4.0 for the mass transport via grain boundary migration. The obtained value of exponent $m = 3.48$ deviates from the expected value of 4.0.

Although one might expect the above analysis of exponents to predict the mass transport mechanism during neck growth, the predictions may be incorrect when more than one mechanism is simultaneously active. A deviation of sintering geometry from spherical particle geometry may also result in incorrect prediction of mass transport mechanism. One example of such incorrect prediction is Kingrey and Berg experiment [12] cited in ref. [6].

6.2.1 Nanoparticles

The consolidation of nanoparticles requires a lower sintering temperature and shorter sintering time as compared to micron size particles [70]. The lower sintering temperature and shorter sintering time may result from the difference in structure between nanoparticles and micron size particles. Nanoparticles have higher surface area per unit volume as compared to the micron size particles. Furthermore, nanoparticles usually have a higher defect density which results in higher diffusivity [71–73].

A shorter sintering time for the nanoparticles has also been predicted by Herring's scaling law [6]. Herring's predictions of shorter sintering time do not include the effects of high diffusivity. The Herring's law suggests that the time taken for obtaining similar geometries depends upon the particle size as given in eq. 6.7.

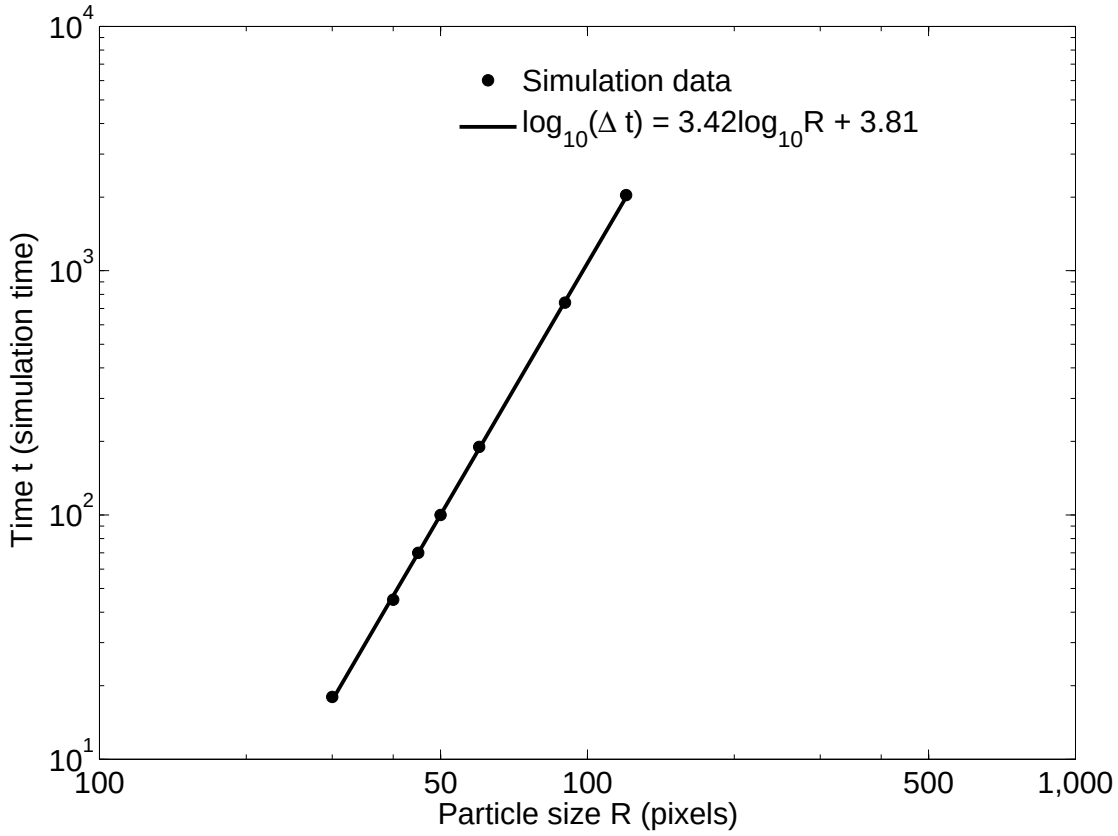
$$\frac{\Delta t_2}{\Delta t_1} = \lambda^m = \left(\frac{R_2}{R_1}\right)^m \tag{6.7}$$

where Δt_1 and Δt_2 are the time required to obtain similar particle geometries for particles with sizes R_1 and R_2, respectively. In eq. 6.7, λ is a numerical factor and m is an exponent. The value of m depends on the sintering mechanism.

The phase field simulation results of two equal particle sintering can be compared with Herring's scaling law to highlight the differences in sintering of nanoparticles. The time required to achieve a 0.3 neck size to particle size ratio is considered for the analysis. The time required to form neck of 30% of the particle radius is shown in Table 6.3. The data in Table 6.3 are ploted in log-log format to obtain the exponent m as shown in Fig. 6.8. In Fig. 6.8, a trend line is added to the sintering times. The trend line shows a good linear fit to sintering times required to achive 0.3 neck size to particle size ratio X/R. The constant slope for all of the data points indicate that the phase field simulations of neck growth in equal size particle sintering follows the Herring's scaling law. The phase field simulation results show a shorter sintering time for smaller particle size. However, this reduction in sintering time is the same as expected by Herring's scaling law.

The comparison of the sintering simulation results with Herring's scaling law shows that the sintering behavior of nanoparticles can be predicted by the scaling law.

Rhodes studied the particle size effect (Herring's scaling law) on sintering of yttria-stablized zirconia [74]. Rhodes found that samples containing nano powders requires longer



6.8: Time required to achieve 0.3 neck size to particle size ratio

sintering time. The sintering of these powders did not follow Herring's scaling law. Rhodes attributed this lower sinterability and deviation from the scaling law to presence of agglomerate in the samples. Rhodes prepared a sample from agglomerate free powder. He found that agglomerate free sample followed Herring's scaling law satisfactorily. Rhodes suggested that difference in sinterablity of nanopowders is a result of difference in initial microstructures. The results of phase field simulations presented in this dissertation also show that Herring's scaling law holds. This shows the agreement between the particle size effect obtained from sintering simulations and sintering experiments.

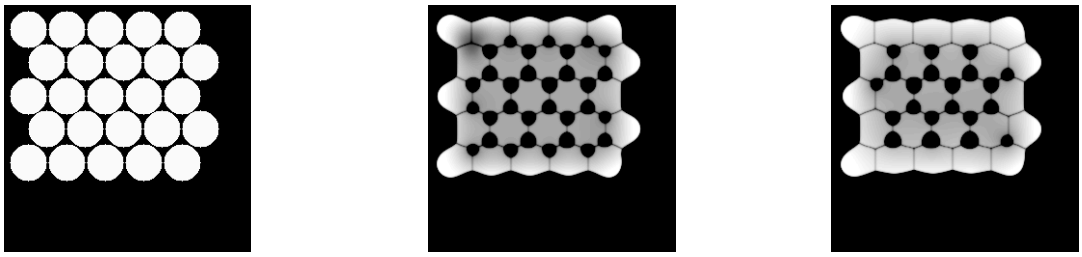
6.3 Closed packed particles

The sintering simulations of closed packed array of equal sized particles were performed using phase field method. The particle sizes for simulations were chosen such that the total solid volume of the particles was similar. The particle sizes for the sintering simulations are given in Table 6.4. Table 6.4 also show that the total solid volume was the same for sintering of 4, 9 and 16 particles. The parameters D_{vol}, D_{vap}, D_{surf}, D_{gb} in eq. 6.3 were chosen as 0.5, 0.23, 38, and 1.15, respectively. These values of variables resulted in diffusion coefficients of 2, 1, 0.5, and 0.25 along surface, grain boundary, surface and vapor phase, respectively.

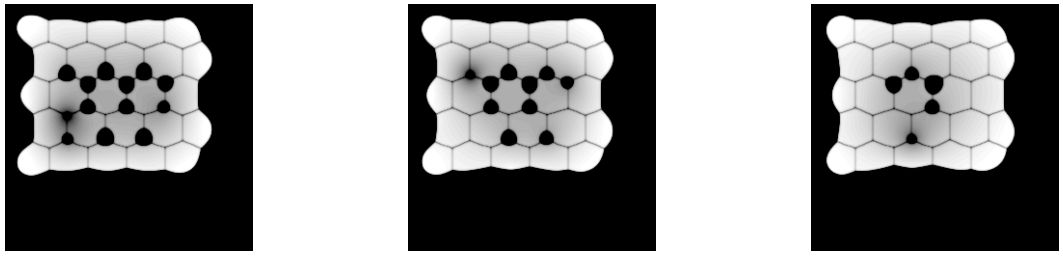
The microstructural evolution of the 25 particle closed pack array sintering is shown in Fig. 6.9. The sintering started with particles touching in a hexagonal arrangement. The neck between the particles was formed significantly by simulation time of 100. It should be noted that the symmetry of the initial microstructure is lost during neck formation. For example, the pores on the upper left corner and lower left corner are symmetrically situated in Fig. 6.9(a). However, Fig. 6.9(b) indicates that one of the pores disappears first resulting in loss of symmetry of the configuration. This loss of the symmetry could be attributed to random fluctuation in phase field due to floating point errors during computation. Fig. 6.9 show that voids closer to the outer surface of the sintering geometry disappeared first. It can also be noted that the voids connected with more grain boundaries for the same distance from the outer surface disappeared rapidly. The voids in the interior of the sintering geometry

6.4: Particle size for sintering simulations of closed packed array of equal sized particles

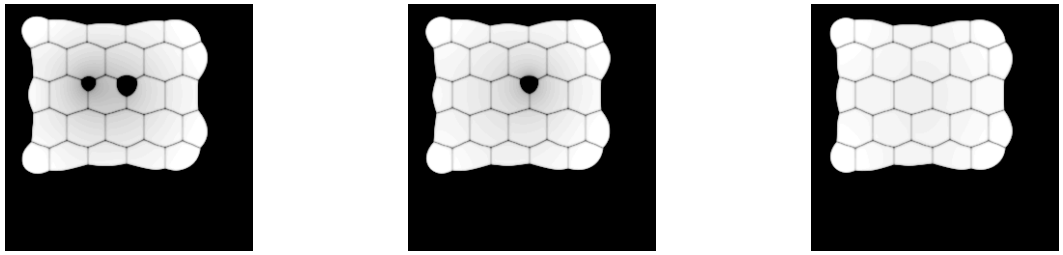
S. no.	Particle diameter (pixels)	No. of particles	Total particle volume (pixel^2)
1	96	4	28393
2	64	9	28393
3	48	16	28393
4	38	25	28353



(a) time 0 (b) time 100 (c) time 200



(d) time 400 (e) time 600 (f) time 800



(g) time 1000 (h) time 1200 (i) time 1500

6.9: Microstructure evolution of sintering of close packed array of 25 particles using phase field simulation method. The number below images refers to coursing time. The density field ρ is shown on a gray scale of [0.95 1] in the figures.

disappeared in the end of the sintering.

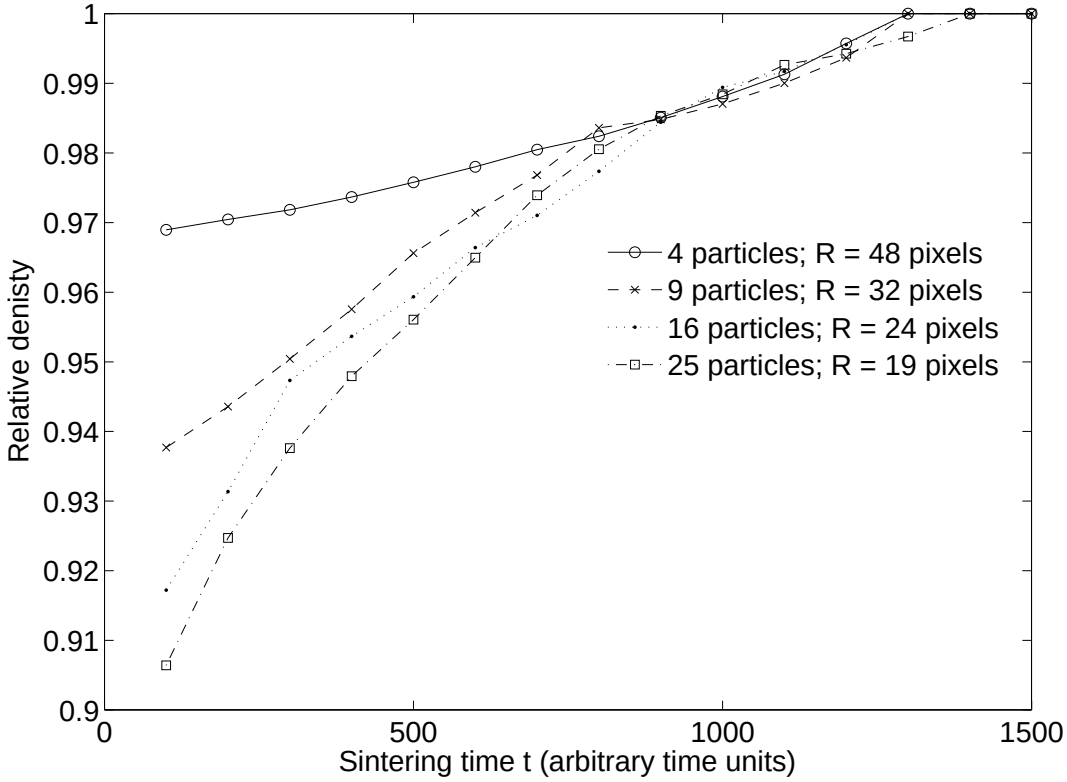
The closed pack array of particles makes a symmetric configuration. The sintering of such symmetric configuration involves densification without interparticle mass transport. The grain boundary in such a configuration may not migrate due to the symmetry. Therefore, the densification in such closed packed array is achieved by change in the particle shapes.

The amount of mass to be transported for densification increases with reduction in particle size for close packed array. At the same time, the grain boundary area and surface area for the mass transport increases. In this set of simulation, effect of the particle size on sintering can be observed. The particle size effect in this case is based upon constant volume. For the constant volume of green compact, the initial density depends upon particle size. Even if the particle arrangement is the same, the compacts with smaller particles have lower green density. The effect of particle size on green density can be observed in Fig. 6.9. Fig. 6.9 shows that particles with 24 pixels radius have lower density than particles with 48 pixel radius. The sintering curves in Fig. 6.9 show that sintering of smaller particles results in higher sintering rates. It should be noted that this effect of particle size excludes effect of particle coarsening and grain boundary migration. The effect of particle size on the sintering kinetics of the close packed array is shown in Fig. 6.10. Fig. 6.10 shows that the initial relative density of the arrays with smaller particles was lower; however, the rate of sintering was faster for the smaller particles. This shows that the effect of increase in void fraction for smaller particles can be outweighed by increase in the surface area and the grain boundary area.

6.4 Randomly arranged particles

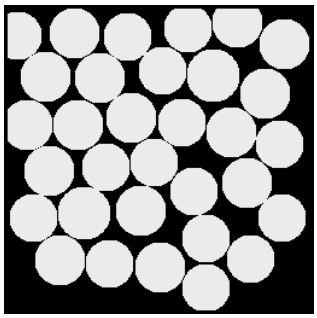
The phase field simulations were performed for randomly distributed particles. The simulations were performed on particles of size 40 pixel in diameter. In order to fit the sintering geometry in simulation domain, a few particles on the edges were truncated. The values of variables for this set of simulations were the same as values in simulations of two unequal size particles described in Section 6.1.

The sintering of multiparticle microstructure started from touching circular particles as shown in Fig. 6.11(a). The circular particles made a point contact. The neck between the particles grew significantly in 10 unit simulation time as shown in Fig. 6.11(b). The 10 unit simulation time is less than 0.1% of total sintering time. The neck formation was also accompanied by a shape change in the particles and voids. The voids also acquired their equlibrium shapes. The pores with smaller coordination numer shrank and disappeared preferentially.

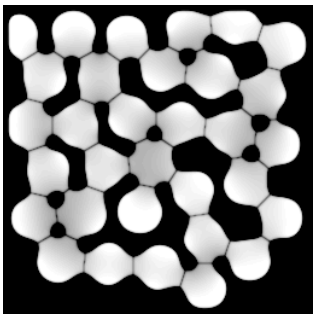


6.10: Effect of particle size on sintering kinetics of equal sized close packed array of particles

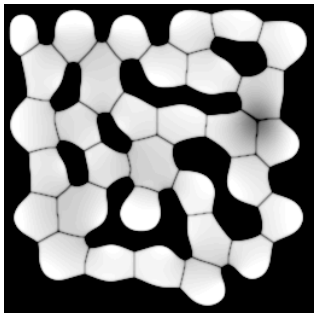
All of the pores with coordination number 3 disappeared by 40 unit simulation time. A grain boundary migration in Fig. 6.11(c)-(d) can also observed at 40-70 unit simulation time. Migration of a grain boundary located at the upper left corner in microstructure during this period of time can be observed in Fig. 6.11(c)-(d). This grain boundary migration resulted in a rapid shrinkage of the partice at the corner. The particle in the corner diappeared by 130 sintering time. The pore size also changed during the sintering. Fig. 6.4 shows that few pores shrank and the rest of the pores grew in size. Fig. 6.11(e)-(f) show that the "U" shaped pore near the bottom of the sintering geometery grew in size between 130-190 unit sintering time. On the other hand, the pore located on the upper right corner reduced during the same period of time. As pores reduced and disappeared, the coordination number of the pores changed. This change in the pores coordination number resulted in increasing rate of pore closure. It should be noted from the Fig. 6.4 that the grain boundary did not cross a pore. Therefore, the pore functioned as grain boundary pinning sites. The number of pores reduced considerably by 430 sintering time. At this time, the grain boundary migration became a predominant factor in evolution of microstructure. The grain boundary migration affected the grain size significantly from 430 unit sintering time until the end of the sintering as shown



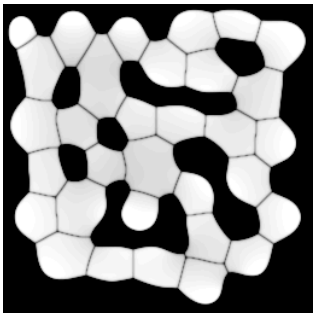
(a) time 0



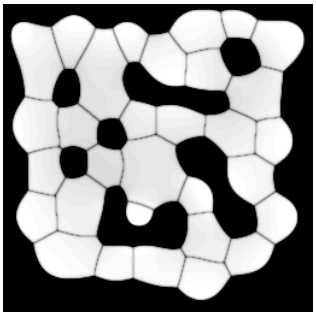
(b) time 10



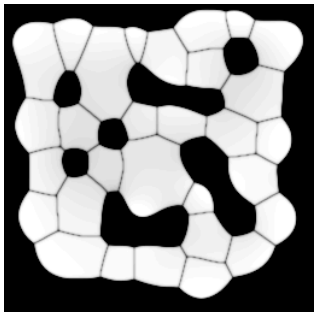
(c) time 40



(d) time 70

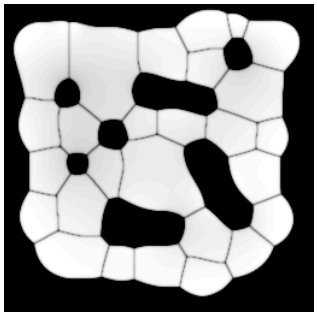


(e) time 130

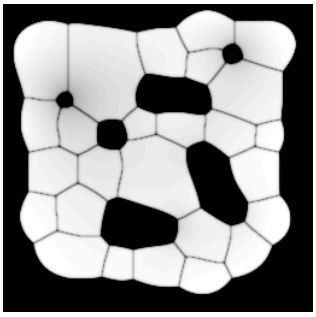


(f) time 190

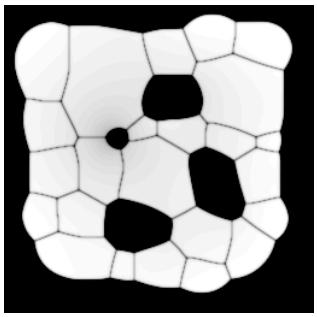
Microstructure evolution of sintering of randomly distributed particles using phase field simulation method. The number below images refers to coursing time. The density field ρ is shown on a gray scale of [0.95 1] in the figures.



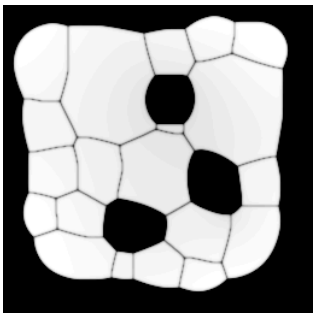
(g) time 310



(h) time 430



(i) time 670



(j) time 910

(k) time 1410

(l) time 1910

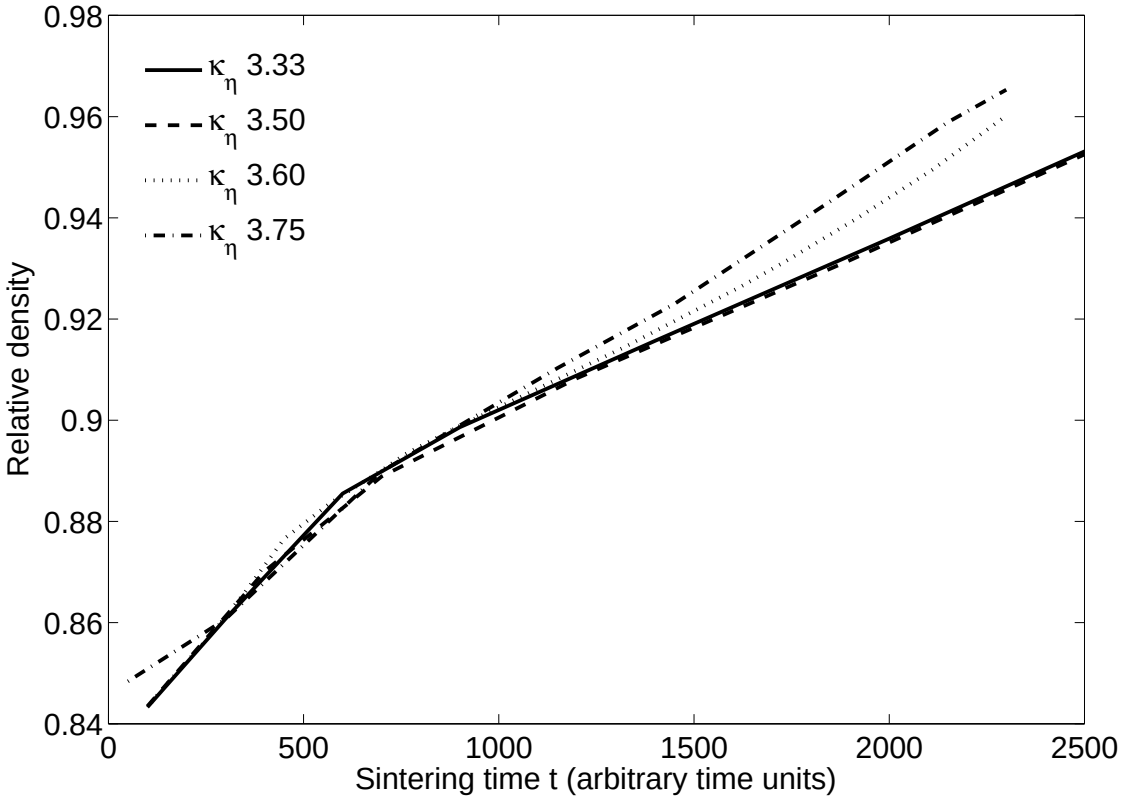
Continued

by Fig. 6.11(h)-(l).

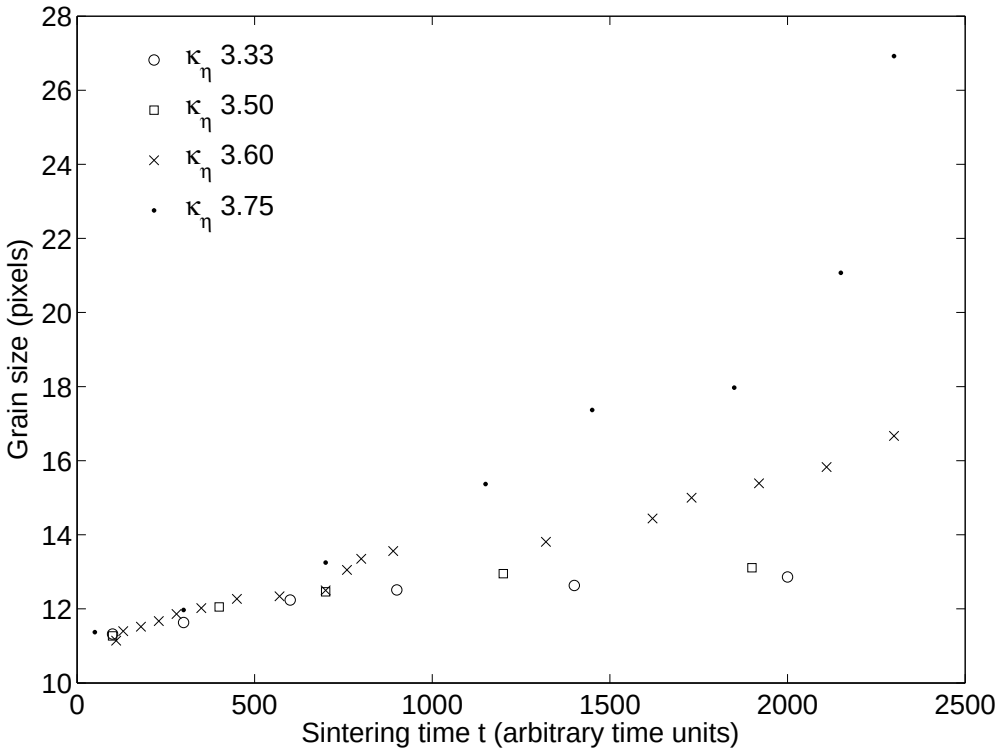
The density of the sintering geometry in Fig. 6.4 increased as sintering progressed. Fig. 6.12 shows the density of the sintering geometry as a function of sintering time.

Fig. 6.12 shows that the rate of densification decreased with sintering time. Fig. 6.12 also shows the density of simulations with various values of κ_η. The kinetic coefficient κ_η in eq. 6.1 was found to be affecting the grain boundary mobility. A higher value of the kinetic coefficient κ_η resulted in increased grain boundary mobility. The grain boundary was observed to be almost immobile with a kinetic coefficient κ_η of 3.33. Relative density curves in Fig. 6.1 show that the rate of densification increased with grain boundary mobility. The increase in the densification could be attributed to reduction in the pore coordination number during the grain boundary migration. It is reported that the pores with small coordination number shrink rapidly [5]. This reduction in the pore coordination number may lead to higher rates of densification.

The mean grain size variation during the sintering is shown in Fig. 6.13. Fig. 6.13 shows that the mean grain size of the sintering geometry increases with time. Fig. 6.13 shows a



6.12: Relative density of multiparticle random microstructures



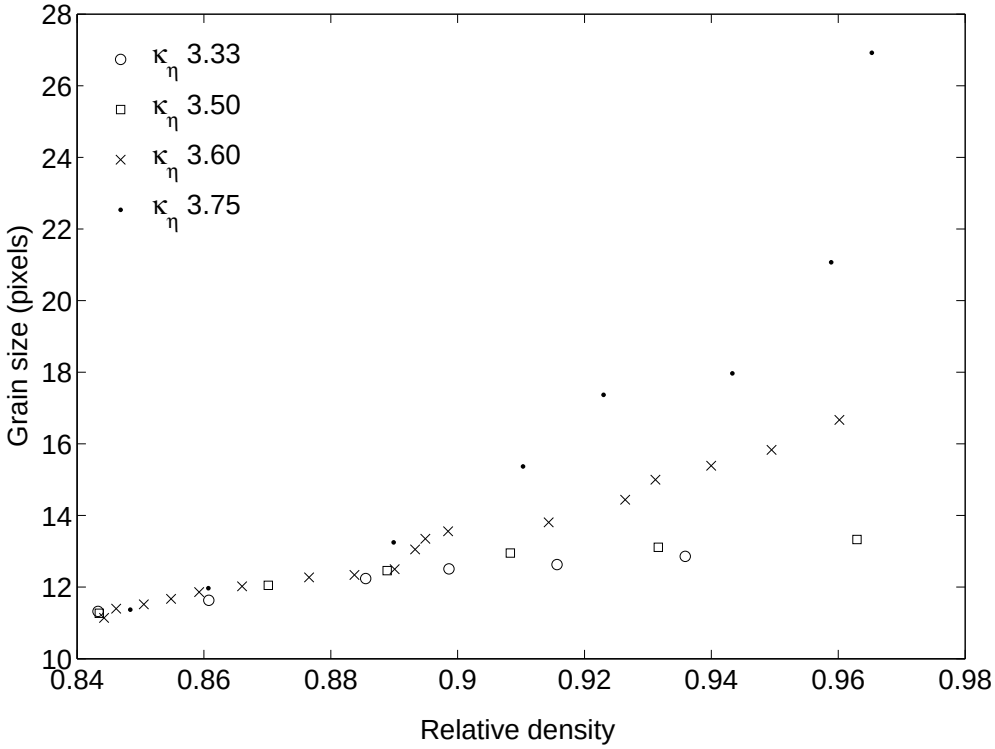
6.13: Mean grain size of multiparticle random microstructures

strong effect of the grain boundary mobility on the grain size during sintering. The grain growth rate reduced with a lower grain boundary mobilities, κ_η 3.33. The rate of grain growth increases with the grain boundary mobility. The grain size increased rapidly in a high grain boundary mobility sintering with kinetic coefficient κ_η 3.75. This showed that final grain size in the sintering simulation was controlled by the grain boundary mobility.

This relative effect of the grain boundary mobility can be observed in Fig. 6.14. Fig. 6.14 indicates that the grain boundary mobility showed a stronger effect on the grain size than on the relative density. The sintering simulation with immobile grain boundaries achieved theoretical density with a minimal grain growth. With high grain boundary mobility κ_η 3.75, the grain size increased by three times before sintering geometry was fully dense. An intermediate value of κ_η 3.6 resulted in a moderate grain growth as shown in Fig. 6.14.

6.5 Pores

The phase field simulations were performed to simulate the closure of pores. The simulation variables used in this set of simulations were the same as variables in sintering to two unequal particles described in Section 6.1. The simulations were performed to understand a relation between shrinkage behavior and pore coordination number. For simulating the pore



6.14: Mean grain size and relative density of multiparticle random microstructures

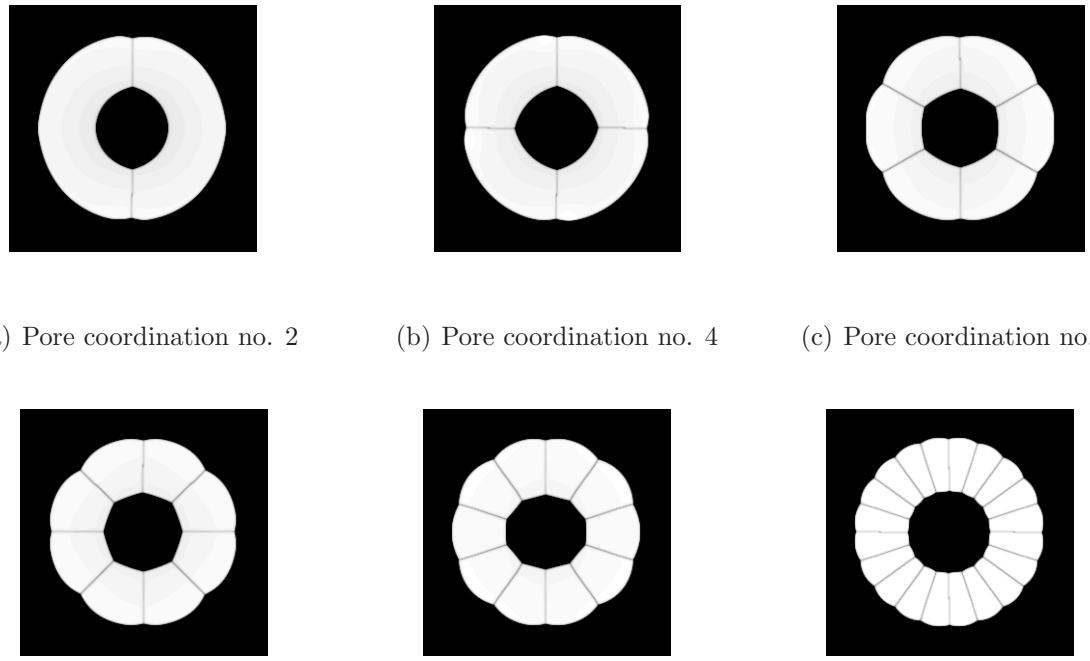
shrinkage, microstructures with a circular pore surrounded by symmetrically arranged grains were simulated. The simulations were performed with pores of coordination number 2, 4, 6, 8, 10, and 20. Figure 6.15 show the microstructures of various pores.

The shrinkage of the pores is quantitatively shown in Fig. 6.16. The radius of a pore in Fig. 6.16 refers to the radius of a circle having area equal to the pore area.

The initial pore size was the same for pores with different coordination numbers. The shrinkage of a pore is found to have a strong correlation with the pore coordination number as shown in Fig. 6.16. Fig. 6.16 shows that the pore with coordination number 2 shrank rapidly. The rate of pore shrinkage reduced as pore coordination number was increased. The pore with the coordination number of 20 did not shrink during shrinkage.

The energy analysis of pore shrinkage provides the critical coordination number below which pores shrink [5]. The critical coordination number n_c depends upon the dihedral angle as

$$n_c = \frac{\pi - \phi_e}{2\pi} \tag{6.8}$$



(a) Pore coordination no. 2 (b) Pore coordination no. 4 (c) Pore coordination no. 6

(d) Pore coordination no. 8 (e) Pore coordination no. 10 (f) Pore coordination no. 20

6.15: Microstructures of pore shrinkage using phase field simulation method. The density field ρ is shown on a gray scale of [0.95 1] in the figures.

where ϕ_e is the equilibrium dihedral angle. The equilibrium dihedral angle for the simulation performed was 140deg. This dihedral angle of 140deg resulted in a critical pore coordination number of 9. The shrinkage of all of the pores follows the prediction of eq. 6.8 except the pore with the coordination number 10. Based on the analysis of eq. 6.8, the pore with coordination number 10 should not shrink. A microstructural analysis indicates that the pore shrinkage in this case was achieved by geometrical distortion. Fig. 6.15(e) indicates that the pore with coordination number 10 acquired an elliptical shape. This shape is not symmetric and can be achieved by small asymmetric fluctuation. The asymmetry in the pore shape resulted in concave and convex radii of curvature at different surfaces of the pore. The pore may shrink in such cases even though it has a coordination number higher than critical pore coordination number n_c.

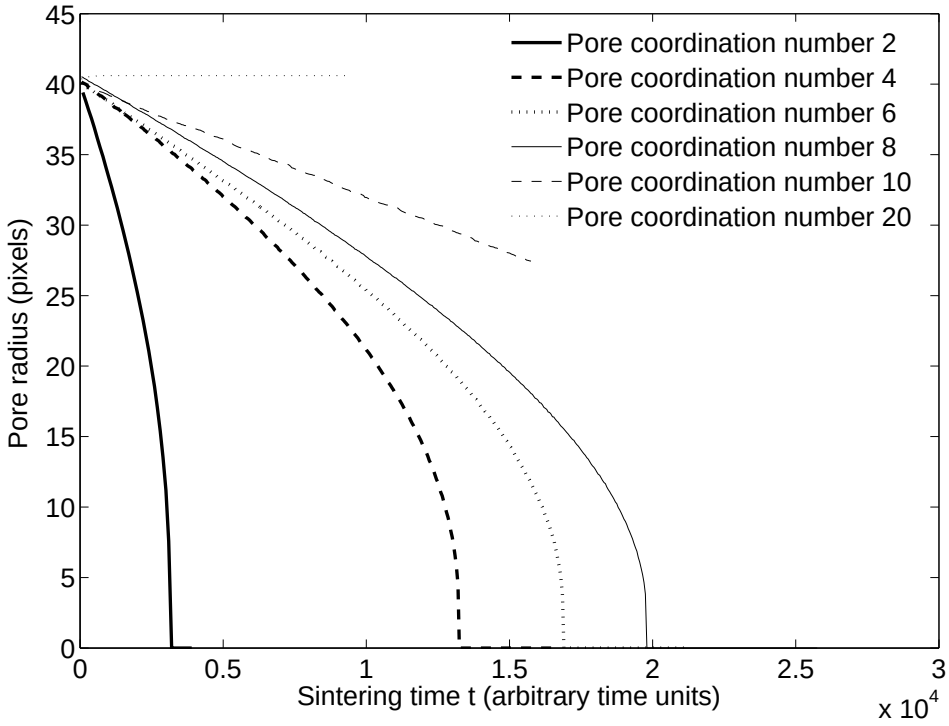
6.6 Limitations of 2D simulations

The phase field simulations described above provide insight into sintering of various microstructures. These simulations were performed in two dimensions. These simulations of circular particles in two dimensions can be correlated with sintering of cylindrical particles in three dimensions. This correlation of circular and cylindrical particles can provide under-

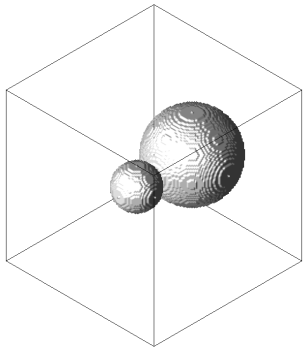
standing of sintering of various microstructures. These microstructures include linear array of particles, pores and symmetric array of particles. However, microstructures containing randomly distributed particles cannot be appropriately simulated using two-dimensional simulation. The inappropriateness results from inability of two-dimensional microstructure to correctly describe microstructure during intermediate stage of sintering. The intermediate stage of sintering in three dimensions is characterized by cylindrical pores long grain edges. These pores are closed by vacancy transport from pores to grain boundaries. The configuration of cylindrical pores along the grain edges and required vacancy transport cannot be described by a two-dimensional microstructure. Therefore, sintering of bulk powders containing randomly oriented particles should be simulated in three dimensions.

6.7 3D simulations of two unequal sized particles

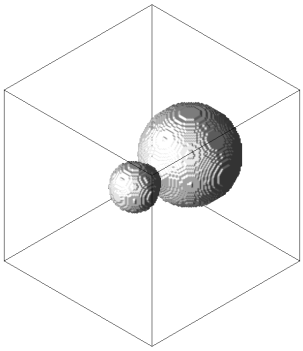
The sintering simulation on two unequal sized particles was performed in three dimensions. The simulation variables in this simulation were the same as variables used in sintering of two unequal sized particles described in Section 6.1. The simulation in three dimensions was performed on a lattice of 128×128×128 voxels. The initial particle sizes were 64 and 32 voxels. The microstructural evolution of the two unequal sized particles in three dimensions is shown in Fig. 6.17.



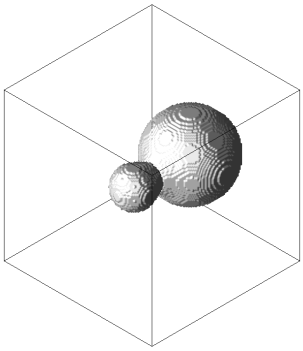
6.16: Pore shrinkage during phase field simulation of sintering



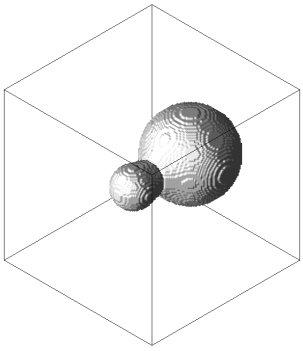
(a) time 0



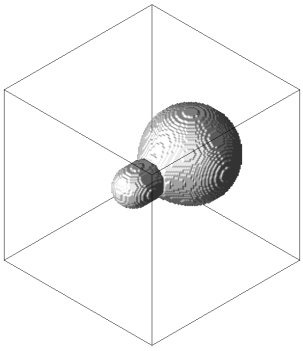
(b) time 10



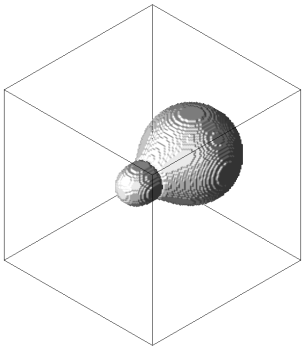
(c) time 50



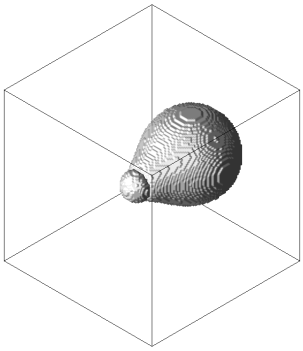
(d) time 100



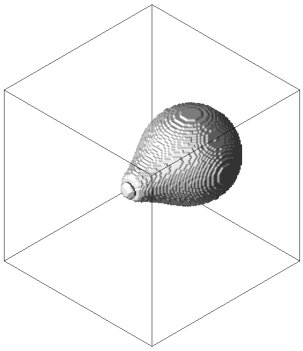
(e) time 300



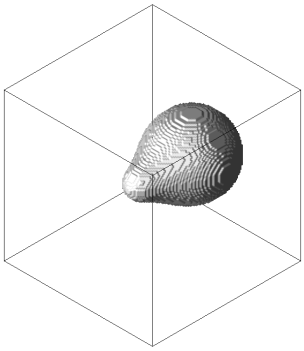
(f) time 500



(g) time 750



(h) time 760



(i) time 800

6.17: Microstructure evolution of sintering of two unequal sized particles in three dimensions using phase field simulation method. The number below images refers to coursing time.

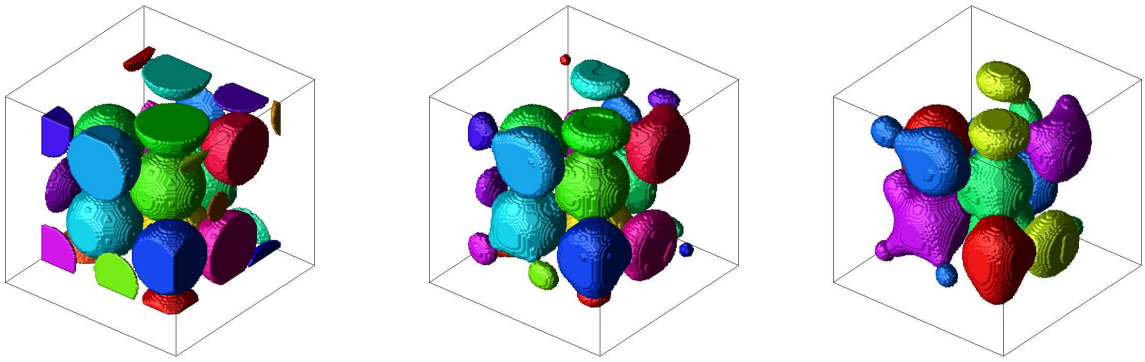
Fig. 6.17(a) shows the particles making point contact at the beginning of the sintering. A neck between the particles was formed rapidly. Fig. 6.17(b) shows that a significant neck size was formed in 10 unit simulation time. The neck growth was reduced as sintering progressed. The neck was reduced considerably by 100 unit simulation time as shown in Fig. 6.17(d). It can be observed that particle size change until 100 unit simulation time was negligible. Fig. 6.17(d) - 6.17(g) show that the particle size change slowly during 100 - 750 unit simulation time. Fig. 6.17(g) - 6.17(h) show a rapid grain boundary migration. This rapid grain boundary migration resulted in a rapid shrinkage of the small particle. The small particle disappeared at the end of the sintering as shown in Fig. 6.17(d).

A similarity between two-dimensional and three-dimensional simulations can be observed from Fig. 6.17 and 6.2. These figures show that the sintering of two particles follows the same microstructural evolution path irrespective of dimensionality of the simulation. Therefore, a two-dimensional simulation of two unequal sized particles can produce a microstructural evolution for quantitative analysis.

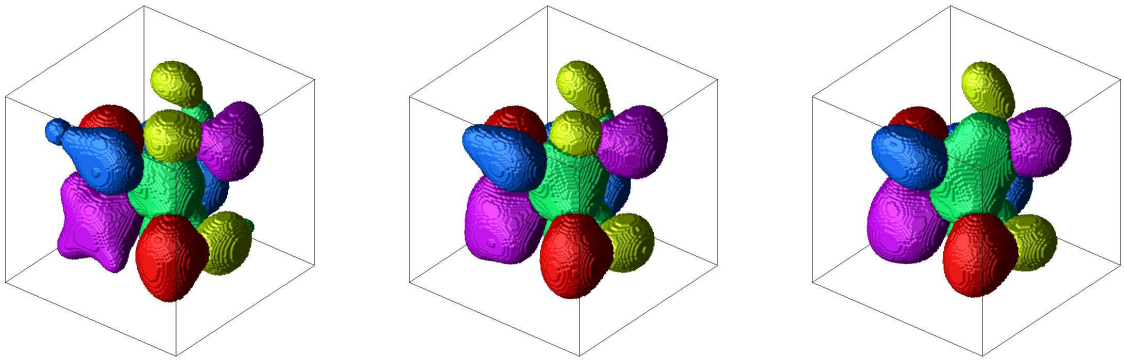
6.8 Randomly arranged particles in 3D

The sintering simulation on randomly distributed particles was performed in three dimensions. The simulation variables in this simulations were the same as variables used in sintering of two unequal sized particles described in Section 6.1. The simulation in three dimensions was performed on a lattice of 128×128×128 voxels. The initial particle sizes were 20-21 voxels. The initial microstructure contained 28 particles. The particles located on the edges of the simulation domain were truncated. The truncation was performed due to a use of periodic boundary conditions. The particle on the one side of edge can bond with particles on the other side of edge due to periodic boundary condition. The truncation of the particles resulted in smaller particle sizes.

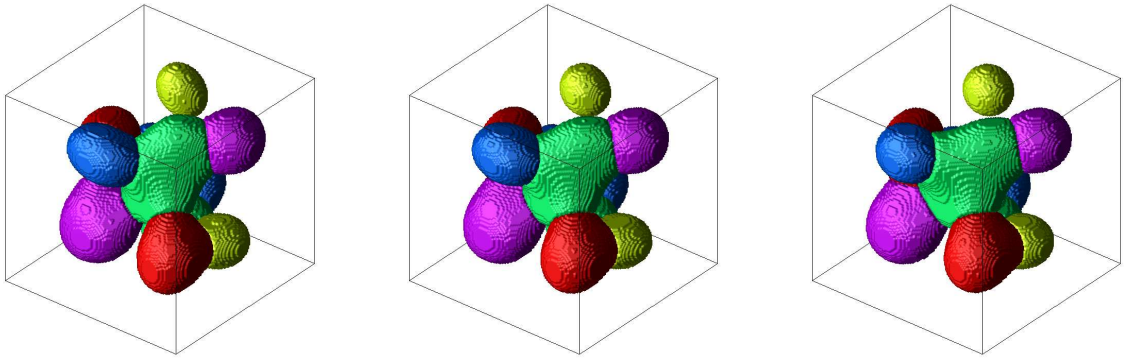
The microstructural evolution of randomly distributed particles in three dimensions is shown in Fig. 6.18. Fig. 6.18(a) shows the initial microstructure. The neck between the particles forms rapidly. Fig. 6.18(b) show that the significant neck formation occurred between 0-10 unit simulation time. The coarsening and grain boundary migration subprocess during simulation can also be seen in Fig. 6.18. A quantitative analysis on this simulation was not performed.



(a) time 0 (b) time 10 (c) time 50



(d) time 100 (e) time 200 (f) time 300



(g) time 400 (h) time 500 (i) time 600

6.18: Microstructure evolution of sintering of randomly distributed particles using phase field simulation method. The number below images refers to coursing time.

CHAPTER 7

GEOMETRICAL MODEL

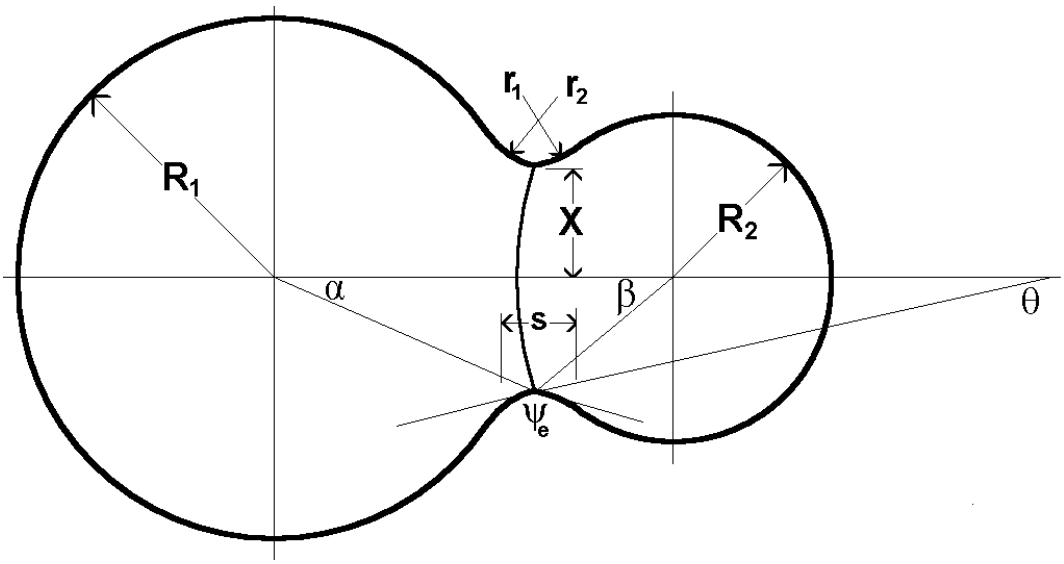
A geometrical sintering model using microstructural evolution results from the Monte Carlo simulation methods is developed. The geometrical model makes assumptions regarding the sintering geometries during the microstructural evolution. The geometrical assumptions introduce errors in deriving the sintering kinetics. Nevertheless, the assumptions help in deriving simple expressions for sintering kinetics. The kinetics expressions of the geometrical model can provide sintering kinetics in terms of real time and length scale. Moreover, geometrical models can be used to obtain the sintering kinetics with sintering parameters such as temperature and diffusion coefficients.

7.1 Description of geometrical model

A geometry of two unequal sized spherical particles is used for obtaining kinetics using a geometrical model. The sintering kinetics are derived for the unequal sized particle geometry shown in Fig. 7.1. The geometry in Fig. 7.1 consists of two spheres bonded by a neck. The particle radii in Fig. 7.1 are R_1 and R_2, $(R_1 \geq R_2)$. The solid vapor surfaces and grain boundary at the neck maintain the equilibrium dihedral angle ϕ_e. The radii of curvature at the neck are r_1 and r_2 as shown in Fig. 7.1. The neck of size X subtends an angle of θ on the curvature center of the grain boundary.

The sintering model using the above geometry is described in three stages. The stages are: (1) neck growth, (2) coarsening and (3) rapid grain boundary migration. These stages brings geometrical changes in the two particle geometry. The time required to bring the geometrical changes depends upon the mass transport path and diffusion coefficient. The kinetics of the sintering is determined by estimating the time required for bringing geometrical changes. The kinetics can be calculated from the following equation.

$$\frac{dV}{dt} = \vec{J} A \Omega \tag{7.1}$$



7.1: The two particle geometry

where $\frac{dV}{dt}$ is the rate of volume transport, $\vec{J}$ is the flux in moles/(m^2 s), A is the area through which the mass is transported and Ω is the molar volume. The flux $\vec{J}$ in equation 7.1 can be written as follows:

$$\vec{J} = \frac{D}{\Omega RT}\frac{\Delta\mu}{\Delta x} \tag{7.2}$$

where D is the diffusion coefficient, R is the universal gas constant, T is the absolute temperature, and $\frac{\Delta\mu}{\Delta x}$ is the chemical potential gradient. Equation 7.1 can be solved numerically to derive the sintering kinetics as:

$$\Delta t = \frac{\Delta V}{\frac{D}{\Omega RT}\frac{\Delta\mu}{\Delta x}A\Omega} \tag{7.3}$$

Eq. 7.3 is used to calculate the time intervals between sintering geometries during microstructural evolution. The time intervals provide the sintering kinetics.

7.1.1 Sintering stages

The sintering kinetics of the three sintering stages can be derived using eq. 7.3. The choice of parameters the equation 7.3 in each stage depends upon the mass transport mechanism.

The kinetics of the three stages of sintering can be described as follows.

7.1.1.1 Neck growth

In the neck growth stage of sintering, a formation of neck binds the particles. The neck between the particles can grow by several mass transport mechanisms. These mechanisms include (1) surface diffusion from particle surfaces, (2) grain boundary diffusion from grain boundaries, (3) lattice diffusion from particle surfaces and (4) lattice diffusion from grain boundary. The choice of parameters in eq. 7.3 is given in Table 7.1 for various mass transport mechanisms. For solving eq. 7.3, the value of dV is taken as the volume added to the neck for the neck growth.

In Table 7.1, D_s, D_{gb}, and D_l are surface diffusion coefficient, grain boundary diffusion coefficient, and lattice diffusion coefficient, respectively. In Table 7.1, Ω, δ_S, and δ_{gb} refer to molar volume, thickness of atomic diffusion layer on surface, and thickness grain boundary, respectively.

7.1.1.2 Coarsening

The particle sizes change during the coarsening stage. The particle size change is driven by the chemical potential difference between the particles. This chemical potential difference depends upon the two particle sizes. The choice of parameters in equation 7.3 for calculation of coarsening kinetics are given in Table 7.1.

7.1.1.3 Grain boundary migration

The grain boundary between the particles migrates in the later stage of sintering. The driving force for the grain boundary migration results from change in volume energy and grain boundary energy. The atoms in the smaller particles are at higher chemical potential due to the pressure difference caused by dissimilarity in particle sizes (cf. eq. 2.2) The atoms in small particle jump across the grain boundary to reduce the volume energy. The jump of the atoms results in the grain boundary migration. The grain boundary may also migrate due to a reduction in the grain boundary area. The velocity of the grain boundary can be derived from changes in volume energy and grain boundary energy per unit volume swept by the boundary.

$$v = mF = -m\frac{\mathrm{d}G}{\mathrm{d}V} = \underbrace{\Delta P}_{\text{Volume energy}} + \underbrace{\gamma_{gb}\frac{\mathrm{d}A}{\mathrm{d}V}}_{\text{Grain boundary energy}} \tag{7.4}$$

7.1: Parameters for estimating the kinetics of neck growth and coarsening subprocesses

mass transport mechanism	Materials source	Materials sink	$\Delta\mu$	Δx	A	D
Neck growth						
surface diffusion	particle surface	neck	$\gamma_s\Omega\left(\frac{1}{R_1}+\frac{1}{R_2}+\frac{2}{r_1+r_2}-\frac{1}{X}\right)$	S	$2\pi X\delta_S$	D_s
grain boundary diffusion	grain boundary	neck	$\gamma_s\Omega\left(\frac{2}{r_1+r_2}-\frac{1}{X}\right)$	X	$2\pi X\delta_{gb}$	D_{gb}
lattice diffusion	particle surface	neck	$\gamma_s\Omega\left(\frac{1}{R_1}+\frac{1}{R_2}+\frac{2}{r_1+r_2}-\frac{1}{X}\right)$	S	$2\pi XS$	D_l
lattice diffusion	grain boundary	neck	$\gamma_s\Omega\left(\frac{2}{r_1+r_2}-\frac{1}{X}\right)$	X	$2\pi XS$	D_l
Coarsening						
surface diffusion	small particle surface	large particle	$\gamma_s\Omega\left(\frac{2}{R_1}-\frac{2}{R_2}\right)$	S	$2\pi X\delta_S$	D_s
lattice diffusion	small particle	large particle	$\gamma_s\Omega\left(\frac{2}{R_1}-\frac{2}{R_2}\right)$	S	πX^2	D_l

R_1 radius of large particle, R_2 radius of small particle, (r_1, r_2) radii of curvature at neck, X neck size, α half of angle subtended by grain boundary on center of large particle, β half of angle subtended by grain boundary on center of small particle, θ half of angle subtended by grain boundary on center of sphere formed by grain boundary, Neck width $S = r_1(\cos(\alpha) - cos(\phi_e/2-\theta)) + r_2(\cos(\beta) - \cos(\phi_e/2+\theta))$, δ_s diffusion thickness of surface, δ_{gb} grain boundary thickness, γ_s specific surface energy, D_s surface diffusion coefficient, D_{gb} grain boundary diffusion coefficient, D_l lattice diffusion coefficient

where m is the grain boundary mobility, F is the net driving force, $\mathrm{d}G/\mathrm{d}V$ is the change in total free energy per unit volume swept by the boundary, ΔP is the pressure across the boundary and $\mathrm{d}A/\mathrm{d}V$ change in grain boundary area per unit volume swept by the boundary. If grain boundary shown in geometry in Fig. 7.1 migrates towards the right by a distance $\mathrm{d}x$, the quantities ΔP and $\mathrm{d}A/\mathrm{d}V$ can be calculated as:

$$\Delta P = \gamma_s \left(\frac{2}{R_1} - \frac{2}{R_2} \right) \tag{7.5}$$

$$\frac{\mathrm{d}A}{\mathrm{d}V} = \gamma_{gb} \frac{\dfrac{2\pi \Delta x X \sec\theta}{(\tan\beta - \tan\theta)}}{\pi \Delta x (X^2 + \Delta x R_2 \cos\beta - \Delta x^2/3)} \tag{7.6}$$

where R_1 and R_2 are radii of curvature of the large and the small particles and ΔX is the change in neck size. Eq. 7.5 is based upon Young-Laplace equation [10] which describes the pressure due to a curved surface. The grain boundary velocity can be approximated from Eq. 7.4 - 7.6 for $\delta X << X$ as:

$$v = -m \left[\gamma_s \left(\frac{2}{R_1} - \frac{2}{R_2} \right) + \gamma_{gb} \frac{2 \sec\theta}{X(\tan\beta - \tan\theta)} \right] \tag{7.7}$$

For a small angle θ in the two particle geometry, eq. 7.7 can be written as follows

$$v = 2m \left[\gamma_s \left(\frac{1}{R_2} - \frac{1}{R_1} \right) - \gamma_{gb} \frac{\cot(\beta)}{X} \right] \tag{7.8}$$

The first term in eq. 7.8 provides the driving force for the grain boundary migration, whereas the second term containing $\cot\beta$ can act as a drag force or driving force. When grain boundary migration results in increase in grain boundary area, the second term acts as drag force. Otherwise it acts a driving force. Eq. 7.8 shows the dependence of grain boundary velocity on geometry via angle β. When subtended angle β becomes greater than $\pi/2$, the drag force term in eq. 7.8 becomes positive. In this situation, both terms in eq. 7.8 drive the grain boundary migration. Therefore, the grain boundary migration is rapid for angle $\beta > \pi/2$.

For $\beta \leq \pi/2$, eq. 7.7 can be used to calculate the kinetics. In order to calculate the volume change due to grain boundary migration ΔV_{gb}, time required for small change in particle size Δt is first calculated using eq. 7.3. The grain boundary velocity v is then calculated from eq. 7.8 using geometrical parameters. The volume change due to the slow grain boundary migration is calculated as

$$\Delta V_{gb} = \pi X^2 v \Delta t \tag{7.9}$$

When angle β becomes greater than $\pi/2$, a rapid grain boundary migration occurs. During the rapid grain boundary migration, the neck size and the volume of particle reduces to zero. The time required to bring a small change in volume ΔV in small particle can be calculated as

$$\Delta t = \frac{\Delta V}{\pi X^2 v} \tag{7.10}$$

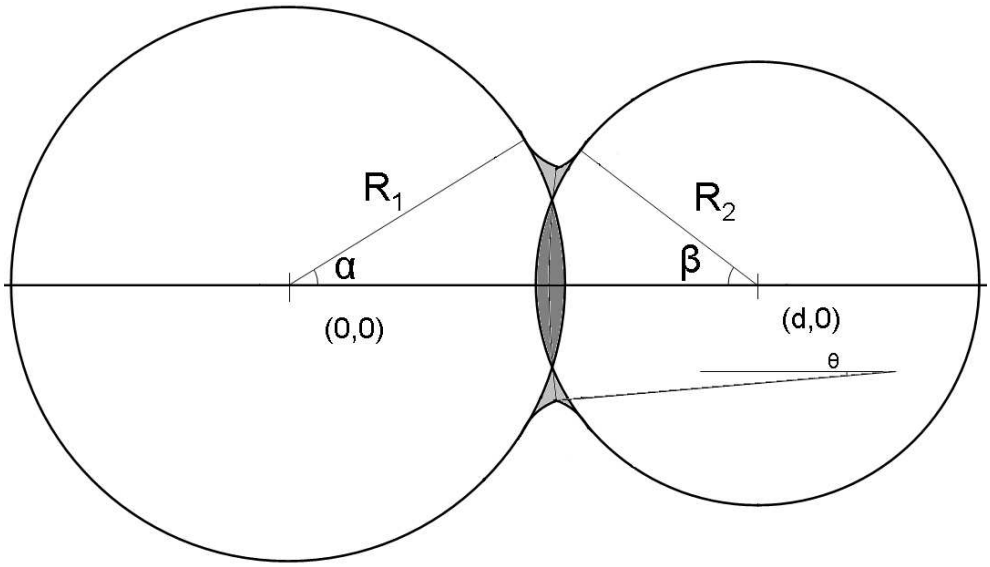
7.1.2 Sintering geometry

As discussed before, the sintering geometry consists of two spherical particles and a neck between them. The neck formed between the particles has two circular profiles intersecting at equilibrium dihedral angle ϕ_e. The angle ϕ_e depends upon the grain boundary and surface energy ratio given by eq. 7.11. The following sections describe the sintering geometry for neck growth, coarsening and grain boundary migration.

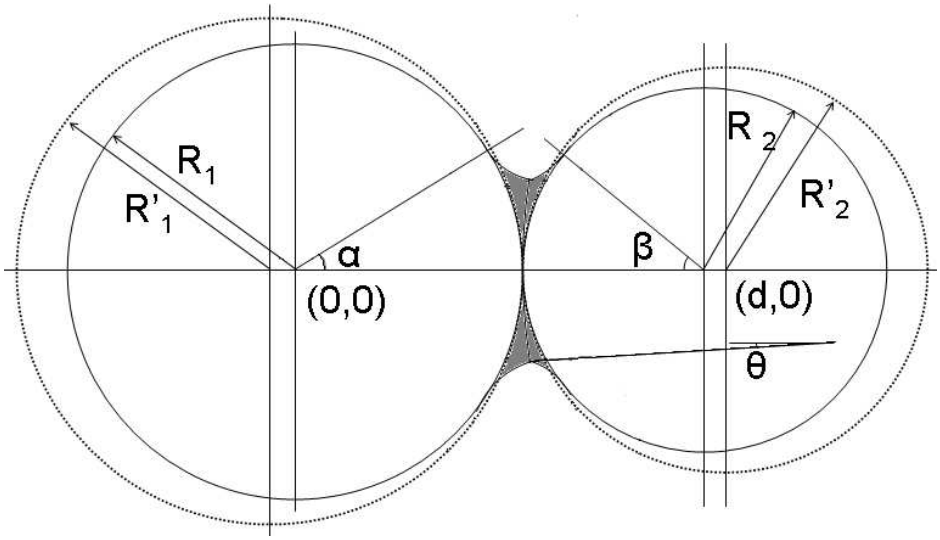
$$\phi_e = 2 \arccos\left(\frac{\gamma_{gb}}{2\gamma_S}\right) \tag{7.11}$$

7.1.2.1 Neck growth geometry

The sintering geometry to derive the neck growth kinetics can be described for sintering cases of (1) with shrinkage and (2) without shrinkage as shown in Fig. 7.2 and 7.3, respectively. Angles 2α and 2β in Fig. 7.2 and 7.3 are angles subtended by the grain boundary on



7.2: Sintering geometry with shrinkage



7.3: Sintering geometry without shrinkage

particle centers of radii R_1 and R_2, respectively. The following equations describe the scheme to obtain the geometrical parameters for the two cases of sintering.

For the case of neck growth with shrinkage, the geometry consists of two intersecting particles. The mass of particle intersection and neck is shown as dark and light gray color in Fig. 7.2. The mass of particle intersection is transported to the neck for the case of neck formation with shrinkage.

In case of neck growth without shrinkage, the mass from the particle surfaces is transported

to the neck. Fig. 7.3 shows the mass transported to the neck by a gray color. The geometries for neck growth with and without shrinkage are used to derive the sintering kinetics.

The geometries described above provide the radii of curvature at the neck r_1, r_2 and the volume of the neck dV for calculating the sintering kinetics. The radii of curvature and volume of the neck are calculated by equating the volume added to neck and volume transported from the particles. For calculating the radius of curvature and neck volume, distance between particle centers d is used as a variable. The particle sizes R_1 and R_2 for a distance d can be given as:

$$\{R_1, R_2\} = \begin{cases} \left\{R_1' - \dfrac{{R_2'}^2(R_1' + R_2' - d)}{{R_1'}^2 + {R_2'}^2}, R_2' - \dfrac{{R_1'}^2(R_1' + R_2' - d)}{{R_1'}^2 + {R_2'}^2}\right\} & \text{without shrinkage} \\ \{R_1', R_2'\} & \text{with shrinkage} \end{cases} \tag{7.12}$$

where R_1' and R_2' are initial radii of the large and small particles, respectively. The particles radii R_1, R_2 can be used to evaluate the grain boundary radius of curvature as given by eq. 7.13.

$$R_{gb} = \frac{\gamma_{gb}}{\gamma_s}\left(\frac{R_1 R_2}{R_1 - R_2}\right) \tag{7.13}$$

The center of the sphere forming the grain boundary is located at distance d_{gb} from the center of the large particle. The value of d_{gb} can be given as

$$d_{gb} = \begin{cases} R_1 + R_{gb} & \text{without shrinkage} \\ \dfrac{1}{2d}(d^2 - R_2^2 - R_1^2) + \sqrt{(\dfrac{1}{2d}(d^2 - R_2^2 - R_1^2))^2 + R_{gb}^2 - R_1^2} & \text{with shrinkage} \end{cases} \tag{7.14}$$

For a given neck size X, the radii of curvature can be calculated as

$$r_1 = \frac{1}{2}\frac{d_{gb}^2 + R_{gb}^2 - 2d_{gb}R_{gb}\cos\theta - R_1^2}{R_1 - (d_{gb} - R_{gb} - \cos\theta)\cos(\frac{\phi_e}{2} - \theta) - R_{gb}\sin\theta\sin(\frac{\phi_e}{2} - \theta)} \tag{7.15}$$

$$r_2 = \frac{1}{2}\frac{(d - d_{gb})^2 + R_{gb}^2 - 2(d - d_{gb})\cos\theta - R_2^2}{R_2 - (d - d_{gb} + R_{gb}\cos\theta)\cos(\frac{\phi_e}{2} + \theta) - R_{gb}\sin\theta\sin(\frac{\phi_e}{2} + \theta)} \tag{7.16}$$

In the above equations, angle θ can be calculated as

$$\theta = \arcsin\frac{X}{R_{gb}} \tag{7.17}$$

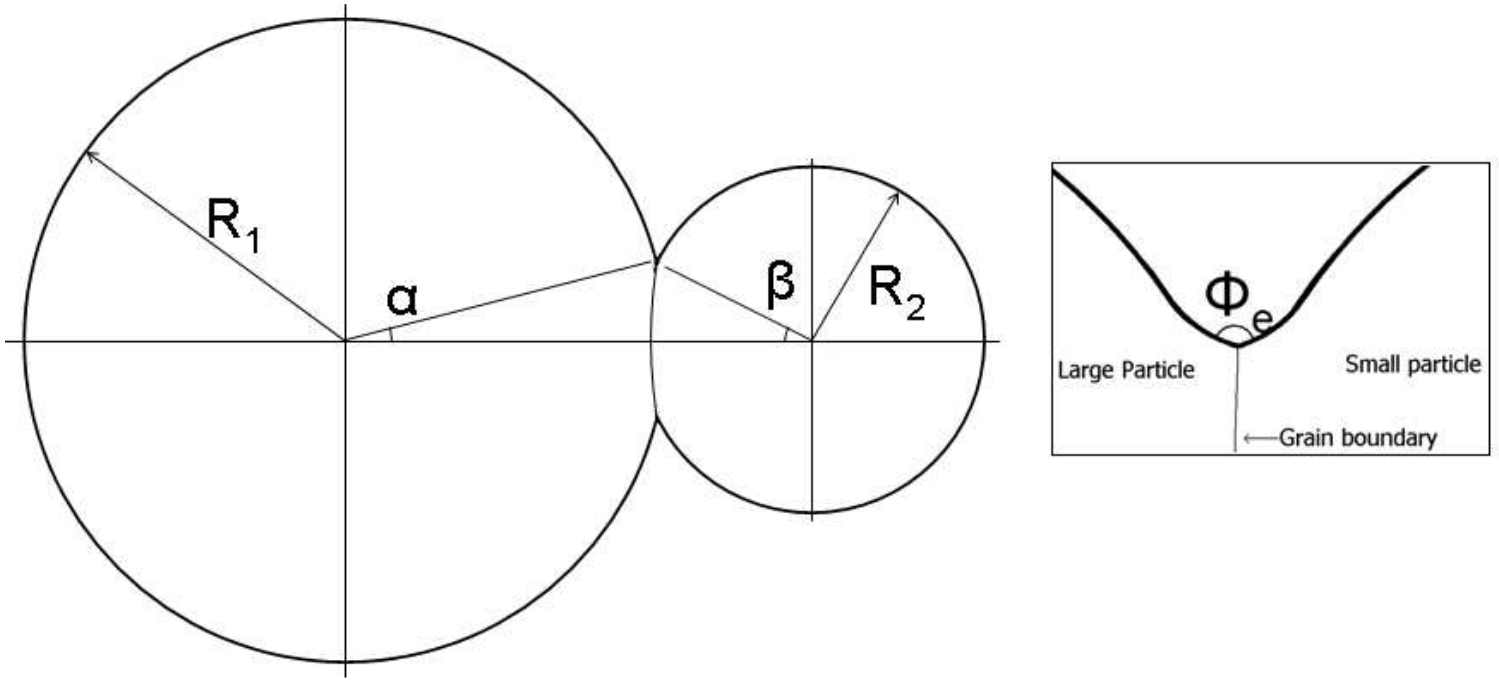
A relation between the neck size X and distance between particle centers d cannot be expressed in closed form for the assumed sintering geometry. The neck size X is numerically determined by a mass conservation condition such that

$$\begin{aligned}\frac{4\pi}{3}({R_1'}^3 + {R_2'}^3) = \pi {R_1}^3\left(\frac{2}{3} + \cos\alpha - \frac{\cos^3\alpha}{3}\right) + \pi {R_2}^3\left(\frac{2}{3} + \cos\beta - \frac{\cos^3\beta}{3}\right) \\ + \pi\int_{R_1\cos\alpha}^{R_1 + r_1(\cos\alpha - \cos(\phi_e/2-\theta))}\left((R_1 + r_1)\sin\alpha - \sqrt{r_1^2 - x^2}\right)^2 \mathrm{d}x \\ + \pi\int_{R_1 + r_1(\cos\alpha - \cos(\phi_e/2-\theta))}^{d - R_2\cos\beta}\left((R_2 + r_2)\sin\beta - \sqrt{r_2^2 - (x-d)^2}\right)^2 \mathrm{d}x\end{aligned} \tag{7.18}$$

The values of geometrical parameters neck size X, radii of curvature at the neck r_1 and r_2, angles α, β and θ can be obtained from the above calculations. These parameters are used to derive the sintering kinetics.

7.1.2.2 Coarsening and grain boundary migration geometry

During the coarsening stage, the small particle shrinks. The neck size is assumed to be the same as achieved at the end of the neck growth stage. Due to the assumption of the invariable neck size, the geometry used for neck growth cannot be used for coarsening. The two particle geometry used for coarsening is similar to the geometry of neck growth with shrinkage. The geometry consists of two intersecting particles as shown in Fig. 7.4. The particle surface intersect at equilibrium dihedral angle at the junction of particle surface and grain boundary. The region between particle surface and grain boundary junctions is assumed



7.4: Geometry for coarsening. The inset shows the local geometry at the grain boundary - surface junction

to be very small. This assumption is equivalent to neglecting circular profiles at the neck. Therefore, the local geometry at the particle surface - grain boundary has been ignored. The particle size change during the coarsening caused the sintering geometry to evolve. During the geometrical evolution, the volume of the particles change while maintaining the spherical shapes shown in Fig. 7.4. For calculation of geometrical parameters, the volume of the small particle V_2 can be calculated from mass conservation in eq. 7.19.

$$V_2 = \frac{4\pi}{3}\left({R_1'}^3 + {R_2'}^3\right) - V_1 \tag{7.19}$$

where V_1 is the volume of the large particle. The radii of the large and the small particles are found such that eq. 7.20 and 7.21 are simultaneously satisfied.

$$V_1 + V_2 = \pi\left(\left(\cos\alpha - \frac{\cos^3\alpha}{3} + \frac{2}{3}\right){R_1}^3 + \left(\cos\beta - \frac{\cos^3\beta}{3} + \frac{2}{3}\right){R_2}^3\right) \tag{7.20}$$

$$V_1 = \pi\left(\left(\cos\alpha - \frac{\cos^3\alpha}{3} + \frac{2}{3}\right){R_1}^3 - \left(\cos(\pi-\theta) - \frac{\cos^3(\pi-\theta)}{3} + \frac{2}{3}\right){R_{gb}}^3\right) \tag{7.21}$$

where R_{gb} is defined by eq. 7.13. The angles α, β, and θ in eq. 7.20 and 7.21 are defined as

$$\alpha = \arcsin\frac{X}{R_1} \tag{7.22}$$

$$\beta = \arcsin\frac{X}{R_2} \tag{7.23}$$

$$\theta = \arcsin\frac{X}{R_{gb}} \tag{7.24}$$

7.1.3 Sequence of the sintering stages

The sintering starts with two touching particles. The neck and the grain boundary between the particles form in the beginning of the sintering. The growth of the neck and the grain boundary is driven by the curvature difference at the neck and at the particle surfaces. The coarsening may occur during the neck growth stage; however, the rate of coarsening may

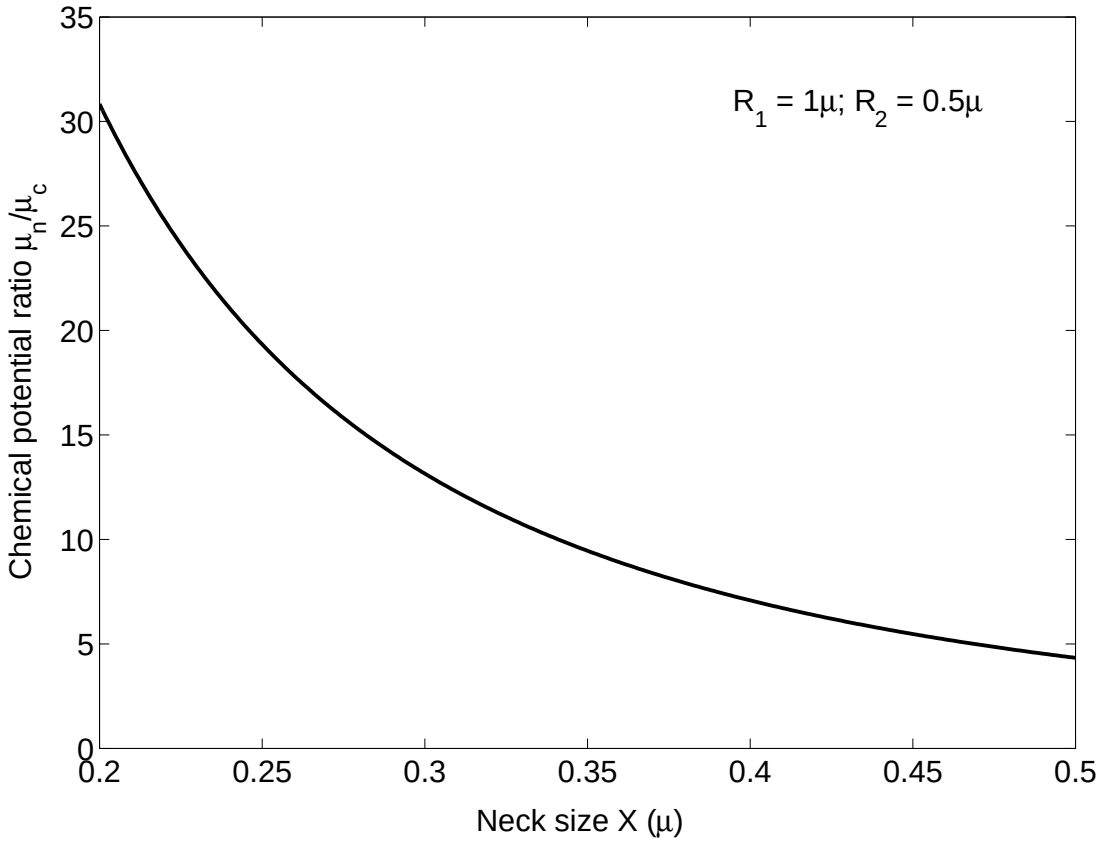
be very low. This is due to the difference in chemical potential for neck growth $\Delta\mu_n$ and chemical potential for coarsening $\Delta\mu_c$. The chemical potential difference for the neck growth and the coarsening can be calculated from eq. 7.25 and 7.26, respectively.

$$\Delta\mu_n = \gamma_s\Omega\left(\frac{2}{\frac{X^2}{4R_1'}+\frac{X^2}{4R_2'}}-\frac{1}{X}\right) \tag{7.25}$$

$$\Delta\mu_c = \gamma_s\Omega\left(\frac{2}{R_2}-\frac{2}{R_1}\right) \tag{7.26}$$

The ratio of the chemical potential difference for the neck growth to coarsening is shown in Fig. 7.5. Fig. 7.5 shows that the chemical potential for neck growth is several orders of magnitude higher than that of coarsening. This supports the arguments that a negligible coarsening occurs until neck grows significantly.

The driving force for the neck growth results from the difference in curvature at the neck and the particle surfaces. This driving force approaches zero as the mean radii of curvature



7.5: Ratio of driving force for neck growth to driving force for coarsening

at neck $(r_1+r_2)/2$ approaches infinite. When mean radius of curvature at the neck $(r_1+r_2)/2$ approaches infinity, the neck growth ceases and coarsening dominates.

During coarsening, the neck size does not change. The neck between the particles provides a path for interparticle diffusion. The coarsening stage may overlap with the slow grain boundary migration. The slow grain boundary migration occurs when the subtended angle on small particle β is less than $\pi/2$.

When angle β becomes larger than $\pi/2$, the grain boundary migrates rapidly. This rapid grain boundary migration involves short range atomic jumps as opposed to long range diffusion of coarsening. Therefore, a negligible coarsening is expected to occur simultaneously with rapid grain boundary migration.

7.2 Application of geometrical model

The geometrical model described above has been applied to sintering of two equal particles, two unequal sized particles and sintering of a linear array of particles. Neck growth and coarsening by surface diffusion is considered to be mass transport mechanism during sintering. The results of the sintering kinetics are compared with existing sintering theories and experimental data produced in the author's research group. The kinetics results of two equal sized particles are compared with existing sintering theories. The kinetics results of two unequal sized particles are compared with Monte Carlo and phase field simulation results. The sintering kinetics of linear array of particles are compared with experimental data obtained from sintering of tungsten powders.

7.2.1 Two equal sized particles

The geometrical model described in Section 2.2 is applied to sintering of two equal sized particles. The sintering kinetics of equal sized particles are used to compare the model against sintering theories. These sintering theories are described in Section 2.2. The parameters used to obtain the sintering kinetics are given in Table 7.2. The geometrical model describes sintering process in three stages. These stages are (1) neck growth, (2) coarsening, and (3) grain boundary migration. The sintering of equal sized particle occurs only by neck growth. The stages of coarsening and grain boundary migration are absent in sintering of equal particles.

The neck growth kinetics for particles of 1 μ are obtained for the two cases: (1) neck growth with shrinkage and (2) neck growth without shrinkage. The neck growth with shrinkage can be accomplished by a mass transporting mechanism of grain boundary diffusion. The

7.2: Materials properties and sintering parameters

Materials property/sintering parameter	Value
Sintering Material	Tungsten (W)
Surface energy per unit area γ_s	2.8 joul/m^2 [75]
Grain boundary energy per unit area γ_{gb}	2.37 joul/m^2
Equilibrium dihedral angle ϕ_e	130° [76]
Surface diffusion coefficient D_s	$4.0 \times 10^{-4} * \exp\left(\frac{-300000}{RT}\right)$ m^2/s [77]
Grain boundary diffusion coefficient D_{gb}	$54\times10^{-4}*\exp\left(\frac{-504000}{RT}\right)$ m^2/s [77]
Gas constant R	8.3114 joul.mol^{-1}K^{-1}
Surface diffusion layer thickness δ_s	0.3 nm [78]
Grain boundary diffusion layer thickness δ_s	1 nm
Molar volume Ω	9.55×10^{-6} m^3/mol
Grain boundary mobility m	3.5×10^{-21} m^{4}s^{-1}Pa^{-1} [79]
Sintering temperature T	950 °C

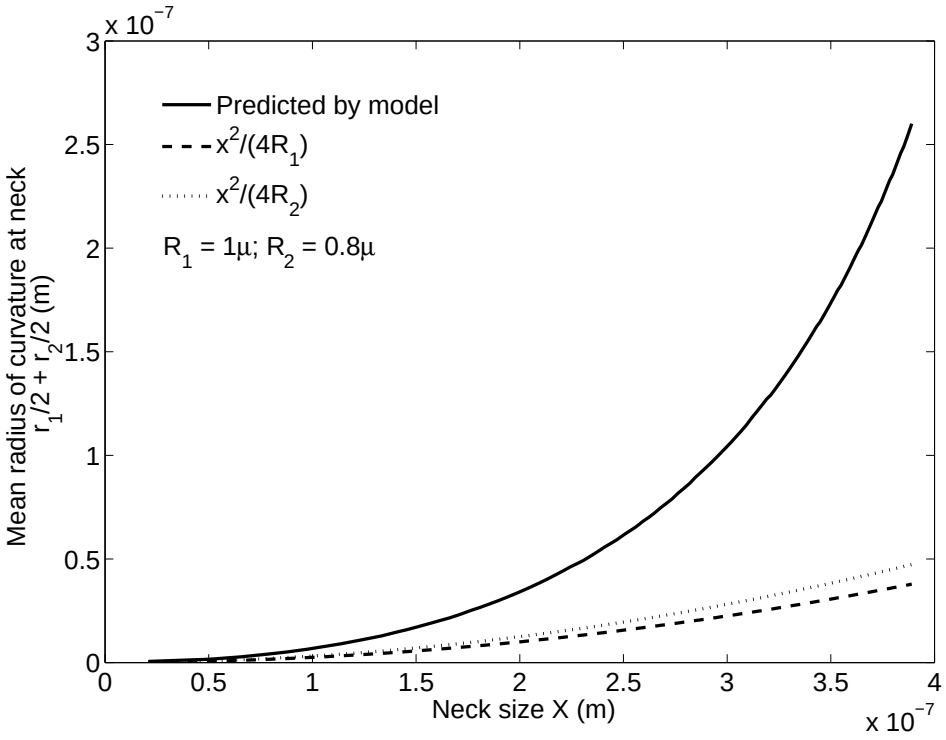
mass transporting mechanism for neck growth without shrinkage is considered to be surface diffusion.

7.2.1.1 Neck growth with shrinkage

The neck growth with shrinkage may occur due to a mass transport from the particles through the grain boundary. The driving force for the neck growth originates from curvature difference at the neck and at the center of the grain boundary. Fig. 7.6 shows the variation of the radius of curvature r_1 at neck as a function of the neck size X. The figure also shows the classical approximation of the radius of curvature $r_1 = X^2/(4R_1)$. The classical approximation is valid for small neck sizes due to assumptions made during derivation the expression. Both of the estimations of radius curvature provide similar radius of curvature for small neck sizes. The difference in the prediction of scheme of Section 7.1.2.1 and the classical approximation increases at larger neck sizes. The classical approximation shows that the radius of curvature increases slowly and attains a maximum value of $R_1/4$ when the neck size become the particle size ($X = R_1$). On the other hand, the scheme described in Section 7.1.2.1 show that the radius of curvature approaches infinite when neck size to particle size ratio (X/R_1) approaches 0.66 for equal size particles ($R_1 = R_2$).

The radius of curvature and the volume transported calculated following the scheme described in Section 7.1.2.1 are used to derive the neck growth kinetics. The kinetics derived using equations described in Section 7.1 are shown in Fig. 7.7.

Fig. 7.7 shows that the neck growth kinetics are rapid as neck growth starts. The neck



7.6: Radius of curvature at neck for case of neck growth with shrinkage

grows to 30% of the particle radius in almost no time. The rate of neck growth slows down and neck grows slowly to about 40% of the particle radius. It should be noted that the neck growth ceases when mean radius of curvature at the neck $r_1/2 + r_2/2$ approcaches the neck size X. When $2/(r_1 + r_2)$ becomes equal to $1/R_1 + 1/R_2 - 1/X$, the neck growth ceases as driving force for the neck growth reduces to zero.

Fig. 7.7 also shows the neck growth kinetics for classical model of Coble. The sintering kinetics of Coble's model are given in eq. 7.27. Fig. 7.7 shows that neck growth kinetics in Coble's model are faster than predictions of geometrical model developed in the dissertation. It should be noted that the Coble's model is only valid for small neck sizes. If Coble's model is used for large neck sizes, the models predicts that the neck should grow indefinitely. Nevertheless, Coble's model can be compared against the geometrical model for small neck sizes with help of Fig. 7.8. Fig. 7.8 shows sintering kinetics on a log-log scale.

$$t = \frac{X^6}{48 D_{gb} \delta_{gb} \gamma_S \Omega R_1^2 / RT} \tag{7.27}$$

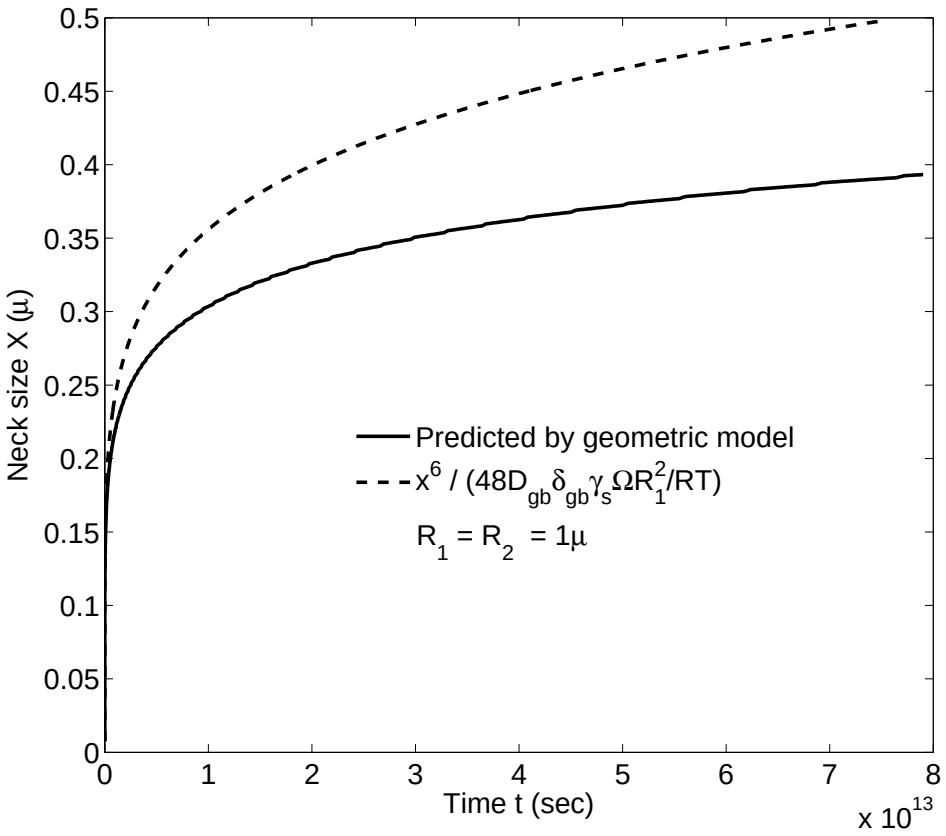
The slopes of curves in Fig. 7.8 indicate the exponent between time and neck size terms in eq. 7.27. It can be seen that the slopes of Coble's equation and the geometrical model are

similar for small neck sizes. The exponent of the geometrical model increases for large neck sizes indicating a reduction in neck growth rate.

7.2.1.2 Neck growth without shrinkage

The neck growth without shrinkage may occur by a transport of material through particle surfaces. The particle surface acts as a source of material whereas the neck acts as a sink. The mass transport from particles causes a reduction in the particle sizes. This particle size change is negligible. However, due to this size change, this case of neck growth does not occur without absolute shrinkage. The neck growth kinetics without shrinkage are calculated using scheme described in Section 7.1. The scheme of deriving sintering kinetics require an estimation of the radius of curvature at the neck. Fig. 7.9 show the estimated radius of curvature at the neck.

The estimation in Fig. 7.9 is based on methods described in Section 7.1.2.1. Fig. 7.9 also shows the conventional approximation of the neck size ($r = X^2/2R$). It can be observed that the prediction of the radius of curvature using numerical methods is similar to conventional approximations for small neck sizes. The differences in the prediction of the numerical method



7.7: Neck growth kinetics with shrinkage for equal sized particles

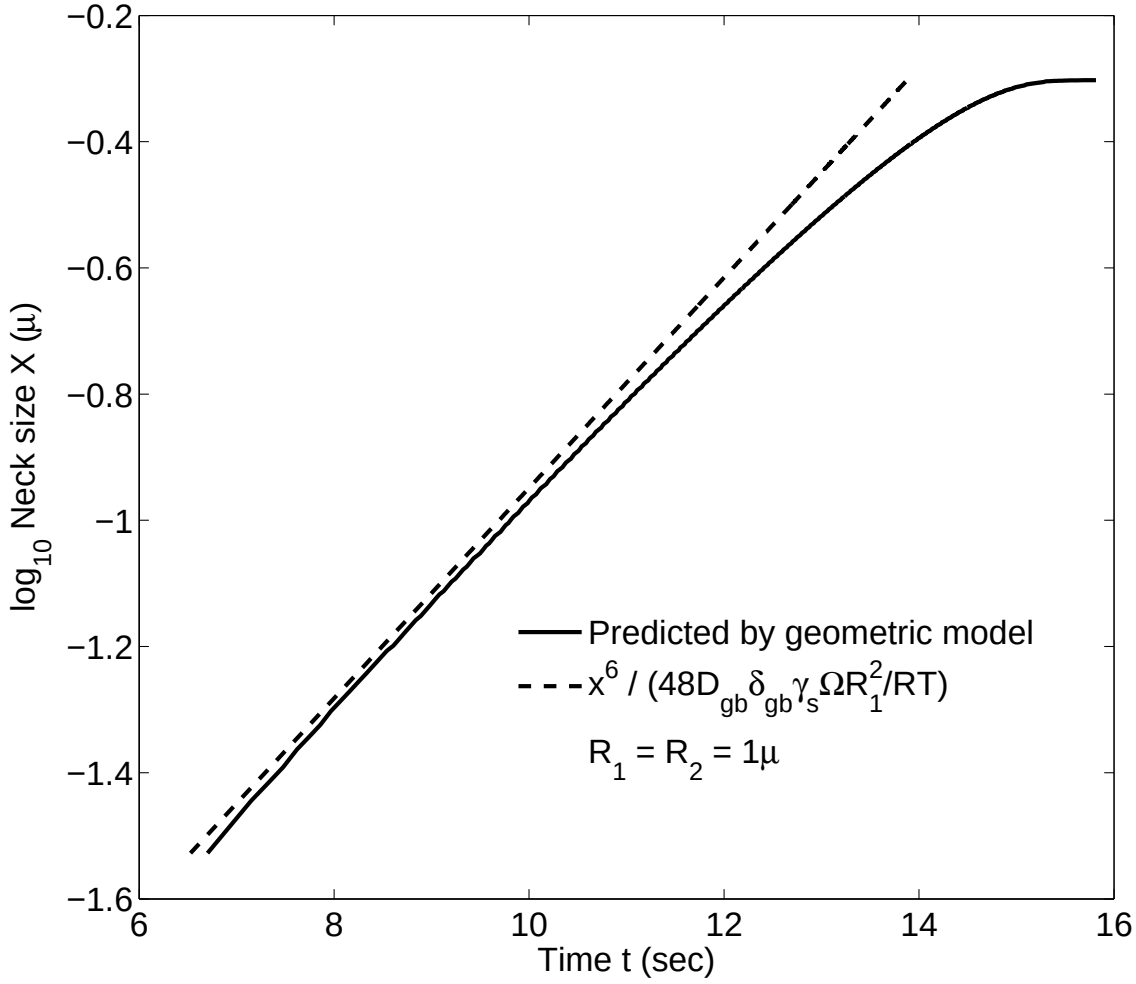
and conventional approximation increases as neck size increases.

The radius of curvature at the neck is used to evaluate the neck growth kinetics for equal particle of size 1μ. Fig. 7.10 shows the numerically evaluated neck growth kinetics.

Fig. 7.10 shows that neck growth kinetics slows down as neck grows. The numerical estimation of the neck growth kinetics are compared with Coble's model. The neck growth kinetics of Coble's equal particle model are given eq. 7.28.

$$t = \frac{X^7}{56 D_s \delta_s \gamma_S \Omega R_1^3 / RT} \tag{7.28}$$

Fig. 7.10 show that Coble's model predicts a faster neck growth kinetics. The differences of



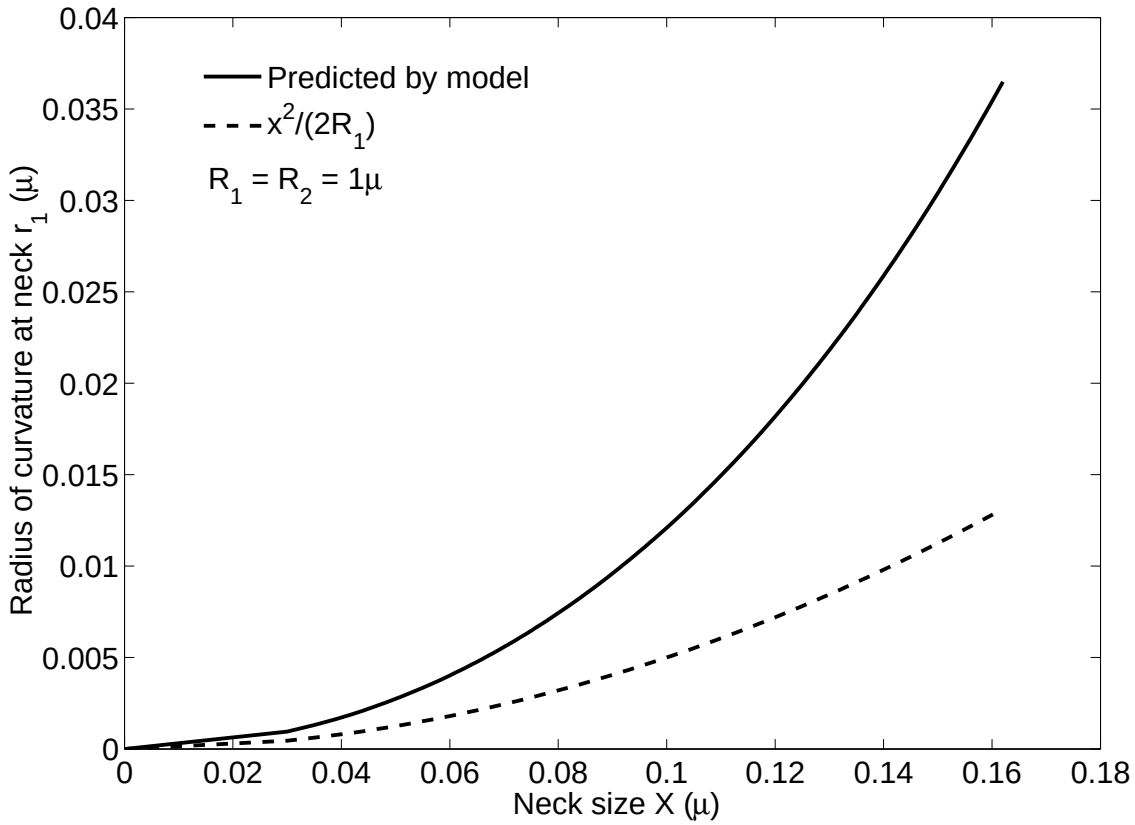
7.8: Neck growth kinetics with shrinkage for equal sized particles on log-log scale

the analytical model and numerical estimation can be highlighted in the log-log plot. Fig. 7.11 shows the log-log plot of the sintering time and neck size.

The slope of sintering kinetics in the log-log plot refers to the exponent between time and neck size terms in kinetics equation. Fig. 7.11 and Eq. 7.28 show that the conventionally used Coble's model predicts an exponent of 7. The neck growth kinetics obtained from the geometrical model show that the slope of sintering kinetics in Fig. 7.11 decrease. This indicates that the exponent increases as the neck grows. The increase in the exponent results in slower neck growth kinetics. Therefore, the rate of the neck growth reduces as the neck grows.

7.2.2 Two unequal sized particles

The geometrical model was applied to the sintering of two unequal sized particles of initial size 100 nm and 50 nm. The choice of particle sizes resulted in a ratio of 2:1. This particle size ratio is the same as the particle size ratio in Monte Carlo simulation and phase field simulations. The sintering parameters and materials properties used for obtaining the



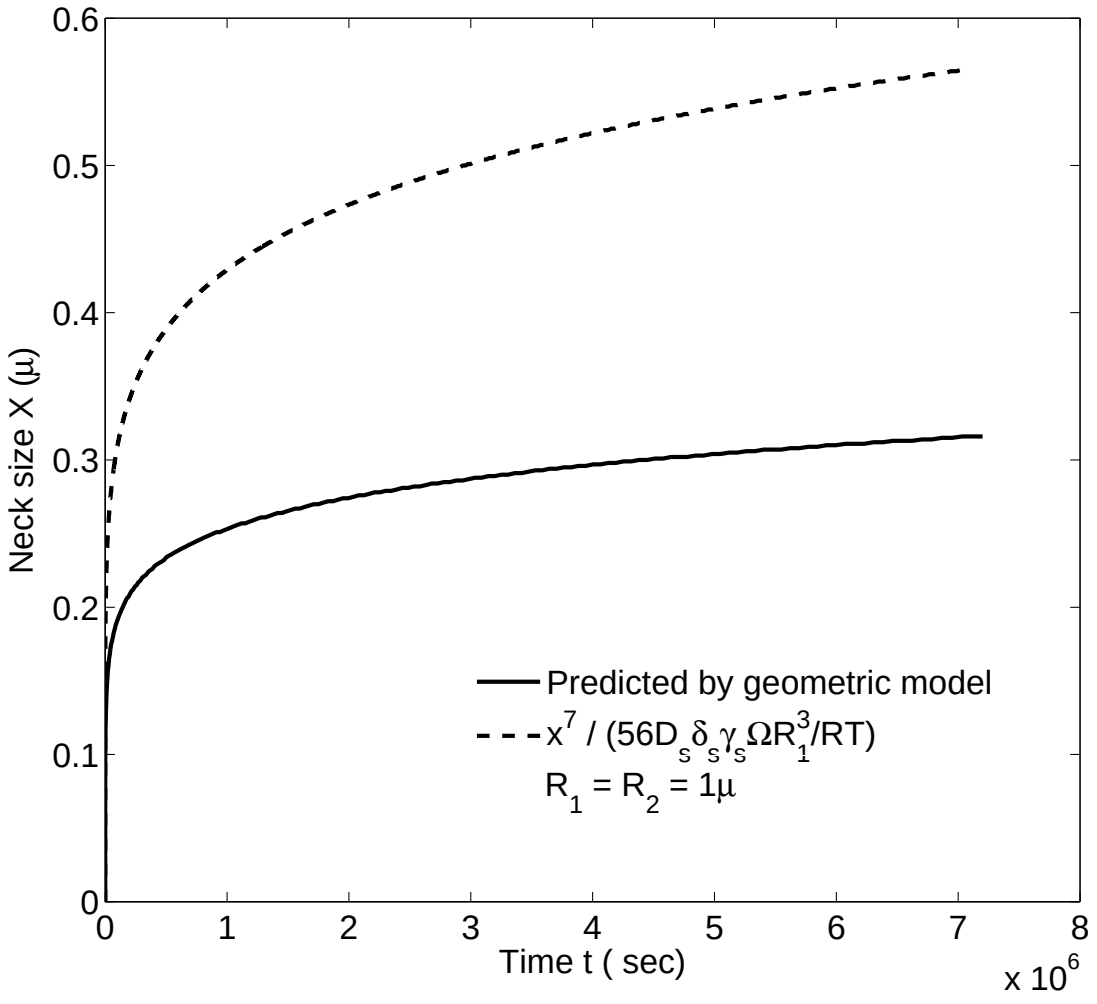
7.9: Radius of curvature at neck for equal size particles for neck growth without shrinkage

sintering kinetics are given in Table 7.2.

The microstructural evolution obtained using geometrical model is shown in Fig.7.12.

The sintering of two unequal sized particles began with two touching particles as shown in Fig. 7.12(a). Fig. 7.12(b) depicts that a neck between the particles formed rapidly. The neck growth slowed down as neck grew. This can be inferred from Fig. 7.12(c) which shows that neck growth was completed in 4.5 hours. The coarsening continued until 32 hours resulting in the particles size change. Microstructure in Fig. 7.12(e) shows the onset of grain boundary migration. A grain boundary migration resulted in rapid shrinkage of small particle as shown in Fig. 7.12(f) - 7.12(g).

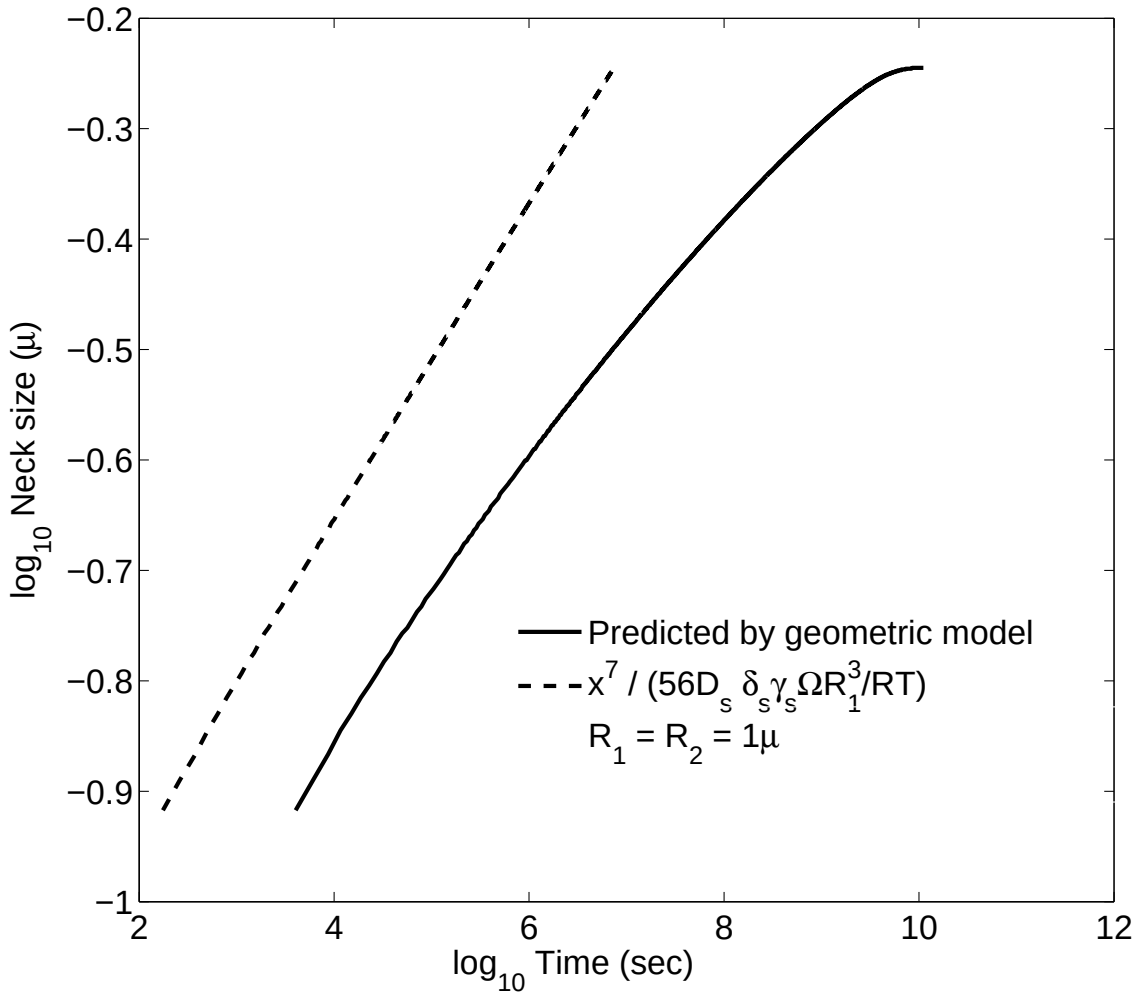
The geometrical changes during sintering of two unequal sized is shown quantitatively in Fig. 7.13. Neck size evolution in Fig. 7.13 shows that the neck grew to about 60% of its maximum size instantaneously. However neck growth slowed down towards the end of neck



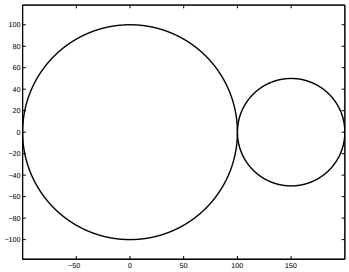
7.10: Neck growth kinetics without shrinkage for equal sized particles

growth subprocess. The neck size remained constant during coarsening subprocess. A reduction in small particles size can also be observed in Fig. 7.13. The rate of coarsening increased with reduction in particle size. A slow grain boundary migration occured towards the end of sintering process. This slow grain boundary migration resulted in an increase in grain boundary area. The slow grain boundary migration was followed by a rapid grain boundary migration. The grain boundary area was reduced by rapid grain boundary migration until the small particle disappeared.

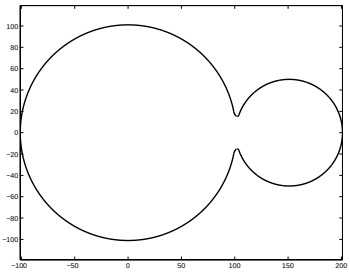
The sintering kinetics obtained using geometrical model can be compared with kinetics obtained using Monte Carlo and phase field simulation methods described in Chapters 5 and 6, respectively. A similarity between microstructural evolution obtained from geometrical model (Fig. 7.12) and microstructural evolution obtained from simulation methods (Fig. 5.1 and 6.2) can be noticed. A quantitative comparison of neck size and small particle size



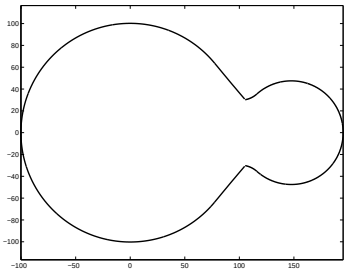
7.11: Neck growth kinetics without shrinkage for equal sized particles



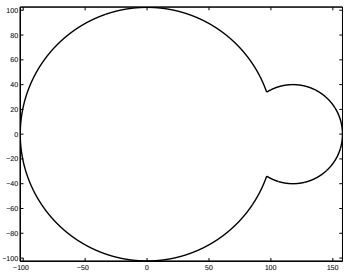
(a) time 0 second



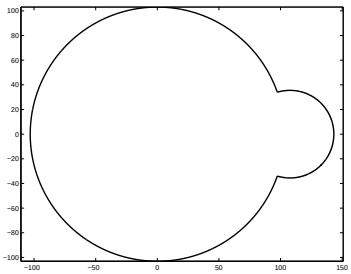
(b) time 4 seconds



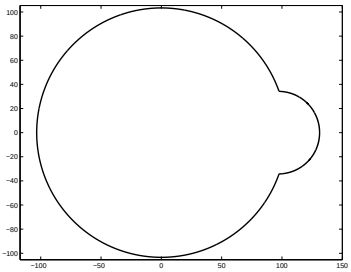
(c) time 4.5 hours



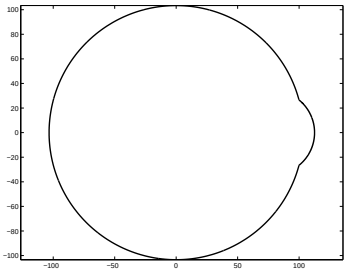
(d) time 26 hours



(e) time 32 hours

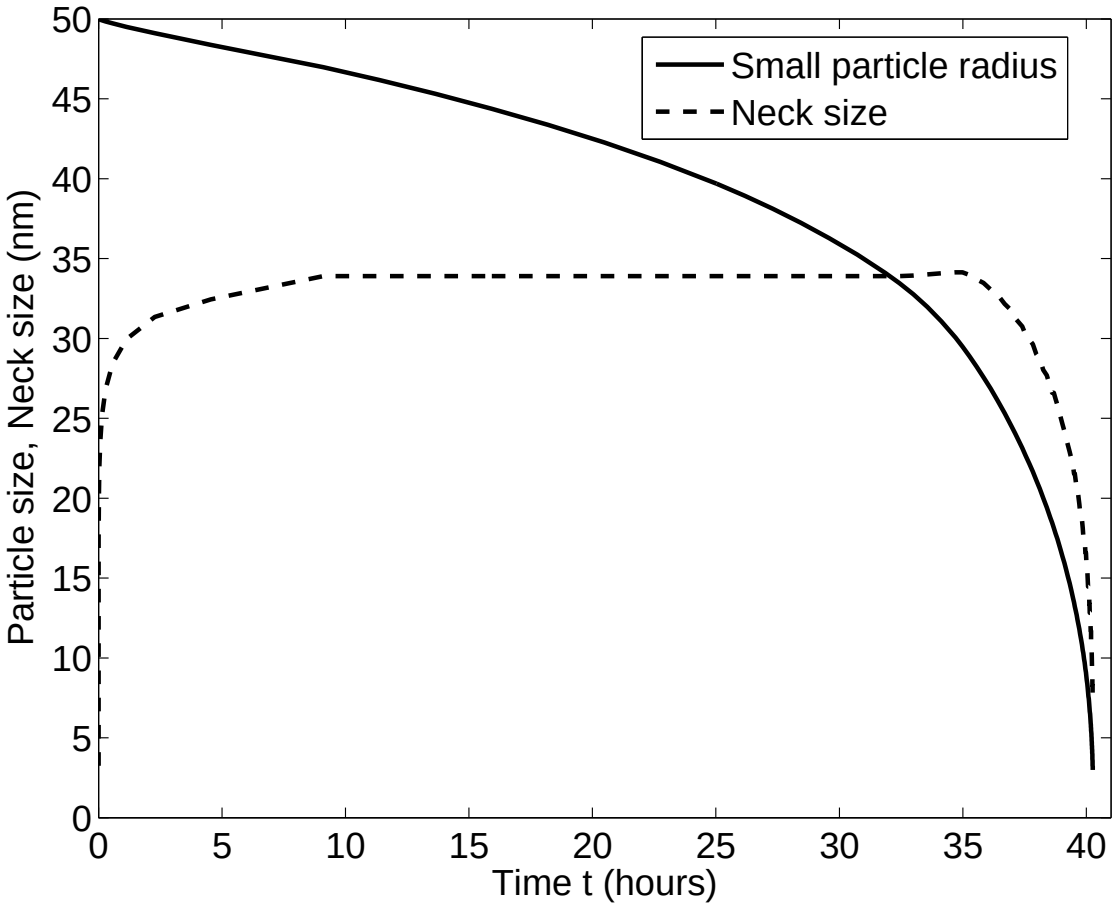


(f) time 35 hours



(g) time 39 hours

7.12: Microstructure evolution of two unequal sized particles obtained using geometrical model. The distances in the subfigures are in units of nanometers. Time below subfigures indicates the sintering time.



7.13: Small particle size and neck size during sintering of two unequal sized particles

evolution obtained from geometrical model and simulations in Fig. 7.13, 5.2 and 6.3 indicate that all three methods are in good agreement.

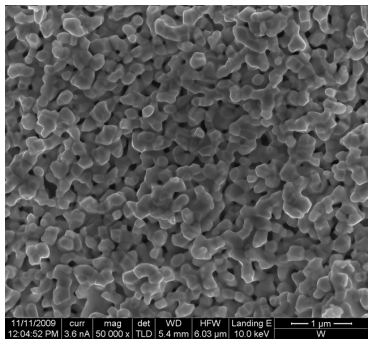
7.2.3 Linear array of particles

The geometrical model was extended to sintering of a linear array of particles. The extension of the geometrical model includes simultaneous subprocesses or stages of neck growth, coarsening, and grain boundary migration. The simultaneous subprocesses require a suitable time interval for microstructural evolution. The suitable time interval ensures that the geometrical changes due to subprocess of neck growth, coarsening and grain boundary migration are small. The suitable time interval can be adaptively chosen for estimation of geometrical changes. The results of geometrical model were compared against experimental sintering results of tungsten powders. These experiments were performed in author's research group. Details of sintering experiments on tungsten powders can be found in ref. [80]. The results of the sintering experiments and predictions of geometrical model are given below.

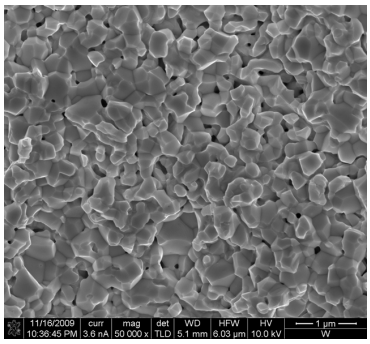
The sintering experiments at various temperatures were performed on nano sized tungsten powders. Fig. 7.14 shows the microstructural evolution during sintering at 1050 °C and

various times. The initial density for the sample was 69% of theoretical density.

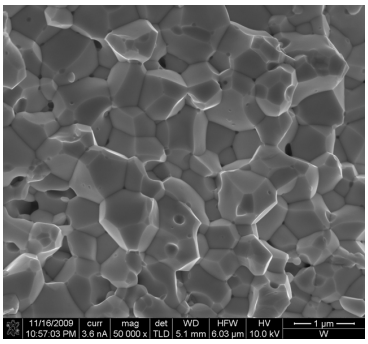
Fig. 7.14(a) shows the microstructure after 5 minutes of sintering. This microstructure can be considered to be in initial stage of sintering (cf. 2.2.1). Necks between particles can be observed in Fig. 7.14(a). The pores between the particles make an interconnecting network. The porosity between the particles is open porosity. Fig. 7.14(b) shows that pores are located at grain edges. This signifies the intermediate stage of sintering (cf. 2.2.2).



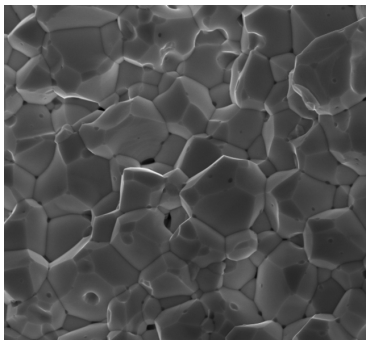
(a) time 5 min



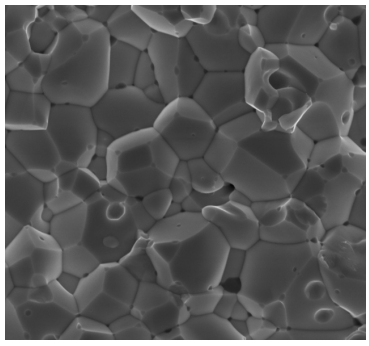
(b) time 15 min



(c) time 30 min



(d) time 45 min

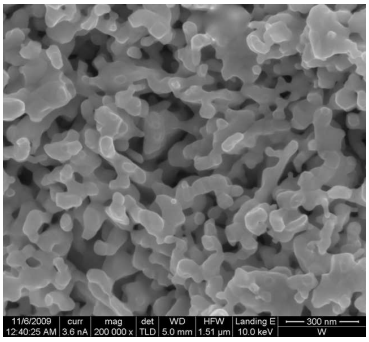


(e) time 60 min

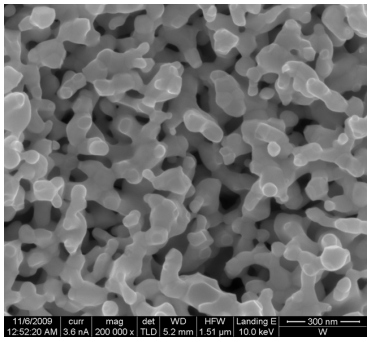
7.14: Microstructure evolution during sintering of tungsten powder at 1050 °C. Time below subfigures indicates the sintering time.

During intermediate stage of sintering, a vacancy transport from pores located at grain edges to grain faces occurs. Pores become rounded at sintering time of 45 minutes as shown in Fig. 7.14(d). The pore structure at this time becomes closed pore structure. This pore structure characterize the final stage of sintering.

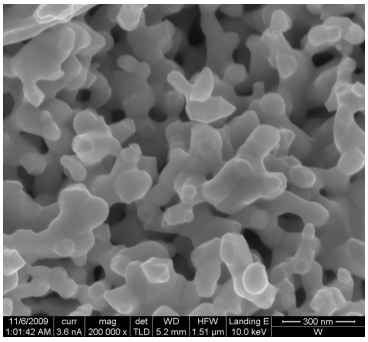
Fig. 7.15 shows a microstructure evolution of sintering at temperature 950 °C. The initial density in the beginning of sintering was 42% of theoretical density. The microstructure in



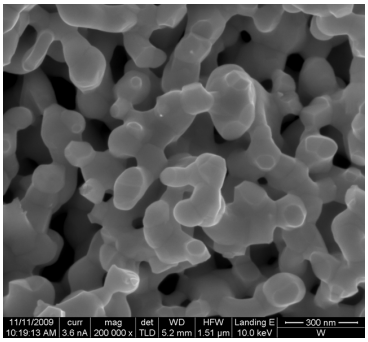
(a) time 5 min



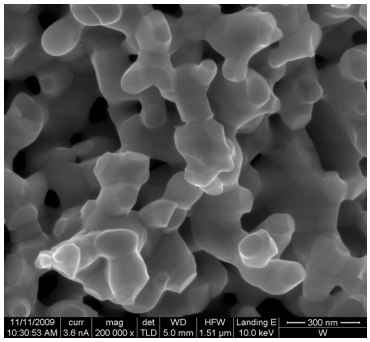
(b) time 15 min



(c) time 30 min



(d) time 45 min



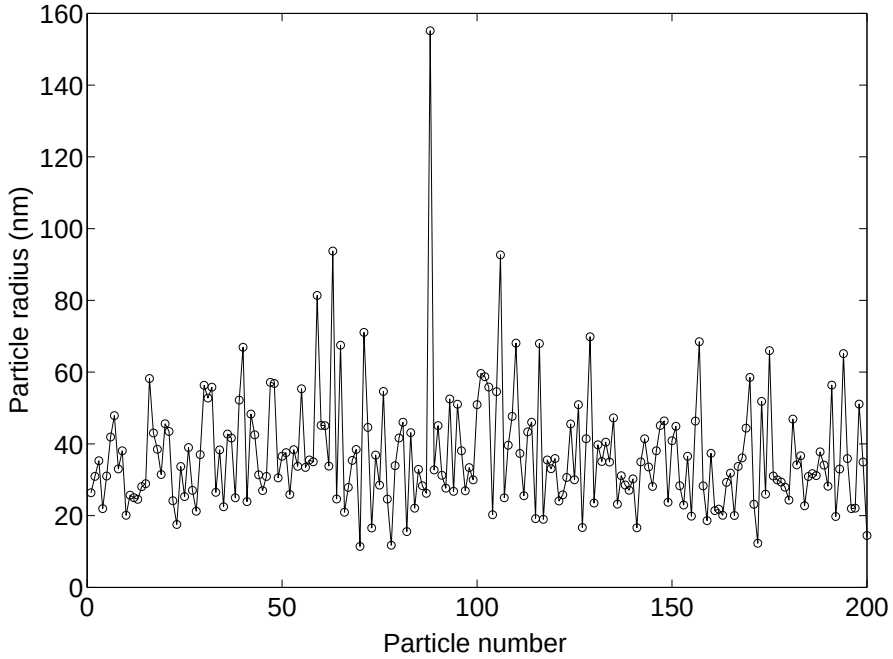
(e) time 60 min

7.15: Microstructure evolution during sintering of tungsten powder at 950 °C. Time below subfigures indicates the sintering time.

the beginning of the sintering (Fig. 7.15(a)) shows that particles are connected by necks. The pores at this time are interconnected. The interconnected pores form an open pore structure. Fig. 7.15(b) - 7.15(e) show that pore structure remains the same from 5-60 minute sintering times. Therefore, the microstructure containing particles connected by necks maintains morphology during sintering.

This maintenance of morphology results in similarity between the sintering of sample with 42% green density and sintering of linear array of partilces. In sintering of linear array of particles, the particles are connected by a neck. When a particle disappears due to grain boundary migration, a neck between the adjacent particles forms. After this neck formation, all of the particles are connected by necks and the microstructure maintains its morphology. These similarities of microstructures are the basis of comparison between sample with 42% green density and sintering of linear array of particles.

The geometrical model was applied to linear array of particles. The predictions of geometrical model on linear array of particles were compared with sintering of sample sintered at 950 °C. The initial particle size distribution for geometrical model was same as distribution of starting tungsten powder. Two hundred particles from the particle size distribution of tungsten powder were randomly chosen and arranged to form a linear array as shown in Fig. 7.16. The geometrical model was applied on this array of particles using materials properties and sintering parameters given in Table 7.2.



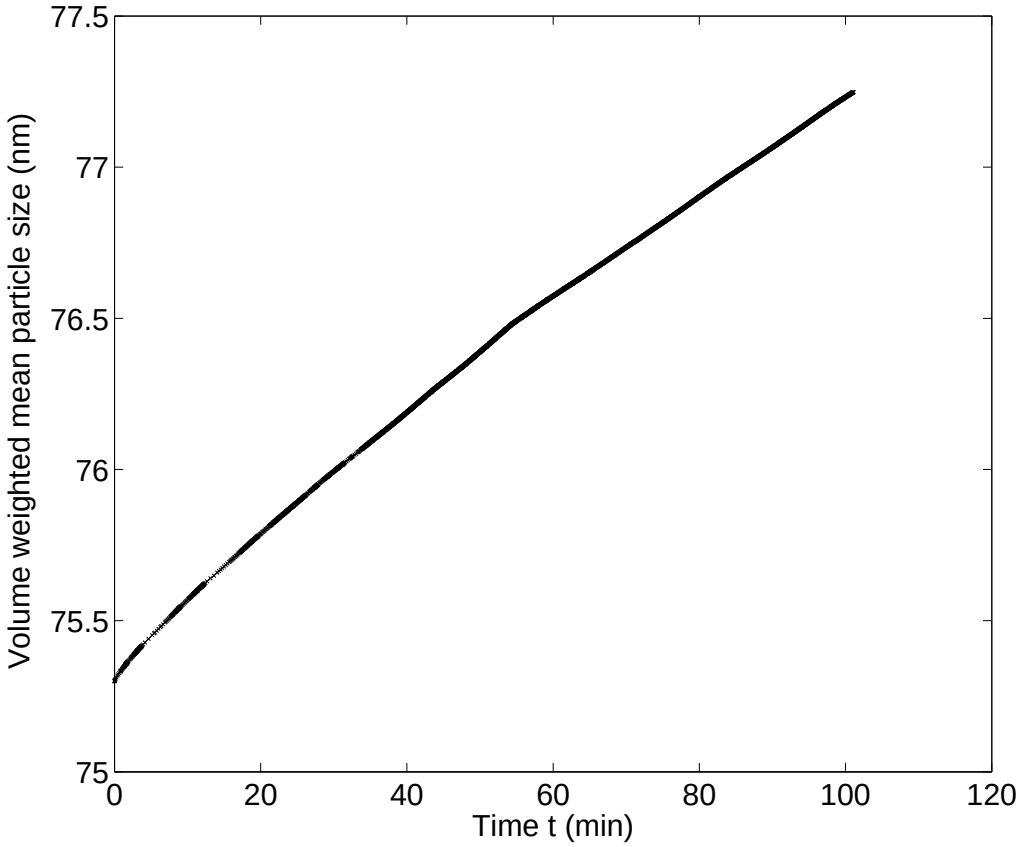
7.16: Initial particle arrangement used in application of geometrical model to linear array of particles

Figure 7.17 shows volume weighted mean grain size of linear array of particles during sintering. For particle size analysis during sintering, volume weighted mean grain size is chosen to describe particle size change. The algebraic mean grain size decreases for during neck growth and coarsening subprocesses. Volume weighted mean grain size R_v given by eq. 7.29 increases during the subprocesses of neck growth, coarsening and grain boundary migration.

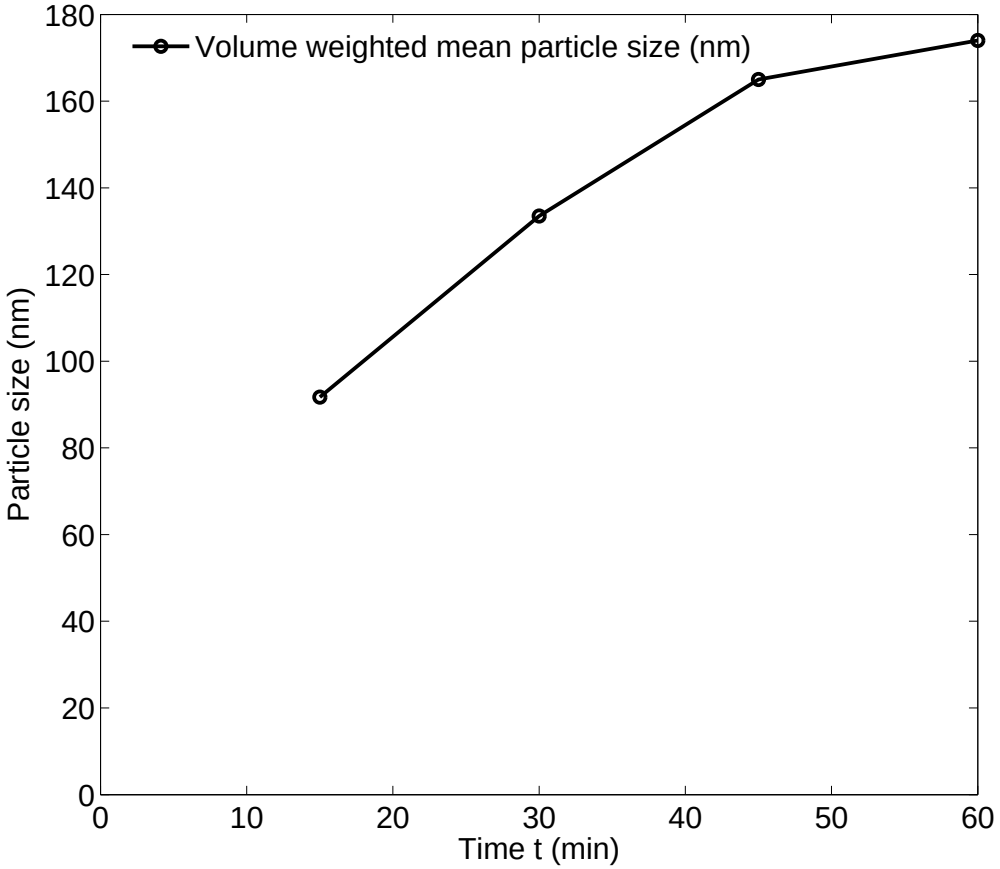
$$R_v = \frac{\sum_i V_i R_i}{\sum_i V_i} \tag{7.29}$$

where, V_i is the volume of i^{th} particle with radius R_i. Fig. 7.17 shows the volume weighted mean grain size obtained from geometrical model.

A comparison between particle size change obtained using experiments shown in Fig. 7.18 and using geometrical model (Fig. 7.17) indicate a qualitative agreement. However, a



7.17: Volume weighted mean grain size during sintering of linear array of particles



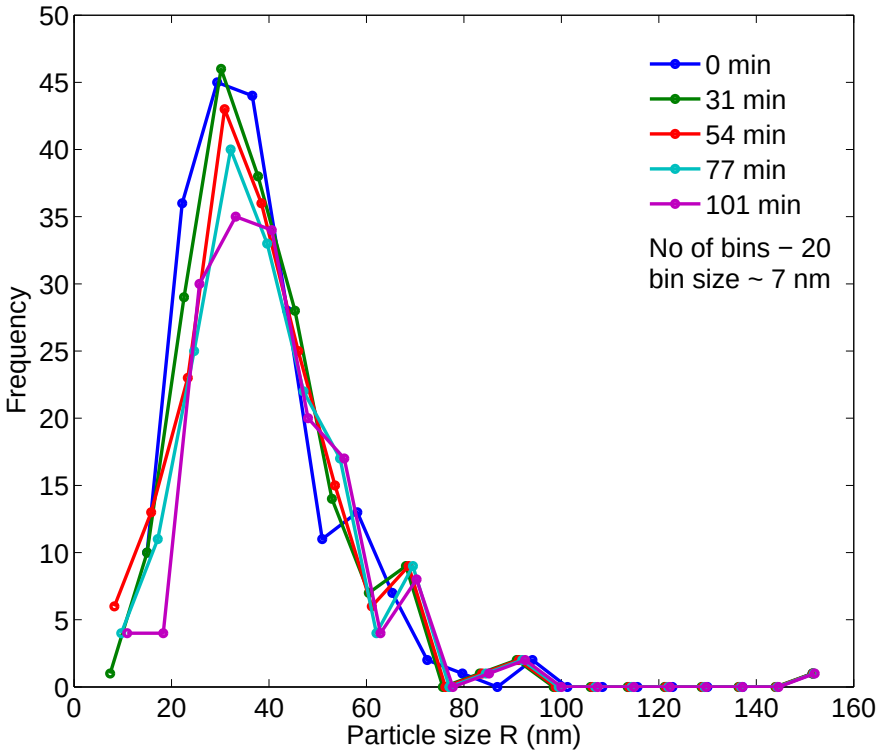
7.18: Mean grain size during sintering of tungsten powder at 950 °C

larger particle size change was experimentally observed. This difference can be attributed to following factors:

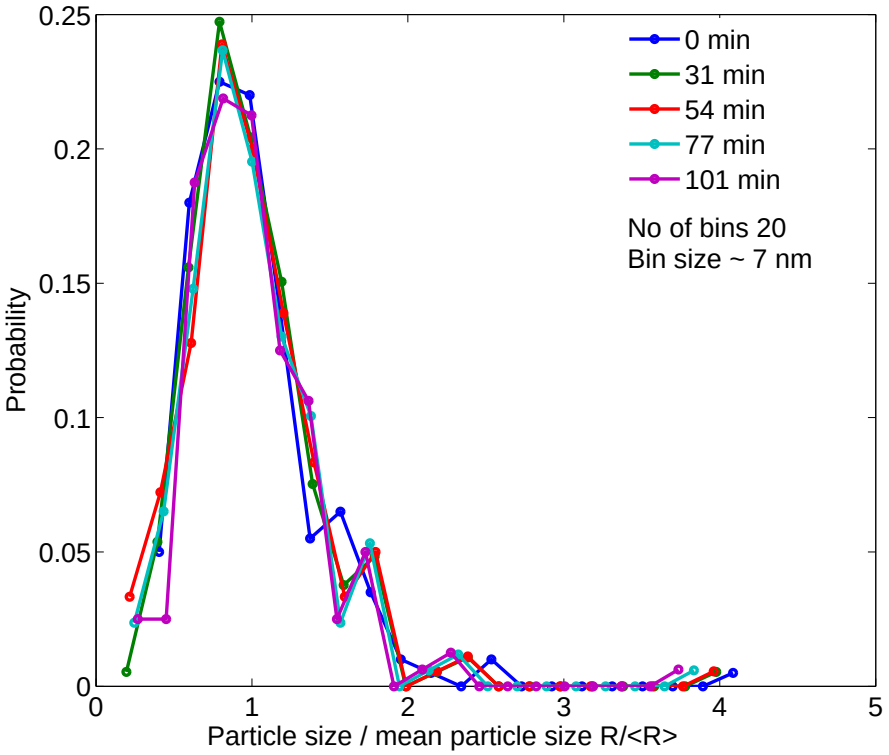
1. The sintering experiments involved three-dimensional network of particles resulting in larger area per particle for interparticle mass transport.

2. The sintering experiments involved a rearrangement process which was not considered in geometrical model.

The particle size distribution during sintering can also be used to compare with experimentally observed trends. Particle size distributions at various times are shown in Fig. 7.19. Fig. 7.19 shows that the grain size distribution shifted to the larger grain size as sintering progresses. This observation is expected due to the disappearance of small particles. A normalized grain size distribution is shown in Fig. 7.20. Normalized grain size distributions normalized with mean grain size show that shape of the grain size distribution does not change during coarsening. This observation is in agreement with experiments [81].



7.19: Grain size distributions during application of geometrical model to sintering of linear array of particles



7.20: Normalized grain size distributions during application of geometrical model to sintering of linear array of particles

CHAPTER 8

SUMMARY, CONCLUSION AND FUTURE WORK

In this dissertation, we successfully applied Monte Carlo method to simulate solid state sintering process. For the first time, we demonstrated three subprocesses of sintering of unequal sized particles with help of numerical simulations. These subprocesses are neck growth, coarsening, and grain boundary migration. In addition, we applied phase field method to simulate sintering of various particle geometries. The three subprocesses of sintering of unequal sized particles were again demonstrated by phase field simulation. We also observed two types of grain boundary migration, namely slow and rapid grain boundary migration. The rapid grain boundary migration has been quantitatively discussed in the literature; however, slow grain boundary migration that we observed has neither been observed experimentally nor proposed theoretically in sintering process. In this dissertation, we derived an equation to quantitatively describe the slow and rapid grain boundary migration.

In addition to unequal sized particles, we applied phase field method to simulate sintering of two equal sized particle, nanoparticles, closed packed array of particles, randomly distributed particles and pores. We found that phase field simulation of sintering of two equal sized particles follow neck growth kinetics of Kingery and Berg's model. We further extrapolated the neck growth kinetics to infer particle size effect in sintering of nanoparticles. We found that sintering of nanoparticles follows Herring's scaling law. We also studied the effect of pore distribution in sintering of regular array of particles using phase field simulation. We observed that a larger amount of porosity for a given solid volume results in higher densification rates. We attributed this higher densification rate to larger grain boundary area which acts as a path for mass transport. Further, we studied the effect of grain boundary mobility during sintering of randomly distributed particles. We deduced that a higher grain boundary mobility supports particle coarsening.

We did not find a model in the literature that can analytically explain the results obtained from Monte Carlo and the phase field simulation. Therefore, we developed a geometrical model to analytically describe the sintering of two unequal sized particle. This model is based

upon idealized geometries to describe three sintering subprocess: neck growth, coarsening, and grain boundary migration. For the first time, this model provided a quantitative description for the three subprocesses of the sintering. This model also successfully describes the overlap between the three subprocess. This overlap has been ignored in models reported in the literature.

Furthermore, the geometrical model developed in this dissertation is used to obtain neck growth kinetics of two equal size particles. We found that the developed model is capable of predicting the maximum neck size, a result that cannot be obtained from conventionally used geometric models. Further, an application of the model to unequal sized particles reproduced microstructural evolution similar to that obtained by Monte Carlo and phase field simulations. We also applied the geometrical model developed in this dissertation to a row of particles and compared the results with sintering experiments on tungsten powders. The geometrical model predicted a grain growth trend similar to the trend observed during sintering experiment. An analysis of particle size distributions obtained from the geometrical model predicts that the normalized particle size does not change during sintering. This prediction has been experimentally observed during sintering experiments.

8.1 Conclusion

We conclude the following based upon the studies performed in this dissertation:

1. The sintering of unequal sized particles involves three subprocesses: neck growth, coarsening, and grain boundary migration. These subprocesses transition from one to another with a finite overlap.

2. The coarsening is the rate determining subprocess during the sintering process.

3. The pressure difference between the particles drives the grain boundary migration. The change in grain boundary area may apply drag force or driving force on the grain boundary migration. The grain boundary migration is possible before grain boundary area reduction occurs.

4. The sintering of nanoparticles follow Herring's scaling law as long as material properties remain the same.

5. During sintering of a given volume of solid particles, reduction in particle size results in higher densification rates.

6. The grain boundary migration during sintering of randomly distributed particles supports the particle growth.

7. The rate of pore closure depends upon the pore coordination number.

8.2 Future Work

The sintering process in practice involves rearrangement of particles. This rearrangement of particles involves bonding and debonding of particles due to stress between particles. The rearrangement of particles during sintering has not been studied extensively due to difficulties in experimental observation as well as mathematical formulation. The numerical simulation method applied in this dissertation may be helpful in observing rearrangement of particles in sintering simulation. This observation could be helpful in mathematically describing the rearrangement phenomenon.

In addition to rearrangement, the sintering stress may change the rate of densification and coarsening. The geometrical model developed in this dissertation is applicable to only one-dimensional arrangement because this model does not incorporate the effect of stress between particles. The geometrical model presented in this dissertation may be improved by considering the effect of stress between particles on properties of the sintered product. This stress analysis should result in extension of the model to two and three dimensions, and therefore it can be more directly compared with sintering experiments.

APPENDIX A

MONTE CARLO SIMULATION

The code of Monte Carlo simulation of two unequal particle is given below:

Listing A.1: Algorithm for Monte Carlo simulation of sintering

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <time.h>

#define xDim 512
#define yDim 512
#define nsp 512*512
#define Q 5
#define liqFrac 1.0
#define solFrac 0.0

#define nNeigh 8
// constants for procedure ran3
#define MBIG 1000000000
#define MSEED 161803398
#define MZ 0
#define FAC (1.0/MBIG)

#define liqSolSurfEnergy 0.6;
#define solSolSurfEnergy 1.0;
#define kTss 0.0
#define kTsl 0.7

#define maxMCS 2000000000000
#define MCSPeriod 100000

struct LP{
  int Id;
  int neigh[8];
  int orient;
};
struct orientation{
  float phi1;
  float phi;
  float phi2;
};
```

```
void InitLP(struct LP *p);
double ran3(long *idum);
int q_ran(int q);
void MCGrowthBySolReppt(struct LP *p);
float SurfaceEnergyByPhase(int oId1, int oId2);
void OutputData(struct LP *p,struct orientation *O, int filenameExt
   );
void InitOrientTable(struct orientation *O);
void MakeCircle(struct LP *p, int x, int y, int radius, int cirOri)
   ;
main(){
  struct LP *particle;
  struct orientation *ori;
  int mcStep, i;
  particle = (struct LP *)malloc ((nsp)*sizeof(struct LP));
  ori = (struct orientation *) malloc ((Q+1)*sizeof(struct
     orientation));
  InitOrientTable(ori);

  InitLP(particle);
  MakeCircle(particle, 144, 256,128 , 1);
  MakeCircle(particle, 336, 256,64, 2);

  for (mcStep = 0; mcStep< maxMCS+1; mcStep++){
    for (i = 0; i<nsp; i++){
      MCGrowthBySolReppt(particle);
    }
    if (mcStep%MCSPeriod==0){
      printf("OutputData started for mcStep = %d\n", mcStep);
      OutputData(particle, ori, mcStep);
      printf("output data end\n");
    }
  }
  return 0;
}

double ran3(long *idum){
  static int inext,inextp;
  static long ma[56];
  static int iff=0;
  long mj,mk;
  int i,ii,k;
  if (*idum < 0 || iff == 0) {
    iff=1;
    mj=labs(MSEED-labs(*idum));
    mj %= MBIG;
    ma[55]=mj;
    mk=1;
```

```
      for (i=1;i<=54;i++) {
        ii=(21*i) % 55;
        ma[ii]=mk;
        mk=mj-mk;
        if (mk < MZ) mk += MBIG;
        mj=ma[ii];
      }
      for (k=1;k<=4;k++)
        for (i=1;i<=55;i++) {
          ma[i] -= ma[1+(i+30) % 55];
          if (ma[i] < MZ) ma[i] += MBIG;
        }
      inext=0;
      inextp=31;
      *idum=1;
    }
    if (++inext == 56) inext=1;
    if (++inextp == 56) inextp=1;
    mj=ma[inext]-ma[inextp];
    if (mj < MZ) mj += MBIG;
    ma[inext]=mj;
    return mj*FAC;
}
int q_ran(int q){

    static int itemp;
    static long seed2;
     seed2 = 2;
     itemp = (int)(q*ran3(&seed2) + 1);
     return itemp;
}
void InitLP(struct LP *p){
    int i, x, y, x1, y1;
    long seed1;
    float ran1;

    for (i = 0; i<nsp; i++){
      p[i].Id = i;
      ran1 = (float)ran3(&seed1);
      if (ran1 < liqFrac){
        p[i].orient = -1;
      }else{
        p[i].orient = q_ran(Q);
      }
      x = i%xDim;
      y = (i-i%xDim)/xDim;

      x1 = (x-xDim)%xDim + xDim -1;
      y1 = (y-yDim)%yDim + yDim -1;
```

```
    p[i].neigh[0] = y1*xDim + x1;
    x1 = x;
    y1 = (y-yDim)%yDim + yDim -1;
    p[i].neigh[1] = y1*xDim + x1;
    x1 = (x+1)%xDim;
    y1 = (y-yDim)%yDim + yDim -1;
    p[i].neigh[2] = y1*xDim + x1;
    x1 = (x-xDim)%xDim + xDim -1;
    y1 = y;
    p[i].neigh[3] = y1*xDim + x1;
    x1 = (x+1)%xDim;
    y1 = y;
    p[i].neigh[4] = y1*xDim + x1;
    x1 = (x-xDim)%xDim + xDim -1;
    y1 = (y+1)%yDim;
    p[i].neigh[5] = y1*xDim + x1;
    x1 = x;
    y1 = (y+1)%yDim;
    p[i].neigh[6] = y1*xDim + x1;
    x1 = (x+1)%xDim;
    y1 = (y+1)%yDim;
    p[i].neigh[7] = y1*xDim + x1;
  }
}

void MCGrowthBySolReppt(struct LP *p){

  int cSite, cOrient;
  int nSite, nOrient;
  int i;
  double prob;
  long seed3;
  int  tcOrient, tnOrient;
  //int tcSite, tnSite;
  float E1, E2, E3, E4, dE;
  float threshold;
  float kT;
  int neighborRan;

  cSite   = q_ran(nsp)-1;
  cOrient = p[cSite].orient;

  neighborRan = q_ran(nNeigh)-1;
  nSite   = p[cSite].neigh[neighborRan];
  nOrient = p[nSite].orient;

  if (cOrient == nOrient)
    return;
  E1 = 0.0; E2 = 0.0; E3 = 0.0; E4 = 0.0;
```

```
//Lets assign new orientation for all possible cases.

if (cOrient*nOrient < 0){ // liquid and solid site
  if (cOrient == -1){
    tcOrient = q_ran(Q);
    tnOrient = -1;
  } else {
    tcOrient = -1;
    tnOrient = q_ran(Q);
  }
  kT = (float) kTsl;
} else { //solid solid site
  tcOrient  = q_ran(Q);
  tnOrient  = nOrient;
  kT = (float) kTss;
}

//Lets do Energy calculations
for (i = 0; i <nNeigh; i++){
  E1 = E1 + SurfaceEnergyByPhase(cOrient, p[p[cSite].neigh[i]].
     orient);
  E2 = E2 + SurfaceEnergyByPhase(nOrient, p[p[nSite].neigh[i]].
     orient);
  E3 = E3 + SurfaceEnergyByPhase(tcOrient, p[p[cSite].neigh[i]].
     orient);
  E4 = E4 + SurfaceEnergyByPhase(tnOrient, p[p[nSite].neigh[i]].
     orient);
}
  dE = E3 + E4 - E1 - E2 +
  SurfaceEnergyByPhase(cOrient, nOrient) + SurfaceEnergyByPhase(
     tnOrient, tcOrient)
  -SurfaceEnergyByPhase(tcOrient, nOrient) - SurfaceEnergyByPhase
     (cOrient, tnOrient);

     if (dE > 0){
       prob = exp((-1.0)*dE/kT);
     } else {
       prob = 1.0;
     p[cSite].orient = tcOrient;
     p[nSite].orient = tnOrient;
     return;
     }

     seed3 = 3;
     threshold = (float) ran3(&seed3);

     if (prob < threshold){
       return;
     }
```

```
        p[cSite].orient = tcOrient;
        p[nSite].orient = tnOrient;

        return;
}
float SurfaceEnergyByPhase(int oId1, int oId2){
  if (oId1 == oId2){
    return 0.0;
  }
  if (oId1*oId2 <0){ // if one of them is liquid and other one is
     solid
    return (float)liqSolSurfEnergy;
  } else { // if both are solid -
    return (float)solSolSurfEnergy;
  }
}

void OutputData(struct LP *p, struct orientation *O,int filenameExt
   ){

  char filename[100];
  int i;
  struct tm *l_time;
  char string[20];
  FILE *outputFile;
  time_t now;
  now = time((time_t *)NULL);
  time(&now);
  l_time = localtime(&now);
  strftime(string, sizeof string, "%d-%b-%y", l_time);
  sprintf(filename,"2DMC %s %d.ang", string, filenameExt);

  outputFile = fopen(filename, "w");
  fprintf(outputFile,"# TEM_PIXperUM          1.000000\n# x-star
                  0.467500\n# y-star                0.946700\n# z
     -star                 0.704700\n# WorkingDistance
     12.000000\n#\n# Phase 1\n# MaterialName   Tungsten Carbide\n#
     Formula       WC\n# Info\n# Symmetry              62\n#
     LatticeConstants      2.906 2.906 2.838  90.000  90.000
     120.000\n# NumberFamilies        8\n# hklFamilies      0  0  1
      1 45.000000 1\n# hklFamilies      1  0  0 1 33.299999 1\n#
     hklFamilies      1  0  1 1 16.700001 1\n# hklFamilies      1
     1  0 1 6.700000 1\n# hklFamilies     1  1  1 1 4.200000 1\n#
     hklFamilies     1  0  2 1 5.000000 1\n# hklFamilies     2  0
     1 1 3.300000 1\n# hklFamilies     1  1  2 1 2.300000 1\n#
     Categories16992756 0 16992756 16992640 2009385142 \n#\n# GRID:
      SqrGrid\n# XSTEP: 10.000000\n# YSTEP: 10.000000\n# NCOLS_ODD:
      %d\n# NCOLS_EVEN: %d\n# NROWS: %d\n#\n# OPERATOR:   vineet\n
```

```
    #\n# SAMPLEID:  \n#\n# SCANID:  \n#\n", xDim, xDim, yDim);
  for (i =0; i< nsp; i++){
    if (p[i].orient <0){
      fprintf(outputFile," %2.5f  %2.5f  %2.5f", 4.94880,
         1.95727,  4.37499 );
      fprintf(outputFile,"     %2.5f     %2.5f", (float) (i%xDim)
         , (float) ((i-i%xDim)/xDim));
      fprintf(outputFile," %3.1f %1.3f %d    %d %1.3f \n",
         117.4, 0.000, 0, -1, 1.66);
    }else {
      fprintf(outputFile," %2.5f  %2.5f  %2.5f",O[p[i].orient].
         phi1 , O[p[i].orient].phi, O[p[i].orient].phi2);
      fprintf(outputFile,"    % 2.5f     %2.5f", (float) (i%xDim)
         , (float) ((i-i%xDim)/xDim));
      fprintf(outputFile," %3.1f %1.3f %d    %d %1.3f \n",
         117.4, 0.343, 0, -1, 1.66);
    }
  }
  fclose (outputFile);
}

void InitOrientTable(struct orientation *O){
  int i;
  long seed1 = 2;
  float pi = (float) 3.14159265;
  for (i=1; i<=Q; i++){
    O[i].phi1 = (float) (ran3(&seed1)*pi/2.0);
    O[i].phi =  (float) (acos(ran3(&seed1)*2.0-1.0));
    O[i].phi2 = (float) (ran3(&seed1)*pi/2.0);
  }
  printf("orientation matrix table initalization - Done ...\n");
}

void MakeCircle(struct LP *p, int x, int y, int radius, int cirOri)
   {
  int i, j;
  int isInCir;
  for (i=0; i<xDim; i++){
    for (j = 0; j<yDim; j++){
      isInCir = (i-x)*(i-x) + (j-y)*(j-y) -radius*radius;
      if ( isInCir<=0)
        p[j*xDim+i].orient = cirOri;
    }
  }
}
```

APPENDIX B

PHASE FIELD SIMULATION

The code of phase field simulation of two unequal particle is given below:

Listing B.1: Algorithm for Monte Carlo simulation of sintering

```
#include "stdio.h"
#include "math.h"
#include "stdlib.h"
#include "time.h"
#define MBIG 1000000000
#define MSEED 161803398
#define MZ 0
#define FAC (1.0/MBIG)
#define pi 3.14159265358979323846

struct dimensions{
  int nx, ny;
  int noOfGrains;// noOfGb;
  double deltaT, totalTime;
  //double deltaX, deltaY;
  double randFluctC,randFluctEta, rhoVap, etaVap;
  double rhoSol, etaSol;
  char filename[50];
};
struct makeCircleInfo{
  int centerX, centerY, grainNo;
  double radius;
  double cirVal;
};
struct constWang2005{
  double A, B;
  double betaRho, betaEta;
  double Dvol, Dvap, Dsurf, Dgb;
  //double kappa, rho0;
  //double c, mT, mR;
  double L;
  double particleDensity, vaporDensity;
  //double advFlagEtaIJSum, advFlagEtaij, advFlagRhoMin,
     advFlagRhoMax;
  double GBEFactor;
};
```

```
double* double1D( int n, const char *message );
double **double2D( int nx, int ny, const char *message );
void freeDouble2D( double **m, int nx, int ny);
int **int2D( int nx, int ny, const char *message );
double Ran3(long *idum);
void RandInit3D(double ***M, long seed, struct dimensions dim);
void RandInit2D(double **M, long seed, struct dimensions dim);
void MakeCircle(double *p, struct dimensions dim, struct
    makeCircleInfo cir);

int main(){
  double *rho, **eta, *FPR, **FPE, *vol, **rc;
  double *rhoNew, **etaNew, *rhoSwap, **etaSwap;
  double *D;
  double  oneBy12 = 1.0/12.0;
  int **N, x, y, g, i, j, nxny, outputInterval;
  struct dimensions sim;  struct constWang2005 vars;
  struct makeCircleInfo cir;
  long seed = 1;
  double etaSqSum, etaCubeSum, etaSq,  phi, sum, rhoSq;
  double timeT = 0.0, totalTime = 5000000.0;
  double *variationDeri, *rhoTemp;
  int readFromFile;
  double AT2, BT12, deltaTThalf;
  double gravityCenterX, gravityCenterY, inertia;
  double **ptInfo;
  char string1[200], string2[200];
  int grainSize, g1;
  double eta_ij;
  FILE *f, *fp;

  sim.nx = 256;
  sim.noOfGrains = 2;
  sim.rhoVap = 0.008917;
  sim.randFluctC = 0.0000001;
  sim.rhoSol = 0.99857;
  sim.etaVap = 0;
  sim.randFluctEta = 0.0000;
  sim.etaSol = 1.0;
  sim.deltaT = 0.001;
  outputInterval = 1000;
  vars.A = 16.0;
  vars.B = 1;
  vars.particleDensity = 0.999;
  vars.vaporDensity = 0.01;

  vars.Dsurf = 45;
  vars.Dgb = 4.1;
  vars.Dvol = 0.08;
```

```
vars.Dvap = 0.012;
vars.betaRho = 10;
vars.betaEta = 3.75;
vars.L = 10.0;
vars.GBEFactor = 7;
grainSize = 30;

printf("read from old file  yes==1, from matlabInitFile ==2, no
   == 0\n If yes: make a copy of progressFile\n");
scanf("%d", &readFromFile);
fp = fopen("progressFile.dat","a");
fprintf(fp,"nx = %d, noOfGrains = %d, average C = %f, fluctuation
    in C = %f, average eta = %f, fluctuation in eta = %f, delta t
    = %f\n", sim.nx, sim.noOfGrains, sim.rhoVap, sim.randFluctC,
   sim.etaVap, sim.randFluctEta, sim.deltaT);
fprintf(fp, "A = %f, B = %f, betaRHo = %f, betaEta = %f, ", vars.
   A, vars.B, vars.betaRho, vars.betaEta);
fclose(fp);
nxny = sim.nx*sim.nx;

//memory Allocation
rho = double1D(sim.nx*sim.nx, "rho");
rhoTemp = double1D(sim.nx*sim.nx, "rhoTemp");
rhoNew = double1D(nxny, "ehoNew");
eta = double2D(sim.noOfGrains, sim.nx*sim.nx, "eta");
D = double1D(sim.nx*sim.nx, "D");
etaNew = double2D(sim.noOfGrains, nxny, "etaNew");
etaSwap = (double **) malloc ( sim.noOfGrains * sizeof(double *))
   ;
FPR = double1D(sim.nx*sim.nx, "FPR");
FPE = double2D(sim.noOfGrains, sim.nx*sim.nx, "FPE");
N = int2D(sim.nx*sim.nx, 4, "NeighList");
vol = double1D(sim.noOfGrains, "volume of the partilces");
rc = double2D(sim.noOfGrains, 2, "center of mass");
ptInfo = double2D(sim.noOfGrains, 3,"ptInfo");
variationDeri = double1D(sim.nx*sim.nx,"variDeri");

i = grainSize;
if (readFromFile == 0){
  ptInfo[0][2] = (double)i*2;    ptInfo[0][1] = 2*i+8;
     ptInfo[0][0] = sim.nx/2;
  ptInfo[1][2] = (double)i;    ptInfo[1][1] = 5*i+8+1;
     ptInfo[1][0] = sim.nx/2;

  //Initialization of initial microstructure
  for (i = 0; i<nxny; i++){
    rho[i] = sim.rhoVap + (0.5 - Ran3(&seed))*sim.randFluctC;
    for (g = 0; g<sim.noOfGrains; g++){
      eta[g][i] = sim.etaVap + (0.5 - Ran3(&seed))*sim.
```

```
            randFluctEta;
        }
    }

    fp = fopen("progressFile.dat","a");
    for (g = 0; g<sim.noOfGrains; g++){
        cir.radius = ptInfo[g][2];
        cir.centerX = ptInfo[g][0];
        cir.centerY = ptInfo[g][1];
        cir.cirVal = sim.rhoSol; cir.grainNo = g;
        MakeCircle(rho, sim, cir);
        cir.cirVal = sim.etaSol;  MakeCircle(eta[cir.grainNo] , sim,
            cir);
        fprintf(fp,"circle radius = %f, Cx = %d, Cy = %d, val = %f,
            grain no %d\n", cir.radius, cir.centerX, cir.centerY, cir.
            cirVal, cir.grainNo);
    }
    fclose(fp);
}
if (readFromFile == 1){
    printf("Whats the time\n");   scanf("%lf", &timeT);

    sprintf(sim.filename,"Rho%d.txt",(int)(timeT));
    f = fopen(sim.filename,"r");
    for (y = 0; y<sim.nx; y++){
        for (x = 0; x<sim.nx; x++){
            i = y*sim.nx + x;
            fscanf(f,"%lf\t", &rho[i]);
        }fscanf(f,"\n");
    } fclose(f);
    sprintf(sim.filename,"Eta%d.txt",(int)(timeT));
    f = fopen(sim.filename,"r");
    for (g = 0; g<sim.noOfGrains; g++){
        for (y = 0; y<sim.nx; y++){
            for (x = 0; x<sim.nx; x++){
                i = y*sim.nx + x;
                fscanf(f,"%lf\t", &eta[g][i]);
            }fscanf(f,"\n");
        }
    } fclose(f);
}

sprintf(string1, "scp *.c  128.110.227.36:/media/Elements/cade/
    be3p75/2pt/2pt%d_%d", sim.nx,grainSize );
    //  system(string1);

sprintf(string1, "scp *.txt *.dat  128.110.227.36:/media/Elements
    /cade/be3p75/2pt/2pt%d_%d", sim.nx,grainSize );
```

```
//Initialzation of neighboring list
for (y = 0; y<sim.nx; y++){
  for (x = 0; x<sim.nx; x++){
    i = sim.nx*y + x;
    N[i][0] = ((y+sim.nx)%sim.nx)*sim.nx + ((x+sim.nx+1)%sim.nx);
    N[i][1] = ((y+sim.nx)%sim.nx)*sim.nx + ((x+sim.nx-1)%sim.nx);
    N[i][2] = ((y+sim.nx+1)%sim.nx)*sim.nx + ((x+sim.nx)%sim.nx);
    N[i][3] = ((y+sim.nx-1)%sim.nx)*sim.nx + ((x+sim.nx)%sim.nx);
  }
}

// running the main loop for microstructural evolution
BT12 = vars.B * 12.0;
AT2  = vars.A * 2.0;
deltaTThalf = sim.deltaT * 0.5;

while(timeT < totalTime){
  //Calculation of diffusion coefficient and partial derivatives
     of rho and eta
  for (i = 0; i<nxny; i++){

    rhoSq = rho[i]*rho[i];                          //R = rho[i]*rho[
       i]*rho[i]*(10.0 - 15.0*rho[i] + 6.0*rho[i]*rho[i]);
    phi = rhoSq * rhoSq * (7.0 * rhoSq - 18.0*rho[i] + 12.0);


    etaCubeSum = eta[0][i]*eta[0][i]*eta[0][i] + eta[1][i]*eta
       [1][i]*eta[1][i];
    etaSqSum = eta[0][i]*eta[0][i] + eta[1][i]*eta[1][i];
    eta_ij = eta[0][i]*eta[1][i];

    D[i] = vars.Dvol * phi + vars.Dvap * (1.0 - phi) + vars.Dsurf
        * rhoSq*(1.0-rho[i])*(1.0-rho[i]) +  vars.Dgb * 2 *
       eta_ij;

    FPR[i] = AT2* (rho[i]-vars.vaporDensity) * (vars.
       particleDensity-rho[i]) * (1.0-2.0*rho[i])
      +  2.0*( rho[i] - 3.0*etaSqSum + 2.0*etaCubeSum); // This
         term shouls have been multiplied with vars.B which is
         unity


    etaSq = eta[0][i]*eta[0][i];
    FPE[0][i] = BT12* eta[0][i] * (   (1.0-rho[i]) - (2.0-rho[i
       ])*eta[0][i] + etaSq + vars.GBEFactor*(etaSqSum - etaSq));
    etaSq = eta[1][i]*eta[1][i];
    FPE[1][i] = BT12* eta[1][i] * (   (1.0-rho[i]) - (2.0-rho[i
       ])*eta[1][i] + etaSq + vars.GBEFactor*(etaSqSum - etaSq));
  }
```

```
for (i = 0; i<nxny; i++){
  variationDeri[i] = FPR[i] - vars.betaRho *(-4.0*rho[i] + rho[
    N[i][0]] + rho[N[i][1]] + rho[N[i][2]] + rho[N[i][3]]);
}

for (i = 0; i<nxny; i++){
  rhoNew[i] = rho[i] +  deltaTThalf * (
    D[i]*( -4.0*variationDeri[i] +  variationDeri[N[i][0]] +
       variationDeri[N[i][1]] + variationDeri[N[i][2]] +
       variationDeri[N[i][3]])
    + D[N[i][0]]*(variationDeri[N[i][0]] - variationDeri[i])
    + D[N[i][2]]*(variationDeri[N[i][2]] - variationDeri[i])
    - D[N[i][1]]*(variationDeri[i] - variationDeri[N[i][1]])
    - D[N[i][3]]*(variationDeri[i] - variationDeri[N[i][3]])  )
       ;

  etaNew[0][i] = eta[0][i] + sim.deltaT * (
        (- vars.L * (FPE[0][i] - vars.betaEta * (-4.0*eta[0][i]
           + eta[0][N[i][0]] + eta[0][N[i][1]] + eta[0][N[i
          ][2]] + eta[0][N[i][3]]))) );
  etaNew[1][i] = eta[1][i] + sim.deltaT * (
        (- vars.L * (FPE[1][i] - vars.betaEta * (-4.0*eta[1][i]
           + eta[1][N[i][0]] + eta[1][N[i][1]] + eta[1][N[i
          ][2]] + eta[1][N[i][3]]))) );
}
//Output Data to file
if ((int)(timeT/sim.deltaT)%(outputInterval) == 0){

  sprintf(sim.filename,"D%d.txt",(int)(timeT));
  f = fopen(sim.filename,"w");
  sum = 0.0;
  for (y = 0; y<sim.nx; y++){
    for (x = 0; x<sim.nx; x++){
      i = y*sim.nx + x;
      fprintf(f,"%1.3f\t", D[i]);
    }fprintf(f,"\n");
  } fclose(f);
  //Calculate the deltaRho with each diffusion coefficient
  for (j = 0; j<4; j++){
    switch(j){
      case 0:   // Volume
        sprintf(sim.filename,"RhoVol%d.txt",(int)(timeT));
        for (i = 0; i<nxny; i++){
          phi = rho[i]*rho[i]*rho[i]*rho[i]*(7.0*rho[i]*rho[i]
             -18.0*rho[i] +12.0);
          D[i] = vars.Dvol * phi;
        } break;
      case 1: // vapor
        sprintf(sim.filename,"RhoVap%d.txt",(int)(timeT));
```

```
        for (i = 0; i<nxny; i++){
          phi = rho[i]*rho[i]*rho[i]*rho[i]*(7.0*rho[i]*rho[i]
              -18.0*rho[i] +12.0);
          D[i] = vars.Dvap * (1.0 - phi);
        } break;
      case 2: // Surface
        sprintf(sim.filename,"RhoSurf%d.txt",(int)(timeT));
        for (i = 0; i<nxny; i++){
            D[i] = vars.Dsurf * rho[i]*rho[i]*(1.0-rho[i])
                *(1.0-rho[i]);
        } break;
      case 3: // gb
        sprintf(sim.filename,"RhoGb%d.txt",(int)(timeT));
        etaSqSum = 0.0;
        for (i = 0; i<nxny; i++){
          for (g = 0; g<sim.noOfGrains; g++){
            etaSqSum = etaSqSum + eta[g][i]*eta[g][i];
          }
          D[i] = vars.Dgb * (rho[i]*(1.0-etaSqSum));
        } break;
    }
    for (i = 0; i<nxny; i++){
      rhoTemp[i] = 0.5*(
        D[i]*( -4.0*variationDeri[i] + variationDeri[N[i][0]]
            + variationDeri[N[i][1]] + variationDeri[N[i][2]] +
            variationDeri[N[i][3]])
        + D[N[i][0]]*(variationDeri[N[i][0]] - variationDeri[i
            ])
        + D[N[i][2]]*(variationDeri[N[i][2]] - variationDeri[i
            ])
        - D[N[i][1]]*(variationDeri[i] - variationDeri[N[i
            ][1]])
        - D[N[i][3]]*(variationDeri[i] - variationDeri[N[i
            ][3]])  );
    }
    f = fopen(sim.filename,"w");
    for (y = 0; y<sim.nx; y++){
      for (x = 0; x<sim.nx; x++){
        i = y*sim.nx + x;
        fprintf(f,"%1.3f\t", rhoTemp[i]);
      }fprintf(f,"\n");
    } fclose(f);

  } // End induvidual mechanism contributions

  sprintf(sim.filename,"Rho%d.txt",(int)(timeT));
  f = fopen(sim.filename,"w");
  sum = 0.0;
  for (y = 0; y<sim.nx; y++){
```

```
        for (x = 0; x<sim.nx; x++){
          i = y*sim.nx + x;
          fprintf(f,"%1.3f\t", rho[i]);
          sum = sum + rho[i];
        }fprintf(f,"\n");
      } fclose(f);
      printf (    "\n%f\t%1.10f\t", timeT, sum/nxny);

      sprintf(sim.filename,"Eta%d.txt", (int)timeT);
      f = fopen(sim.filename,"w");
      for (g = 0; g<sim.noOfGrains; g++){
        sum = 0.0;
        for (y = 0; y<sim.nx; y++){
          for (x = 0; x<sim.nx; x++){
            i = y*sim.nx + x;
            sum = sum + eta[g][i];
            fprintf(f,"%1.2f\t", eta[g][i]);
          }fprintf(f,"\n");
        }
        printf("\t%1.10f", sum/nxny);
      } fclose(f);
    //  i = system(string1);
    //  if (i == 0){    system("rm *.txt");      }

      fp =   fopen("progressFile.dat","a");
      printf (    "\n");
      fprintf(fp, "\n%f\t%f\n", timeT, sum/nxny);
      fclose(fp);
    }
    rhoSwap = rho; rho = rhoNew; rhoNew = rhoSwap;
    etaSwap = eta;  for (g = 0; g<sim.noOfGrains; g++){ etaSwap[g]
       = eta[g];  }
    eta = etaNew; for (g = 0; g<sim.noOfGrains; g++){ eta[g] =
       etaNew[g];   }
    etaNew = etaSwap; for (g = 0; g<sim.noOfGrains; g++){ etaNew[g]
       = etaSwap[g]; }
    timeT = timeT + sim.deltaT;
  }
  return 0;
}

double* double1D( int n, const char *message )
{
  double *m;
  m = (double*) malloc( n * sizeof( double ) );
  if( m == NULL ) {
    printf("double1D() cannot allocate memory size=%d: %s\n",
           n, message);
    exit( 0 );
```

```
  }
  printf("memory allocated for %s is %d bytes\n", message, n*sizeof
     (double));
  return( m );
} /* end double1D() */

double **double2D( int nx, int ny, const char *message )
{
  double **m;
  int i;
  m = (double**) malloc( nx * sizeof( double* ) );
  if( m == NULL ) {
    printf("double2D cannot allocate pointers, size=%d: %s\n",
           nx, message );
    exit(0);
  }
  for (i=0; i<nx; i++){
    m[i] = (double *) malloc( ny * sizeof( double ) );
    if( m[i] == NULL ){
      printf("double2D cannot allocate arrays, size=%d: %s\n",
             ny, message );
      exit(0);
    }
  }
  printf("memory allocated for %s is %d bytes\n", message, nx*ny*
     sizeof(double));
  return m;
}  /* end double2D() */

void freeDouble2D( double **m, int nx, int ny){
  int i;
  for (i = 0; i<nx; i++){
    free(m[i]);
  }
  free(m);
}

int **int2D( int nx, int ny, const char *message )
{ int **m;
  int i;
  m = (int**) malloc( nx * sizeof( int* ) );
  if( m == NULL ) {
    printf("int2D cannot allocate pointers, size=%d: %s\n",
           nx, message );
    exit(0);
  }
  for (i=0; i<nx; i++){
    m[i] = (int *) malloc( ny * sizeof( int ) );
    if( m[i] == NULL ){
```

```
      printf("int2D cannot allocate arrays, size=%d: %s\n",
             ny, message );
      exit(0);
    }
  }
  printf("memory allocated for %s is %d bytes\n", message, nx*ny*
     sizeof(int));
  return m;
}  /* end double2D() */

double Ran3(long *idum){
/*------------------------------ ran3()-------------------------------
  This program takes the pointer a long integer and generates
  a random number between 0 to 1
  usage - ran3(&seed)
-------------------------------------------------------------------------

*/
  static int inext,inextp;
  static long ma[56];
  static int iff=0;
  long mj,mk;
  int i,ii,k;

  if (*idum < 0 || iff == 0) {
    iff=1;
    mj=labs(MSEED-labs(*idum));
    mj %= MBIG;
    ma[55]=mj;
    mk=1;
    for (i=1;i<=54;i++) {
      ii=(21*i) % 55;
      ma[ii]=mk;
      mk=mj-mk;
      if (mk < MZ) mk += MBIG;
      mj=ma[ii];
    }
    for (k=1;k<=4;k++)
      for (i=1;i<=55;i++) {
        ma[i] -= ma[1+(i+30) % 55];
        if (ma[i] < MZ) ma[i] += MBIG;
      }
    inext=0;
    inextp=31;
    *idum=1;
  }

  if (++inext == 56) inext=1;
  if (++inextp == 56) inextp=1;
  mj=ma[inext]-ma[inextp];
```

```
  if (mj < MZ) mj += MBIG;
  ma[inext]=mj;
  return mj*FAC;
}

void RandInit3D(double ***M, long seed, struct dimensions dim){
/*-----------------RandInit2Dim()-----------------------------------
  This function initiallizes a three dimensional array to
  average composition +- random fluctuations.
  usage RandInit3D( 3DMatrixPointer, LongIntNo,
     structureDimensionPointer)
------------------------------------------------------------------*/
  double randFluct, avgComp;
  int noOfLayers;
  int i,j,k, nx,ny;

  nx = dim.nx;
  ny = dim.ny;

  noOfLayers = dim.noOfGrains;
  randFluct = dim.randFluctEta;
  avgComp = dim.etaVap;

  for (i = 0; i<noOfLayers; i++){
    for(k = 0; k<ny; k++){
      for(j = 0; j<nx; j++){
        M[i][k][j] = avgComp + (0.5 - Ran3(&seed))*randFluct;
      } // j - X
    }// k -Y
  }// i
}// RandInit3D()

void RandInit2D(double **M, long seed, struct dimensions dim){
/*-----------------RandInit2Dim()-----------------------------------
  This function initiallizes a two dimensional array to
  average composition +- random fluctuations.
  usage RandInit2D( 3DMatrixPointer, LongIntNo,
     structureDimensionPointer)
------------------------------------------------------------------*/
  int j,k;
  int nx,ny;
  double avgComp, randFluct;

  avgComp = dim.rhoVap;
  randFluct = dim.randFluctC;
  nx = dim.nx;
  ny = dim.ny;

  for(k = 0; k<ny; k++){
```

```
    for(j = 0; j<nx; j++){
      M[k][j] = avgComp + (0.5 - Ran3(&seed))*randFluct;
    }//j
  }//k
}// RandInit2D()

void MakeCircle(double *p, struct dimensions dim, struct
   makeCircleInfo cir){
  /* This funsction initalize the mircostructure by making a circle
     .
  This is designed for simulations in fourier space where we
  shall be working in 1 dimensional arrays */

  int x, y;

  for (y = 0; y<dim.nx; y++){
    for (x = 0; x<dim.nx; x++){
      if ((x - cir.centerX)*(x - cir.centerX) +
        (y - cir.centerY)*(y - cir.centerY) <= cir.radius*cir.
           radius){
          p[x+y*dim.nx] = cir.cirVal;
      }
    }
  }
}
```

APPENDIX C

GEOMETRICAL MODEL

The code of geometrical model is given below:

Listing C.1: Algorithm for Monte Carlo simulation of sintering

```
%% Initialize the data and provide values of materials consatnts
clc;      clear;    %close all;
format short e;
gs = dlmread('initialGS.txt');
gs = gs(:,1)'/1000;
sizeGs = size(gs);
a = randperm(sizeGs(2));
gs = gs(a); clear a;

%gs =[0.0278    0.0181    0.0180    0.0302    0.0240];
gs = gs(1:min([200 sizeGs(2)]));
sizeGs = size(gs);


%% Materials properties
deltaS = 0.3E-3; % micron - Ref: J am ceram soc vol 71 (2) pp
    113-20 (1988) 1E-3;%
gammaS = 2.8; %j/m^2 Ref: J of Less Common Metals vol20 (2) pp
    93-103 (1970) 1;%
gammaGb = 2.3667;%1.0715; %j/m^2
psiBy2 = acos( 0.5*gammaGb/gammaS); %radians Dihedral Angle The
    effect of thermal etching on emmisivity of tungsten, T J Quinn
% Brit. J Appl Phys, 1965, Vol 16, p 973 http://iopscience.iop.org
    /0508-3443/16/7/310
mobility = 1E-18;
Ds = 4.0 * exp((-300 * 1000)/((273 + 950)*8.3144)) * 1E-4; %in m^2/
    s at 950C       5E-11;%
RT = 8.3144*(950+273); %in J/mol
molarVol = 183.84/19.25 * 1E-6; %m^3/mol (mol weight / density) 102
    E-3/1522;%
%% Initialization of various arrays and flags
vol = 4/3*pi * gs.^3;
deltaTi = 1e-3; deltaT = deltaTi; timeT = 0;

gb = zeros(1, sizeGs(2)-1); sizeGb = size(gb);
gbMax = zeros([1, sizeGb(2)]);
```

```
flagNeckGrowth = ones([1, sizeGb(2)]);
flagGbMigration = zeros([1, sizeGb(2)]);
rapidGbMigrationGeometry = zeros([5, sizeGb(2)]);    %R1 R2 Rgb
    beta in each row
neckVolMat = zeros([1, sizeGb(2)]);
neckSizeMat = zeros([1, sizeGb(2)]);
flagFirstRunNeckGrowth = ones([1, sizeGb(2)]);

timeMat = [];
gsMat = [];
gbMat = [];
volWeightedMean = [];
neckSizeMatrix = [];
flagVolChange = 1;
flagGrainDisappear = 0;
flagEvolve = 1;


%% Calculate gb Max to evaluate the presence of neck growth
for i = 1:sizeGb(2)
    R1P = max([gs(i) gs(i+1)]);
    R2P = min([gs(i) gs(i+1)]);
    R = R2P/R1P;
    gbMax(i) = -.11*R^3 + .084*R^2 + 0.6*R + 0.7E-5;
    gbMax(i) =  gbMax(i) * R1P;
    gb(i) = 1E-2*(round(gbMax(i)*100)/10);
end

%%
while sizeGs(2) > 1 % until all grain boundaries disappear

    %% if a grain disappears
    while flagGrainDisappear == 1
        [i j] = min(vol);
        if j == 1
            vol(2) = vol(2) + vol(1);
            vol = vol(2:end);
            gb = gb(2:end);
            flagNeckGrowth = flagNeckGrowth(2:end);
            flagNeckGrowth(1) = 1;
            flagFirstRunNeckGrowth = flagFirstRunNeckGrowth(2:end);
            flagFirstRunNeckGrowth(1) = 1;
            flagGbMigration = flagGbMigration(2:end);
            flagGbMigration(1) = 0;
            gbMax = gbMax(2:end);
            gs = (3*vol/(4*pi)).^(1/3);
            R = min([gs(1) gs(2)])/max([gs(1) gs(2)]);
            gbMax(1) = -.11*R^3 + .084*R^2 + 0.6*R + 0.7E-5;
            gbMax(1) = gbMax(1) * max([gs(1) gs(2)]);
```

```
        gb(1) = 1E-2*(round(gbMax(1)*100)/10);
    elseif j == sizeGs(2)
        vol(sizeGs(2) -1) = vol(sizeGs(2) - 1) + vol(sizeGs(2))
            ;
        vol = vol(1:end-1);
        gb = gb(1:end-1);
        flagNeckGrowth = flagNeckGrowth(1:end-1);
        flagNeckGrowth(end) = 1;
        flagGbMigration = flagGbMigration(1:end-1);
        flagFirstRunNeckGrowth = flagFirstRunNeckGrowth(1:end
            -1);
        flagFirstRunNeckGrowth(end) = 1;
        flagGbMigration(end) = 0;
        gbMax = gbMax(1:end-1);
        gs = (3*vol/(4*pi)).^(1/3);
        R = min([gs(end) gs(end-1)])/max([gs(end) gs(end-1)]);
        gbMax(end) = -.11*R^3 + .084*R^2 + 0.6*R + 0.7E-5;
        gbMax(end) = gbMax(end) * max([gs(end) gs(end-1)]);
        gb(end) = 1E-2*(round(gbMax(end)*100)/10);
    else
        if vol(j+1) > vol(j-1)
            vol(j+1) = vol(j+1) + vol(j);
        else
            vol(j-1) = vol(j-1) + vol(j);
        end
        vol = [vol(1:j-1) vol(j+1:end)];
        gb = [gb(1:j-1) gb(j+1:end)];
        flagNeckGrowth = [flagNeckGrowth(1:j-1) flagNeckGrowth(
            j+1:end)];
        flagNeckGrowth(j-1) = 1;
        flagGbMigration = [flagGbMigration(1:j-1)
            flagGbMigration(j+1:end)];
        flagGbMigration(j-1) = 0;
        gbMax = [gbMax(1:j-1) gbMax(j+1:end)];
        flagFirstRunNeckGrowth = [flagFirstRunNeckGrowth(1:j-1)
            flagFirstRunNeckGrowth(j+1:end)];
        flagFirstRunNeckGrowth(j-1) = 1;
        gs = (3*vol/(4*pi)).^(1/3);
        R = min([gs(j-1) gs(j)])/max([gs(j-1) gs(j)]);
        gbMax(j-1) = -.11*R^3 + .084*R^2 + 0.6*R + 0.7E-5;
        gbMax(j-1) = gbMax(j-1) * max([gs(j-1) gs(j)]);
        gb(j-1) = 1E-2*(round(gbMax(j-1)*100)/10);
    end
    sizeGs = size(gs);
    sizeGb = size(gb);
    flagEvolve = 1;
    flagGrainDisappear = 0;
    flagVolChange = 1;
    if sizeGs(2) == 1
```

```
        flagEvolve = 0;
        break;
    end
end

%%

deltaVCoarsen = zeros(1, sizeGb(2));
deltaVGbMigration = zeros(1, sizeGb(2));
deltaVneckGrowth = zeros(1, sizeGb(2));

while flagEvolve == 1
    %% Calculate driving force for microstructural evolution

    if flagVolChange == 1


        for i = 1:sizeGb(2)
            %% gb migration
            %If beta < pi/2, soap bubble geometry subroutine
               used to
            %calculate the geometry
            if (flagGbMigration(i) == 1 ||flagGbMigration(i) ==
                0)
                R1P = max([gs(i) gs(i+1)]);
                R2P = min([gs(i) gs(i+1)]);
                R = R2P/R1P;
                volP = 4*pi/3;
                [R1i, R2i, Rgbi, alphai, betai, thetai,
                   errorR1i, errorR2i]= soapBubbleGeom(1, R, gb
                   (i)/R1P, volP, gammaS, gammaGb);
                R1i = R1i * R1P;     R2i = R2i * R1P; Rgbi =
                   Rgbi * R1P;errorR2i = errorR2i*R1P;
                gbVelocity = 2 * mobility * ( gammaS * (1/R2i -
                    1/R1i) - gammaGb * sec(thetai) / (gb(i)* (
                   tan(betai) - tan(thetai))));
                gbVelocity = (gbVelocity > 0) * gbVelocity * 1
                   E6;
                flagGbMigration(i) = (gbVelocity > 0);

                if (betai >= pi/2) ||abs(errorR2i/R2i) > 0.1
                    flagGbMigration(i) = 2;
                    gbVelocity = 1E20;
                else
                    if betai > 85*pi/180
                        partcle_no_and_angle_beta = [i betai];
                    end
                    rapidGbMigrationGeometry(1,i) = R1i;
                    rapidGbMigrationGeometry(2,i) = R2i;
```

```
            rapidGbMigrationGeometry(3,i) = Rgbi;
            rapidGbMigrationGeometry(4,i) = betai;
            rapidGbMigrationGeometry(5,i) = gb(i);
        end
    end
    if flagGbMigration(i) == 2
        R1P = rapidGbMigrationGeometry(1,i);
        R2P = rapidGbMigrationGeometry(2,i);
        volP = min([vol(i) vol(i+1)]);
        [R1i, R2i, Rgbi, X, alphai, betai, thetai,
           errorBetai]= soapBubbleGeomForRapidGBM(R1P,
           R2P, volP, gammaS, gammaGb);
        gbVelocity = 1E6* 2 * mobility * ( gammaS * (1/
           R2i - 1/R1i) + gammaGb * sec(thetai) / (X* (
           tan(betai) - tan(thetai))));
        rapidGbMigrationGeometry(4,i) = betai;
    end

    %% neckGrowth
    if flagNeckGrowth(i) == 1
        R1P = max([gs(i) gs(i+1)]);
        R2P = min([gs(i) gs(i+1)]);
        R = R2P/R1P;
        dMax = -0.027*R^2 + R + 0.99;
        findNeckFlag = 0;
        center = (1+R + dMax)/2;
        delta = (1+R - dMax)/2;
        while findNeckFlag < 0.5
            xStepFlag = 0;
            if gb(i)/gbMax(i) < 0.2
                xStepFlag = 1;
            end
            [R1, R2, r1,r2, x, alpha, beta, theta,
               neckVol, error] = noShrink3D_1(1, R,
               center, gb(i)/R1P, xStepFlag);
            x = x * R1P;
            delta = delta/2;
            if gb(i) > x
                center = center - delta;
            else
                center = center + delta;
            end
            findNeckFlag = (abs(x - gb(i))/gb(i) <= 1E
               -3 *(flagFirstRunNeckGrowth(i)==1) + 1E
               -5 *(flagFirstRunNeckGrowth(i) == 0)   )
               ;%
            if delta < 1E-15
                %x = gb(i);
                findNeckFlag = 2;
```

```
                    flagNeckGrowth(i) = 0;
                end
            end
            R1 = R1*R1P;     R2 = R2*R1P;     r1 = r1*R1P;
                r2 = r2*R1P;  neckVol = neckVol*R1P^3;
            deltaVneckGrowth(i) = 0;
            if findNeckFlag ~=2 % if we find a neck size
               larger than the previous one
                flagFirstRunNeckGrowth(i) = 0;
                neckVolMat(i) = neckVol;
                deltaMu = gammaS * molarVol * (1/R1 + 1/R2
                   + 2/(r1+r2) - 1/x)*1E6;
                if deltaMu <= 0
                    deltaMu = 0;
                    findNeckFlag = 2;
                    flagNeckGrowth(i) = 0;
                end
                deltaX = min([r1*cos(psiBy2 - alpha - theta
                   ) + r2*cos(psiBy2 - beta + theta),
                   0.25*(R1+R2)])*1E-6;
                neckSizeMat(i) = deltaX;
                area = 2*pi*x*deltaS*1E-12;
                deltaVneckGrowth(i) = 1E18 * (Ds/RT) *
                   deltaMu / deltaX * area;
            end
        end

        %% Coarsening: R1i and R2i come from gb migration
           calculation.
        deltaMu = gammaS*molarVol * (1/R2i - 1/R1i)*1E6;
        area = 2*pi*gb(i)*deltaS*1E-12;
        deltaX = neckSizeMat(i);
        deltaVCoarsen(i) = 1E18 * (Ds/RT) * (deltaMu /
           deltaX) * area;
        dx = gbVelocity;
        deltaVGbMigration(i) = pi*gb(i)*gb(i)*dx*1E18; %in
           micron cube
    end
    deltaVCoarsenTmp = deltaVCoarsen;
    deltaVneckGrowthTmp = deltaVneckGrowth;
    deltaVGbMigrationTmp = deltaVGbMigration;
end
clear R R1 R1P R1i R2 R2i R2P Rgb Rgbi alpha alphai beta
   betai area center delta dMax deltaMu
clear deltaX drivingForce dx error errorR1i errorR2i
   findNeckFlag i theta thetai volP x
clear r1 r2

deltaVCoarsen = deltaVCoarsenTmp * deltaT;
```

```
deltaVneckGrowth = deltaVneckGrowthTmp * deltaT;
deltaVGbMigration = deltaVGbMigrationTmp * deltaT;

%% Make changes in the volume
volTmp = vol;
gbTmp = gb;
if max([deltaVneckGrowth deltaVGbMigration  deltaVCoarsen])
    < max(vol)
    for i = 1:sizeGb(2)
        [volMin j1] = max([vol(i) vol(i+1)]);
        [volMax j2] = min([vol(i) vol(i+1)]);
        vol(i + j1 -1) = vol(i + j1 - 1) + deltaVCoarsen(i)
            ;
        vol(i + j2 -1) = vol(i + j2 - 1) - deltaVCoarsen(i)
            ;

        vol(i + j1 -1) = vol(i + j1 - 1) +
           deltaVGbMigration(i);
        vol(i + j2 -1) = vol(i + j2 - 1) -
           deltaVGbMigration(i);


        if flagNeckGrowth(i) == 1
            R1P = max([gs(i) gs(i+1)]);
            R2P = min([gs(i) gs(i+1)]);
            R = R2P/R1P;
            dMax = -0.027*R^2 + R + 0.99;

            neckVoli = neckVolMat(i);
            [R1, R2, r1,r2, x, alpha, beta, theta, neckVol,
                error] = noShrink3D_1(1, R, dMax, gbTmp(i)/
               R1P, 1);
            neckVol = neckVol*R1P^3;
            neckVolRatio = (neckVoli + deltaVneckGrowth(i)-
               neckVol)/(neckVoli + deltaVneckGrowth(i));
            gb(i) = 1E9;

            if neckVolRatio < 0
                findNeckFlag = 0;
                center = (1+R + dMax)/2;
                delta = (1+R - dMax)/2;
                while findNeckFlag == 0
                    xStepFlag = 0;
                    if gb(i)/gbMax(i) < 0.2
                        xStepFlag = 1;
                    end
                    [R1, R2, r1,r2, x, alpha, beta, theta,
                       neckVol, error] = noShrink3D_1(1, R,
                        center, gbTmp(i)/R1P, xStepFlag);
```

```
                        neckVol = neckVol*R1P^3;    x = x*R1P;
                        neckVolRatio = (neckVoli +
                           deltaVneckGrowth(i)-neckVol)/(
                           neckVoli + deltaVneckGrowth(i));

                        delta = delta/2;
                        if neckVolRatio > 0%  neckVoli +
                           deltaVneckGrowth(i) > neckVol*R1P^3
                            center = center - delta;
                        else
                            center = center + delta;
                        end
                        findNeckFlag = abs(neckVolRatio) <
                           0.001;
                        if delta <1E-20
                            x = 1E9;
                            findNeckFlag = 1;
                        end
                    end
                    if gb(i) == x
                        str = strcat('No neck growth at
                           boundary no', num2str(i)); disp(str)
                           ; clear str;
                    end
                    gb(i) = x;
                end
            end
        end
    else
      gb = 1000000000*ones(1, sizeGb(2)); %if max volume change
          > largest particle volume
    end

    flag1 = max(gb) > 100000;
    flag2 = max(abs(vol - volTmp)./volTmp) > 0.2 ;
    flag3 = sum(vol<0) >0;
    flag4 =  max(abs(gb - gbTmp)./gbTmp) > 0.2;
    flag5 = min(gb./gbMax)>0.2;
    disp([mean(gs) mean(gb) deltaT sizeGs(2)])
    if flag1 || flag2 || flag3 || (flag4 && flag5)
        vol = volTmp;
        gb = gbTmp;
        deltaT = deltaT*0.5;
        flagVolChange = 0;
        pause(2);
    else
        if sum( gb < gbTmp) > 0
            letsHaveABreak = 0;
        end
```

```
            flagVolChange = 1;
            gs = ((3*vol)/(4*pi)).^(1/3);
            timeT = timeT + deltaT;
            timeMat = [timeMat; deltaT timeT];
            sizeTimeMat = size(timeMat);
            gsMat(sizeTimeMat(1), 1:sizeGs(2)) = gs;
            gbMat(sizeTimeMat(1), 1:sizeGb(2)) = gb;
            volWeightedMean = [volWeightedMean; sum(gs.*vol)/sum(vol
               )];


             neckSizeMatrix = [neckSizeMatrix; neckSizeMat];
                dlmwrite('neckSize.txt', neckSizeMatrix, '\t');
             dlmwrite('gsMat.txt', gsMat,'\t'); dlmwrite('gbMat.txt
                ', gbMat,'\t'); dlmwrite('timeMat.txt', timeMat,'\t'
                );dlmwrite('volWeightedMean.txt', volWeightedMean,'\
                t');
            pause(0.1);
             deltaT = deltaT*4;
        end

        if deltaT< 1E-10
            flagGrainDisappear = 1; flagEvolve = 0;
            deltaT = max([deltaTi deltaT*1000]);
        end
    end
end


function[R1, R2, Rgb, alpha, beta, theta, errorR1, errorR2] =
   soapBubbleGeom(R1P, R2P, X, vol1, gammaS, gammaGb)

lhs = 4*pi/3 * (R1P^3 + R2P^3); %Total volume

matR = [];

DeltaR = 0.0005;
R1 = R1P;
diff1 = 1;
while diff1 > 0
    if X/R1<=1
        alpha = asin(X/R1);
        mat = [];
        R2 = 0;
        diff = 1;
        while diff>0
            if X/R2 <= 1
                beta = asin(X/R2);
                rhs = pi * (R1^3 * (cos(alpha) - (1/3)* (cos(alpha)
```

```
                    )^3 + 2/3)) + ...
                        pi * (R2^3 * ( cos(beta) - (1/3)*(cos(beta))
                            ^3 + 2/3));
                diff = (lhs - rhs)/lhs*100;
                mat = [mat; diff alpha beta R2];
            end
            R2 = R2 + DeltaR;
        end
        [R2 i] = min(abs(mat(:, 1)));
        R2 = mat(i,4);
        alpha = mat(i, 2);
        beta = mat(i,3);
        error = mat(i,1);
        Rgb = (gammaGb/gammaS) * (R1*R2)/(R1 - R2);
        theta = asin(X/Rgb);
        vol = pi * R1^3 * (cos(alpha) - (1/3)*(cos(alpha))^3 + 2/3)
            ...
                - pi * Rgb^3 * (cos(pi-theta) - (1/3)*(cos(pi-theta
                    ))^3 + 2/3);
        diff1 = (vol1 - vol)/vol1 * 100;

        if (abs(error) <1) ||(beta < pi/2)
            matR = [matR; diff1 R1 R2 Rgb alpha beta theta error];
        end

    end
    R1 = R1 + DeltaR;

end

[R1 i] = min(abs(matR(:,1)));
errorR1 = matR(i,1);
R1 = matR(i,2);
R2 = matR(i,3);
Rgb = matR(i,4);
alpha = matR(i,5);
beta = matR(i,6);
theta = matR(i,7);
errorR2 = matR(i,8);

sizeMat = size(matR);
if sizeMat(1) < 1
    alpha = 0;
    beta = pi/2;
    theta = 0;
    errorR1 = 1E10;
    errorR2 = 1E10;
end
```

```
function [R1, R2, Rgb, X, alpha, beta, theta, errorBeta]=
   soapBubbleGeomForRapidGBM(R1, R2, volP, gammaS, gammaGb)

Rgb = (gammaGb/gammaS) * (R1*R2)/(R1 - R2);


beta = 0;
deltaBeta = 1*pi/180;
volDiff = 0;
mat = [];
volOld =0;
volNew = 0;
while volDiff <= 0
    X = R2*sin(beta);
    theta = asin(X/Rgb);
    vol = pi * R2^3 * (cos(pi-beta) - (1/3)*(cos(pi-beta))^3 + 2/3)
       ...
                + pi * Rgb^3 * (cos(pi-theta) - (1/3)*(cos(pi-theta
                   ))^3 + 2/3);
    volDiff = vol - volP;
    mat = [mat; volDiff beta];
    beta = beta + deltaBeta;
    volOld = volNew;
    volNew = volDiff;
end
if abs(volNew) < abs(volOld)
    beta = beta - deltaBeta;
else
    beta = beta - 2*deltaBeta;
end
X = R2*sin(beta);
theta = asin(X/Rgb);
errorBeta = abs(min([volNew, volOld]))/volP;
alpha = asin(X/R1);
function[R1, R2, r1,r2, x, alpha, beta, theta, neckVol, error] =
   noShrink3D_1(R1P, R2P, d, xOld, xStepFlag)

mat1 = 1E20*ones(2,7);

psiBy2 = pi/3;
dR2 = (R1P + R2P - d)/( 1 + R2P^2/R1P^2);
dR1 = (R2P^2/R1P^2)*dR2;
R1 = R1P - dR1;
R2 = R2P - dR2;

vol1 =  4*pi/3*(R1P^3 + R2P^3);
R3 = 100000;
if R1>R2
    R3 = R1*R2/(R1 - R2);
```

```
end
d3 = R1 + R3;

x = 0;%xOld;
deltaX = 0.0001;
if xStepFlag == 1
    deltaX = 0.00001;
end
while mat1(2,1) > 0 && x < 1%volDiff > -1E-10
      theta = asin(x/R3);
    A = R1 + R3 - R3*cos(theta); B = x;
    r1 = 0.5 * ( A^2 + B^2 - R1^2)/( R1 - A*cos(psiBy2 - theta) - B
       *sin(psiBy2- theta));

    A = d - d3 + R3*cos(theta);
    B = x;
    r2 = 0.5*(A^2 + B^2 - R2^2)/(R2 - A*cos(psiBy2+theta) - B*sin(
       psiBy2 + theta));

    alpha = asin ( (x + r1*sin(psiBy2 - theta))/(R1 + r1));
    beta = asin(    (x + r2*sin(psiBy2 + theta))/(R2 + r2));

    cy2 = (R1+r1)*sin(alpha);

    v1 = pi * R1^3 * (cos(alpha) - (1/3)*(cos(alpha))^3  + 2/3);

    A = pi - alpha; b = cy2;
    v2L = b^2*r1*cos(A) + r1^3*(cos(A) - (1/3)*(cos(A))^3) + b*r1*
       r1*(A-0.5*sin(2*A));
    A = pi - psiBy2 + theta;
    v2R = b^2*r1*cos(A) + r1^3*(cos(A) - (1/3)*(cos(A))^3) + b*r1*
       r1*(A-0.5*sin(2*A));
    v2 = pi*(v2R - v2L);

    A = psiBy2 + theta; b = (R2+r2)*sin(beta);
    v3L = b^2*r2*cos(A) + r2^3*(cos(A) - (1/3)*(cos(A))^3) + b*r2*
       r2*(A-0.5*sin(2*A));
    A = beta;
    v3R = b^2*r2*cos(A) + r2^3*(cos(A) - (1/3)*(cos(A))^3) + b*r2*
       r2*(A-0.5*sin(2*A));
    v3 = pi*(v3R - v3L);

    v4 = pi * R2^3*( cos(beta) - (1/3)*(cos(beta))^3 + 2/3);

    vol2 = v1 + v2 + v3 + v4;
    volDiff = vol1 - vol2;
  if r1>=0 && r2>=0
    mat1(1,:) = mat1(2,:);
    mat1(2,:) = [volDiff x r1 r2 alpha  beta theta];
```

```
  end
   x = x + deltaX;

end

[r f] = min(abs(mat1(:,1)));
r1 = mat1(f,3);
r2 = mat1(f,4);

alpha = mat1(f,5);
beta = mat1(f,6);

x = mat1(f, 2);
theta = mat1(f,7);
error = mat1(f,1);
if x == 1
    x = 1E20;
end
neckVol = 4*pi/3 * (R1P^3 + R2P^3 - R1^3 - R2^3);


function[r,x, alpha, beta,theta, neckVol, error] = aaha(R1, R2, d)

mat = [];
h1 = (R1^2 -R2^2 + d^2)/(2*d);
k1 = sqrt(R1^2 - h1^2);

psiBy2 = pi/3;

vol1 =  4*pi/3*(R1^2 + R2^2);
R3 = 1000;
if R1>R2
    R3 = R2*R2/(R1 - R2);
end
d3 = h1 + sqrt(R3^2 - k1^2);

for x = k1:0.0001:R1
    theta = asin(x/R3);
    A = d3 - R3*cos(theta); B = x;
    r = 0.5 * ( A^2 + B^2 - R1^2)/( R1 - A*cos(psiBy2 - theta) - B*
       sin(psiBy2- theta));
    alpha = asin ( (x + r*sin(psiBy2 - theta))/(R1 + r));
    beta = asin(    (x + r*sin(psiBy2 + theta))/(R2 + r));
    cx1 = 0; cy1 = 0;
    cx2 = (R1+r)*cos(alpha); cy2 = (R1+r)*sin(alpha);
    cx3 = d - (R2+r)*cos(beta); cy3 = (R2 +r)*sin(beta);
    cx4 = d; cy4 = 0;
```

```
    h = 0.5 * ( d + (R1+r)*cos(alpha) - (R2+r)*cos(beta) - 2*r*sin(
        psiBy2)*sin(theta));

    h1_l = -R1; h1_r = R1*cos(alpha);
    h2_l = R1*cos(alpha); h2_r = h;
    h3_l = h;    h3_r = d-R2*cos(beta);
    h4_l = d-R2*cos(beta); h4_r = d + R2;

    x1 = linspace(h1_r, h1_l, 10000); z1 = (h1_r-h1_l)/10000;
    x2 = linspace(h2_r, h2_l, 10000); z2 = (h2_r-h2_l)/10000;
    x3 = linspace(h3_r, h3_l, 10000); z3 = (h3_r-h3_l)/10000;
    x4 = linspace(h4_r, h4_l, 10000); z4 = (h4_r-h4_l)/10000;

    y1_2 = R1^2 - x1.^2;
    y2_2 = (cy2 - sqrt(r^2 - (x2-cx2).^2)).^2;
    y3_2 = (cy3 - sqrt(r^2 - (x3-cx3).^2)).^2;
    y4_2 = R2^2 - (x4-cx4).^2;

    v1 = z1*pi*trapz(y1_2);
    v2 = z2*pi*trapz(y2_2);
    v3 = z3*pi*trapz(y3_2);
    v4 = z4*pi*trapz(y4_2);

    vol2 = v1 + v2 + v3 + v4;
    volDiff = vol1 - vol2;

    mat = [mat; volDiff x r alpha beta theta h vol1 vol2];

end

[r f] = min(abs(mat(:,1)));
r = mat(f,3);
alpha = mat(f,4);
beta = mat(f,5);
x = mat(f, 2);
theta = mat(f,6);
error = mat(f,1);

cx2 = (R1+r)*cos(alpha); cy2 = (R1+r)*sin(alpha);
cx3 = d - (R2+r)*cos(beta); cy3 = (R2 +r)*sin(beta);

h = 0.5 * ( d + (R1+r)*cos(alpha) - (R2+r)*cos(beta) - 2*r*sin(
    psiBy2)*sin(theta));

h2_l = R1*cos(alpha); h2_r = h;
h3_l = h;    h3_r = d-R2*cos(beta);


x2 = linspace(h2_r, h2_l, 1000); z2 = (h2_r-h2_l)/1000;
```

```
x3 = linspace(h3_r, h3_l, 1000); z3 = (h3_r-h3_l)/1000;

y1_2 = R1^2 - x2.^2;
y2_2 = (cy2 - sqrt(r^2 - (x2-cx2).^2)).^2;
y3_2 = (cy3 - sqrt(r^2 - (x3-cx3).^2)).^2;
y4_2 = R2^2 - (x3-cx4).^2;

v1 = z2*pi*trapz(y1_2);
v2 = z2*pi*trapz(y2_2);
v3 = z3*pi*trapz(y3_2);
v4 = z3*pi*trapz(y4_2);

neckVol = -v1 + v2 + v3 - v4;
```

REFERENCES

[1] R. L. Coble, "Initial sintering of Alumina and Hematite," *Journal of American Ceramic Society*, vol. 41, no. 2, pp. 55–62, 1958.

[2] R. L. Coble and J. E. Burke, "Sintering in ceramics," in *Progress in Ceramics Science*, J. E. Burke, Ed. MacMillan, NY, 1963, pp. 197–251.

[3] R. L. Coble, "Effects of particle size distribution on initial stage sintering," *Journal of American Ceramic Society*, vol. 56, no. 9, pp. 461–466, 1973.

[4] F. F. Lange and B. J. Kellett, "Thermodynamics of densification: 1, sintering kinetics of simple particle arrays, equilibrium configurations, pore stability, and shrinkage," *Journal of American Ceramic Society*, vol. 72, no. 5, p. 72534, 1989.

[5] ——, "Thermodynamics of densification: 2, grain growth in porous compacts and relation to densification," *J. Am. Cera. Soc.*, vol. 72, no. 5, p. 73541, 1989.

[6] M. N. Rahman, *Sintering of Ceramics*. NY: CRC Press, 2007.

[7] F. Wakai, M. Yoshida, Y. Shinoda, and T. Akatsu, "Coarsening and grain growth in sintering of two particles of different sizes," *Acta Materialia*, vol. 53, pp. 1361–1371, 2005.

[8] J. Pan, H. Le, S. Kucherenko, and J. A. Yeomans, "A model for the sintering of spherical particles of different sizes by solid state diffusion," *Acta Materialia*, vol. 46, no. 13, pp. 4671–4690, 1998.

[9] S. C. Colbeck, "Sintering of unequal grains," *Journal of Applied Physics*, vol. 89, no. 5, pp. 4612–4618, 2001.

[10] P. S. Laplace, *Mecanique Celeste, supplement to the tenth book*. Paris: Duprat, 1806.

[11] M. N. Rahman, *Ceramic processing and sintering*. NY: Marcel Dekker, 1995.

[12] W. D. Kingery and M. Berg, "Study of initial stages of sintering solids by viscous flow, evaporation-condensation, and self-diffusion," *Journal of Applied Physics*, vol. 26, p. 1205, 1955.

[13] S.-J. L. Kang, *Sintering - Densification, grain growth and microstructure*. Burlington, MA: Elsevier, 2005.

[14] R. L. Coble, "Sintering of crystalline solids I. intermediate and final stage diffusion models," *Journal of Applied Physics*, vol. 32, p. 787, 1961.

[15] J.-W. Noh, S.-S. Kim, and K.-S. Churn, "Collapse of interconnected open pores in solid stage sintering of W-Ni," *Metallurgical Transactions A*, vol. 23A, pp. 2141–2145, 1992.

[16] G. C. Kuczynski, "The mechanism of densification during sintering of metallic particles," *Acta Metallurgica*, vol. 4, no. 1, pp. 58–61, 1956.

[17] C. Greskovich and K. W. Lay, "Grain growth in very porous Al_2O_3 compacts," *Journal of American Ceramic Society*, vol. 55, no. 3, pp. 142–46, 1972.

[18] T. K. Gupta, "Possible correlation between density and grain size during sintering," *Journal of American Ceramic Society*, vol. 55, no. 5, pp. 276–77, 1972.

[19] C. A. Bruch, "Sintering kinetics for the high densityalumina process," *American Ceramic Society Bulletin*, vol. 41, no. 12, pp. 799–806, 1962.

[20] M. Braginski, V. Tikare, and E. A. Olevsky, "Numerical simulation of solid state sintering," *International Journal of Solids and Structures*, vol. 42, no. 2, pp. 621–636, 2005.

[21] M. I. Mendelev, D. J. Srolovitz, L. S. Shvindlerman, and G. Gottstein, "Interface mobility under different driving forces," *Journal of Materials Research*, vol. 17, no. 1, pp. 234–245, 2002.

[22] A. D. Rollett and P. Manohar, "The Monte Carlo method," in *Continuum Scale Simulation of Engineering Materials: Fundamentals Microstructures Process Applications.*, D. Raabe, F. Roters, F. Barlat, and L.-Q. Chen, Eds. Wienhem: Wiley-VCH Verlag GmbH & Co. KGaA, 2004, pp. 76–113.

[23] R. B. Potts, "Some generalized order-disorder transitions," *Proc. Cambridge Philosophical Society*, vol. 48, pp. 106–109, 1952.

[24] E. Ising, "Theorie des ferromagnetismus," *Zeitschrift für Physik*, vol. 31, p. 253, 1925.

[25] M. P. Anderson, G. S. Grest, P. S. Sahni, and D. J. Srolovitz, "Computer simulation of grain growth–1." *Acta Metallurgica*, vol. 32, no. 5, pp. 783–791, 1984.

[26] D. Turnbull, "Theory of grain boundary migration rates," *Transactions of the American Institute of Mining, Metallurgical and Petroleum Engineers*, vol. 191, pp. 661–665, 1951.

[27] I. M. Lifshitz, "Kinetics of ordering during second order phase transitions," *Soviet Physics, JETP*, vol. 15, pp. 939–942, 1962.

[28] S. M. Allen and J. W. Cahn, "A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening," *Acta Metallugica*, vol. 27, pp. 1085–1095, 1979.

[29] P. S. Sahni, G. S. Grest, and S. A. Safran, "Temperature dependence of domain kinetics in two dimensions," *Physical Review Letters*, vol. 50, no. 1, pp. 60–63, 1983.

[30] D. J. Srolovitz, M. P. Anderson, P. S. Sahni, and G. S. Grest, "Computer simulation of grain growth - II. grain size distribution, topology, and local dynamics," *Acta Metallurgica*, vol. 32, no. 5, pp. 793–802, 1984.

[31] P. A. Beck, "Annealing of cold worked metals," *Advances in Physics*, vol. 3, no. 11, pp. 245–324, 1954.

[32] P. Feltham, "Grain growth in metals," *Acta Metallaurgica*, vol. 5, no. 2, pp. 97–105, 1957.

[33] D. J. Srolovitz, M. P. Anderson, G. S. Grest, and P. S. Sahni, "Computer simulation of grain growth - III. influence of a particle dispersion," *Acta Metallurgica*, vol. 32, no. 9, pp. 1429–1438, 1984.

[34] G. S. Grest, D. J. Srolovitz, and M. P. Anderson, "Computer simulation of grain growth - IV. anisotropic grain boundary energies," *Acta Materialia*, vol. 33, no. 3, pp. 509–520, 1984.

[35] D. J. Srolovitz, G. S. Grest, and M. P. Anderson, "Computer simulation of grain growth - V. abnormal grain growth," *Acta Metallurgica*, vol. 32, no. 12, pp. 2233–2247, 1985.

[36] E. A. Holm, D. J. Srolovitz, and J. W. Cahn, "Microstructural evolution in two dimensional two phase ploycrystals," *Acta Metallurgica et Materialia*, vol. 41, no. 4, pp. 1119–1136, 1993.

[37] P. L. Liu and S. T. Lin, "Three dimensional modeling of the grain growth by coalescence in the initial stage of liquid phase sintering," *Materials Transactions*, vol. 44, no. 5, pp. 924–930, 2003.

[38] D. Zhang, G. Weng, and D. Zhou, "Computer simulation of grain growth of intermediate and final stage sintering and Ostwald ripening of $BaTiO_3$ based PTCR ceramics," *Material Science and Engineering B*, vol. 99, no. 1-3, pp. 428–432, 2003.

[39] W. D. Kingery, H. K. Brown, and D. R. Uhlmann, *Introduction to Ceramics*, 2nd ed. New York: Wiley, 1976.

[40] A. M. Meizoso, M. T. Reparáz, and J. G. Sevillano, "Un generador aleatorio de microestructuras virtuales 3d," *Revista de Metalurgia*, vol. 34, pp. 314–318, 1998.

[41] J. Aldazabal, A. M. Meizoso, and J. M. M. Esnaola, "Simulation of liquid phase sintering using the Monte Carlo method," *Materials Science and Engineering A*, vol. 365, pp. 151–155, 2004.

[42] J. Kishino, H. Nomura, S. G. Shin, H. Matsubara, and T. Tanase, "Computational study on grain growth in cemented carbides," *International Journal of Refractory Metals and Hard Materials*, vol. 20, no. 1, pp. 31–40, 2002.

[43] S. G. Shin and H. Matsubara, "Grain growth of carbide based cermets during liquid phase sintering," in *Sintering Technology*, R. M. German, G. L. Messing, and R. G. Cornwell, Eds. Marcel Dekker, NY, 1996, pp. 157–172.

[44] V. Tikare, M. Braginski, and E. A. Olevsky, "Numerical simulation of solid state sintering: 1, sintering of three particles," *Journal of American Ceramic Society*, vol. 86, no. 1, pp. 49–53, 2003.

[45] V. Tikare and E. A. Holm, "Simulation of grain growth and pore migration in a thermal gradient," *Journal of American Ceramic Society*, vol. 81, no. 3, pp. 480–484, 1998.

[46] V. Tikare, E. A. Moidowink, and E. A. Holm, "Three dimensional simulation grain growth in presence of mobile pores," *Journal of American Ceramic Society*, vol. 84, no. 6, pp. 1379–1385, 2001.

[47] J. W. Cahn and J. E. Hilliard, "Free energy of nonuniform system. i. interfacial free energy," *Journal of Chemical Physics*, vol. 28, p. 258, 1958.

[48] J. W. Cahn, "On spinodal decomposition," *Acta Metallurgica*, vol. 9, pp. 795–801, 1961.

[49] D. Raabe, *Computational Materials Science: The Simulation of Materials Microstructure and Properties*. Wienhem: Wiley-VCH, 1999.

[50] K. G. F. Janssens, D. Raabe, E. Kozeschnik, M. A. Miodownik, and B. Nestler, *Computational Materials Engineering: An Introduction to Materials Evolution*. San Diego, CA: Elsevier Academic Press, 2007.

[51] D. Fan, S. P. Chen, L. Q. Chen, and P. W. Voorhees, "Phase-field simulation of 2-d ostwald ripening in the high volume fraction regime," *Acta Metallurgica*, vol. 50, pp. 1895–1907, 2002.

[52] V. Tikare, E. A. Holm, D. Fan, and L. Q. Chen, "Comparison of phase field and potts models for coarsening process," *Acta Materialia*, vol. 47, no. 1, pp. 363–371, 1998.

[53] Y. U. Wang, "Computer modeling and simulation of solid state sintering, a phase field approach," *Acta Materialia*, vol. 54, pp. 953–961, 2006.

[54] L. D. Landau and E. M. Lifshitz, *Statistical Physics*. Oxford: Butterworth-Heinemann, 1980.

[55] V. L. Ginzburgh and L. D. Landau, "On the theory of superconductivity," *Soviet Physics – Journal of Experimental and Theoretical Physics*, vol. 20, pp. 1064–1082, 1950.

[56] L. Q. Chen and J. Shen, "Applications of semi-implicit fourier-spectral method to phase field equations," *Computer Physics Communications*, vol. 108, no. 2-3, pp. 147–158, 1998.

[57] L. Q. Chen and Y. Wang, "The continuum field approach to modeling microstructural evolution," *JOM*, vol. 48, no. 12, pp. 13–18, 1996.

[58] J. W. Cahn and J. E. Hilliard, "On spinodal decomposition. A reprise," *Acta Metallurgica*, vol. 19, no. 2, pp. 151–161, 1971.

[59] D. Fan and L. Q. Chen, "Computer simulation of grain growth using continuum field model," *Acta Materialia*, vol. 45, no. 2, pp. 611–622, 1997.

[60] D. Fan, L. Q. Chen, and S. P. Chen, "Effect of grain boundary width in a diffuse-interface field model," *Materials Science and Engineering A*, vol. 283, pp. 78–84, 1997.

[61] N. Ma, A. Kazaryan, S. A. Dregia, and Y. Wang, "Computer simulation of texture evolution during grain growth: Effect of boundary properties and initial microstructure," *Acta Materialia*, vol. 52, pp. 3869–3879, 2004.

[62] W. T. Read and W. Shockley, "Dislocation models of crystal grain boundaries," *Physical Review*, vol. 78, pp. 275–289, 1950.

[63] Y. Huang and H. J. Humphreys, "Subgrain growth and low angle boundary mobility in aluminium crystals of orientation {110}[001]," *Acta Metallurgica*, vol. 48, no. 8, pp. 2017–2030, 2000.

[64] K. J. Ko, P. R. Cha, J. T. Park, J. K. Kin, and N. M. Hwang, "Phase field model simulation of grain growth on three dimensions under isotropic and anisotropic grain boundary energy conditions," *Materials Science Forum*, vol. 538-559, pp. 1101–1106, 2007.

[65] P. W. Voorhees, "Ostwald ripening of two-phase mixtures," *Annual Review of Materials Science*, vol. 22, pp. 197–215, 1992.

[66] L. Q. Chen and D. Fan, "Computer simulation model for coupled grain growth and Ostwald ripening application to Al_2O_3-Zro_2 two-phase systems," *Journal of American Ceramic Society*, vol. 79, no. 5, pp. 1163–1168, 1996.

[67] K. B. Alexander, P. F. Becher, S. B. Waters, and A. Bleier, "Grain growth kinetics in alumina-zirconia (CeZTA) composites," *Journal of American Ceramic Society*, vol. 77, p. 939, 1994.

[68] K. Asp and J. Agren, "Phase field simulation of sintering and related phenomena - A vacancy diffusion approach," *Acta Materialia*, vol. 54, pp. 1241–1248, 2006.

[69] V. Kumar, Z. Z. Fang, and P. C. Fife, "Phase field simulations of grain growth during sintering of two unequal sized particles," *Materials Science and Engineering A*, vol. 528, no. 1, pp. 254–259, 2010.

[70] Z. Z. Fang and H. Wang, "Densification and grain growth during sintering of nanosized particles," *International Materials Reviews*, vol. 53, no. 6, pp. 326–352, 2008.

[71] L. G. Kornelyuk, A. Y. Lozovoi, and I. M. Razumovskii, *Diffusion Defect Data A*, vol. 143A, no. 1, p. 1481, 1997.

[72] R. Wuerschum, K. Reimann, and P. Farber, *Diffusion Defect Data A*, vol. 143A, no. 1, p. 1463, 1997.

[73] A. Nazarov, "Grain-boundary diffusion in nanocrystals with a time-dependent diffusion coefficient," *Physics of the Solid State*, vol. 45, no. 6, pp. 1166–1169, 2003.

[74] W. H. Rhodes, "Agglomerate and particle size effects on sintering of yttria-stabilized zirconia," *Journal of American Ceramic Society*, vol. 64, no. 1, pp. 19–22, 1981.

[75] E. N. Hodkin, M. G. Nicholas, and D. M. Poole, "The surface energies of solid molybdenum, niobium, tantalum and tungsten," *Journal of Less Common Metals*, vol. 20, no. 2, pp. 93–102, 1970.

[76] T. J. Quinn, "The effect of thermal etching on emissivity of tungsten," *British Journal of Applied Physics*, vol. 16, p. 973, 1965.

[77] E. Lassner and W.-D. Shubert, *Tungsten: Properties, Chemistry, Technology of the Element, Alloys, and Chemical Compounds*. NY: Springer, 1999.

[78] J. Zhao and M. P. Harmer, "Effect of pore distribution on microstructure development: I, matrix pores," *Journal of American Ceramic Society*, vol. 72, no. 2, pp. 113–120, 1988.

[79] S. J. Dillon and M. P. Harmer, "Intrinsic grain boundary mobility in alumina," *Journal of American Ceramic Society*, vol. 89, no. 12, pp. 3885–3887, 2006.

[80] H. Wang, "Sintering and grain growth of nanasized powder," Ph.D. dissertation, University of Utah, 2011.

[81] Z. Z. Fang, B. R. Patterson, and M. E. Turner, "Influence of particle size distribution on coarsening," *Acta Metallurgica et Materialia*, vol. 40, no. 4, pp. 713–722, 1992.