2007

Computer-aided modeling of controlled release through surface erosion with and without microencapsulation

Stephanie Tomita Wong
University of South Florida

Computer-Aided Modeling of Controlled Release through Surface Erosion with and without Microencapsulation

by

Stephanie Tomita Wong

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Chemical Engineering
Department of Chemical Engineering
College of Engineering
University of South Florida

Major Professor: Aydin K Sunol, Ph.D.
John T. Wolan, Ph.D.
Sermin G. Sunol, Ph.D.

Date of Approval:
November 2, 2007

Keywords: encapsulation, release rate, mathematical modeling, MATLAB, COMSOL

Dedication

This thesis is dedicated to my parents whose love, guidance and sacrifice have given me the opportunity to live my dreams.

This thesis is also dedicated in loving memory of my Grandmother May. Even though she was not able to see the completion of this thesis, she always knew I would finish.

Acknowledgments

Four years ago, I returned to school unsure of whether I had the ability to complete my academic goals. It was only with the support and help of a special few that this thesis could be completed.

My thanks and appreciation to Dr. Aydin Sunol and Dr. Sermin Sunol for their encouragement and guidance throughout the time it took for me to complete this research and to write this thesis. My committee member, Dr. John Wolan, has provided years of inspiration. His integrity, respect for students' scientific liberty and kindness to those around him has filled me with admiration. Many thanks go to the past and present members of the Environmentally Friendly Engineering Systems research group, who shared their knowledge and experience with me: Dr. Naveed Aslam, Raquel Carvallo, Keyur Patel, Haitao Li, and Wade Mack. Special thanks is given to Brandon Smeltzer, whose constant help throughout these years has been immeasurable. I also appreciate the many contributions from members of Chemical Engineering Department at USF: Dr. Vinay Gupta, Dr. Scott Campbell, Dr. Babu Joseph, Ms. Carla Webb, Ms. Cay Pelaez, Mr. Ed Van Etten and Mr. Jamie Fargen. I would also like to acknowledge the use of the services provided by Research Computing at the University of South Florida.

Finally, I would like to thank my family and friends to whom this thesis is dedicated to. My father has always encouraged and motivated me to learn and has been my sole inspiration to thinking critically. He has instilled within me the importance making meaningful contributions using these skills. My life would not be complete without the unconditional support and love of my mother. Her dedication and keen perspective are a constant inspiration to me. I also have deep appreciation for my sister, who is the most amazing person I know. She continues to fill my life with compassion as my closest confidant and with happiness as my best friend. I am especially grateful to Sharon, whose exceptional friendship has lasted throughout the most dynamic years of my life. This thesis is also dedicated to Dat, whose enduring support, patience and belief in me has given me the strength to become who I am.

Note to Reader

The original of this document contains color that is necessary for understanding the data. The original thesis is on file with the USF library in Tampa, Florida.

Table of Contents

List of Tables v

List of Figures vi

List of Symbols ix

List of MATLAB/COMSOL Script Codes xi

Abstract xii

Chapter 1 : Introduction... 1

1.1 Background 1

1.2 Applications 3

1.2.1 Pharamaceuticals... 3

1.2.2 Foods 3

1.2.3 Fertilizers, Pesticides and Detergents 4

1.3 Review of Dissolution Models... 5

1.3.1 Historical Background 6

1.3.2 Additional Considerations... 8

1.3.3 Computer-Aided Modeling... 8

1.4 Limitations 10

1.5 Purpose of Study 10

Chapter 2 : Overview... 12

2.1 General Dissolution... 12

2.2 Governing Equations... 13

2.3 Modeling Assumptions... 14

Chapter 3 : Monodisperse Particle Dissolution ... 15

3.1 MATLAB ... 15

3.2 Program Design ... 15

3.2.1 Starting the Program ... 16

3.2.2 Determination of Diffusion Coefficient for a System ... 17

3.2.3 Calculation of the Number of Particles based on Mass ... 17

3.2.4 Radius Relationship to Time ... 18

3.2.5 Relationship of Radius to Concentration and Conversion ... 19

3.2.6 Visualization of Results ... 20

3.3 An Example ... 20

3.3.1 Simulation Setup ... 20

3.3.2 Some Results ... 21

3.4 COMSOL Script ... 26

3.5 Visualization ... 27

3.6 Summary of Results ... 33

Chapter 4 : Polydisperse Particle Dissolution ... 35

4.1 Programming Modifications ... 35

4.1.1 Calculation of Number of Particles with Varying Size Distribution .. 36

4.1.2 Calculation of Constants for All Size Distributions ... 38

4.1.3 Determination of Radii for each Group at Specified Times ... 39

4.1.4 Relating Radius to Concentration and Conversion for Each Group ... 41

4.1.5 Visualizing the Results of the Polydisperse Model ... 42

4.2 Examples using the Polydisperse Model ... 42

4.2.1 Simulation for Test Cases ... 42

4.2.2 Determination of Experimental Radius Size Distribution ... 49

4.2.3 MATLAB Output ... 54

4.2.4 Results for Various Initial Concentrations 57

4.3 Visualization for Multiple Particles 59

4.4 Summary 66

Chapter 5 : Encapsulated Monodisperse Particles 67

5.1 Model for Encapsulation 69

5.1.1 Calculation of Diffusion Coefficient for the Encapsulated Layer....... 69

5.1.2 Determining the Radii of Encapsulated Layer 70

5.1.3 Concentration and Conversion of Encapsulation Material 71

5.1.4 Visualization of Results for Encapsulated Model 72

5.2 Example for Encapsulated Model 72

5.2.1 Simulation Setup for Encapsulation Model 72

5.2.2 Effect of Encapsulation Thickness 74

5.2.3 COMSOL Visualization for Encapsulated Model 76

5.2.4 Encapsulation Thickness Effect 84

Chapter 6 : Encapsulated Polysdisperse Particle Dissolution 86

6.1 Program Build Up 86

6.1.1 Calculating Equivalent Number of Particles 87

6.2 Encapsulated Polydisperse Particle Example 88

6.3 Effect of Encapsulation Thickness 91

Chapter 7 : Results and Discussion 99

7.1 Experimental Validation 99

7.1.1 Reaction Microcalorimeter Experiments 100

7.1.2 Additional Experimental Tests 105

7.2 Interpreting Discrepancies 111

7.2.1 Modeling Assumptions 111

7.2.2 Experimental Inaccuracies 114

7.3 Encapsulated Models 115

7.3.1 Sources of Error 117

7.4 Comparison of Methods 118

Chapter 8 : Conclusion 119

8.1 Implications 119

8.2 Future Work 121

References 124

Appendices 130

Appendix A: Derivation of Dissolution of Solid Particles in a Liquid 131

Appendix B: MATLAB Sample Source Code 137

Appendix C: COMSOL Sample Source Code 142

Appendix D: Experimental Procedure using Reaction Microcalorimeter 144

Appendix E : Sample Data 150

List of Tables

Table 3.1: Parameters for citric acid used in test case 20

Table 3.2: Results from simulation for various radii sizes for 0.10 grams citric acid 21

Table 4.1 : Size distributions percents for five test cases.. 42

Table 4.2: Average mass percent of each radius size.. 54

Table 4.3: Number of particles in each radius size distribution. 55

Table 5.1: Parameters for glucose coating material. 72

Table 5.2: Effect of encapsulation thickness on glucose amount and dissolution time. 84

List of Figures

Figure 1.1: Schematic illustration of the principle of surface and bulk erosion [52]. 2

Figure 1.2: Common active ingredient application versus controlled release application [6]. ... 5

Figure 1.3: Dissolution of monodisperse powder [11]. 7

Figure 3.1: Program algorithm for monodisperse particles model. 16

Figure 3.2: Monodisperse model for 0.10 grams citric acid for various radius sizes. 22

Figure 3.3: Monodisperse model for 0.10 grams citric acid with initial radius of 0.059 cm. ... 23

Figure 3.4: Monodisperse model for 0.10 grams citric acid at different temperatures. 23

Figure 3.5: Monodisperse model for citric acid with different initial amounts. 25

Figure 3.6: Visualization with corresponding conversion at a) t = 0 min, b) t = 2 mins, c) t = 4 mins, d) t = 6 mins and t = 8 mins. 28

Figure 3.7: Visualization of 3D particle at a) t = 0 min, b) t = 2 mins, c) t = 4 mins and d) t = 6 mins 33

Figure 4.1: Program algorithm for polydisperse particles model. 36

Figure 4.2: Schematic illustration of polydisperse particle dissolution. 37

Figure 4.3: Conversion versus time graphs and radius size distributions for a) Test 1, b) Test 2, c) Test 3, d) Test 4 and e) Test 5. 43

Figure 4.4: Polydisperse model for citric acid at different initial amounts. 48

Figure 4.5: Equivalent spherical radius. 49

Figure 4.6: SEM and particle numbering for collection of equivalent area for six samples a)-f) 50

Figure 4.7: Radius size distributions for six samples a)-f). 52

Figure 4.8: Average radius size distribution of six samples using equivalent area from SEM .54

Figure 4.9: Concentration versus time for each radius size group. 56

Figure 4.10: Total concentration change of polydisperse particles. 56

Figure 4.11: Total conversion of polydisperse particles. 57

Figure 4.12: Polydisperse conversion versus time for citric acids at different initial concentrations. 58

Figure 4.13: Polydisperse particle dissolution for a) t = 0 min, b) t = 2 mins, c) t = 6 mins, d) t = 10 mins e) t = 15 mins and f) t =20 mins ... 60

Figure 5.1: Encapsulated particle a) full view b) side section view c) front section view ... 67

Figure 5.2: Program algorithm for encapsulated monodisperse particles model. ... 68

Figure 5.3: Inner particle radius, $r_{0,}$ with encapsulation layer thickness, h, yields radius of encapsulated particle r_{0enc} ... 69

Figure 5.4: Schematic illustration for encapsulated monodisperse particle dissolution ... 70

Figure 5.5: Concentration versus time for encapsulation thickness of 0.0010 cm. ... 73

Figure 5.6: Conversion versus time for encapsulation thickness of 0.0010 cm ... 74

Figure 5.7: Concentration versus time for encapsulation thickness of 0.010 cm. ... 75

Figure 5.8: Conversion versus time for encapsulation thickness of 0.010 cm. ... 75

Figure 5.9: Encapsulated particle conversion for a) t = 0 min, b) t = 1 mins, c) t = 3 mins, d) t = 5mins, e) t = 7 mins and f) t = 9 mins. ... 77

Figure 5.10: Effect of encapsulation thickness on the delayed conversion of citric acid. ... 85

Figure 6.1: Schematic illustration for encapsulated polydisperse particles. ... 86

Figure 6.2: Program algorithm for encapsulated polydisperse particles model. ... 87

Figure 6.3: Concentration changes for encapsulation and inner particle for all size distributions ... 88

Figure 6.4: Concentration change for various size distributions. ... 89

Figure 6.5: Total concentration versus time for polydisperse encapsulated particles. ... 90

Figure 6.6: Total conversion versus time for polydisperse encapsulated particles ... 90

Figure 6.7: Increased encapsulation layer thickness for polydisperse model. ... 91

Figure 6.8: Concentration change versus time for all size distributions with increased coating thickness. ... 91

Figure 6.9: Encapsulated polydisperse particle dissolution for a) t = 0 mins, b) t = 2 mins, c) t = 4 mins, d) t = 6 mins, e) t = 8 mins, f) t = 10 mins and g) t = 12 mins ... 92

Figure 7.1: OmniCal SuperCRC reaction microcalorimeter ... 100

Figure 7.2: Reaction microcalorimeter heat flow and conversion graph ... 102

Figure 7.3: Experimental data versus monodisperse model for varying initial radii ... 103

Figure 7.4: Experimental data versus polydisperse model for 0.10 grams of citric acid ... 104

Figure 7.5: Comparison of experimental data for 0.10 grams citric acid with both models ... 105

Figure 7.6: Experimental results for different amounts of citric acid ... 106

Figure 7.7: Comparison of experimental data for 0.02 grams citric acid with both models ... 107

Figure 7.8: Comparison of experimental data for 0.04 grams citric acid with both models ... 107

Figure 7.9: Comparison of experimental data for 0.06 grams citric acid with both models ... 108

Figure 7.10: Comparison of experimental data for 0.08 grams citric acid with both models ... 108

Figure 7.11: Comparison of experimental data for 0.20 grams citric acid with both models ... 109

Figure 7.12: Comparison of experimental data for 0.30 grams citric acid with both models ... 109

Figure 7.13: Comparison of experimental data for 0.40 grams citric acid with both models ... 110

Figure 7.14: Comparison of experimental data for 0.50 grams citric acid with both models ... 110

Figure 7.15: SEM photos of non-spherical citric acid particles. 111

Figure 7.16: Stirred experimental data for citric acid for different initial amounts 113

Figure 7.17: Comparison plot for stirred data using 0.10 grams of citric acid 114

Figure 7.18: Experimental results for citric acid encapsulated with 0.01 and 0.10 grams of glucose. 116

Figure 7.19: Monodisperse model for citric acid encapsulated with 0.01 and 0.10 grams of glucose……... 116

Figure 7.20: Polydisperse model for citric acid encapsulated with 0.01 and 0.10 grams of glucose. 117

Figure 8.1: Comprehensive flowsheet for all four programs developed in this work. 120

Figure 8.2: Other non-spherical geometric shapes of particles 122

Figure 8.3: Surface area changes between a) low concentration and b) high concentration of solid particles……….. 123

Figure B.1: MATLAB resulting plot for concentration for monodisperse model 140

Figure B.2: MATLAB resulting plot for conversion for monodisperse model 141

Figure C.1: Command prompt window in COMSOL Script..142

Figure C.2: COMSOL Script visualization of spherical particle 143

Figure D.1: WinCRC Turbo microcalorimeter program setup window 145

Figure D.2: Heat flow curve from reaction..145

Figure D.3: Dynamic correction using heat curve option 146

Figure D.4: Applying tau correction to reaction heat flow curve 147

Figure D.5: Trimmed corrected heat flow curve..148

Figure D.6: Integrated heat flow curve and conversion 149

List of Symbols

Symbol	Description	Unit
A	Area	cm^2
C	Concentration	g/mol
C	Constant	
$c(g)$	Constant for group g	
C_o	Initial concentration	
C_s	Saturation solubility	g/ cm^3
D	Diffusion coefficient	cm^2/sec
D_{12}	Wilke-Chang diffusion coefficient	cm^2/sec
H	Encapsulation thickness	cm
H	Thickness of diffusion layer	cm
G	Group	
i, j	Counter variables in MATLAB	
K	Dissolution rate constant, integration constant	
$k(g)$	Integration constant for group g	
k_1	Brunner et al. dissolution rate constant	
k_2	Nieberall et al. square-root law dissolution rate constant	
k_3	Hixson-Crowell cubic-root law dissolution rate constant	
M	Mass of particles remaining	g
M_d	Mass dissolved at time t	g
M_o	Initial mass	g
M_t	Dissolved material at time t	g
M_T	Total mass of sample	g
M_2	Molecular weight of solvent	g/mol
μ_2	Viscosity of solvent solution	centipoises
N	Number of particles	
$N(g)$	Number of particles in group g	
$P_s(g)$	Percent of particles in group g	
$P_v(g)$	Percent by volume of each size group g	
P	Density	g/ cm^3
R	Radius of particle	cm
r_o	Initial particle radius	cm
$r_o(g)$	Average radius size for group g	
T	Temperature	K
T	Time	s
$W(g)$	Weight of each particle distribution	
W	Particle weight	g
w_0	Initial particle weight	
V	Volume of spherical particle	cm^3
$V(g)$	Volume of single particle in group g	

V_m	Dissolution medium volume	cm^3
V_P	Volume of single particle	cm^3
$V_p(g)$	Volume percent of size distribution	
$V_s(g)$	Volume of each size distribution	
V_T	Total volume	cm^3
V_1	Molar volume solute at normal boiling point	cm^3/g mol
X	Conversion	
Φ	Association parameter of solvent	
θ	Azimuthal coordinate	
φ	Polar coordinate	

List of MATLAB/COMSOL Script Codes

MATLAB code 1: Calculation of constants and insertion into dissolution function. 19

MATLAB code 2: Loop created to solve for radius at each specified time. 19

COMSOL code 1: Parametric conversion of sphere...26

MATLAB code 3: Calculating constants for various size groups.. 39

MATLAB code 4: Extended function for various size groups... 40

MATLAB code 5: Loop created to find radius for all size groups at given time........................ 40

MATLAB code 6: Assignment of mass percent for each radius size group. 55

Computer-Aided Modeling of Controlled Release through Surface Erosion

with and without Microencapsulation

Stephanie Tomita Wong

ABSTRACT

Predictive models for diffusion-controlled particle dissolution are important for designing advanced and efficient solid products for controlled release applications. A computer-aided modeling framework was developed to derive the effective dissolution rates of multiple particles as the solid surface material eroded gradually into the surrounding liquid phase. The mathematical models were solved with numerical methods using the computational software MATLAB. Results from the models were imported into COMSOL Script to create three-dimensional plots of the particle size data as a function of time. The release model found for the monodispersed particles was manipulated to incorporate polydisperse solids, as these are found more frequently in chemical processes. The program was further developed to calculate the particle size as a function of time for particles encapsulated for use in controlled release. The parameters, such as radius size, coating material and encapsulation thickness, can be altered in the computer models to aid in the design of particles for different desired applications. Simulations produced conversion profiles and three-dimensional visualizations for the dissolution processes. Experiments for the dissolution of citric acid in water were performed using a reaction microcalorimeter to verify results found from the computer models.

Chapter 1 : Introduction

1.1 Background

Controlled release technologies are emerging as a novel solution to producing efficient specialty products for applications extending far beyond their conventional use in medicine. The worldwide market for controlled release materials in areas such as agriculture, food processing and other consumer products is expected to grow to an estimated $787 million by 2012 [1]. Research and development of controlled release systems has concentrated on searching for innovations which improve the efficiency of products while staying cost effective.

The dissolution of solid particles is frequently encountered in the chemical processing industry and has been studied quantitatively for over a century [2, 3]. Pharmaceuticals, foods, fertilizers, pesticides and detergents are examples of formulated solid products whose application is reliant on the dissolution behavior [4-7]. Controlled release, the concept of sustaining or prolonging the release of beneficial agents for a specified time is a novel approach to produce safe and effective uses of active ingredients [8, 9]. In the 1960s, technology in controlled release rapidly evolved as a solution for the application of active agents which had desired objectives while avoiding adverse side effects [10]. Within the last decade, research in the dissolution of solid substances has increased with the advent of sophisticated instrumentation and computer access [11].

Mathematical models which quantitatively describe the transport mechanisms of dissolution have many applications, including aiding in the prediction of release rate profiles, understanding the effect of important formulation/processing parameters and optimizing advanced delivery systems [12]. The knowledge of the fundamental factors which influence dissolution is important for both the manufacturer and administrator of solid dosage forms [13]. Utilizing the essential models for dissolution, advanced models for controlled release can be developed. Controlled release is used in various fields to supply an effective amount of necessary material at a desired time and is most commonly attained using encapsulated particles [14]

Among the most popular classifications of controlled release formulations are erodible devices [15]. This category is defined by the release of the active agent as the carrier is eroded away by the surrounding environment though physical processes [16]. A distinct advantage of this device is that over time, the device disappears into the systems and no retrieval of the remains is necessary after activation. There are two main erosion cases, surface and bulk erosion. Figure 1.1 illustrates these principal cases. In surface erosion, the solid surface degrades much faster than the liquid intrusion into the bulk. Therefore, the resulting degradation occurs mainly on the outer layers. In contrast, in bulk erosion, particles degrade slowly while the liquid uptake by the system occurs rapidly. Erosion occurs throughout the particle, and is not restricted to the surface. This paper focuses on surface erosion, in which during degradation, the physical integrity of the particle, such as device shape or molecular weight, is maintained [17].

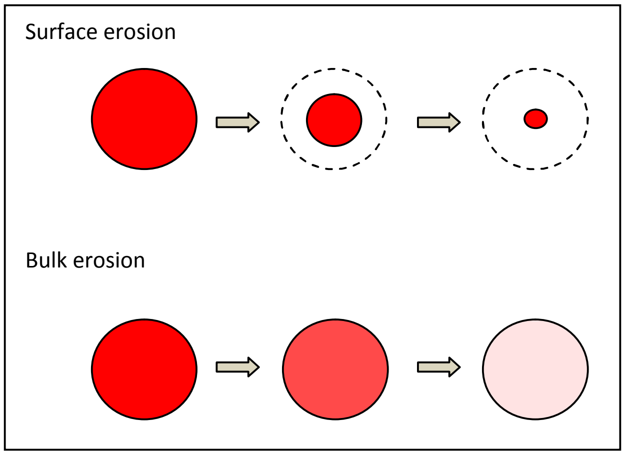


Figure 1.1: **Schematic illustration of the principle of surface and bulk erosion [52].**

In this work, the variations in the application of controlled release devices in the fields of pharmaceuticals, foods and pesticides will be reviewed. Then, the mathematical models for dissolution will be summarized followed by a discussion on the recent advances using computer-aided modeling. As with any mathematical model, limitations based on the simplifying assumptions exist, and will then be discussed. Through this examination, the need and importance for the development of advanced models for controlled release will be demonstrated.

1.2 Applications

1.2.1 Pharmaceuticals

Between the 1960s and the 1970s, there was a significant movement to research and develop microencapsulation techniques, resulting in numerous patents on microencapsulation innovation [18]. Microencapsulation involves surrounding tiny particles with a coating, such as spraying a polymer on a fluidized bed of solid drug particles [19]. Utilization of these coated particles allow drug release within the human body to be controlled and distributed over a broad time period, allowing the active agent to be absorbed continuously [19]. Controlled release of drugs have several advantages over conventional dosage forms, such as avoiding drug release in specific organs [19], reduced dose frequency, minimizing adverse side effects, improved pharmacological activity and prolonging a constant therapeutic effect [20]. In traditional drug delivery systems, such as tablets or intravenous injections, the entire dose is administered at one time. This results in suddenly elevated, close to toxic, concentrations of the drug in the plasma which ultimately lead to adverse reactions [21]. The short duration times require the patient to inconveniently repeat the administration, resulting in strong fluctuating drug levels in the body. In contrast, controlled release offer a systematic release of the appropriate drug concentration for a sustained time period. By providing the drug only where and when it is needed, drug delivery is more predictable and efficient [21]. Erodible devices are especially useful in pharmaceuticals as no surgery is required to remove the device from the body after the drug is depleted [17].

1.2.2 Foods

Encapsulation developed in the food industry as a technique to protect materials, such as food ingredients or enzymes, from moisture, heat or other harsh conditions [22]. One of the most common applications is for the incorporation functional foods, ingredients which exhibit functional benefits beyond basic nutrition. These ingredients can be used to impart negative properties like taste, such as bitterness or oxidation, or physical texture, like sedimentation or phase separation [23] . Flavors are one of the most valuable ingredients in food formulas and are a prime example of the utilization of encapsulation. Food manufactures are often concerned with protecting flavors, such as aroma substances which are expensive and are usually delicate and volatile. Coating the active ingredient can enhance the stability and viability by providing additional protection against evaporation or an undesired reaction in the food [5]. Food manufacturers are also utilizing encapsulation in foods to mask odors and tastes, since flavor control is important in food quality and acceptability. Encapsulation can also be used to release active ingredients over prolonged periods of time, reduce the loss of ingredients through cooking processes, and separate reactive or incompatible components [5].   However, due to the high costs of specialized manufacturing and unavailability of food-grade materials, food encapsulation techniques have been limited and remain as area of needed research [22].

1.2.3 Fertilizers, Pesticides and Detergents

In agriculture, controlled release techniques are used to produce accurate, reproducible and predictable rates of administration of fertilizers and pesticides. Conventionally, fertilizers and pesticides are distributed in periodic intervals which create sharp rises in the concentration levels. The high concentrations may cause undesirable side effects to the target site of the system and/or the surrounding environment. Following the initial peak of the active agent, the concentration diminishes due to natural processes such as elimination from the system, consumption or deterioration [16].   In the case of pesticides, protection from pests is required for extended periods of time. Sustained release can be achieved using encapsulated devices which reduce both the amount of pesticide used as well as the number of times it must be applied to the crop. Membrane-regulated devices are used to slowly release fertilizers and pesticides through the erosion of the membrane. These devices can also be formulated to include compounds with low water solubility, creating low dissolution rates, and the use of nitrogenous compounds which are activated by microbial action [8].   Figure 1.2 shows the improved application using controlled release compared to conventional method of delivery. As shown, the conventional application initially surpasses toxic concentration levels and then drop over time below the minimum effective level. In contrast, controlled release provides sustained release within the desired concentration range [6].

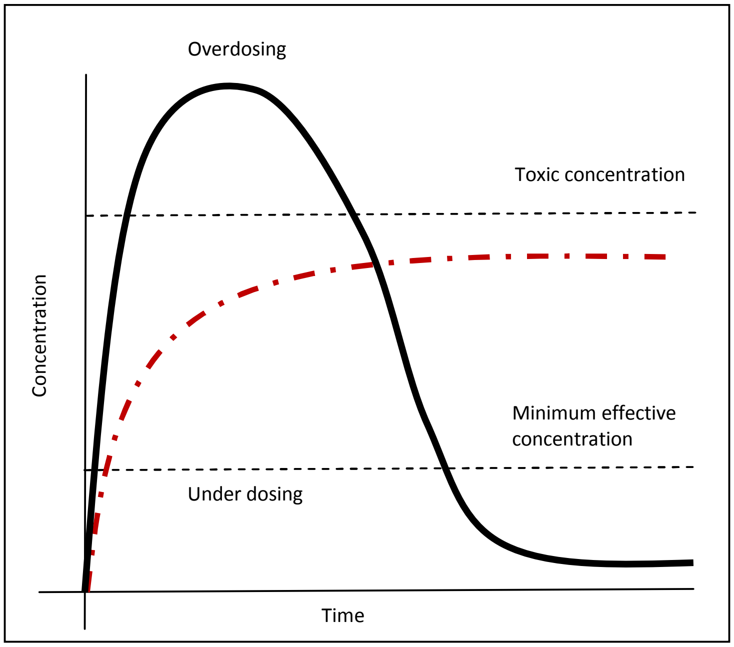


Figure 1.2: Common active ingredient application versus controlled release application [6].

Controlled release methods are also being applied to manufacture specialized detergents. Encapsulation is used to protect and control the delivery of fragrances and softeners to clothes as they are being washed. Techniques for encapsulation are currently being developed which aim at protecting volatile fragrance materials in dry environments and then releasing them when moisture is present [24]. Automatic dishwater detergents are also being encapsulated to release rinse aids, water softeners, and shine additives at elevated temperatures [24]. Formulated models for controlled release for fertilizers, pesticides and detergents are important for predicting the delivery of the active ingredients aid in the design and marketing of these products [6].

1.3 Review of Dissolution Models

Dissolution research has been developing for about a century, and since then, several approaches have been used to evaluate the release rate and dissolution behavior of substances. Mathematical models for dissolution have been developed to aid in design of more sophisticated and effective solid products. The principles of the actual processes occurring at the microscopic level must therefore be understood in order to achieve accurate dissolution models which can be applied in controlled release research. More sophisticated modeling and analysis is necessary in the study of the dissolution of particles, because unlike tablets, the surface area and/or shape changes as the dissolution process proceeds. Systems where surface recedes with time, also classified as moving boundaries problems, are solved using numerical methods [25].

1.3.1 Historical Background

The basic diffusion-controlled model for solid dissolution was developed in 1897 by Noyes and Whitney. The equation stated that the dissolution rate is proportional to the difference between the instantaneous concentration, *C* at time *t*, and the substances saturation solubility, .
The change in concentration, *-dC/dt,* can be written as

$$\frac{\mathrm{dC}}{\mathrm{dt}} = k \cdot (C_s - C) \qquad (1.1)$$

where *k* is the intrinsic dissolution rate constant. Generally, this expression stated that the rate at which the solid substance dissolved in its own solution is proportional to the difference between the concentration of the solution and the concentration of the saturated solution [26].

The equation for dissolution was expanded in 1900, when research by Erich Brunner and Stanislaus von Tolloczko showed that the rate of dissolution depends of the surface are exposed, the structure of the surface, the stirring rate, temperature and arrangement of the apparatus [27]. The Noyes-Whitney was modified by allowing $k=k_1A$, where *A* is the surface area. The model was expressed in Eq. (1.2).

$$\frac{\mathrm{dC}}{\mathrm{dt}} = k_1 \cdot (C_s - C) \qquad (1.2)$$

Then in 1904, Brunner worked with Walther Nerst to include specific relations between the constants [28, 29]. Based on the diffusion layer concept and Fick's second law, the Nerst-Brunner equation was derived by letting $k_1=D/(Vh)$, where D is the diffusion coefficient, h is the thickness of the diffusion layer and *V* is the dissolution medium volume. The equation can be written in terms of the change in concentration, such as

$$\frac{\mathrm{dC}}{\mathrm{dt}} = \frac{DA}{Vh}(C_s - C) \qquad (1.3)$$

or in terms of change in mass of solid material, *M*, as shown in Eq. (1.4).

$$\frac{\mathrm{dM}}{\mathrm{dt}} = -\frac{D}{h}A(C_s - C) \qquad (1.4)$$

Modeling of dissolution kinetics of solid particles in a liquid is often described using the Nerst-Brunner equation, as the concept of a diffusion layer of liquid on the solid surface allows the complex dissolution process to be analyzed in a tractable fashion [3].

In 1931, Hixson and Crowell developed another diffusion-controlled model for single spherical particle dissolution under sink conditions. Hixson and Crowell expressed the surface area in terms of particle weight, w, using the assumption that the surface area, A, was proportional to $w^{2/3}$ [30]. When this assumption is applied to Eq. (1.2), integration of the expression yields what is known as the cubic-root law, stated as

$$w_0^{1/3} - w^{1/3} = k_3 t \tag{1.5}$$

where w_0 is the initial weight of the particle and k_3 is a constant. According to resulting equation, the cubic root of the weight of the particle is linear with the slope k_3, which is a property of the solid and hydrodynamic characteristic of the release system. Hixson and Crowell derived this equation for single or monodisperse spherical particles under sink conditions. Monodisperse particles, which are particles with the same initial radius, undergo dissolution at same rate as shown in Figure 1.3. The cubic-root law is the widely accepted and most commonly used dissolution model because of its simplicity and general applicability to a wide range of particulate studies [31].

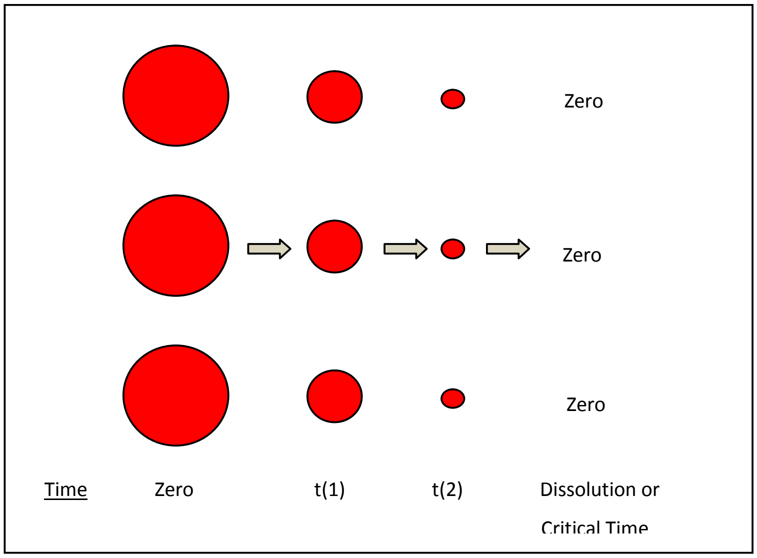


Figure 1.3: Dissolution of monodisperse powder [11].

A semi empirical expression was developed by Niebergall, Milosovich and Govan in 1963 which related time with the square-root of the particle weight [32]. This equation was written as

$$w_0^{1/2} - w^{1/2} = k_2 t \tag{1.6}$$

Through their research, they had found that a relatively constant dissolution constant, k_2, could be found by taking the square-root of the undissolved dissolution profiles. They also suggested that the diffusion layer thickness may have a square-root dependence on the particle size.

In 1961, Higuchi published a mathematical model which described the release of a solute in a diffusion-controlled system. Higuchi based his analysis on the pseudo steady-state of the kinetic release of a drug homogeneously dispersed in a planar matrix into a medium under perfect sink conditions [33]. The model states

$$M_t = K\sqrt{t} \tag{1.7}$$

where M_t is the dissolved material at a given time, t, and K is kinetic constant for the Higuchi model, represented as a composite constant with dimension time$^{-1/2}$. The Higuchi model agrees well with the experimental data for the dissolution of fine particles, under 25μm [31]. However, given the assumptions used to derive this model, it is recommended to only be used for the initial 60% of the release curves [34].

1.3.2 Additional Considerations

Polydisperse media, characterized by a wide spectrum of particle sizes, require the population distribution to be incorporated into the model for dissolution.

Coated particles also add to the complexity of the model. Given the diffusion coefficient, the release rate as a function of particle size and coating thickness can be determined.

1.3.3 Computer-Aided Modeling

The increasingly complicated mathematical models for dissolution can therefore best be solved with numerical methods implemented in computational software. The management of variables, data import, calculations, visualization and file development involved in the quantitative analysis can be easily handled with computer programming the models for dissolution.

Computer-aided modeling has been an attractive tool in solving release models due to its ability handle complex systems. A numerical solution was presented by Mauger and Howard in

1976, in which a System 360 Continuous System Modeling Program (CSMP) was developed to solve the Higuchi-Hiestand equations for a log-normal distribution [35]. The CSMP was used to solve time-variant problems, where the limits of integration where zero to time *t* or zero to infinity. However, after the CMSP was developed it also could be easily applied to solve statistical populations with various limits determined by the specific population being studied, for example, a particle-size population.

Within the last decade, computer-aided dissolution research has continued to evolve. In 2004, Frenning developed a series of FORTRAN routines to solve coupled partial differential equations (PDEs) used to describe the drug release and dissolution processes [36]. The FORTRAN routine, provided by The Numerical Algorithms Group in the United Kingdom, performed special discretization using finite differences to reduce the PDEs to a system of ordinary differential equations (ODEs). Then using a backward differentiation formula method, the resulting ODE system was solved. The work done by Stepanek in 2004, also demonstrated novel computer-aided design methodology for dissolution studies. Stepanek conducted a systematic computational study which related the granule structure to dissolution behavior [7]. The effects of granule microstructure and ingredient properties were investigated and it was discovered that the release rate could be fine-tuned to a desired release rate by controlling the granule porosity and binder-solids ratio. In 2005, Muro-Sune et al. developed predictive models utilizing a computer-aided modeling framework to analyze the release models of pesticide products. The research highlighted the benefits of incorporating controlled release models into a computer platform. This included the ability to generate and test various formulations of the product, prior to performing the final steps experimentally [6].

Research in computer-aided release modeling suggests a continued need for the development of predictive models. The development of computer programs which incorporate dissolution models allows for extended analysis of complex systems, since the solutions generated for release models can be modified for various applications. In the area of controlled release, computer models can optimize product formulation by identifying the effects of the physical parameters of the active ingredient and predicting the results of multiple coating layer alternatives. These advancements invoke the possibility of running virtual (*in silico*) experiments rather than physical ones. Virtual dissolution experiments can be performed quickly and have the unprecedented advantage in that the ‘experimental’ conditions can be completely controlled [7]. Advanced numerical methods are presenting new analytical short-time approximations for dissolution models, providing an increasingly more descriptive analysis than available from model formulations developed previously [37].

1.4 Limitations

Despite the significant progress in the development of dissolution models, discrepancies between theory and experimental data are still present [3]. Research has not distinguished whether the problems originate from experimental factors or limitations in the mathematical models. Models formulated through mathematical modeling are always based on simplifying assumptions, which in effect, influence the accuracy of the model [38]. For example, the release of a drug often involves two mechanisms, diffusion and dissolution. However, since the verification of a model involving both mechanisms would be complex, mathematical models are often simplified by modeling only dominating mechanism while ignoring the other less predominant one [39].

Polydispersity and encapsulation further complicate the models for dissolution. In the case of polydisperse solids, assumptions which ignore the size distributions of the particles lead to significant errors in dissolution calculations. Heterogeneous systems, such as those with multiple particle sizes, need special consideration in the dissolution modeling process [40]. The incorporation of encapsulated particles into dissolution models also demands specialized attention. In the preparation of the coated particles, it is necessary to balance the release rate with an appropriate coating thickness. Designing an effective encapsulated particle requires accounting for the active ingredient present, the mechanism of release, and the final fate of the combined ingredients [41].

1.5 Purpose of Study

As previously stated, dissolution modeling has applications in a wide variety of fields, making in an important area of research for advanced particulate technology. Historically, there have been several classical dissolution rate expressions which have been used to interpret particle dissolution rate phenomena, most commonly described by the Noyes-Whitney, Nernst-Brunner and Hixson-Crowell equations. While each of these models provides insight into the dissolution rate behavior, limitations based simplifying assumptions are still present.

In addition, polydisperse solids and encapsulated materials significantly complicate the existing models. For that reason, computer–aided modeling appears to be the only suitable platform to perform such vigorous computations found in these advanced models.

The scope of this thesis involves developing a general programming code to solve complex time-variant dissolution problems which can be modified and extended for various applications. The computer programming development of this project was structured as follows:

1. Develop a general code using the programming platform MATLAB to solve the dissolution model for monodisperse particles (Chapter 3).

2. Modify the code in Step 1 for monodisperse particles to account for polydisperse particles, where varying particle size distributions are present (Chapter 4).

3. Modify the original code in Step 1 for the dissolution monodisperse particles to account for encapsulated particles, in which two different materials are present, the active core ingredient and the coating material (Chapter 5).

4. Combine the models developed in Steps 2 and 3 to produce a unifying program accounting for the dissolution of polydisperse encapsulated particles (Chapter 6).

The aim of this work was to incorporate the most descriptive mathematical model for dissolution into the interactive programming environment of MATLAB. Once this was accomplished, a second software, COMSOL Multiphysics Script, was utilized to produce various graphs and 3D visualizations of the dissolution processes. The results of the computer models were then compared to the experimental results found for the dissolution of citric acid in water using a reaction microcalorimeter in Chapter 7. This chapter compares the theoretical results to those found experimentally, and discusses the discrepancies found. The final chapter, Chapter 8, examines the implications of computer-aided modeling for particle design and discusses the future work possible in this area. It was hoped that the simulations developed could be used in facilitating the design of specialized controlled release systems, whose release profile would be predictable and whose composition parameters could be manipulated to achieve the most optimal design.

Chapter 2 : Overview

2.1 General Dissolution

Mathematical models for the dissolution of solid particles involve accounting for the complicated changes in the surface area and/or shape which occur during dissolution [3]. Solid particles in liquids can be modeled using Nernst-Brunner type kinetics [28]:

$$\frac{\mathrm{dM}}{\mathrm{dt}} = -\frac{D}{h} \cdot A \cdot (C_s - C) \tag{2.1}$$

where M is the mass of solid material at time t, k is the dissolution rate constant, A is the area available for mass transfer, D is the diffusion coefficient of the dissolving material, h is the diffusion boundary layer thickness, C is the concentration and C_s is the concentration solubility.

In addition, the following considerations were used to model the dissolution of monodisperse particles [42] :

1. The surface area of the particles changes as the particle dissolves.
2. Dissolution of all the particles in the sample contributes to the overall concentration of the solute.
3. The diffusion boundary layer thickness has been shown to decrease for particles below a certain size, depending on the material dissolving and the dissolution conditions. Dissolution is modeled more accurately when the boundary layer thickness during particulate dissolution is approximated by the particle radius.

2.2 Governing Equations

A general mathematical model was derived from Nernst-Brunner type kinetics which could be used to predict the theoretical time required for dissolution of monodispersed particles. The surface area of a spherical particle is given by

$$A = 4\pi r^2 \tag{2.2}$$

where r is the radius of the particle. The volume of the spherical particle is

$$V = \frac{4}{3}\pi r^3 \tag{2.3}$$

The change in volume can be written as

$$dV = A \cdot dr = 4\pi r^2 dr \tag{2.4}$$

where dV is the change in volume and dr is change in radius. The mass of the particles is given by

$$M = N \cdot \rho \cdot V \tag{2.5}$$

where M is the total mass of the particles, N is the number of particles of radius r and ρ is the density. Substitution into Eq. (2.1) yields

$$\frac{dM}{dt} = \frac{N \cdot \rho \cdot dV}{dt} = \frac{N \cdot \rho \cdot 4 \cdot \pi \cdot r^2 \cdot dr}{dt} = -\frac{D}{r} \cdot N \cdot 4 \cdot \pi \cdot r^2 \cdot (C_s - C) \tag{2.6}$$

Cancelation of like terms gives

$$\frac{\rho \cdot dr}{dt} = -\frac{D}{r}(C_s - C) \tag{2.7}$$

The concentration C can be derived from a mass balance which finds the total mass dissolved at a given time

$$C = \frac{M_d}{V_m} = \frac{M_0 - M}{V_m} = \frac{N \cdot \rho \cdot \frac{4}{3} \cdot \pi \cdot r_0^{\,3} - N \cdot \rho \cdot \frac{4}{3} \cdot \pi \cdot r^3}{V_m} \tag{2.8}$$

where M_d is the mass dissolved at a given time, M_0 is the initial mass, M is the mass remaining, V_m is the dissolution medium volume, and r_0 is the initial particle radius. Replacing C in Eq. (2.7) by (2.8) gives

$$\frac{\rho \cdot dr}{dt} = -\frac{D}{r} \cdot (C_s - \frac{N \cdot \rho \cdot \frac{4}{3} \cdot \pi \cdot r_0^{\,3} - N \cdot \rho \cdot \frac{4}{3} \cdot \pi \cdot r^3}{V_m}) \tag{2.9}$$

Rearrangement, shown in full detail in Appendix A, yields

$$\frac{r\cdot dr}{c^3 - r^3} = \frac{D\cdot N\cdot 4\cdot \pi}{3\cdot V_m}\cdot dt \tag{2.10}$$

where

$$c = \left(r_0{}^3 - \frac{3\cdot C_s\cdot V_m}{N\cdot \rho\cdot 4\cdot \pi} \right)^{\frac{1}{3}} \tag{2.11}$$

Integration of differential Eq. (2.10), which describes the rate of change of the radius with respect to time leads to

$$\int \frac{r\cdot dr}{c^3 - r^3} = \frac{1}{6c}\ln\frac{c^3 - r^3}{(c-r)^3} - \frac{1}{\sqrt{3}c}\tan^{-1}\frac{2r+c}{\sqrt{3}c} - k = \frac{D\cdot N\cdot 4\cdot \pi}{3\cdot V_m}\cdot t \tag{2.12}$$

where

$$k = \frac{1}{6c}\ln\frac{c^3 - r_0{}^3}{(c-r_0)^3} - \frac{1}{\sqrt{3}c}\tan^{-1}\frac{2r_0+c}{\sqrt{3}c} \tag{2.13}$$

The derived Eq. (2.12) describes the relationship of time and particle radius and will be used in dissolution calculations. Rearranging Eq. (2.12) provides the relationship between time and radius as shown in Eq. (2.14).

$$t = \frac{\frac{1}{6c}\ln\frac{c^3 - r^3}{(c-r)^3} - \frac{1}{\sqrt{3}c}\tan^{-1}\frac{2r+c}{\sqrt{3}c} - k}{\frac{D\cdot N\cdot 4\cdot \pi}{3\cdot V_m}} \tag{2.14}$$

2.3 Modeling Assumptions

As with all models, simplifications and assumptions were made in the derivation of mathematical equations describing the dissolution of solid particles in a liquid. All particles are assumed to be spherical in shape, where all surface area of the sphere is exposed to the liquid. The particles are considered isotropic spheres where the geometric shape does not change. The assumption is that the solution is well stirred, however, no convection forces are accounted for in the model. In addition, both the solubility and diffusion coefficient are assumed to remain as constants throughout the process.

Chapter 3 : Monodisperse Particle Dissolution

3.1 MATLAB

MATLAB was selected as the main programming and computational platform for solving the multiple equations involved in particle dissolution. The interactive environment of MATLAB allowed for advanced algorithm development, management of variables, import and export of data, numeric computation, data analysis and visualization [43]. In addition, MATLAB has the ability to solve technical computing problems much faster than traditional programming languages such as C, C++ or Fortran [43]. The software environment allowed codes, files and data to be managed while performing advanced mathematical functions such as linear algebra, statistics and numerical integration. MATLAB also supports the vector and matrix operations needed to implement the programming code effectively.

3.2 Program Design

Computational modeling of dissolution was achieved by designing a program in MATLAB which solved for the particle radius as a function of time as developed in Section 2. The properties for a given substance were to be inputted into the program, which would then be used to calculate various transport phenomena parameters, that in turn would be used to solve the governing dissolution equation. The process is outlined in Figure 3.1 and was implemented in MATLAB using the following algorithm:

1. First the program is started using a call function which invokes the first m-file to begin calculations.

2. Based on temperature and properties of solute and solvent, the diffusion coefficient is determined.

3. The total number of particles is calculated using the given initial particle radius, total grams and density of material.

4. Using the calculated values from the previous programs, the radius at incremented times is found using the procedure outlined in Section 2.

5. The calculated radius at each given time is converted to concentration which is then changed into conversion.

6. Finally, the results are plotted as concentration and conversion versus time graphs. Additional visualization using 3D plots are used to show the overall change in particle size over time.

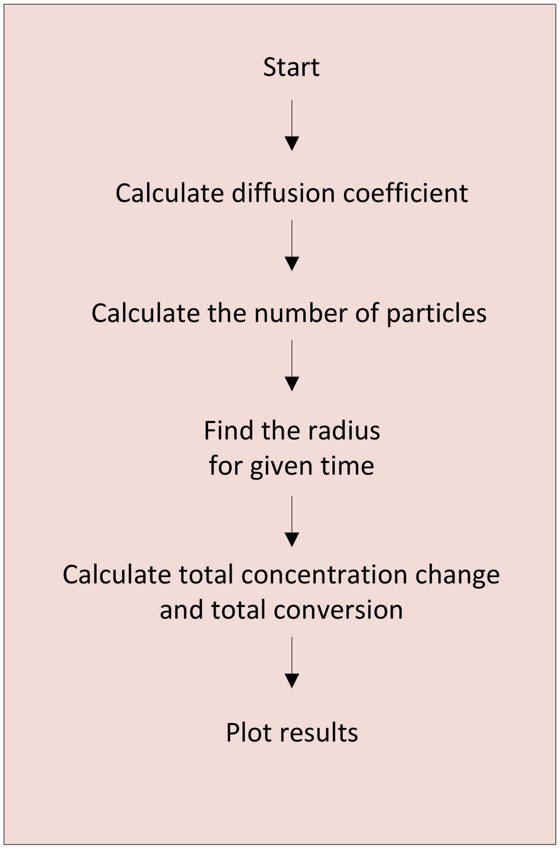

Start
Calculate diffusion coefficient
Calculate the number of particles
Find the radius
for given time
Calculate total concentration change
and total conversion
Plot results

Figure 3.1: : Program algorithm for monodisperse particles model.

The following sections will discuss the implementation of the particle dissolution problem in MATLAB, revealing the programming code necessary for the calculations. Then the program will be used to solve the specific problem of citric acid particles dissolving in water.

3.2.1 Starting the Program

Since this program involved running multiple subroutines, an initial call function was created under the name of *Start.m*. The purpose of this m-file was to organize and sequence the following sub-programs: The first program is *DiffPart.m*, which calculates the diffusion coefficient, then the program *NumPart.m* calculates the total number of particles. The values from these programs are implemented into *FindRadius.m*, which finds the radius at each given time. Next, *ConcConv.m* converts the radius to concentration and conversion. The final program *PrintResults.m* plots the results in graphs. Once the m-file *Start.m* is called, the rest of the programs are run automatically and initiated. The complete source code for the monodisperse particle dissolution model is given in Appendix B.

3.2.2 Determination of Diffusion Coefficient for a System

The diffusion coefficient of the solid particle in the liquid solute was estimated using the Wilke and Chang correlation [45] as shown in Eq. (3.1).

$$D_{12} = 7.4 \cdot 10^{-8} \left[\frac{(\phi M_2)^{\frac{1}{2}} T}{\mu_2 V_1^{0.6}} \right] \qquad (3.1)$$

M_2 is the molecular weight of the solvent (g/mol), T is the temperature (Kelvins), μ_2 is the viscosity of the solvent solution (centipoises), V_1 is the molar volume of the solute at normal boiling point (cm^3/g mol) and φ is the association parameter of the solvent.

3.2.3 Calculation of the Number of Particles based on Mass

In general, the numbers of particles in a given sample is rarely known. This is due to the small, sometimes microscopic, sizes of the particles being examined as well as the large sample amounts used in experiments which drastically increase the number of particles to an amount which is impractical to physically count. To obtain an estimated number of particles, calculations involving the volume of the particle and sample were used.

The parameters of r_0, the initial radius of the particle, M_T, the total mass of sample and ρ, the substances density must first be defined. The total volume, V_T, of the sample can then be calculated as

$$V_T = M_T \cdot \frac{1}{\rho} \qquad (3.2)$$

The volume of a single particle, V_P, is then calculated using the initial radius and the assumption that the particle is spherical in shape

$$V_p = \frac{4}{3} \pi r_0{}^3 \qquad (3.3)$$

The number of particles, N, in the sample is then found by dividing the total volume of the sample by the volume of a single particle as given in Eq. (3.4).

$$N=\frac{V_T}{V_P} \tag{3.4}$$

3.2.4 Radius Relationship to Time

The expression relating the radius of the particle at a given time was derived in Section 2 and given as

$$\frac{1}{6c}\ln\frac{c^3-r^3}{(c-r)^3}-\frac{1}{\sqrt{3}c}\tan^{-1}\frac{2r+c}{\sqrt{3}c}-k=\frac{D\cdot N\cdot 4\cdot \pi}{3\cdot V_m}\cdot t \tag{3.5}$$

In order to solve the given equation, the value for the constant c must first be calculated using the provided r_0, initial particle radius (cm), V_m, dissolution medium volume(mL), C_s, solubility of the solute (g/mL), and ρ, density of particles (g/cm^3). The number of particles calculated earlier in this section is then used in the equation

$$c=\left(r_0{}^3-\frac{3\cdot C_s\cdot V_m}{N\cdot \rho\cdot 4\cdot \pi}\right)^{\frac{1}{3}} \tag{3.6}$$

to solve for the constant c.

Using the constant c, the constant of integration, k, can then be determined using the formula

$$k=\frac{1}{6c}\ln\frac{c^3-r_0{}^3}{(c-r_0)^3}-\frac{1}{\sqrt{3}c}\tan^{-1}\frac{2r_0+c}{\sqrt{3}c} \tag{3.7}$$

Once the constant c and the integration constant k is found, Eq. (2.12) can be solved at various times. The constants derived are then used in the main equation F, as shown in

$$F=\frac{1}{6c}\ln\frac{c^3-r^3}{(c-r)^3}-\frac{1}{\sqrt{3}c}\tan^{-1}\frac{2r+c}{\sqrt{3}c}-k-\left[\frac{D\cdot N\cdot 4\cdot \pi}{3\cdot V_m}\cdot t\right] \tag{3.8}$$

MATLAB can then be implemented to solve at various times of t for when the function F equals zero, to calculate the radius r at the given time.

```
% CALCULATION OF CONSTANT, c
    %Solubility of the solute, Cs
    Cs = 1.33;          % g/mL
    %Dissolution Medium Volume, Vm
    Vm = 1;             % 1 mL H2O
    %Constant with respect to time, c
    c =nthroot((r0.^3.-(3.*Cs*Vm)./(N.*rho*4*pi)),3);

% CONSTANT OF INTEGRATION, k
    k = (1./(6*c)).*log((c.^3-r0.^3.)/(c-r0).^3.)-
(1./(sqrt(3)*c)).*(atan((2*r0+c)./(sqrt(3)*c)));

% FINAL FUNCTION RELATING TIME AND RADIUS
    F=(((1./(6*c)).*log((c.^3-r.^3.)./(c-r).^3.)-
(1./(sqrt(3)*c))*(atan((2*r+c)./(sqrt(3)*c))))-k)-(((D12*N*4*pi)*t)/(3*Vm));
```

MATLAB code 1: Calculation of constants and insertion into dissolution function.

```
% Initial time to start loop
t = 0;
% Loop set from 1-100
for i = 1:100
    % Assign value for time
    Time(i)= t;
    % Solve function for radius at each given time
    CalR = fzero(@Function,r0);

    % Check that calculated radius is not negative, and if it is, assign
    % the particle radius to equal zero.
    if CalR < 0
        r(i) = 0;
    else
        r(i) = CalR;
    end
% Update increment t
t = t + 100;
```

MATLAB code 2: Loop created to solve for radius at each specified time.

3.2.5 Relationship of Radius to Concentration and Conversion

The derived radii are then converted to concentration and conversion using *ConcConv.m* , making the data easier to analyze. The initial mass of the particles, M_0, in group g, can be calculated using the initial radius of the particle using the equation

$$M_0 = N \cdot \rho \cdot \frac{4}{3} \cdot \pi \cdot {r_0}^3 \tag{3.9}$$

Similarly, the mass of the particle remaining, M, at a given time t can be found using equation

$$M = N \cdot \rho \cdot \frac{4}{3} \cdot \pi \cdot r^3 \tag{3.10}$$

Therefore, the mass of the material dissolved, M_d, is can be found as the difference of the initial mass and the mass remaining. This can be written as

$$M_d = M_0 - M \tag{3.11}$$

To convert to concentration, the equation from Section 2 relating concentration to mass of material dissolved in the dissolution medium volume is used. Concentration at a given time is given as

$$C = \frac{M_d}{V_m} = \frac{M_0 - M}{V_m} = \frac{N \cdot \rho \cdot \frac{4}{3} \cdot \pi \cdot r_0^{\,3} - N \cdot \rho \cdot \frac{4}{3} \cdot \pi \cdot r^3}{V_m} \qquad (3.12)$$

The initial concentration is defined as

$$C_0 = \frac{M_0}{V_m} \qquad (3.13)$$

Finally the conversion X can be calculated by Eq. (3.14).

$$X = \frac{C}{C_0} \qquad (3.14)$$

3.2.6 Visualization of Results

The function *PrintResults.m* first converts the time in seconds to time in minutes. Then plots of concentration and conversion versus time are displayed.

3.3 An Example

The following section will used the designed program in the case study of citric acid particles dissolving in water at room temperature.

3.3.1 Simulation Setup

For this test study, the sample of citric acid is assumed to be monodispersed particles, indicating that the particles are uniform in shape and size. The parameters used in the simulation are as tabulated in Table 3.1.

Table 3.1: Parameters for citric acid used in test case.

Total grams of citric acid	M_T	= 0.1086 g
Density of citric acid	rho	= 1.665 g/cm^3
Molecular weight of solvent	M_2	= 18.015 g/mol
Temperature	T	= 298 K
Viscosity of solution (solvent)	μ_2	= 0.91 centipoises
Molar volume of solute	V_1	= 319.88 g/cm^3·mol
Association parameter of solvent	Φ	= 2.6
Solubility of the solute	C_s	= 1.33 g/mL
Dissolution medium volume (water)	V_m	= 1 mL

3.3.2 Some Results

The simulation was carried out for various initial radii sizes ranging 0.05-0.20 cm. The number of particles for each size, diffusion coefficient, constants c and k are listed in Table 3.2. The conversion of citric acid for varying radius sizes was plotted in Figure 3.2.

Table 3.2: Results from simulation for various radii sizes for 0.10 grams citric acid.

r_0 (cm)	N	D_{12} (cm^2/s)	C	k
0.04	243	5.2086 x 10^{-6}	-0.0896	2.2992
0.05	125		-0.1120	1.8393
0.06	72		-0.1344	1.5328
0.07	45		-0.1568	1.3138
0.08	30		-0.1792	1.1496
0.10	16		-0.2240	0.9197

As expected, the results show that for larger particle radii, the time for complete conversion (X=1) is much longer. The slope is very steep for the particles of small radii, indicating that for particle sizes 0.01-0.07 cm, total dissolution occurs in under 10 minutes. Using the average radius size of 0.059 cm, the conversion versus time was found for the given amount of 0.1086g CA, as shown in Figure 3.3. The second simulation was done for the initial radius size of r_0 = 0.06 cm for temperatures of 298K, 308 K and 318K. The program produced results seen in Figure 3.4.

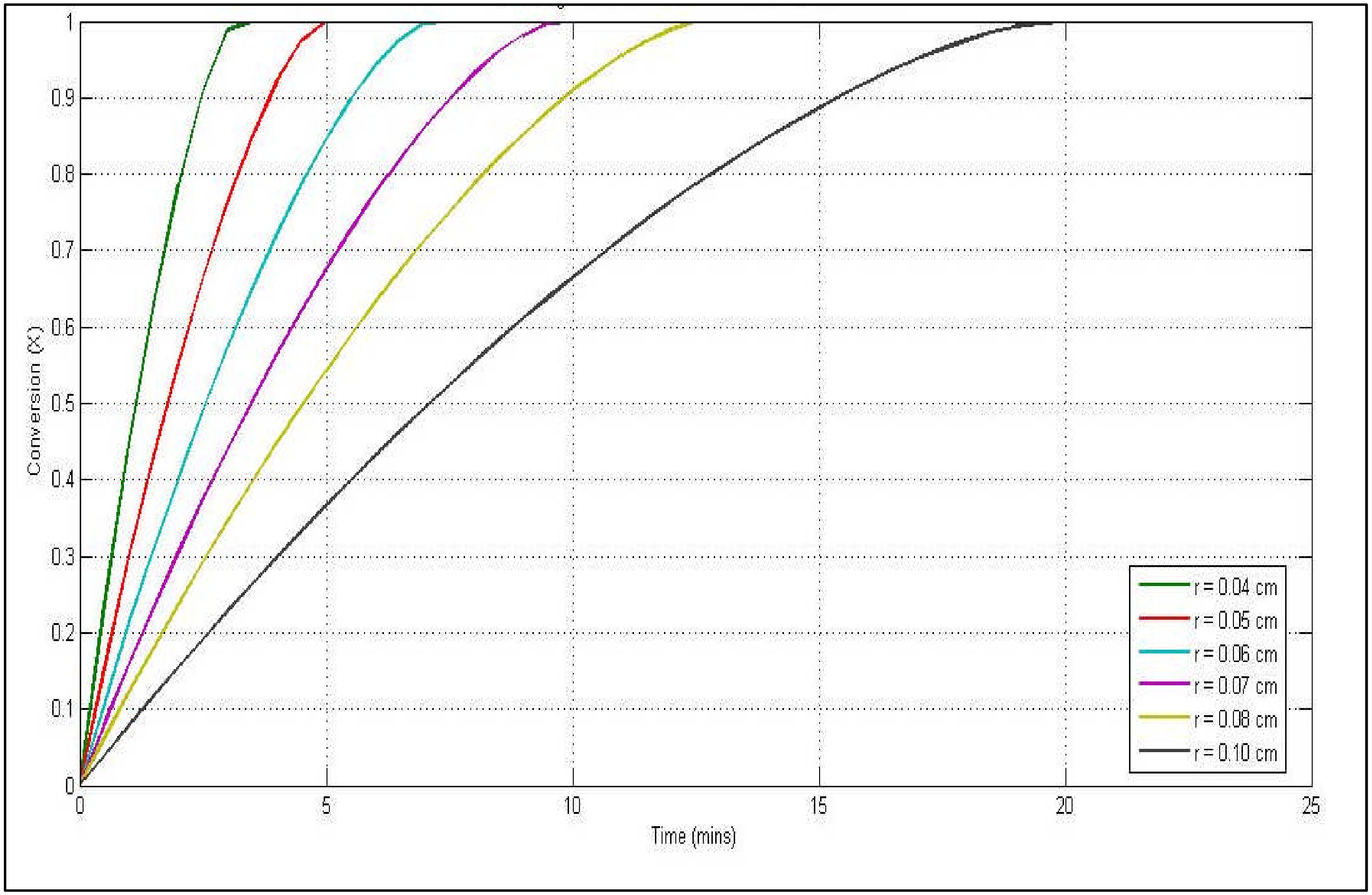


Figure 3.2: Monodisperse model for 0.10 grams citric acid for various radius sizes.

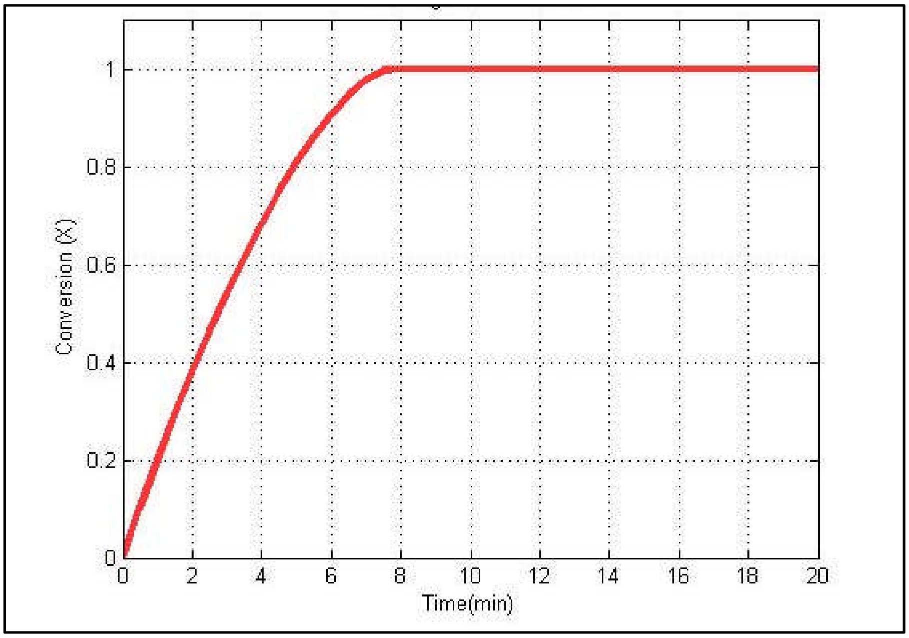


Figure 3.3: Monodisperse model for 0.10 grams citric acid with initial radius of 0.059 cm.

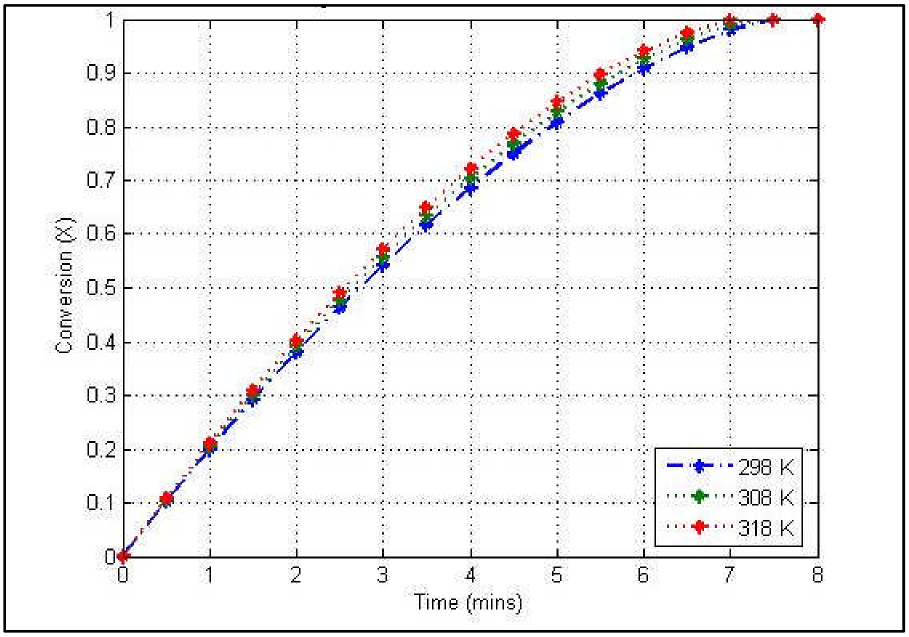


Figure 3.4: Monodisperse model for 0.10 grams of citric acid at different temperatures.

The graph shows that increased temperatures increase the rate of dissolution of the particles, which agrees with thermodynamic predictions. Increasing the temperature increases the calculated diffusion coefficient. For 298 K the diffusion coefficient is 5.2086 cm^2/s, whereas at a high temperature of 318 K the diffusion coefficient is found to be 5.5581 cm^2/s.

The simulation was then run for the average particle size of 0.059 cm for varying initial amounts. A plot, Figure 3.4, was created showing the conversion versus time for amounts of 0.02-0.50g of citric acid in 1 mL H2O at 298K.

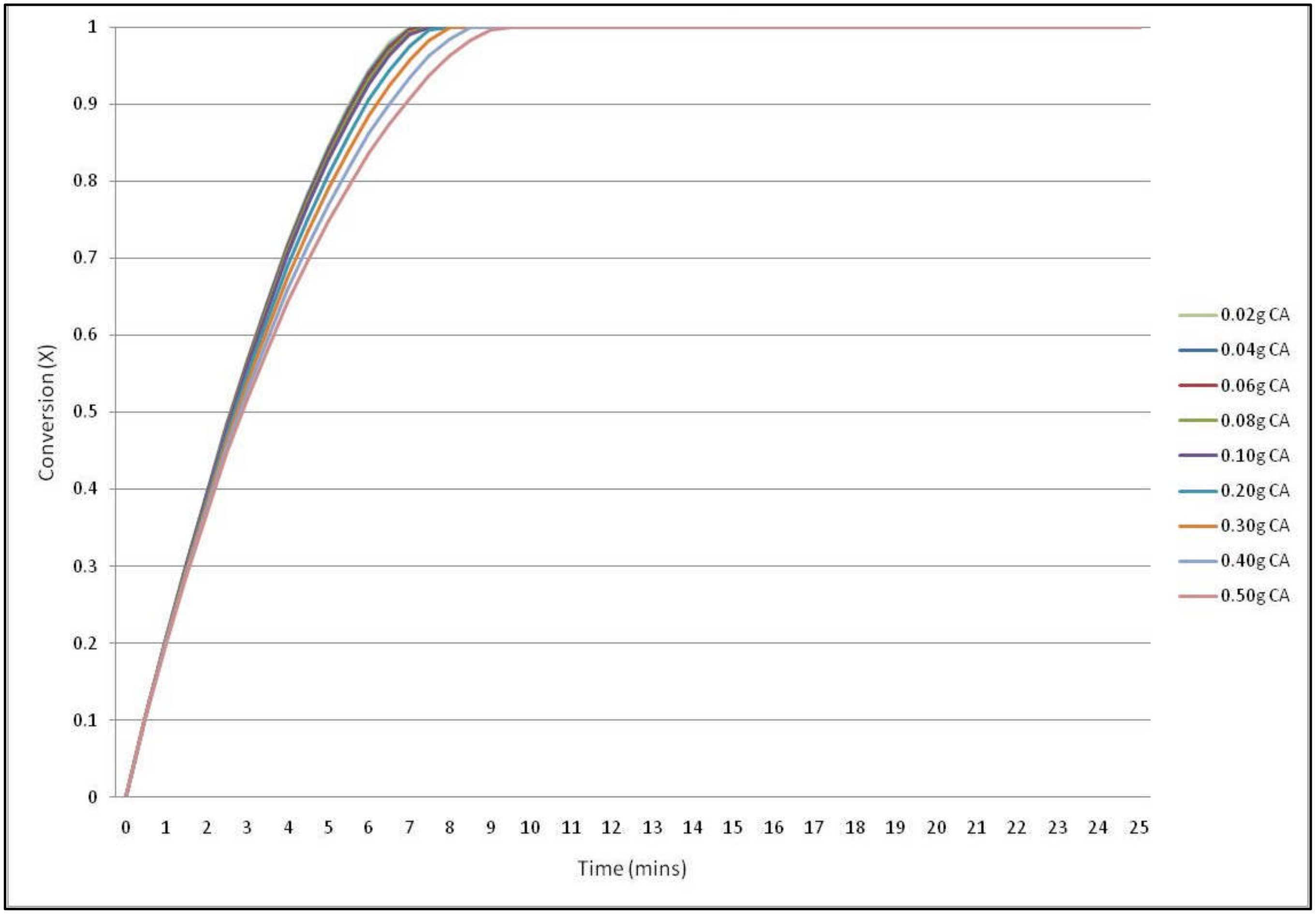


Figure 3.5: Monodisperse model for citric acid with different initial amounts.

3.4 COMSOL Script

COMSOL Script was chosen as an additional scripting language for its advanced technical computing and visualization capabilities [46]. The powerful modeling capabilities allow for more advanced visualization of the calculated data found from the MATLAB program. A graphical function was written in COMSOL Script which converted the radius data found in the MATLAB into a 3D plot in a new figure display window. The first part of the program parametrically converts the sphere of radius, *r*, centered at the origin. This is accomplished using the following equations for spherical coordinates.

$$x = \rho \cos \theta \sin \varphi \tag{3.15}$$

$$y = \rho \sin \theta \sin \varphi \tag{3.16}$$

$$z = \rho \cos \varphi \tag{3.17}$$

For these equations, θ is the azimuthal coordinate from 0 to 2π, φ is the polar coordinate from 0 to π and ρ is the radius. The parametric conversion and meshgrid are implemented in COMSOL Script using the code seen in COMSOL code 1.

```
phi=0:pi/20:pi;
theta=0:pi/10:2*pi;
[Phi,Theta]=meshgrid(phi,theta);

%Next we use the parametrization above.
X=r*sin(Phi).*cos(Theta);
Y=r*sin(Phi).*sin(Theta);
Z=r*cos(Phi);
```

COMSOL code 1: Parametric conversion of sphere.

The radius, r, for a give time is found from the MATLAB data in the variable vector Pr. Importing the initial radius, r0, time,t, and particle radius for at all given times, Pr, into COMSOL Script enables the creation of 3D animations showing the change in a spherical particle as a function of time. A sample source code for developing the animation in COMSOL Script is given in Appendix C.

3.5 Visualization

The data from MATLAB for the dissolution of particles with the average initial radius of *r0* = 0.059 cm was imported into COMSOL Script to create 3D visualizations of the process at various times. The 3D plots generated are shown with the corresponding conversion versus time graphs in Figure 3.6 for 0.10 grams of citric acid from 0 to 8 minutes.

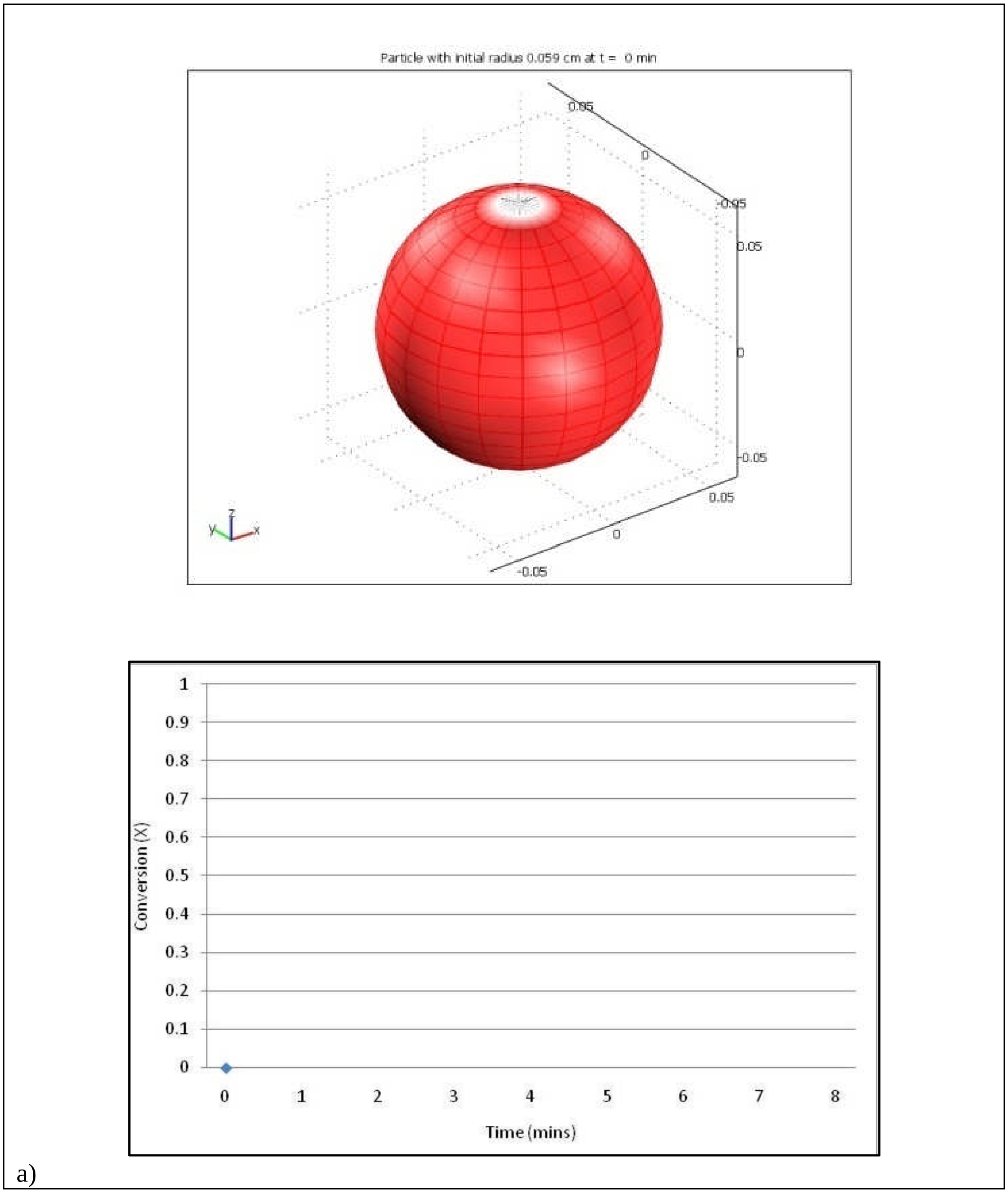


Figure 3.6: Visualization with corresponding conversion at a) t = 0 min, b) t = 2 mins, c) t = 4 mins, d) t = 6 mins and e) t = 8 mins.

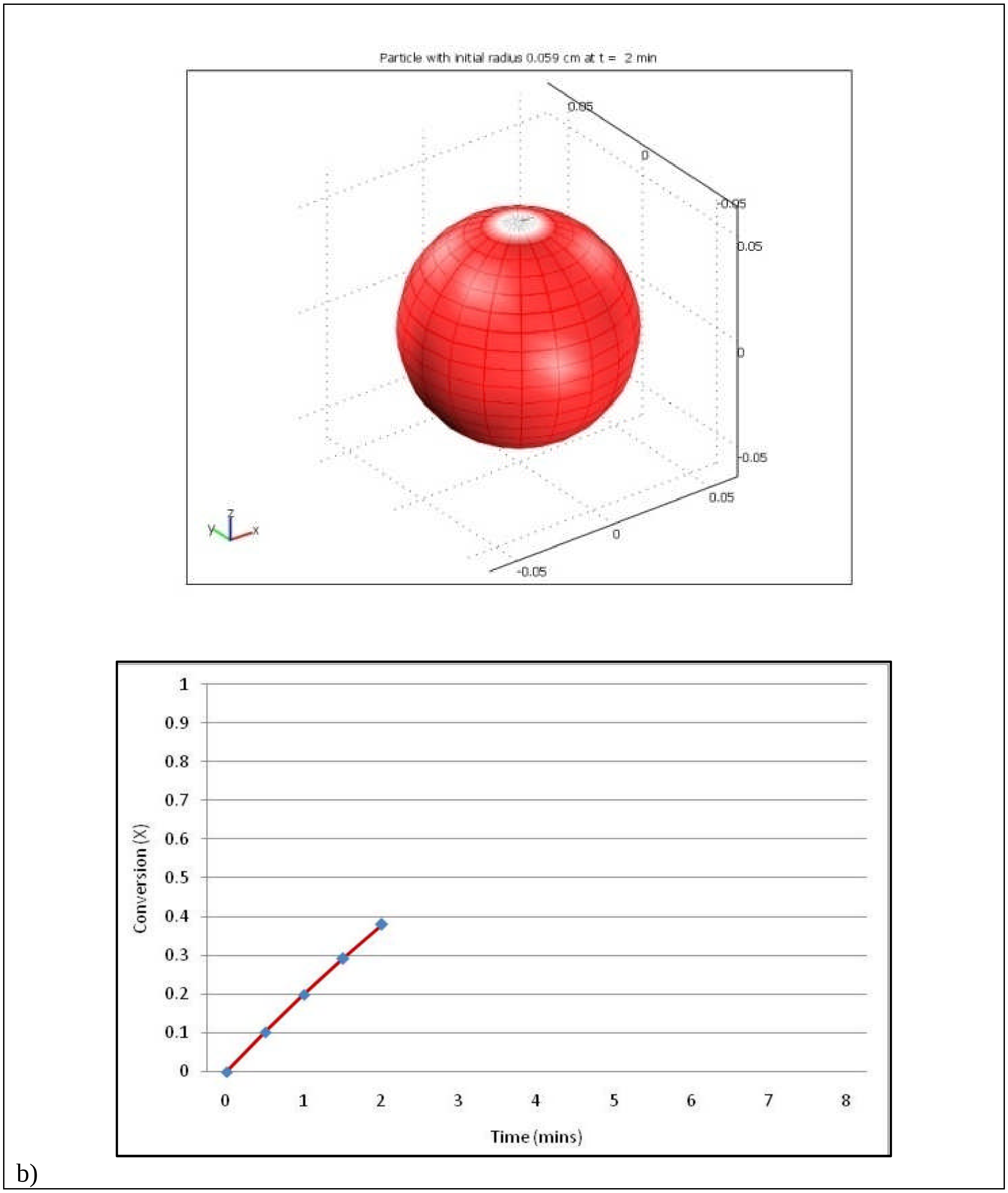
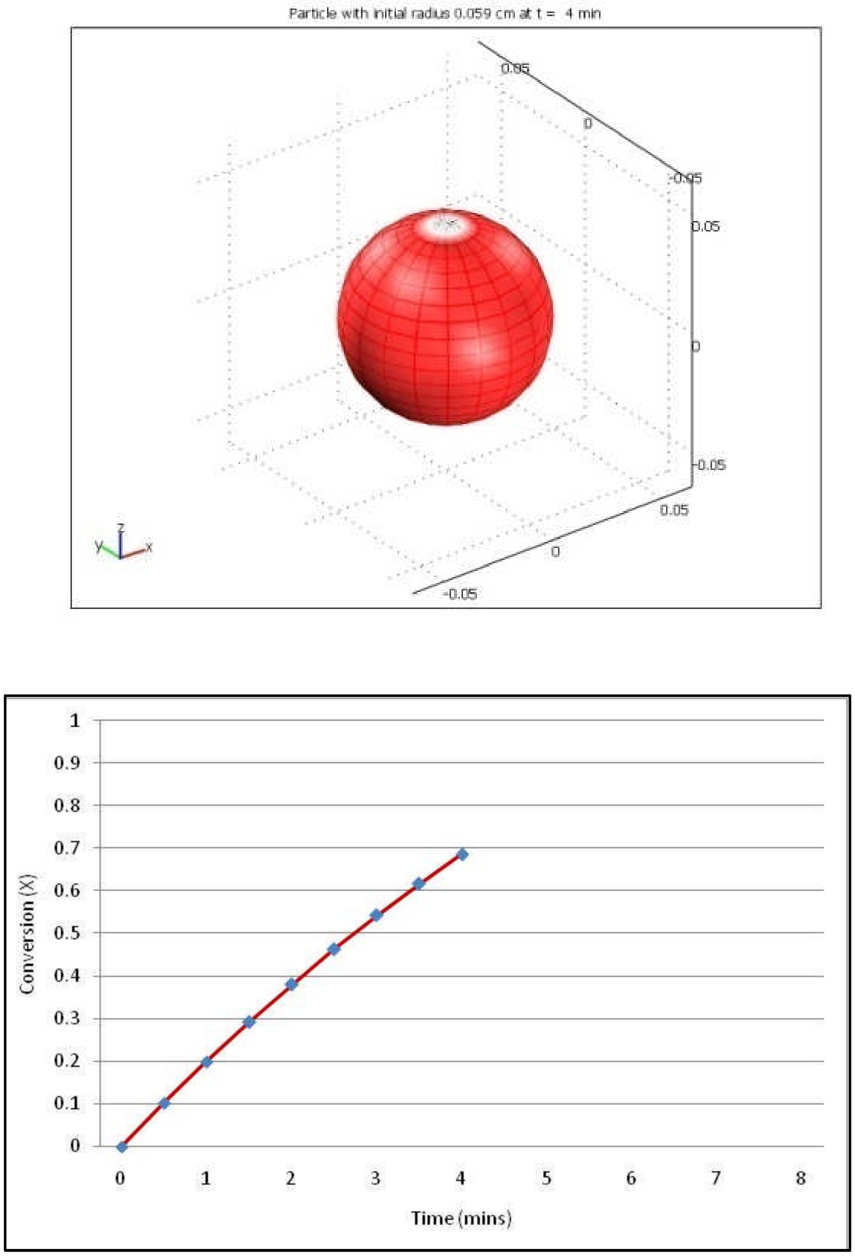


Figure 3.6 (Continued).



c)

Figure 3.6 (Continued).

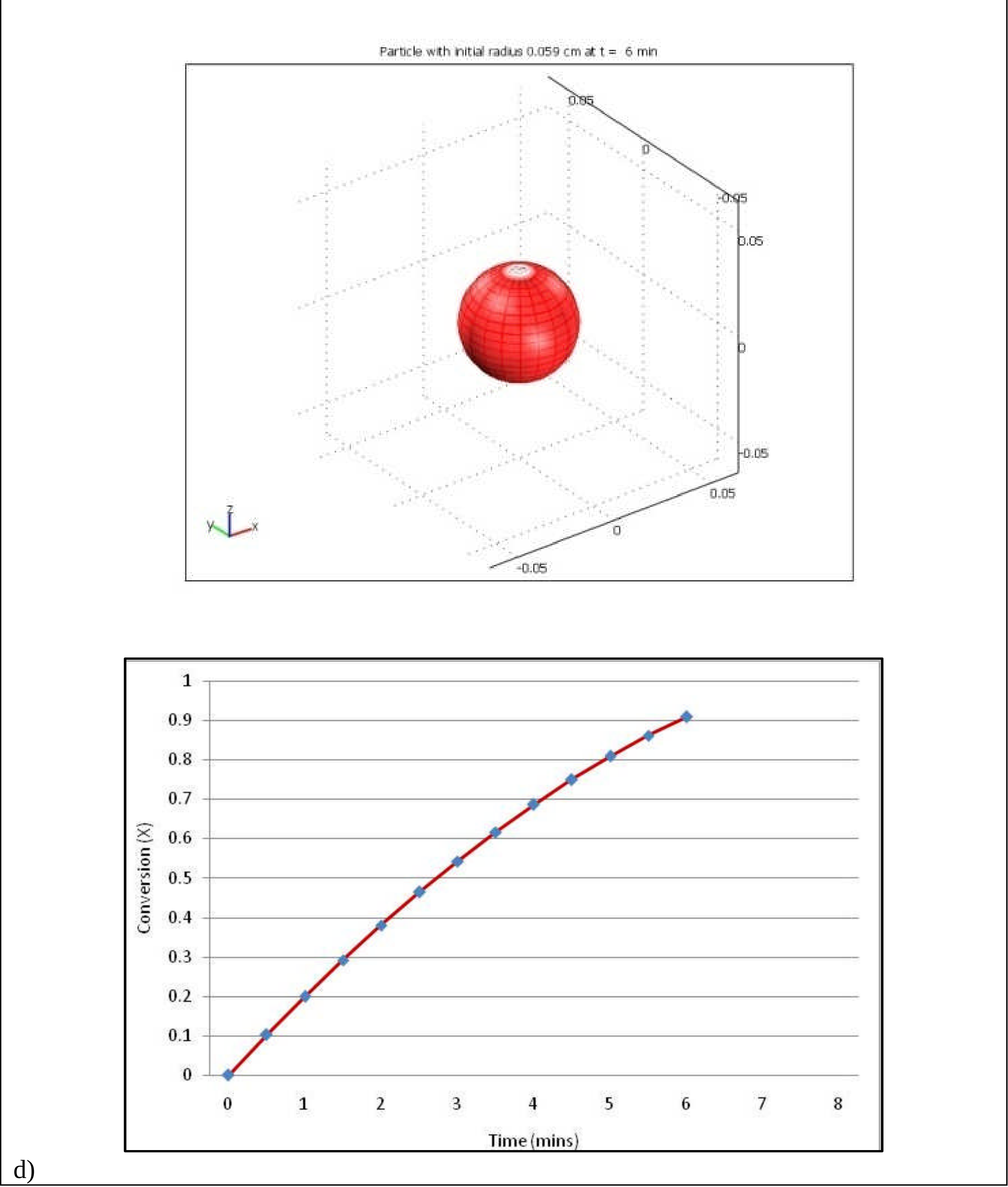
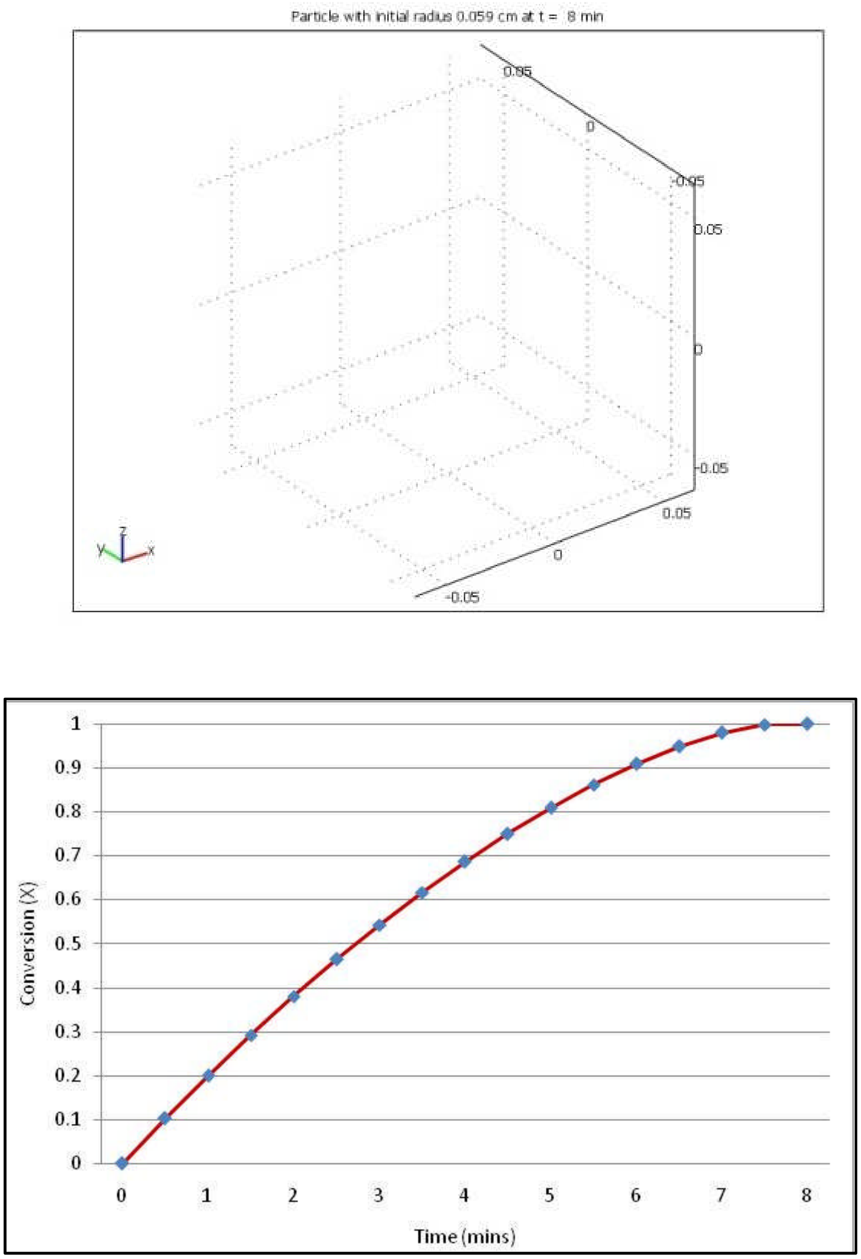
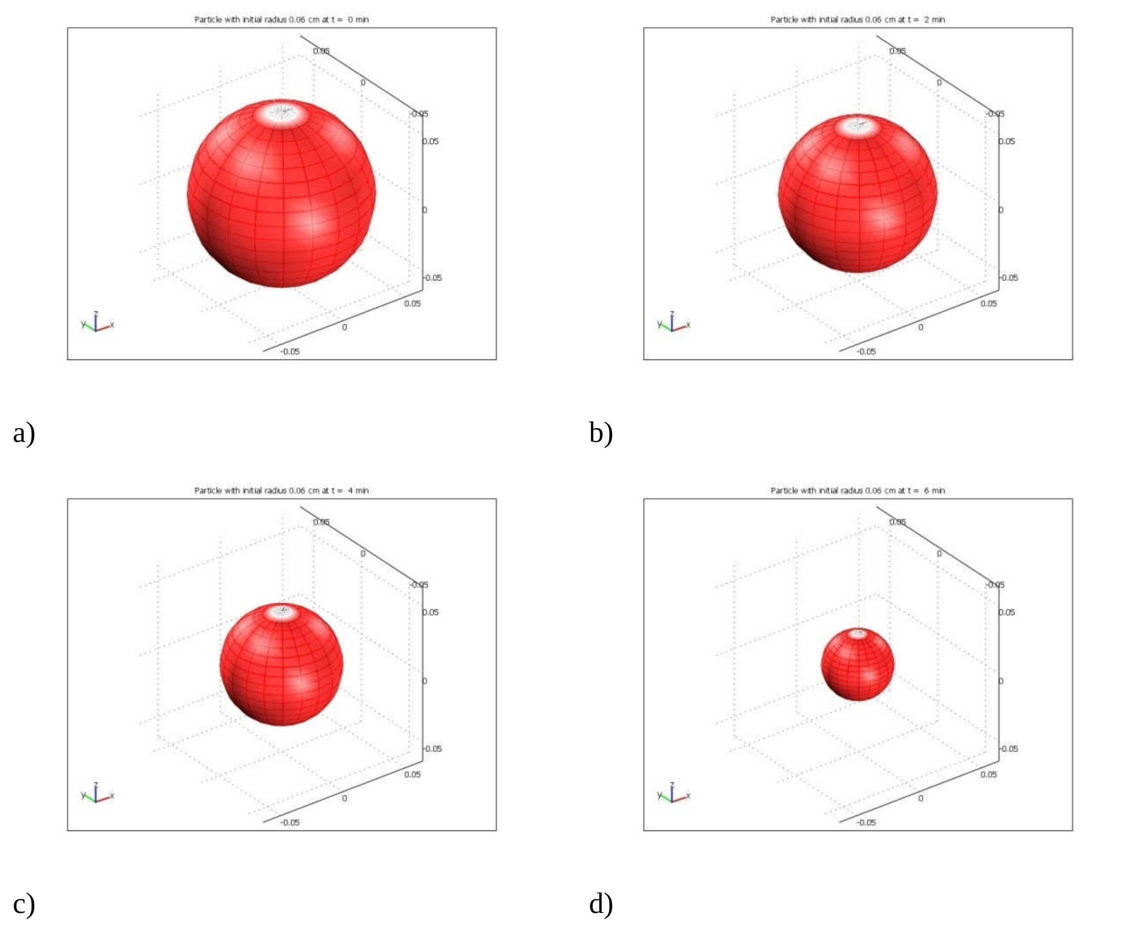


Figure 3.6 (Continued).



e)

Figure 3.6 (Continued).


Particle with initial radius 0.06 cm at t = 0 min
Particle with initial radius 0.06 cm at t = 2 min
Particle with initial radius 0.06 cm at t = 4 min
Particle with initial radius 0.06 cm at t = 6 min
0.05
0
-0.05
z
y
x

a)

b)

c)

d)

Figure 3.7: Visualization of 3D particle at a) t = 0 min, b) t = 2 mins, c) t = 4 mins and d) t = 6 mins.

The particle is completely dissolved at about t = 7.5 mins. COMSOL Script enables the user to visually monitor the decreasing radius size of the dissolving particle over a period of time. All particles are assumed to be monodisperse, and so the change in a single particle is assumed to represent the change of all the particles in the system.

3.6 Summary of Results

Several assumptions were made in developing the MATLAB code in this section. First, the particles were assumed to be monodispersed. The initial particle radius for the monodisperse particles was estimated using the average initial radii of all the particles. In addition, the particles were assumed to be spherical in shape, as this greatly simplified the mathematical computations.

Second, the diffusion layer thickness of the particle is assumed to remain constant throughout the particle dissolution. Third, sink conditions were considered when analyzing the experimental data in order to simplify the mathematical analysis of a changing bulk solution concentration. Lastly, parameters used in the calculations, such as the diffusion coefficient of citric acid in water were assumed to be constant throughout the process. Based on the Brunner's proposed model, the constant *k*, in the equation lumps the effects of exposed surface area, rate of stirring, temperature, structure of the surface and arrangement of the apparatus on the rate of dissolution [34].

The results of the computer model provide a reasonable estimate for the dissolution rate found experimentally. Calculations based on the theoretical model assume ideal conditions, such as even exposed surface area of the particles to the solvent. These conditions are most likely not found in reality, for instance, particles are often in contact with each other, which reduces the exposed surface area of the particle making the process non-ideal. The assumption of monodisperse particles also limits the accuracy of the model, since the influence of particle size distributions are ignored. This aspect will be further explored in Chapter 4.

Chapter 4 : Polydisperse Particle Dissolution

In the chemical processing industry, the dissolution of most solid particles in liquids involve a wide distribution of particle sizes [2]. It has also been showed that models formulated in terms of the average particle size can led to substantial errors in calculating dissolution behavior [40]. The program design from Chapter 3 modeled a collection of monodisperse spheres having an average initial particle size. This chapter will modify the existing program to account for the polydispersity found in most solid particle populations. The results of this model will then be compared to those previously generated.

4.1 Programming Modifications

The method used for calculating polydisperse particles followed the same program flow as developed previously with modifications made to account for particle size distributions. The same call function *Final.m* was used to initiate the same sub-routines developed before. Modifications were made to categorize the particles into several groups based on their average particle sizes. The program was updated to evaluate each of these particle size categories separately, allowing for dissolution data to be found for each of the individual groups. The data was then complied so that the overall dissolution behavior resulting from all size distributions could be accounted for.

The program has an algorithm similar to the monodisperse model developed in Chapter 3. After the program is started, the diffusion coefficient is calculated. Next the number of particles in a particular size group is calculated. Then the radius is calculated for a given time and if additional size groups exist, the program is looped to recalculate the number of particles in that given size group and the radius is found for the same time. After the radii for all sizes are calculated, the concentration and conversion for each group is found. For a given time, the total concentration and total conversion is found by summing the contributions from each of the size groups. The final plots include the individual size group concentration changes as well as the total concentration and total conversion plots. The program structure is diagramed in Figure 4.1.

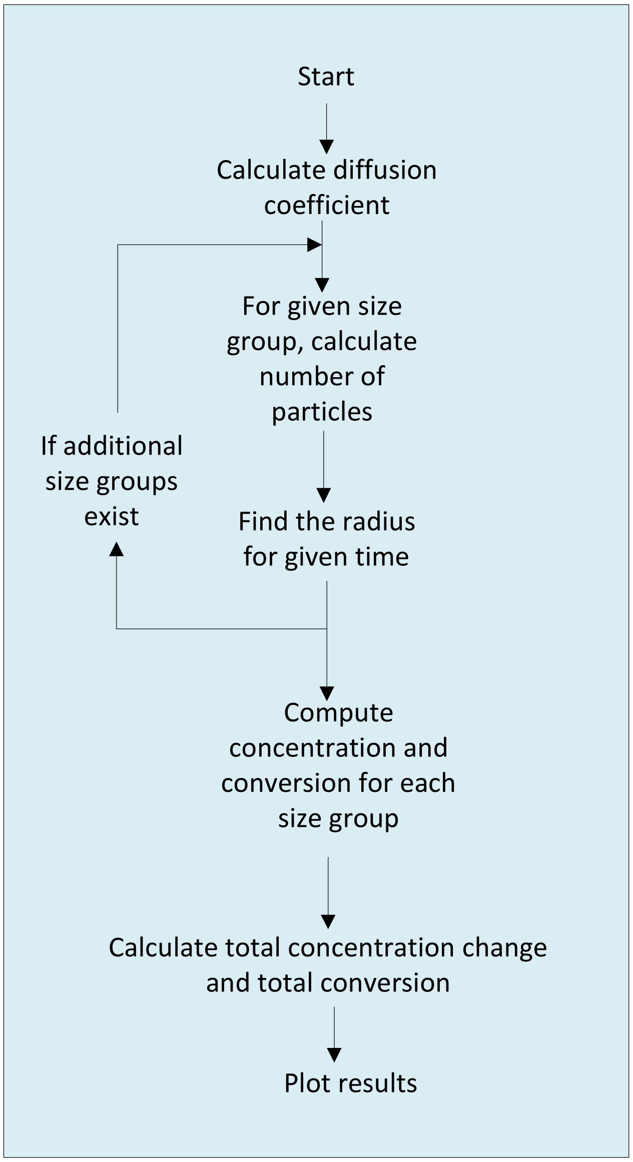

Start
Calculate diffusion coefficient
For given size group, calculate number of particles
If additional size groups exist
Find the radius for given time
Compute concentration and conversion for each size group
Calculate total concentration change and total conversion
Plot results

Figure 4.1: Program algorithm for polydisperse particles model.

4.1.1 Calculation of Number of Particles with Varying Size Distribution

The program file *PartDist.m* was modified to account for the size distribution changes. This was achieved by separating the particles into groups based on the particle radius size. Size ranges for the groups were designated, along with a weight of particles in that size range, indicating the percent of the total particles that were in the designated group. Groups are assigned integer values, beginning at 1, and are designated with the variable g. For each group, the percent size and average radius are assigned as follows: $P_s(g)$ is the percent of particles with radius size range in group g and $r_0(g)$ is the average radius size for particles in group g.

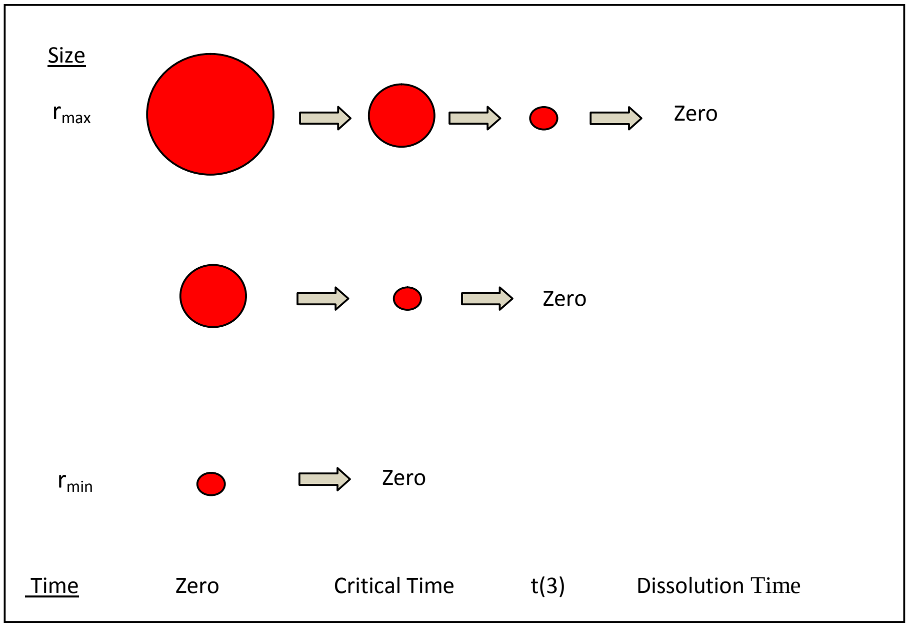


Figure 4.2: Schematic illustration of polydisperse particle dissolution.

Next, it is necessary to calculate the volume percent of each particle size distribution in order to find the total number of particles in each group. The percentages of particles with a given average radius are converted to volume percents which are then used to find the percent mass in each size distribution. First, the volume of a single particle in each group is calculated using the average initial radius, $r_0(g)$, by

$$V(g)=\frac{4}{3}\cdot\pi\cdot r_0(g)^3 \tag{4.1}$$

where V(g) is the volume of a particle in group g. The volume percent of each size distribution, $V_p(g)$, is given by Eq. (4.2).

$$V_p(g)=V(g)\cdot P_s(g) \tag{4.2}$$

The total volume,V_T, is calculated by summing the volume percents in each size distribution as expressed as Eq. (4.3).

$$V_T=\sum_{i=1}^{g}V_p(i) \tag{4.3}$$

The percent by volume, $P_V(g)$, of each size group is then given as Eq. (4.4).

$$P_V(g) = \frac{V_p(g)}{V_T} \tag{4.4}$$

Using this percent by volume, the weight of each particle size distribution, W(g), can be found from Eq. (4.5).

$$W(g) = P_V(g) \cdot M_T \tag{4.5}$$

Converting weight of each size distribution to volume, $V_s(g)$ is achieved by using Eq. (4.6).

$$V_s(g) = W(g) \cdot \frac{1}{\rho} \tag{4.6}$$

Dividing the total volume in each size distribution by the volume of a single particle in the given group gives the total number of particles, N(g). The number of particles in a given size distribution group is given by Eq. (4.7).

$$N(g) = \frac{V_s(g)}{V(g)} \tag{4.7}$$

4.1.2 Calculation of Constants for All Size Distributions

After the numbers of particles in each size distribution have been calculated, the constant c is recalculated for each group using the formula shown in Eq. (4.8).

$$c(g) = \left(r_0(g)^3 - \frac{3 \cdot C_s \cdot V_m}{N(g) \cdot \rho \cdot 4 \cdot \pi} \right)^{\frac{1}{3}} \tag{4.8}$$

The integration constant k is also calculated for each group given by Eq. (4.9).

$$k(g) = \frac{1}{6c(g)} \ln \frac{c(g)^3 - r_0(g)^3}{(c(g) - r_0(g))^3} - \frac{1}{\sqrt{3}c(g)} \tan^{-1} \frac{2r_0(g) + c(g)}{\sqrt{3}c(g)} \tag{4.9}$$

The calculations of the constants for each of the size groups are implemented in the program as shown in MATLAB code 3. The initial mass, $M_0(g)$, is calculated for each size distribution using Eq. (4.10).

$$M_0(g) = \frac{4}{3} \cdot \pi \cdot r_0(g)^3 \tag{4.10}$$

The initial concentration, $C_0(g)$, is calculated for each size distribution is given by Eq. (4.11).

$$C_0(g) = \frac{M_0(g)}{V_m} \tag{4.11}$$

The total initial concentration, C_{0T}, is the sum of all size distribution as shown in Eq. (4.12).

$$C_{OT} = \sum_{i}^{g} C_0(g) \tag{4.12}$$

```
% CALCULATION OF CONSTANT,c
   % Solubility of the solute, Cs
    Cs = 1.33;          % g/mL
    % Constant with respect to time, c
    % Creates loop the length of the size distribution
    for g=1:length(Ps)
        % Checks if the percent of the size distribution is greater than
        % zero, and if so calculates the constant c
        if Ps(g) > 0;
            c(g) = nthroot((r0(g)^3.-(3.*Cs*Vm)./(N(g)*rho*4*pi)),3);
        % If the percent in a size distribution is zero, c is assigned the
        % value of zero
        else
            c(g) = 0
        end
    end
% CONSTANT OF INTEGRATION, k
% Creates loop the length of the size distribution
 for g=1:length(Ps)
        % Checks that the percent of the size distribution is greater than
        % zero, and calculates the constant of integration
        if Ps(g) > 0;
            k(g) = (1./(6*c(g))).*log((c(g)^3-r0(g).^3.)/(c(g)-r0(g)).^3.)-
(1./(sqrt(3)*c(g))).*(atan((2*r0(g)+c(g))./(sqrt(3)*c(g))));
        else
        % Otherwise assigns the constant of integration to be equal to zero
            k(g) = 0;
        end
 end
% INITIAL MASS OF PARTICLES IN EACH SIZE DISTRIBUTION, Mo
    Mo = (r0.^3)*(4/3)*rho.*N*pi;

% INITIAL CONCENTRATION OF PARTICLES IN EACH SIZE DISTRIBUTION, Co
    Co = Mo./Vm;
```

MATLAB code 3: Calculating constants for various size groups.

4.1.3 Determination of Radii for each Group at Specified Times

A loop in MATLAB was generated to include all size distributions groups which solved for the radius as a function of time. The general equation for F(g), which again was solved for the radius, r, when it was equated to zero at a given time, t, is given by Eq. (4.13).

$$F(g)=\frac{1}{6c(g)}ln\frac{c(g)^3-r(g)^3}{(c(g)-r(g))^3}-\frac{1}{\sqrt{3}c(g)}tan^{-1}\frac{2r(g)+c(g)}{\sqrt{3}c(g)} \\ -k(g)-\left[\frac{D\cdot N(g)\cdot 4\cdot \pi}{3\cdot V_m}\cdot t\right] \tag{4.13}$$

In the MATLAB code, it was necessary to write the function groups using the variable j, since g had previously been defined as a local variable and could not be used as a global variable.

```
% TIME AS FUNCTION OF PARTICLE RADIUS
    F=(((1./(6*c(j))).*log((c(j).^3-r.^3.)./(c(j)-r).^3.)-
(1./(sqrt(3)*c(j)))*(atan((2*r+c(j))./(sqrt(3)*c(j)))))-k(j))-
(((D12*N(j)*4*pi)*t)/(3*Vm));
```

MATLAB code 4: Extended function for various size groups.

Due to the increased data size, a matrix was created including the following

$$P_r(t,g)=F(g) \tag{4.14}$$

where $P_r(t,g)$ is the particle radius of the size group g at time t. The particle radius for each group is then used to find the mass and concentration at a given time.

```
% Creates loop to include all size distributions
for j = 1:length(r0) % Initial time set to zero
    t = 0;
    % Checks that the percent in size distribution is greater than zero
    if N(j) > 0
        for i = 1:100
        % Assigns value for time
        Time(i) = t;
        % Calculates the radius at each time given
        CalR(j,i) = fzero(@Function2,r0(j));
        % Checks that the value of radius
        check=isnan(CalR(j,i));
        % If the radius value can not be found assigns zero for the mass
        % and concentration
        if CalR(j,i)<0 | check == 1
            Pr(j,i) = 0;
            M(j,i) = 0;
            C(j,i) = 0;
        else
        % If the radius is found it assigned as the particle radius and the
        % mass and concentration are calculated
            Pr(j,i) = CalR(j,i);
            M(j,i) = (Pr(j,i)^3)*(4/3)*rho*N(j)*pi;
            C(j,i) = M(j,i)/Vm;
        end
        % Updates the value of t
           t = t + 10;
        end
```

MATLAB code 5: Loop created to find radius for all size groups at given time.

```
% When the percent in size distribution is less than zero
    else
        % Creates loop which assigns zero for the values of time, particle
        % radius, mass and concentration
        for i = 1:100
            Time(i) = t;
            Pr(j,i) = 0;
            M(j,i) = 0;
            C(j,i) = 0;
            t = t + 10;
        end
    end
end
```

MATLAB code 5 (Continued).

4.1.4 Relating Radius to Concentration and Conversion for Each Group

The mass of size group g at time t is given as M(t,g) as

$$M(t,g) = \frac{4}{3} \cdot \pi \cdot (P_r(t,g))^3 \cdot \rho \cdot N(g) \tag{4.15}$$

The concentration, C(t,g), for a size group *g* at a given time is found by

$$C(t,g) = \frac{M(t,g)}{V_m} \tag{4.16}$$

while the total concentration of the solution, $C_T(t)$, is calculates by summing all size distributions at each given time in Eq. (4.17).

$$C_T(t) = \sum_{i}^{g} C(t,g) \tag{4.17}$$

The concentration of dissolved material, $C_d(t)$, at a given time is found by

$$C_d(t) = C_0 - C_T(t) \tag{4.18}$$

and the total conversion is then found by Eq. (4.19).

$$X(t) = \frac{C_d(t)}{C_0} \tag{4.19}$$

4.1.5 Visualizing the Results of the Polydisperse Model

The function *PrintResults*.m first converts the time in seconds to time in minutes. Then plots of concentration and conversion versus time are displayed. The first graph displays the concentration versus time results for each size distribution.

4.2 Examples using the Polydisperse Model

The following section will use the designed program in several case studies of citric acid particles dissolving in water at room temperature.

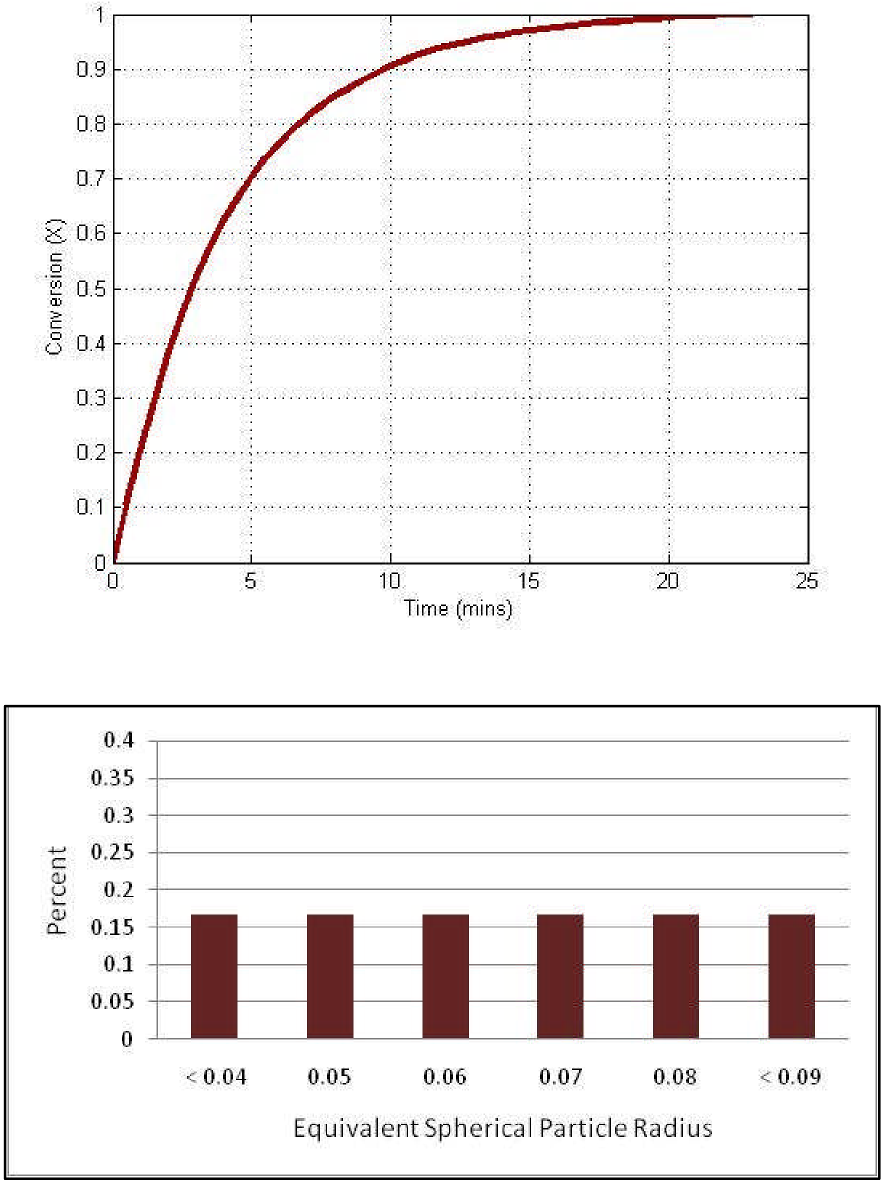
4.2.1 Simulation for Test Cases

First, the MATLAB code was tested to see the results of changing the particle size distribution. The parameters used for citric acid (Table 3.1) were used in the test runs. All cases used an initial amount of citric acid of 0.25g. Five test cases were run, each with different size distribution shapes, as shown in Table 4.1.

Table 4.1 : Size distributions percents for five test cases.

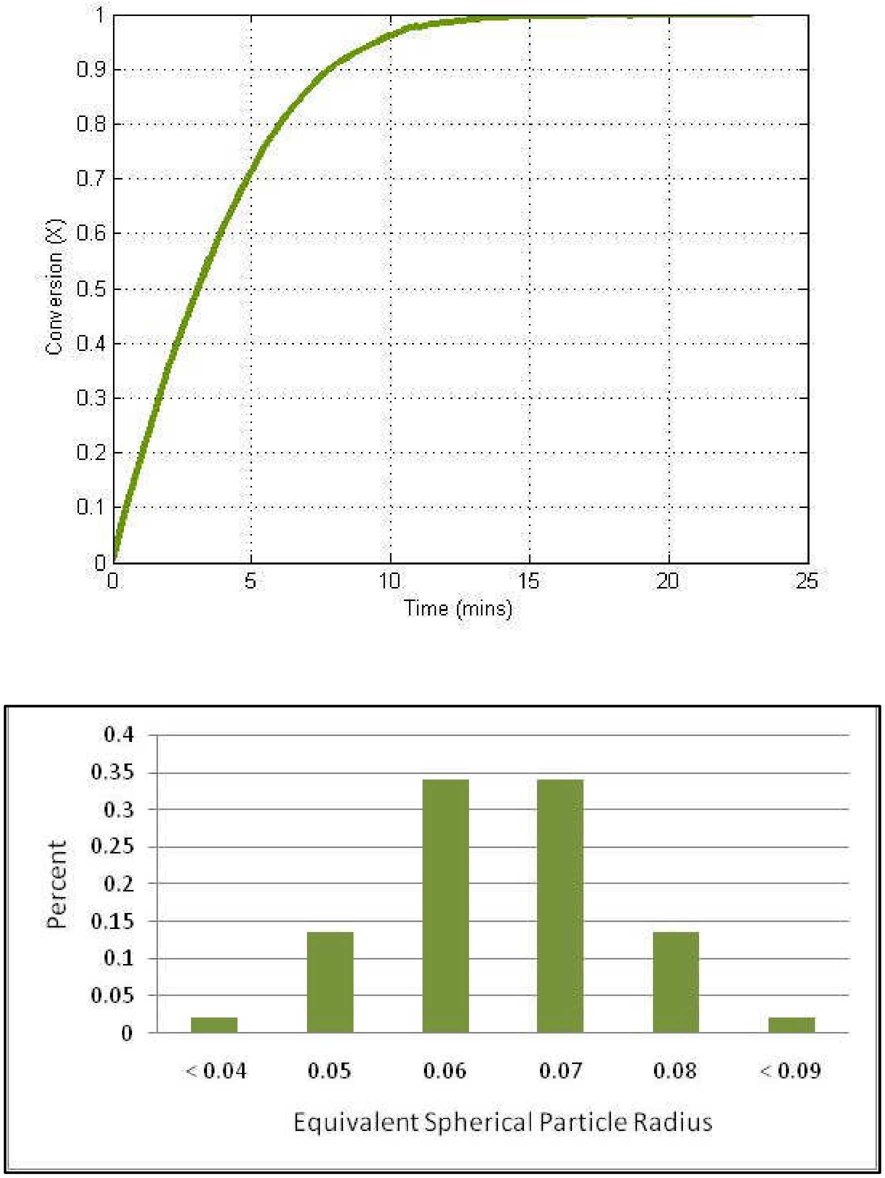
Radius Size (cm)	Test 1	Test 2	Test 3	Test 4	Test 5
> 0.04	16.67%	2.15%	34%	2.15%	34%
0.05	16.67%	13.6%	34%	2.15%	13.6%
0.06	16.67%	34%	13.6%	13.6%	2.15%
0.07	16.67%	34%	13.6%	13.6%	2.15%
0.08	16.67%	13.6%	2.15%	34%	13.6%
< 0.09	16.67%	2.15%	2.15%	34%	34%

Test 1 had an equal percents of all radius sizes, test 2 had a standard normal distribution, test 3 had a high percent of radii in the low range, test 4 had a high percent of radii in the high range and test 5 had a U-shaped distribution. The percents were graphed to illustrate the size distribution and the results of the conversion versus time graphs from MATLAB were plotted.



a)

Figure 4.3: Conversion versus time graphs and radius size distributions for a) Test 1, b) Test 2, c) Test 3, d) Test 4 and e) Test 5.



b)

Figure 4.3 (Continued).

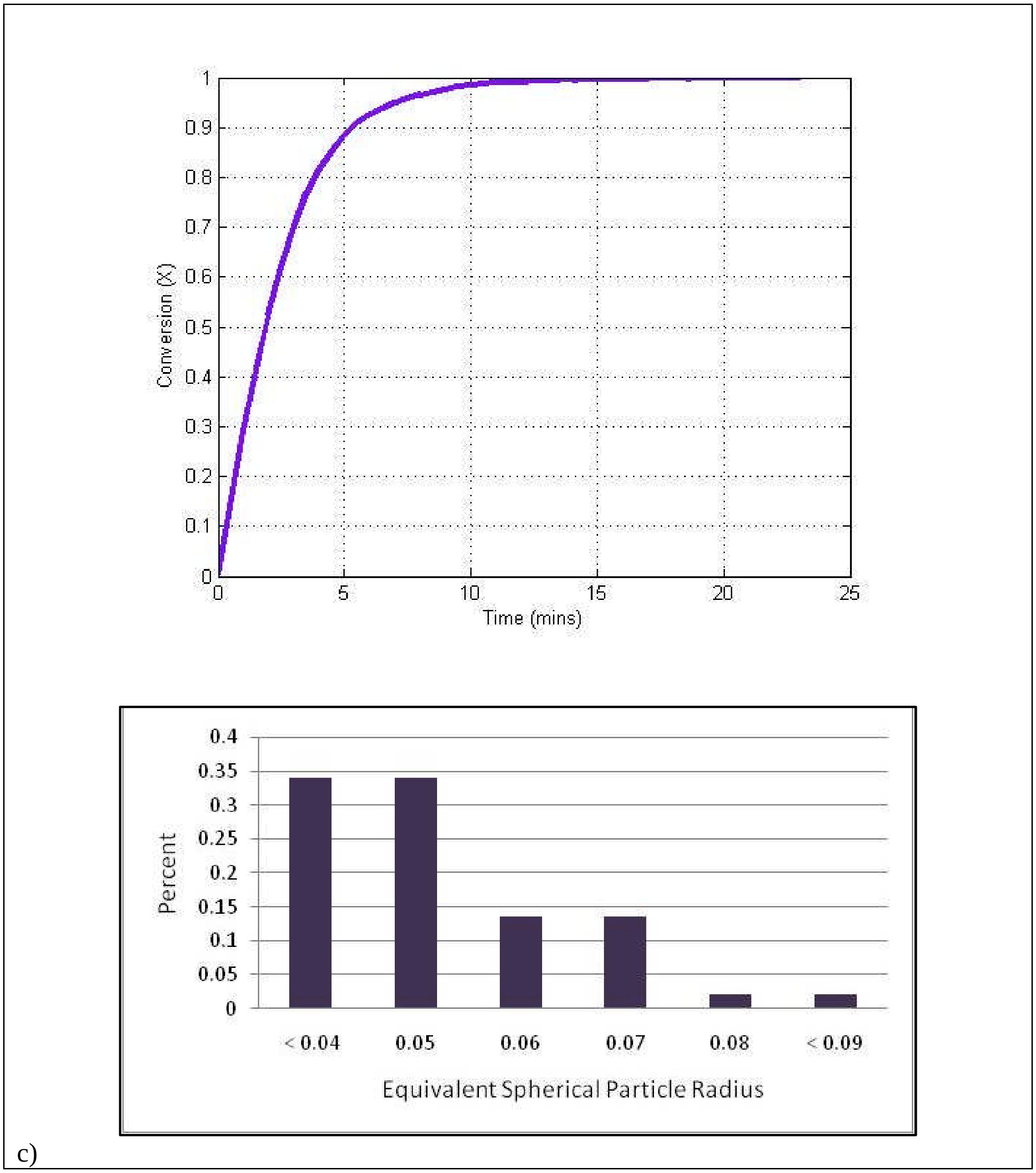
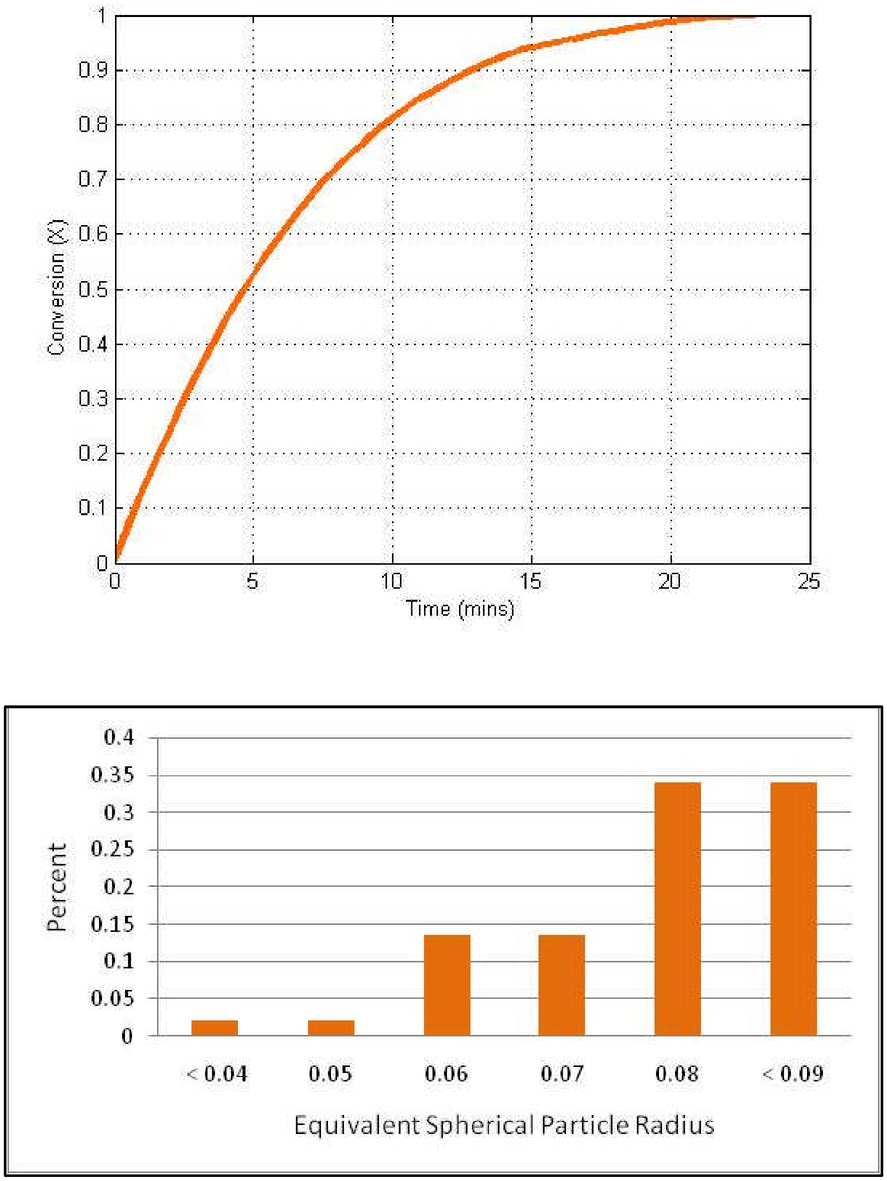
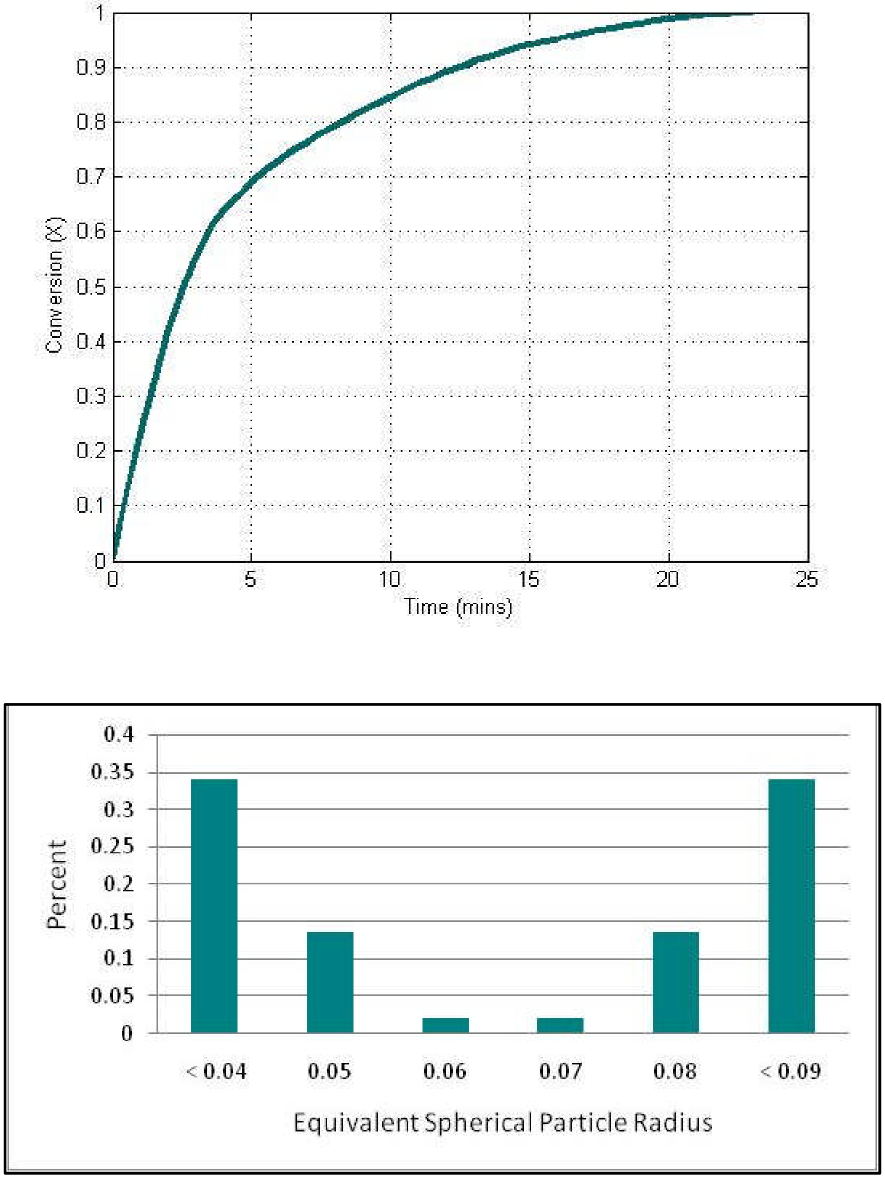


Figure 4.3 (Continued).



d)

Figure 4.3 (Continued).



e)

Figure 4.3 (Continued).

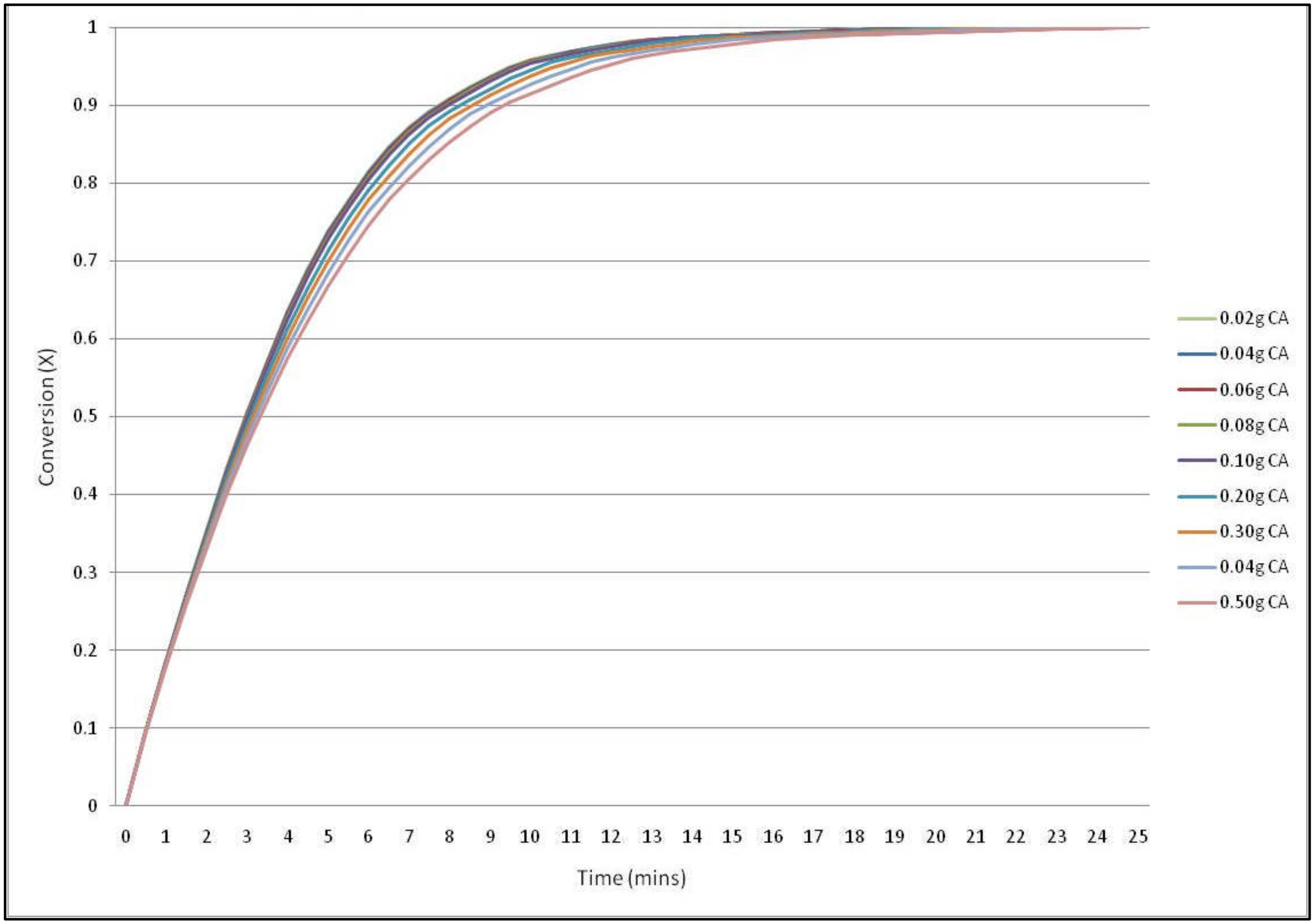


Figure 4.4: Polydisperse model for citric acid at different initial amounts.

4.2.2 Determination of Experimental Radius Size Distribution

As part of the particle size analysis, particles are estimated to be spherical in shape so that an equivalent spherical diameter (ESD) can be derived [47-49]. The citric acid obtained from Fisher Chemicals had crystal geometries that were not well defined, therefore, it was convenient to estimate the structure of the citric acid used in the experiments in terms of equivalent spherical shape. For this analysis, the equivalent radius was defined as the radius of a sphere with equal surface area. The measurements were made using a scanning electron microscope (Hitachi S-800) to image the surface of the particles of citric acid. Images of six samples of < 30 particles were taken at a magnification of 13X.

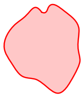
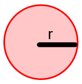
Shape	Actual Shape	Equivalent spherical shape	Surface area of shape	Equivalent spherical radius using equivalence by surface area
Undefined geometry-Crystal Structure			S_A	$r = (S_A/\pi)^{1/2}$

Figure 4.5: Equivalent spherical radius.

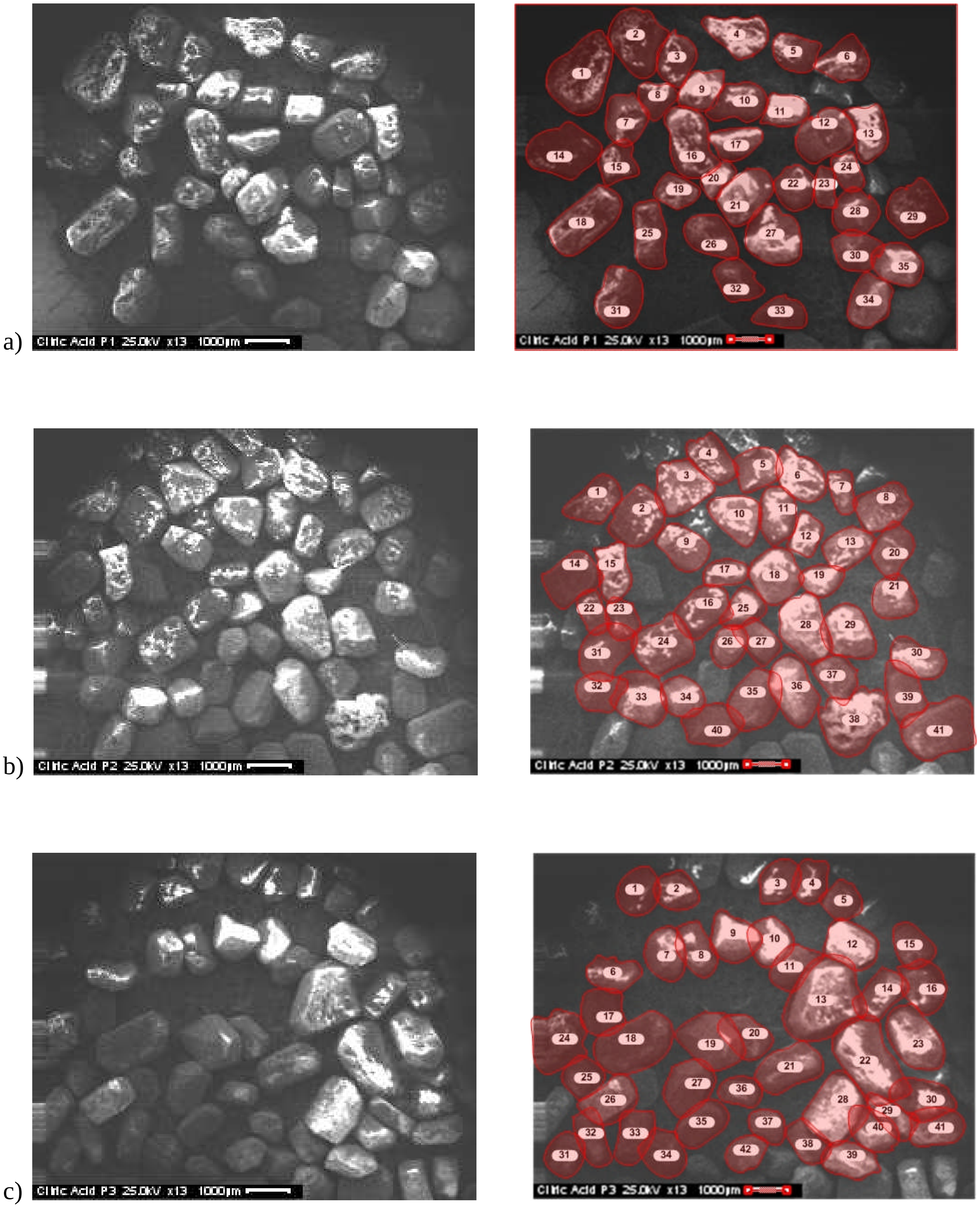

a)
Citric Acid P1 25.0kV x13 1000µm
Citric Acid P1 25.0kV x13 1000µm
b)
Citric Acid P2 25.0kV x13 1000µm
Citric Acid P2 25.0kV x13 1000µm
c)
Citric Acid P3 25.0kV x13 1000µm
Citric Acid P3 25.0kV x13 1000µm

Figure 4.6 : SEM and particle numbering for collection of equivalent area for six samples a)-f).

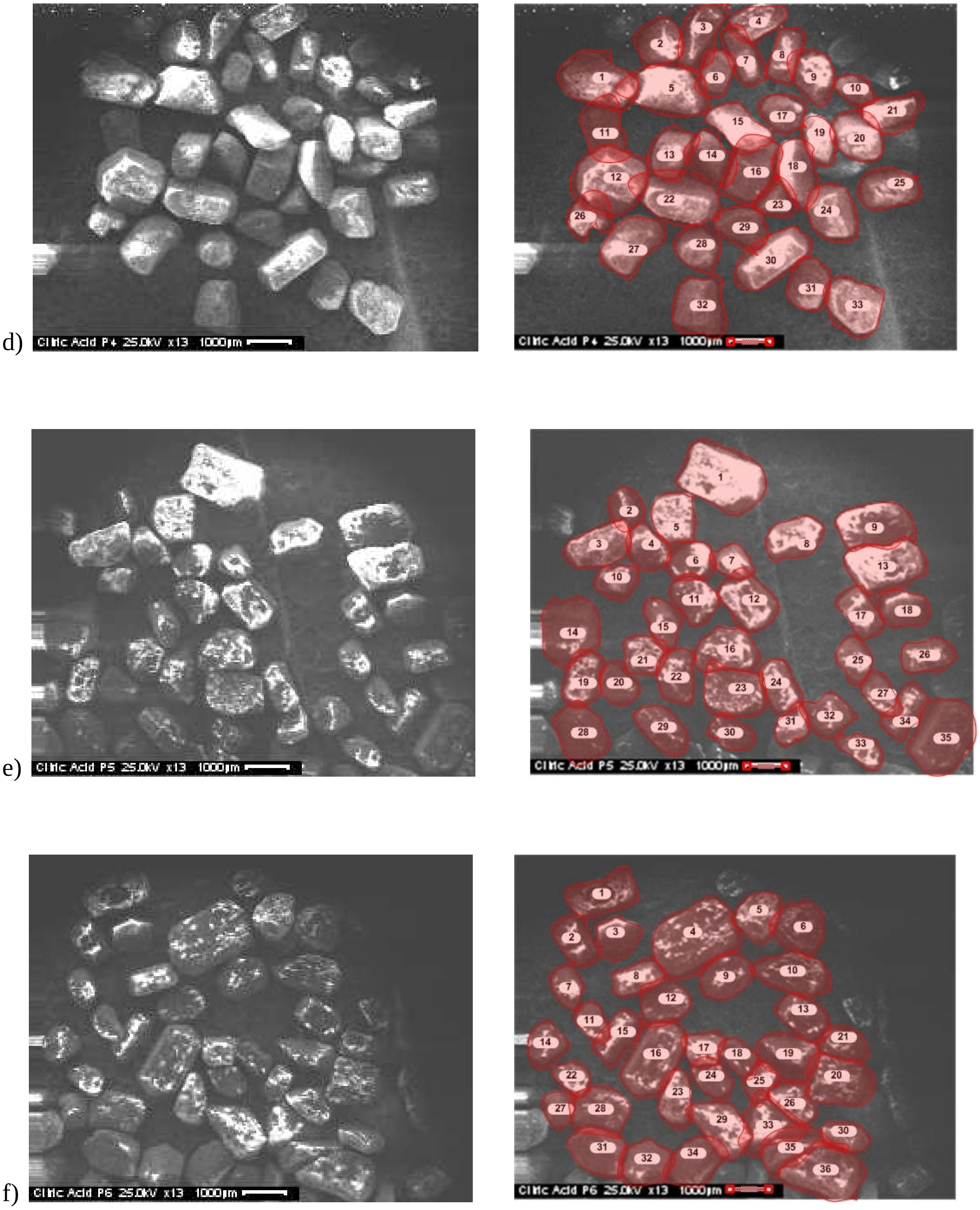
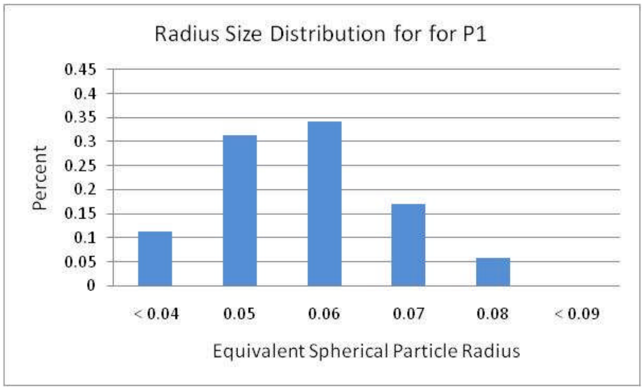
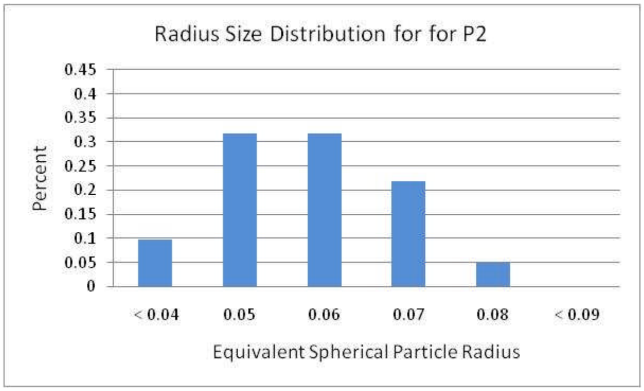


Figure 4.6 (Continued).



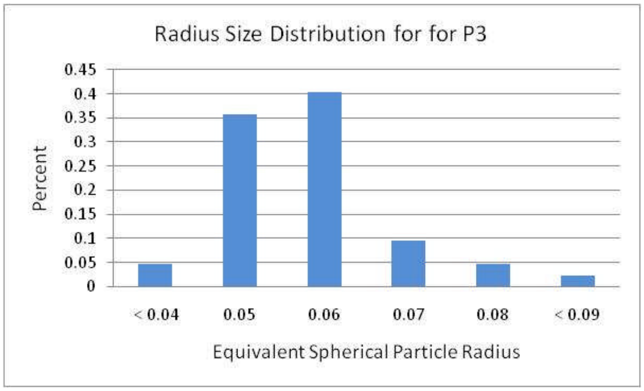
a)

Radius Size	Percent
> 0.04	11.4%
0.05	31.4%
0.06	34.3%
0.07	17.1%
0.08	5.7%
< 0.09	0 %



b)

Radius Size	Percent
> 0.04	9.8%
0.05	31.7%
0.06	31.7%
0.07	22.0%
0.08	4.9%
< 0.09	0 %

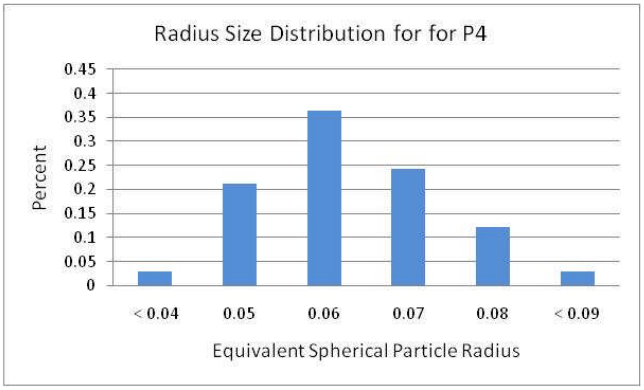


c)

Radius Size	Percent
> 0.04	4.8%
0.05	35.7%
0.06	40.5%
0.07	9.5%
0.08	4.8%
< 0.09	4.8%

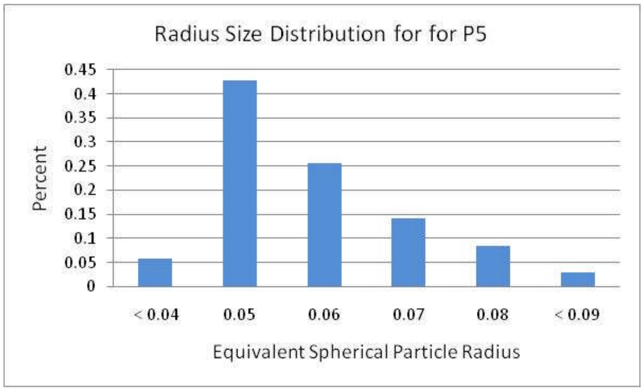
Figure 4.7: Radius size distributions for six samples a)-f).

d)



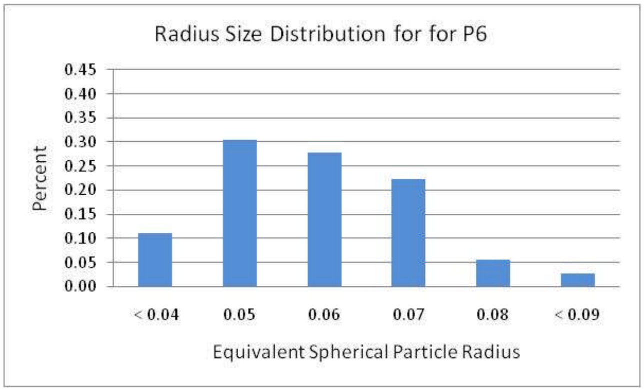
Radius Size	Percent
> 0.04	3.0%
0.05	21.2%
0.06	35.4%
0.07	24.2%
0.08	12.1%
< 0.09	3.0%

e)



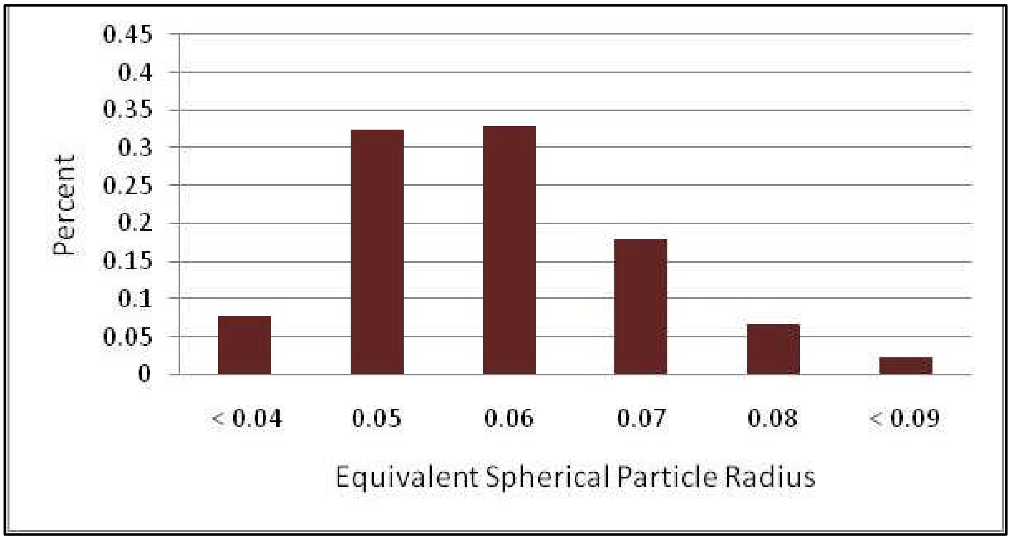
Radius Size	Percent
> 0.04	5.7%
0.05	42.9%
0.06	25.7%
0.07	14.3%
0.08	8.6%
< 0.09	2.9%

f)



Radius Size	Percent
> 0.04	11.1%
0.05	30.6%
0.06	27.8%
0.07	22.2%
0.08	5.6%
< 0.09	2.8%

Figure 4.7 (Continued).



Radius Size	> 0.04	0.05	0.06	0.07	0.08	< 0.09
Percent	7.7%	32.4%	32.9%	18.0%	6.8%	2.3%

Figure 4.8: Average radius size distribution of six samples using equivalent area from SEM.

4.2.3 MATLAB Output

Since the average area is not a usual measurement that would be known, this was converted to the equivalent mass percent. This would allow the user to adjust the radius size distribution in an experimental setup by measuring the desired mass of citric acid from a given radius size. The equivalent mass percent of the distribution found in Figure 4.8 is given below.

Table 4.2: Average mass percent of each radius size.

Radius Size	< 0.04	0.05	0.06	0.07	0.08	< 0.09
Mass Percent	2.1%	17.8%	30.9%	26.9%	15.1%	7.2%

.

By representing the distribution as a mass percent, the user simply has to multiply the mass percent by the total initial amount of citric acid to find out how much citric acid in a given size distribution must be used.

The mass percent is inputted into MATLAB in the *NumPart.m* section of the program and the average radius for each size distribution is assigned. The program is designed to allow as many size groups as necessary. For this example six size groups are defined.

```
% PARTICLE DISTRIBUTION PERCENT AND AVERAGE PARTICLE RADIUS
% Percent of particles with radius size less than 400 um (micrometers)
    Ps(1) = 0.0214;
    r0(1) = 0.04;            % cm
% Percent of particles with radius between the size of 401-500 um (micrometers)
    Ps(2) = 0.1768;
    r0(2) = 0.05;            % cm
% Percent of particles with radius between the size of 501-600 um (micrometers)
    Ps(3) = 0.3098;
     r0(3) = 0.059;           % cm
% Percent of particles with radius between the size of 601-700 um (micrometers)
    Ps(4) = 0.2696;
     r0(4) = 0.07;           % cm
% Percent of particles with radius between the size of 701-800 um (micrometers)
    Ps(5) = 0.1508;
    r0(5) = 0.08;            % cm
% Percent of particles with radius between the size of 801-1000 um (micrometers)
    Ps(6) = 0.0716;
    r0(6) = 0.1;            % cm
```

MATLAB code 6: Assignment of mass percent for each radius size group.

The program generated for polydisperse particles was more complicated because it involved multiple data sets for the different size groups. Therefore, three graphs were produced to visualize the changes. The first graph, Figure 4.9, showed the concentration change in each size group as function of time. The number of particles for each group was calculated using the average mass percents found in the samples of citric acid and is shown in Table 4.3.

Table 4.3: Number of particles in each radius size distribution.

Initial Radius, r_0	Number of particles, N
> 400 μm	5
401-500 μm	22
501-600 μm	23
601-700 μm	12
701-800 μm	5
< 900 μm	1

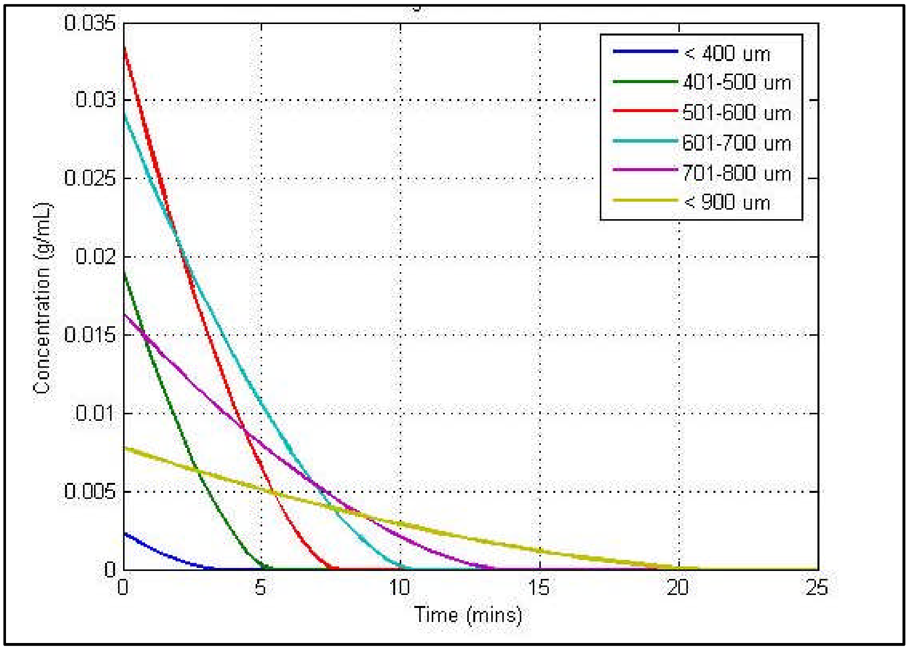


Figure 4.9: Concentration versus time for each radius size group.

The second graph represented the total concentration change. This was calculated by summing the concentration changes from each of the radius size groups.

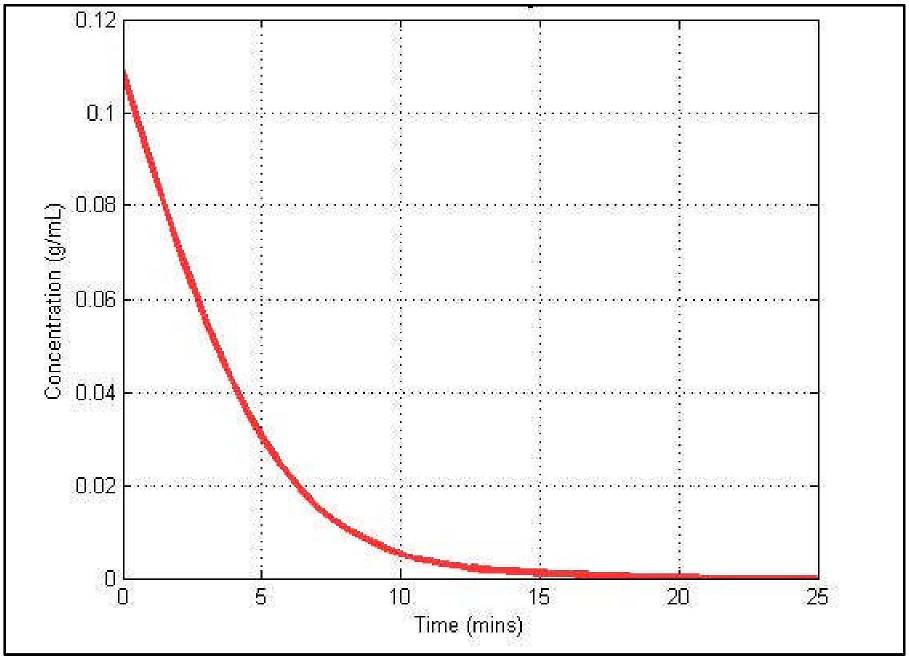


Figure 4.10: Total concentration change of polydisperse particles.

Then by dividing the total concentration for a given time by the total initial concentration, the conversion (X) of the entire system as a function of time was found.

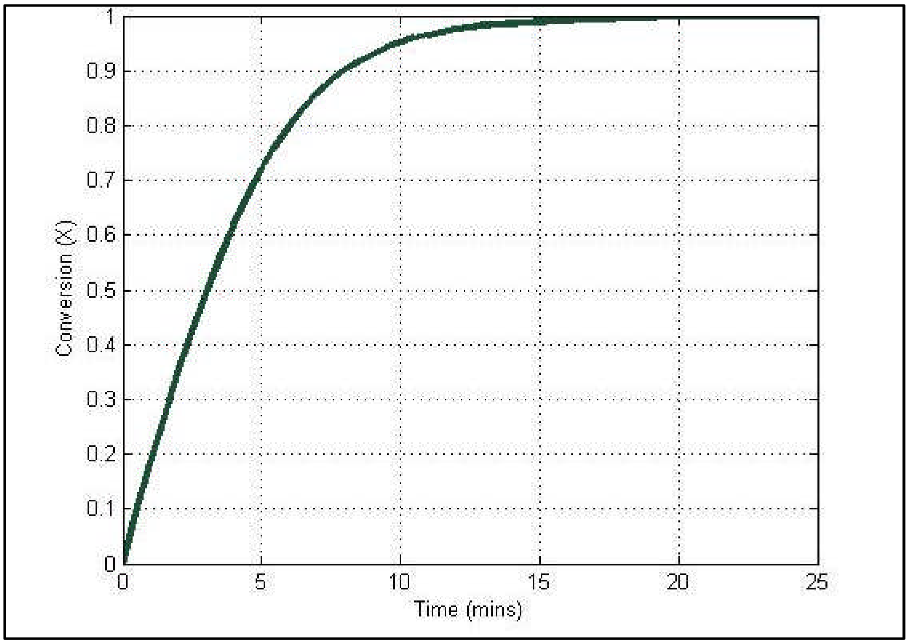


Figure 4.11: Total conversion of polydisperse particles.

The slope of the total conversion graph using the polydisperse was noticeably less steep than the conversion graphs produced using the monodisperse model. For radius sizes of less than 0.1 cm, the time for dissolution to complete was also longer than for the monodisperse particle model.

4.2.4 Results for Various Initial Concentrations

The average radius size distribution was applied to calculate the conversion for initial concentrations varying from 0.2-0.5g CA. The percent of each radius size found in Figure 4.8 was based on the equivalent area found from the SEM.

The program for polydisperse particles was run for initial concentrations of 0.02-0.50 grams of citric acid, the same as done previously for the monodisperse model. The polydisperse model resulted in a smoother and more prolonged conversion. For example, complete conversion of 0.10 grams of citric acid in the monodisperse model occurred at 7.5 minutes, while for the polydisperse model, dissolution was complete in 21 minutes. The polydisperse model resulted in an increase for dissolution time for all concentrations, as seen in Figure 4.12.

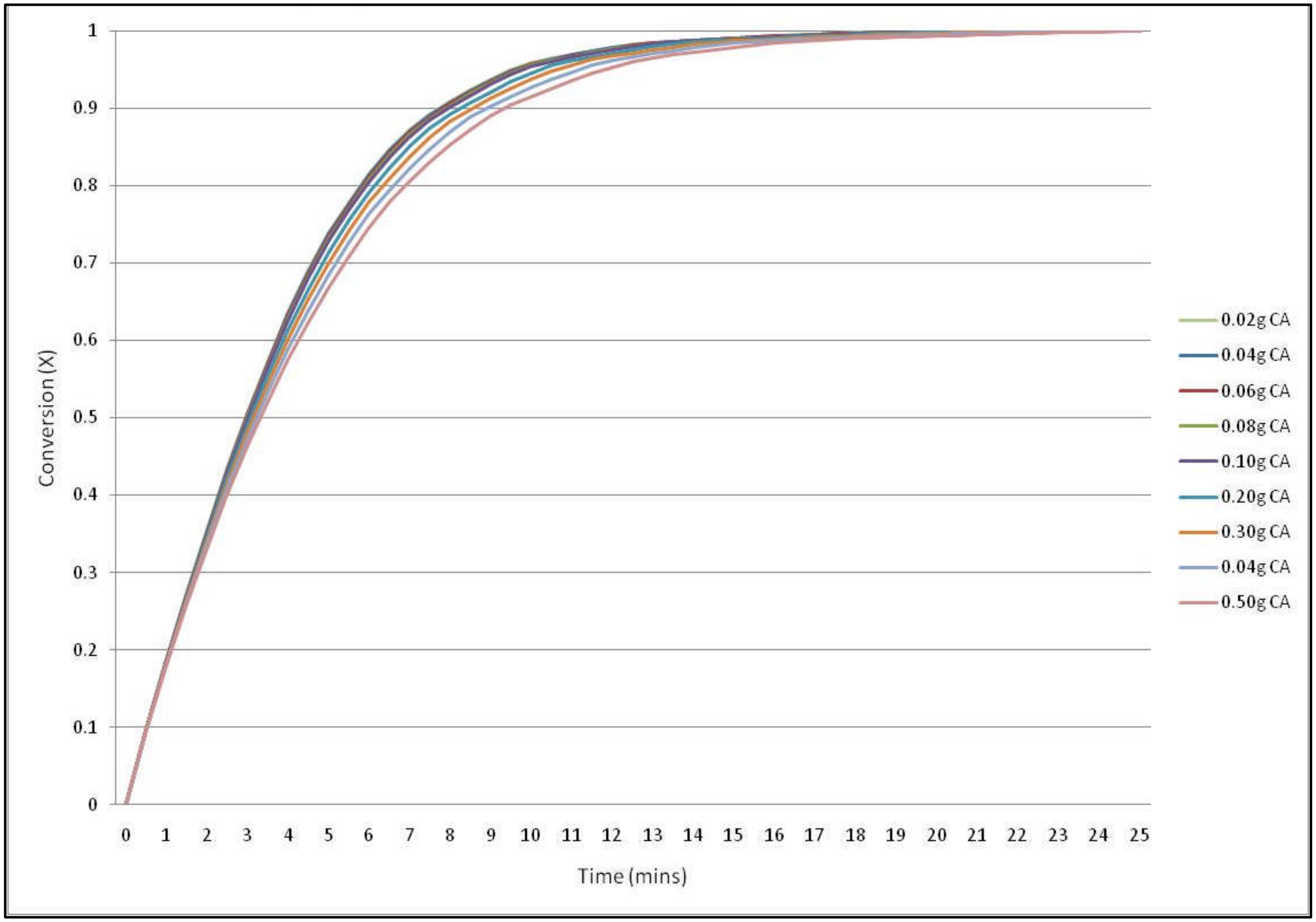
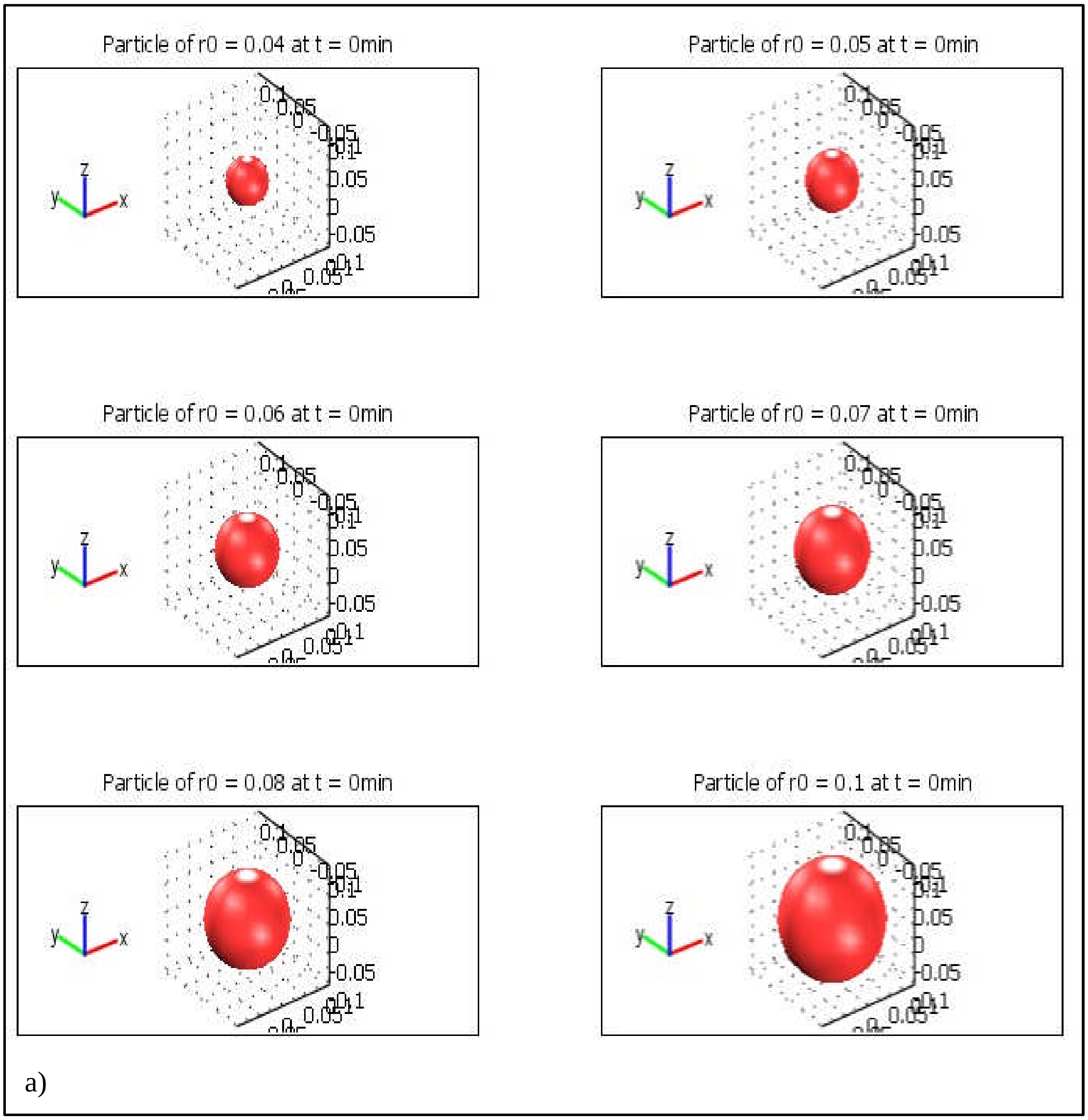


Figure 4.12: Polydisperse conversion versus time for citric acids at different initial concentrations.

4.3 Visualization for Multiple Particles

The COMSOL script code developed in Chapter 3 was adjusted to simultaneously output 3D graphs for particles of six different initial radius sizes. For the test case of 0.10 grams of citric acid and the size distribution found through SEM techniques, the graphs representing the change of radius for a polydisperse particle system produced are shown in Figures 4.13-18. The graphs show the changes for the six size groups from the initial time of 0 minutes to 20 minutes. It can be seen from the polydisperse results that the radius size has a significant effect on the time required for dissolution. While the monodisperse model for 0.10 grams using an average radius of 0.059 cm took only 8 minutes for complete dissolution, the polydisperse model using the average size distribution has a longer dissolution time of almost 20 minutes.



Particle of r0 = 0.04 at t = 0min

Particle of r0 = 0.05 at t = 0min

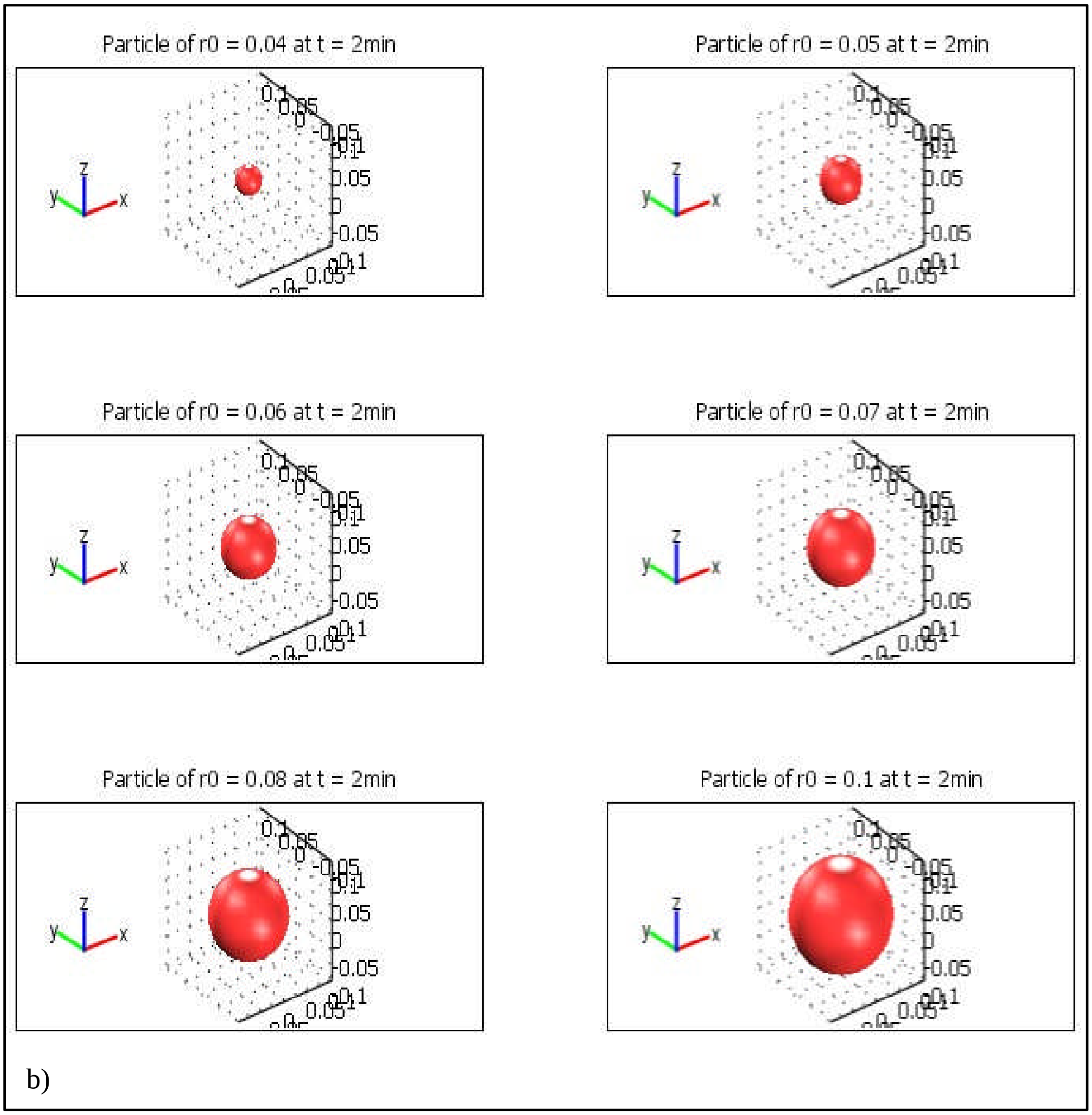
Particle of r0 = 0.06 at t = 0min

Particle of r0 = 0.07 at t = 0min

Particle of r0 = 0.08 at t = 0min

Particle of r0 = 0.1 at t = 0min

Figure 4.13: Polydisperse particle dissolution for a) t = 0 min, b) t = 2 mins, c) t = 6 mins, d) t = 10 mins e) t = 15 mins and f) t =20 mins.



Particle of r0 = 0.04 at t = 2min

Particle of r0 = 0.05 at t = 2min

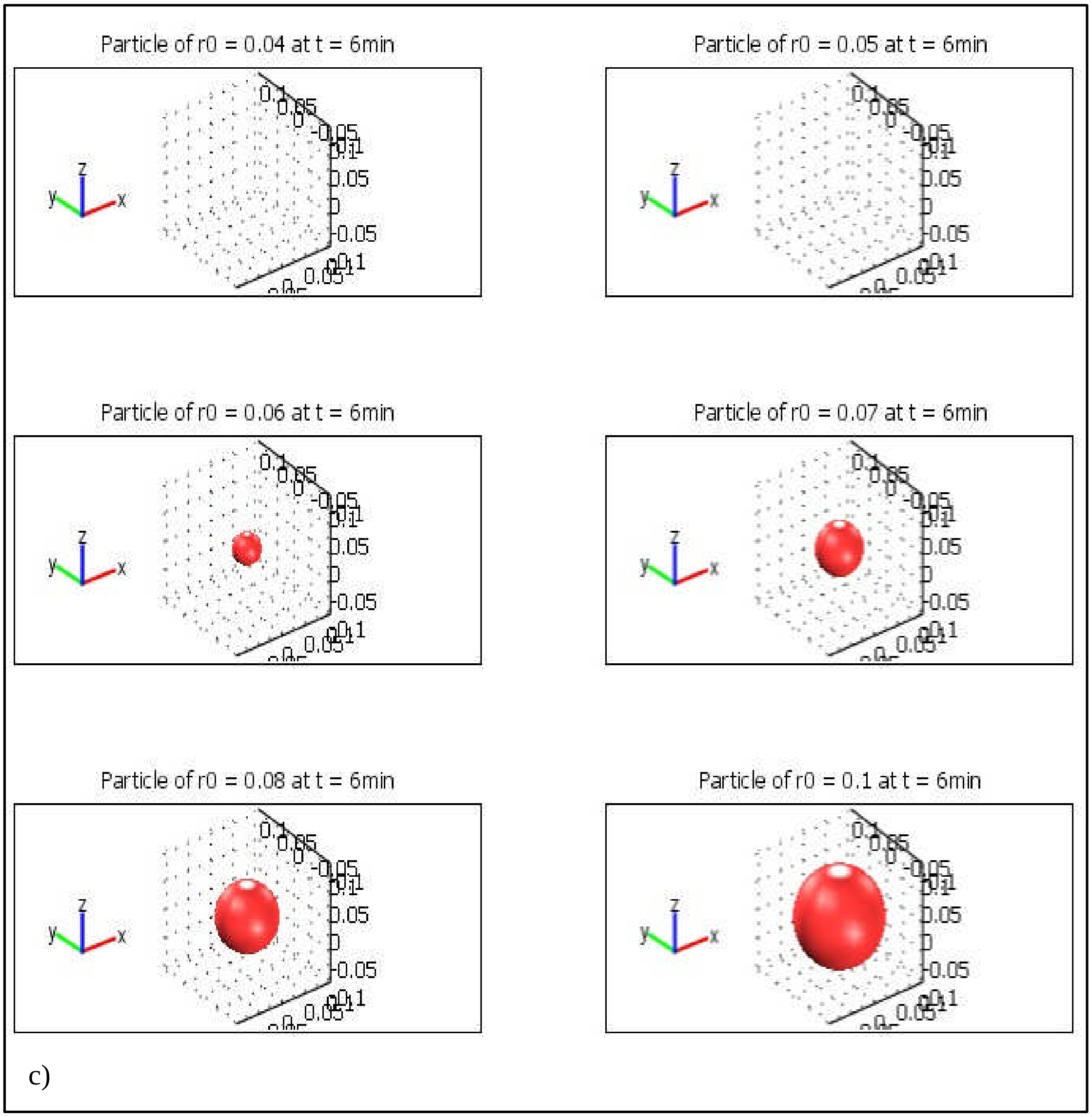
Particle of r0 = 0.06 at t = 2min

Particle of r0 = 0.07 at t = 2min

Particle of r0 = 0.08 at t = 2min

Particle of r0 = 0.1 at t = 2min

Figure 4.13 (Continued).


Particle of r0 = 0.04 at t = 6min

Particle of r0 = 0.05 at t = 6min

Particle of r0 = 0.06 at t = 6min

Particle of r0 = 0.07 at t = 6min

Particle of r0 = 0.08 at t = 6min

Particle of r0 = 0.1 at t = 6min

Figure 4.13 (Continued).

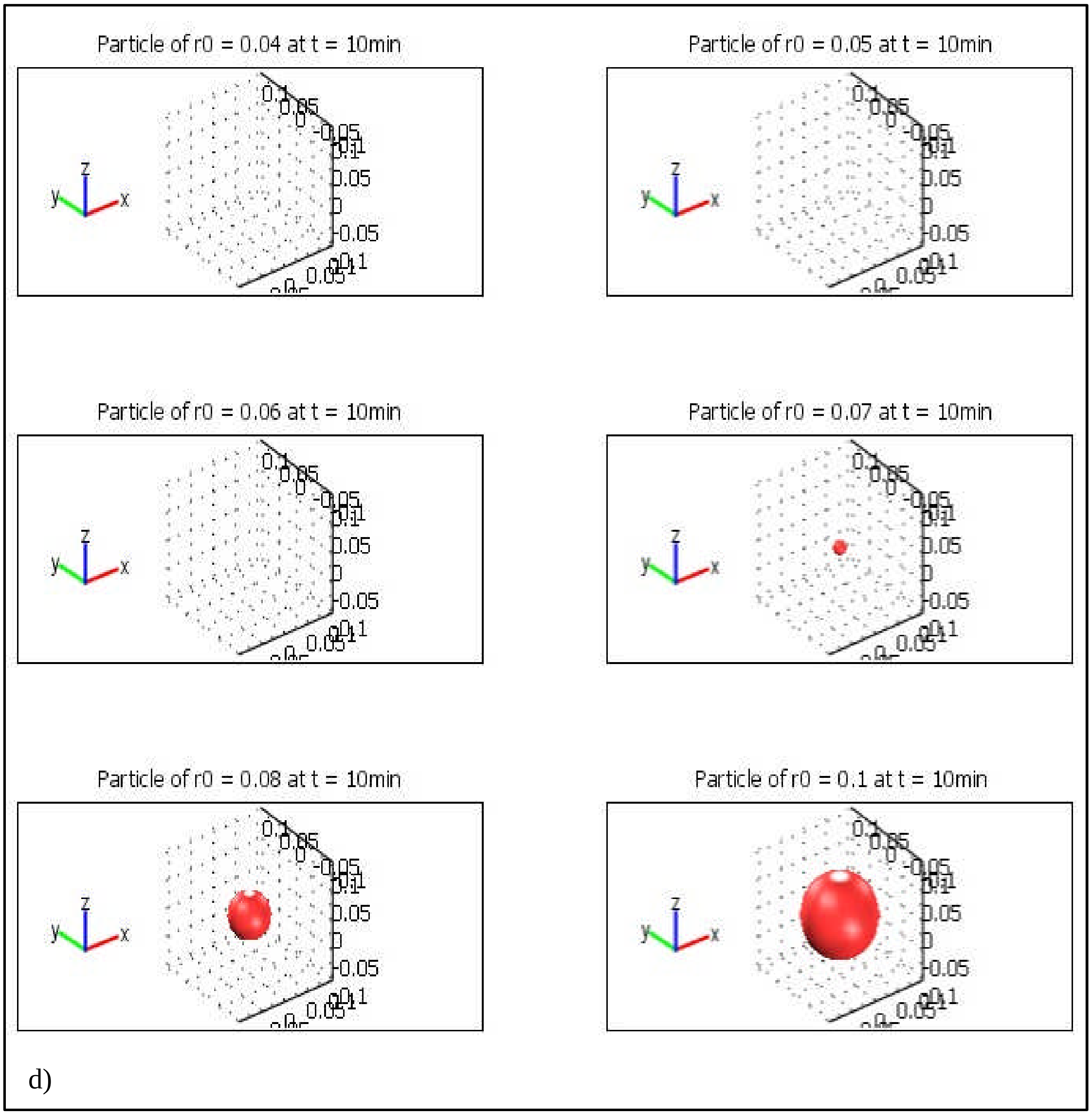
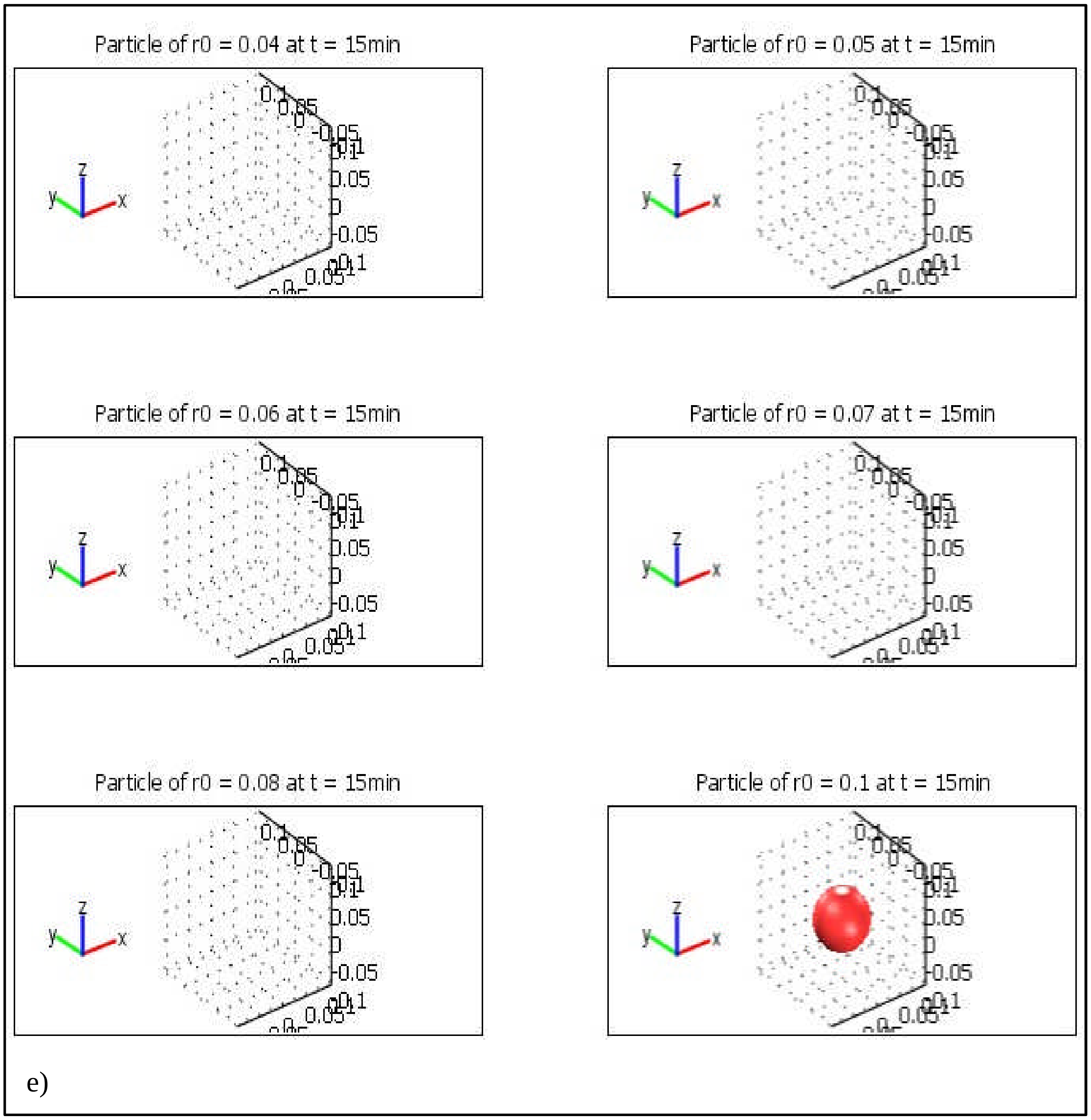

Particle of r0 = 0.04 at t = 10min
Particle of r0 = 0.05 at t = 10min
Particle of r0 = 0.06 at t = 10min
Particle of r0 = 0.07 at t = 10min
Particle of r0 = 0.08 at t = 10min
Particle of r0 = 0.1 at t = 10min
z
y
x
0.1
0.05
0
-0.05
-0.1
d)

Figure 4.13 (Continued).


Particle of r0 = 0.04 at t = 15min

Particle of r0 = 0.05 at t = 15min

Particle of r0 = 0.06 at t = 15min

Particle of r0 = 0.07 at t = 15min

Particle of r0 = 0.08 at t = 15min

Particle of r0 = 0.1 at t = 15min

Figure 4.13 (Continued).

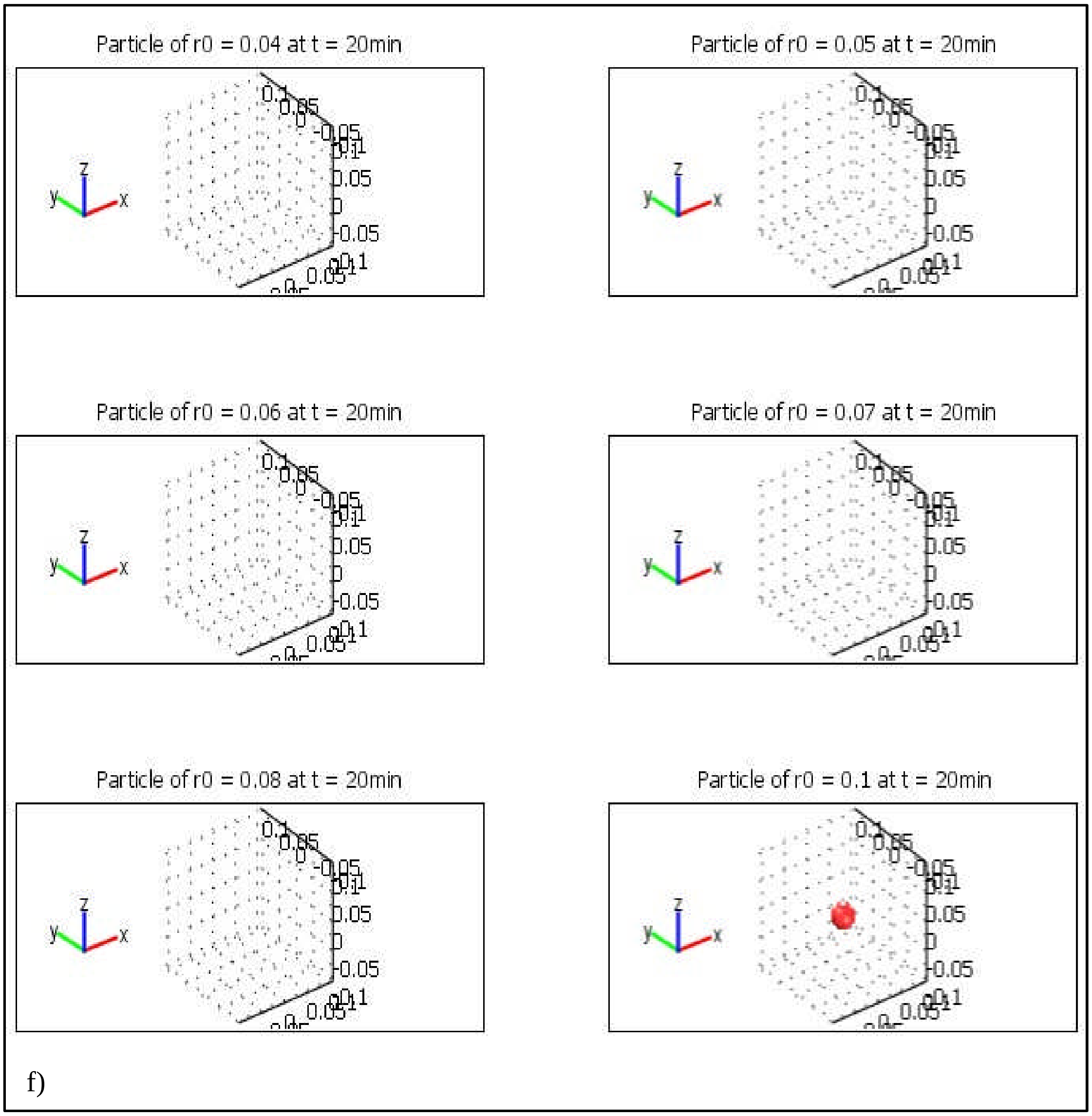

Particle of r0 = 0.04 at t = 20min
Particle of r0 = 0.05 at t = 20min
Particle of r0 = 0.06 at t = 20min
Particle of r0 = 0.07 at t = 20min
Particle of r0 = 0.08 at t = 20min
Particle of r0 = 0.1 at t = 20min
z
y
x
0.1
0.05
0
-0.05
f)

Figure 4.13 (Continued).

4.4 Summary

By incorporating the radius size distribution into the MATLAB program, an increasingly comprehensive code was generated. For every size group created, it was necessary to re-run the initial code, produced in Chapter 3, using the radius size of the specified group. In addition, the results from all radius size groups needed to be combined to describe the overall effect. The polydisperse particles model provided more realistic results for dissolution, since most industrial chemicals involve particle size distributions.

The program produced in this section also has additional benefits in optimization and product design. By manipulating the percentages of each of the radius size distributions, the rate of dissolution can be changed significantly, even when total initial weight remains constant. This allows the manufacturer to determine the best radius size to use for a given application while keeping the amount of resources used to a minimum. When a desired dissolution profile is given, the program can be used to back-calculate what size distributions will produce the desired effect.

Chapter 5 : Encapsulated Monodisperse Particles

Controlled release through encapsulation has been studied for use in pharmaceuticals [17], chemicals [2], cosmetics, foods [5] and pesticides [6]. The encapsulation of citric acid has benefits in manufacturing, storage and use. The encapsulation can increase the processing speed of the particles and allows for more durability by hardening the outer shell. The additional layer also protects the core citric acid material from undergoing undesired reactions during storage. Finally, the encapsulation improves the functionality of the citric acid by controlling of the acid release allowing for the creation products which can be tailor desired to meet the manufacturers needs.

For simplification purposes, the encapsulated particles in this section are assumed to be monodispersed. The core material considered is citric acid and is designated as the red material in Figure 5.1. Glucose was chosen as a suitable coating material and is represented using the color green. Since this application considered the dissolution in water, glucose was an appropriate coating material because it was a water soluble carbohydrate which could be manufactured using spray dried glucose syrup. Glucose is also an approved sweetening food additive, making it a safe choice for use applications in the food industry. Processed citric acid has functional benefits in pharmaceuticals, food applications, health care and detergents.

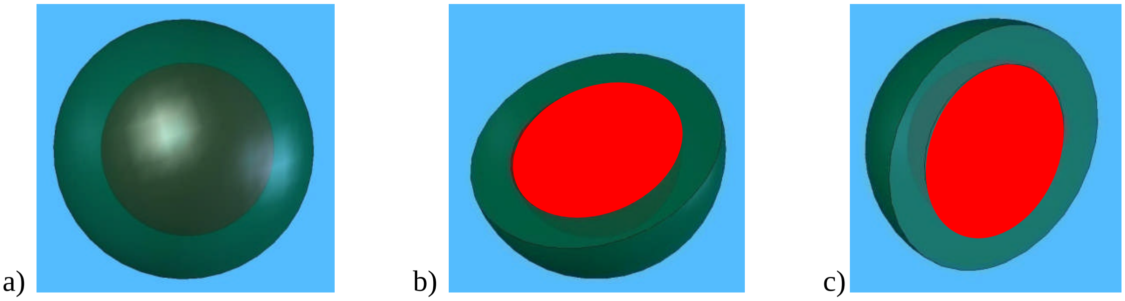


Figure 5.1: Encapsulated particle a) full view b) side section view c) front section view.

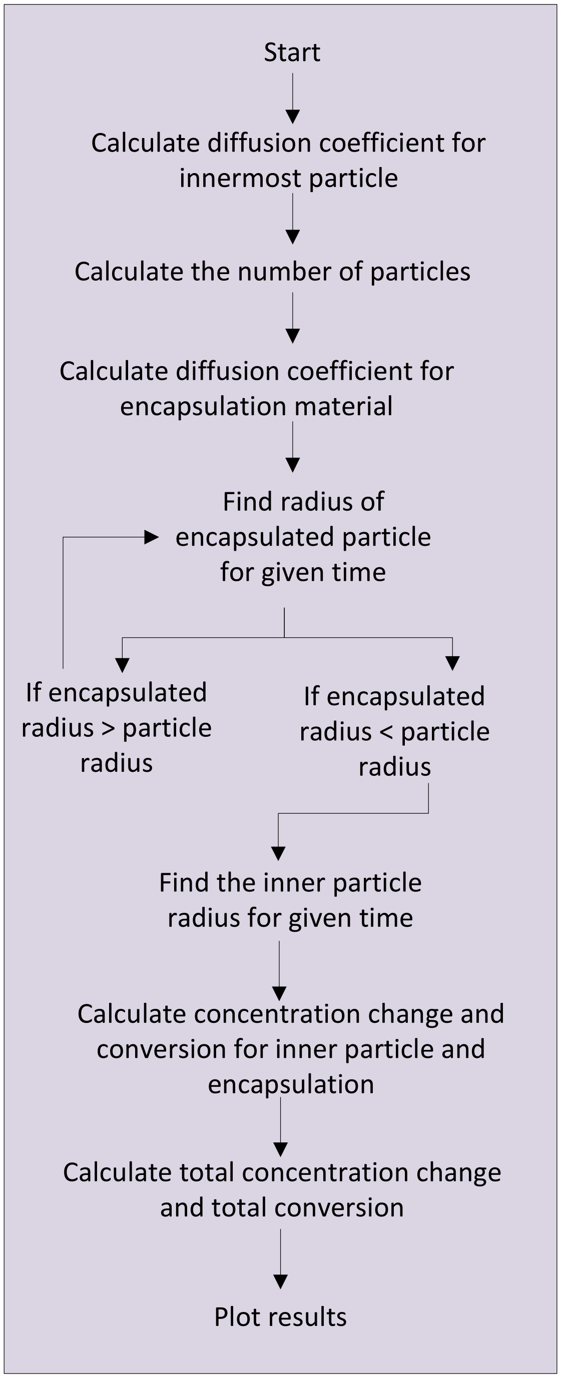

Start
Calculate diffusion coefficient for innermost particle
Calculate the number of particles
Calculate diffusion coefficient for encapsulation material
Find radius of encapsulated particle for given time
If encapsulated radius > particle radius
If encapsulated radius < particle radius
Find the inner particle radius for given time
Calculate concentration change and conversion for inner particle and encapsulation
Calculate total concentration change and total conversion
Plot results

Figure 5.2: Program algorithm for encapsulated monodisperse particles model.

The mathematical model used to describe the dissolution of encapsulated monodisperse particles is similar the one developed in Chapter 3. Since all the particles are assumed to be initially the same size and have the same encapsulation thickness, the dissolution of all the particles occurs at the same time, as seen in Figure 5.4. The encapsulation layer complicates the model by requiring two sets of chemical property parameters, one for the encapsulation layer and the second for the inner particle. Calculations for the eroding encapsulation layer are performed first until all of the coating is dissolved. Then the inner particle calculations are performed in a manner resembling the method developed for monodisperse particle dissolution.

The concentration and conversion must be calculated twice, once for the encapsulation material and the second for the inner particle material. The concentration and conversions for both materials will be totaled and plotted as a function of time. The program algorithm is shown in Figure 5.2.

5.1 Model for Encapsulation

In this section, the program designed in Chapter 3 was extended to include calculations for an erodible layer encapsulated around the original spherical particle. Similar assumptions on the derivation on the dissolution kinetics were made for the layer of encapsulation, however the program was designed to include separate calculations based on the parameters of the material used in the coating layer. The program also had to be designed to check for when the coating layer completely dissolved, and to begin calculations at that time for the inner particle dissolution.

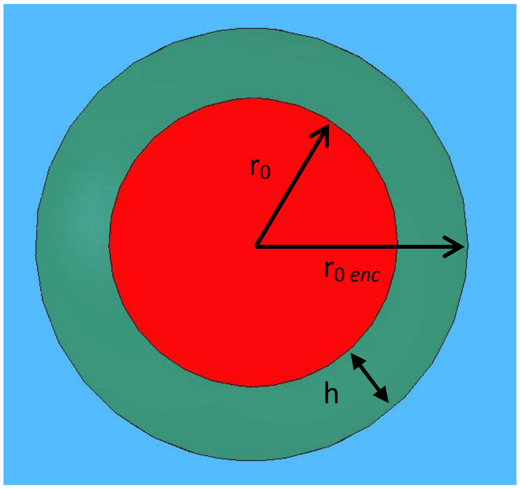


Figure 5.3: Inner particle radius, $r_{0,}$ with encapsulation layer thickness, h, yields radius of encapsulated particle $r_{0enc.}$

5.1.1 Calculation of Diffusion Coefficient for the Encapsulated Layer

The thickness of the encapsulation layer is given as h. The radius of the encapsulated particle, which includes the inner particle and the coating layer, is defined as

$$r_{0enc} = r_0 + h \qquad (5.1)$$

where *$r0_{enc}$* is the initial radius of the encapsulation and particle and *r0* is the initial radius of the particle. Following the procedure outlined in Section 3.2, the diffusion coefficient using the Wilke-Chang method was found for the coating material, *D_{12enc}*, by

$$D_{12enc} = 7.4\cdot10^{-8}\left[\frac{(\phi_{enc}M_2)^{\frac{1}{2}}T}{\mu_{2enc}V_{1enc}^{0.6}}\right] \qquad (5.2)$$

where M_2 is the molecular weight of the solvent (g/mol), T is the temperature (Kelvins), $\mu_{2\ enc}$ is the viscosity of the solvent solution (centipoises), $V_{1\ enc}$ is the molar volume of the solute at normal boiling point (cm^3/g mol) and $\varphi_{\ enc}$ is the association parameter of the solvent for the encapsulation material specified.

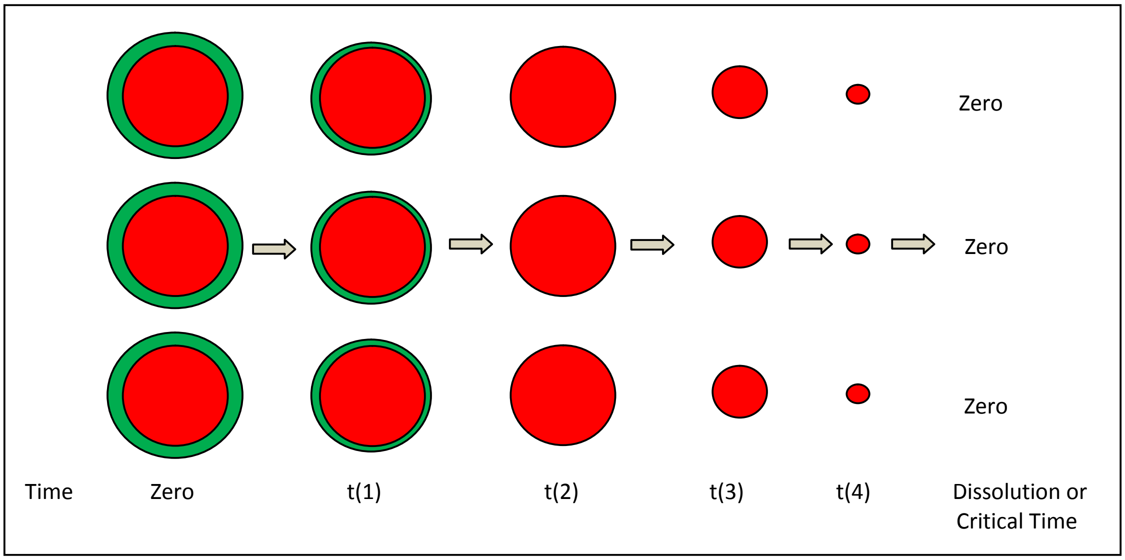


Figure 5.4: Schematic illustration for encapsulated monodisperse particle dissolution.

5.1.2 Determining the Radii of Encapsulated Layer

The addition of the encapsulated layer increases the complexity of the problem by creating two separate radii which must be solved in a sequential order. The dissolution of the outer coating is calculated first. Since the coating material differs from the particle being encapsulated, different constants must be calculated for use in the equation to solve for the radius at a given time. The constant c for the encapsulated material, expressed as c_{enc}, is given by

$$c_{enc} = \left({r_{0enc}}^3 - \frac{3 \cdot C_{s\_enc} \cdot V_m}{N \cdot \rho_{enc} \cdot 4 \cdot \pi} \right)^{\frac{1}{3}} \tag{5.3}$$

where C_{s_enc} is the solubility of the solute (g/mL) and ρ_{enc} is the density of the encapsulation material. Using the constant c_{enc}, the constant of integration, k_{enc}, for the encapsulation material can then be determined using the formula given in Eq. (5.4).

$$k_{enc} = \frac{1}{6c_{enc}} \ln \frac{{c_{enc}}^3 - {r_{0enc}}^3}{(c_{enc} - r_{0enc})^3} - \frac{1}{\sqrt{3}c_{enc}} \tan^{-1} \frac{2r_{0enc} + c_{enc}}{\sqrt{3}c_{enc}} \tag{5.4}$$

The constant c_{enc} and the integration constant k_{enc}, for the encapsulation material are then inserted into the main equation, F_{enc}, which relates the radius of the encapsulated particle, r_{enc}, to each given time as

$$\begin{aligned} F_{enc} &= \frac{1}{6c_{enc}} ln \frac{{c_{enc}}^3 - {r_{enc}}^3}{(c_{enc} - r_{enc})^3} - \frac{1}{\sqrt{3}c_{enc}} tan^{-1} \frac{2r_{enc} + c_{enc}}{\sqrt{3}c_{enc}} \\ &\quad - k_{enc} - \left[\frac{D \cdot N_{enc} \cdot 4 \cdot \pi}{3 \cdot V_m} \cdot t \right] \end{aligned} \tag{5.5}$$

The radius for the encapsulated particle, r_{enc}, is calculated using Eq. (4.24) until the coating is completely dissolved, when r_{enc}=0. When this occurs, the program records the time at which r_{enc}=0, and begins solving the inner particle dissolution using this as the starting time. The procedure for solving the inner particle dissolution using the same program developed in Chapter 3.

5.1.3 Concentration and Conversion of Encapsulation Material

The mass of the encapsulation material, M_{enc}, at a given time t is

$$M_{enc}(t) = \left[\frac{4}{3} \cdot \pi \cdot r_{enc}(t)^3 \cdot \rho_{enc} \cdot N \right] - M_0 \tag{5.6}$$

where $r_{enc}(t)$ is the radius of the encapsulated material at time t and M_0 is the initial mass of the particle as derived in Eq. (3.9). The mass of the encapsulated material can be converted to concentration by

$$C_{enc}(t) = \frac{M_{enc}(t)}{V_m} \tag{5.7}$$

where $C_{enc}(t)$ is the concentration of the encapsulation material at time t. The initial concentration of coating material is calculated based on the initial encapsulated particle radius using

$$C_{0enc} = \frac{\left[\frac{4}{3} \cdot \pi \cdot {r_{0enc}}^3 \cdot \rho_{enc} \cdot N \right]}{V_m} \tag{5.8}$$

The concentration of dissolved encapsulation material, C_{denc}, is found by

$$C_{denc}(t) = C_{0enc} - C_{0enc}(t) \tag{5.9}$$

The amount dissolved is used to find the conversion of the encapsulated material by applying the definition of conversion as given in Eq. (5.10).

$$X_{enc}(t) = \frac{C_{denc}(t)}{C_{0enc}} \tag{5.10}$$

5.1.4 Visualization of Results for Encapsulated Model

The function *PrintResults*.m first converts the time in seconds to time in minutes. Then plots of concentration and conversion versus time are displayed. The encapsulation material and inner particle are represented as different colors and plotted on the same graph.

5.2 Example for Encapsulated Model

The following section will used the designed program in the case study of citric acid particles encapsulated with glucose dissolving in water at room temperature. The properties for citric acid were assumed to be the same as given in Table 3.1 and the coating material of glucose properties were listed in Table 5.1. The thickness of the encapsulation layer must also be specified at this time, and is given by h. For this example, a thickness of 0.001 cm of glucose encapsulation is considered.

Table 5.1: Parameters for glucose coating material.

Density of glucose	ρ	$= 1.54$ g/cm^3
Molecular weight of solvent	M_2	$= 18.015$ g/mol
Temperature	T	= 298 K
Viscosity of solution (solvent)	μ_2	= 0.91 centipoises
Molar volume of solute	V_1	$= 319.88$ g/cm^3·mol
Association parameter of solvent	Φ	= 2.6
Solubility of the solute	C_s	= 0.91 g/mL
Dissolution medium volume (water)	V_m	= 1 mL
Encapsulation layer thickness	h	= 0.001 cm

5.2.1 Simulation Setup for Encapsulation Model

The parameters for glucose were inputted into the MATLAB program in the *DiffusionEnc.m* and *Function2.m* sections. The diffusion coefficient for the glucose encapsulation layer is found to be 5.67 x 10^{-6} cm^2/sec, compared to the diffusion coefficient of citric acid in water of 5.21 x 10^{-6} cm^2/sec. It is important to note that the diffusion coefficient calculated in Chapter 3 is used in this section only as an estimate for citric acid, since the solvent properties will differ due to the glucose dissolved in the solution. Since tests had not be done to determine the new ternary system properties, the values calculated in this section provide a reasonable estimate of the diffusion characteristics. It was not within the scope of this project to experimentally find the changes in the diffusion rate of citric acid in a glucose solution, however, results from this type of investigation could significantly improve the accuracy of the model.

After the program was run, two resulting graphs were produced. The first was showed the change of the encapsulation and inner particle concentration, shown in Figure 5.5, while the second graph monitored the conversion of the two materials, as shown in Figure 5.6. The encapsulation thickness is very thin in this example, so it has little effect on the release rate of citric acid. Within the first minute, the glucose coating is completely dissolved, exposing the inner particle. Once the water is in contact with the citric acid, dissolution occurs in a fashion similar to the one described in Chapter 4.

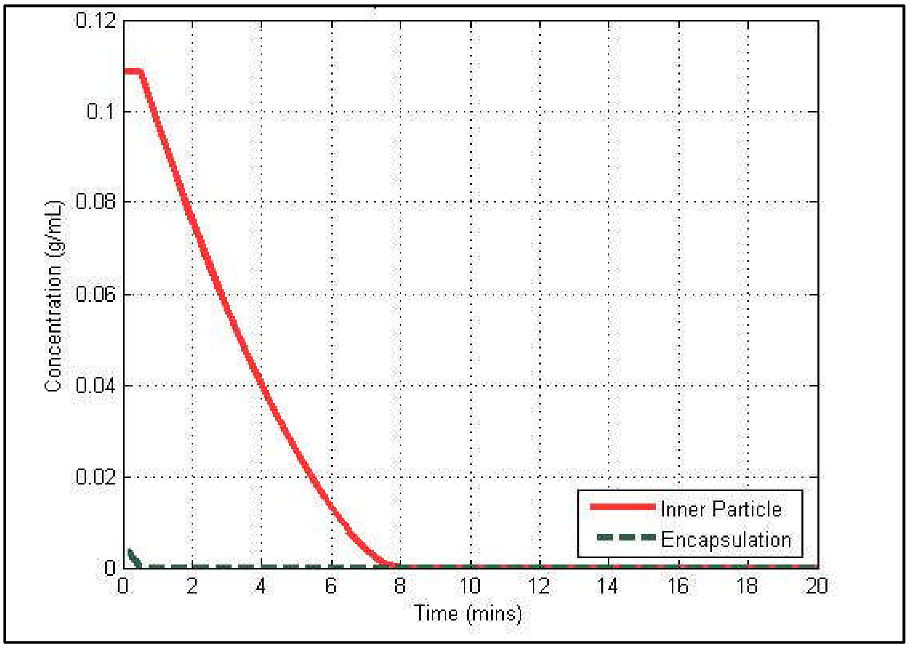


Figure 5.5: Concentration versus time for encapsulation thickness of 0.0010 cm.

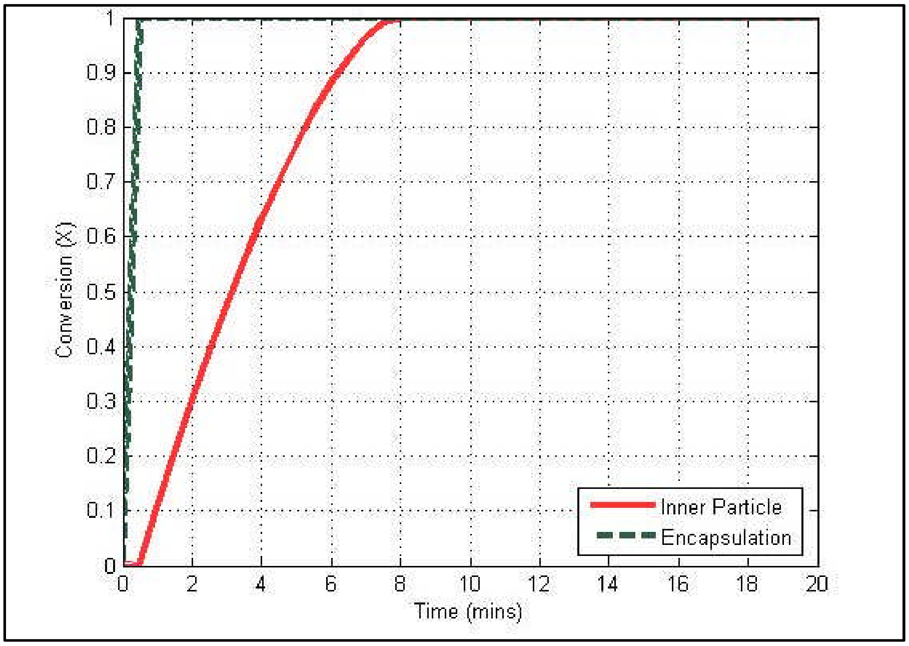


Figure 5.6: Conversion versus time for encapsulation thickness of 0.0010 cm

5.2.2 Effect of Encapsulation Thickness

A similar simulation was completed for an encapsulation thickness of 0.01 cm, a 10% increase from the thickness tested previously. The results from this run are shown in Figures 5.7 and 5.8. The results showed that the concentration of the encapsulating material dropped from 0.06 grams of glucose to 0 grams in about 4 minutes and the concentration of citric acid at this time remained constant. After all of the glucose is dissolved, the citric acid begins dissolution until there is no longer any citric acid is present, which occurs at around 11 minutes. Compared to the unencapsulated particles modeled in Chapter 3, which had a total dissolution time of around 7 minutes, the an encapsulated particle will not complete dissolution for an addition 4-5 minutes. This characteristic could be useful in designing functional particles where the activity of the acid needs to be delayed. For food applications, postponement of acid release can prolong the flavor enhancement of a product. In pharmaceuticals, the active ingredient may want to be protected until it reaches a target site, such as a specific organ, before it is released. The same control is also desired in detergents, since certain cleaning agents should not be released until a specific time in the washing cycle.

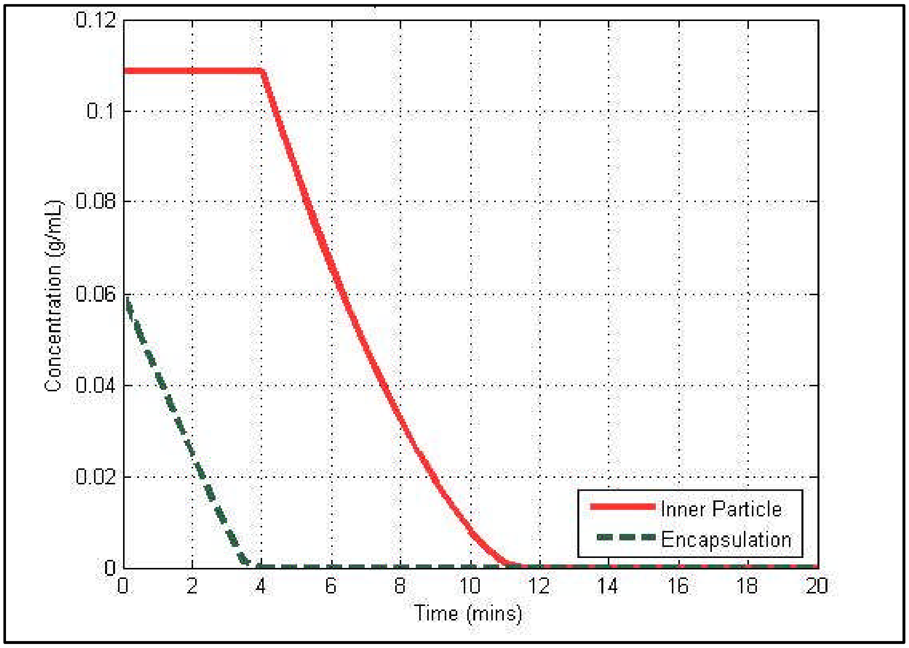


Figure 5.7: Concentration versus time for encapsulation thickness of 0.010 cm.

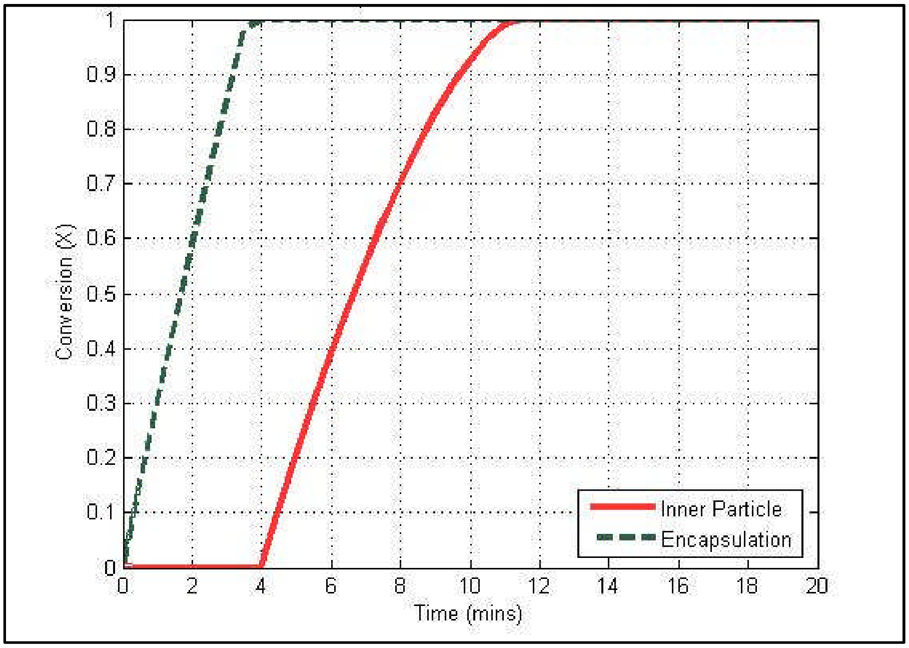


Figure 5.8: Conversion versus time for encapsulation thickness of 0.010 cm.

5.2.3 COMSOL Visualization for Encapsulated Model

COMSOL was used to visualize the changes in radius of the particle as dissolution proceeded. Data calculated from the MATLAB program was imported into COMSOL Script and 3D graph of the spherical particle was generated. The code was written to display the encapsulation material in green and in the inner core particle as red.

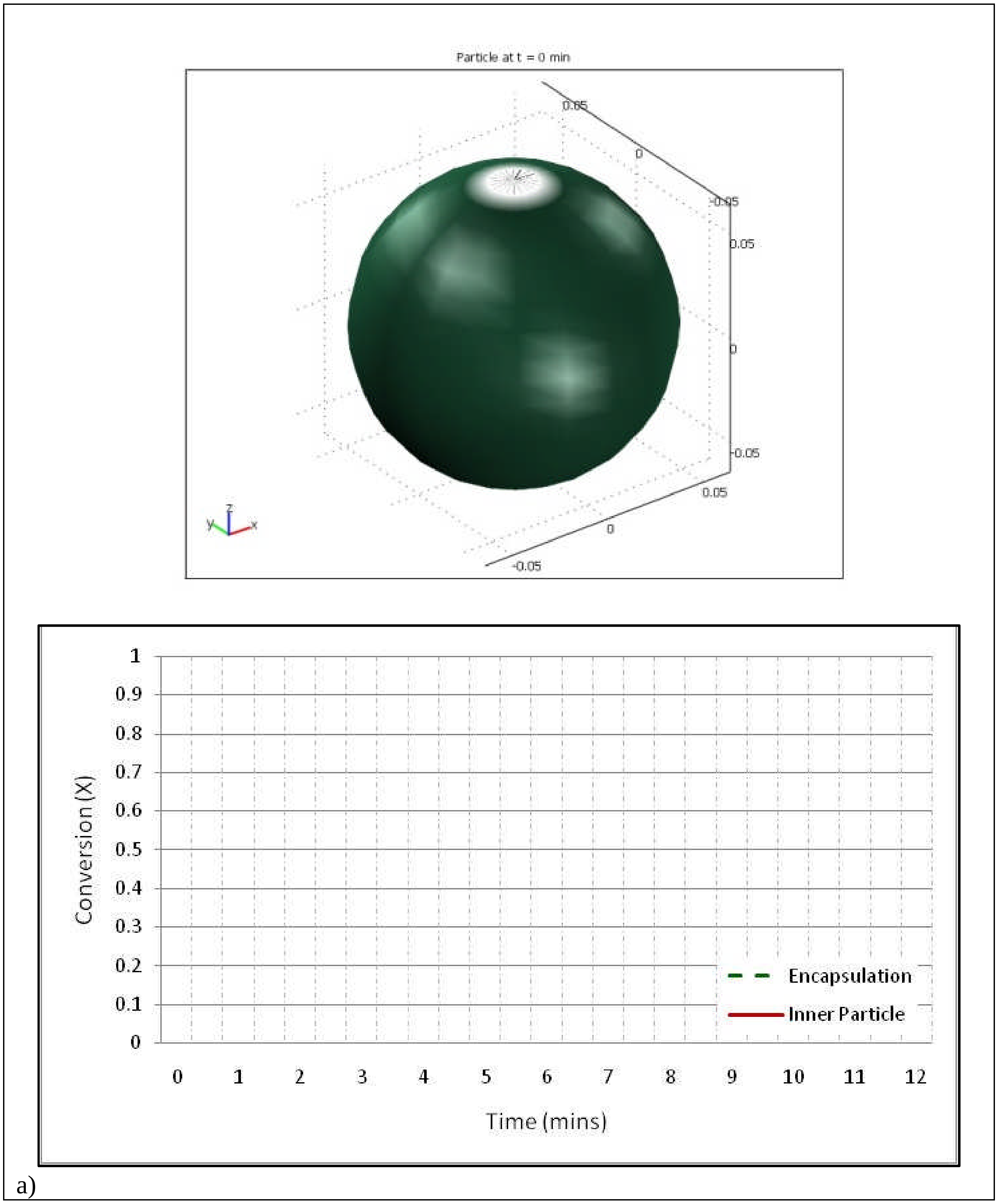


Figure 5.9: Encapsulated particle conversion for a) t = 0 min, b) t = 1 mins, c) t = 3 mins, d) t = 5 mins, e) t = 7 mins and f) t = 9 mins.

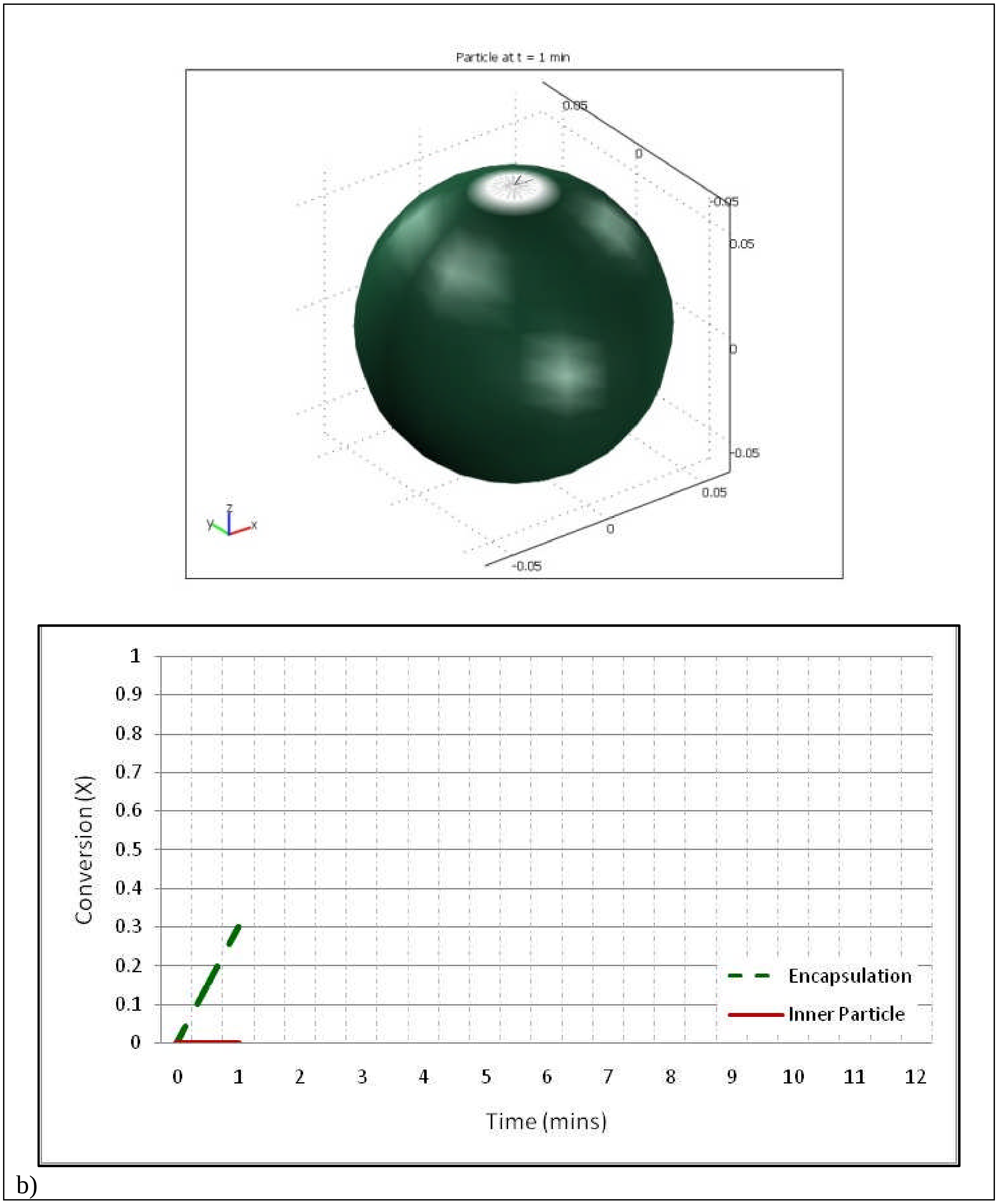


Figure 5.9 (Continued).

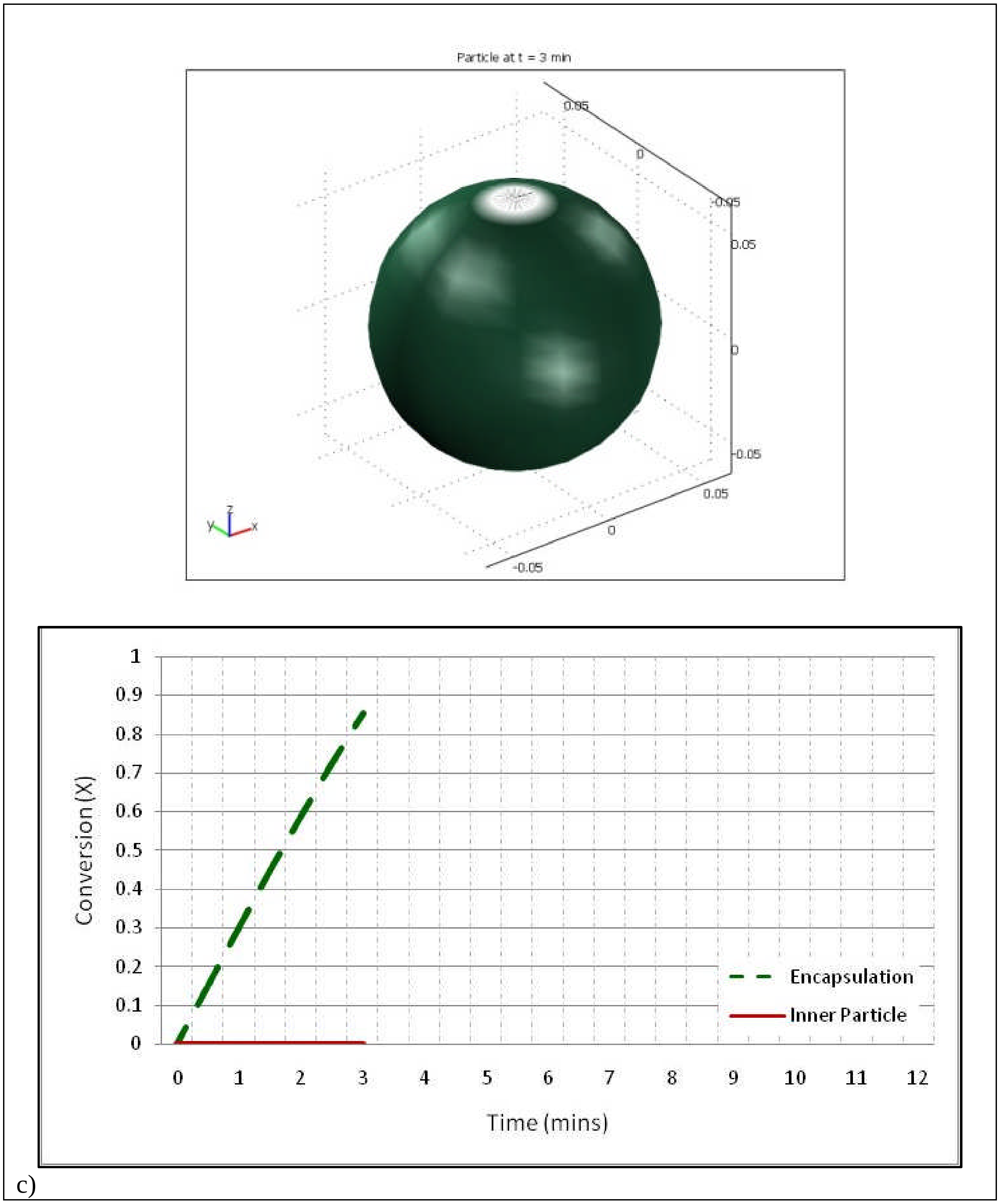


Figure 5.9 (Continued).

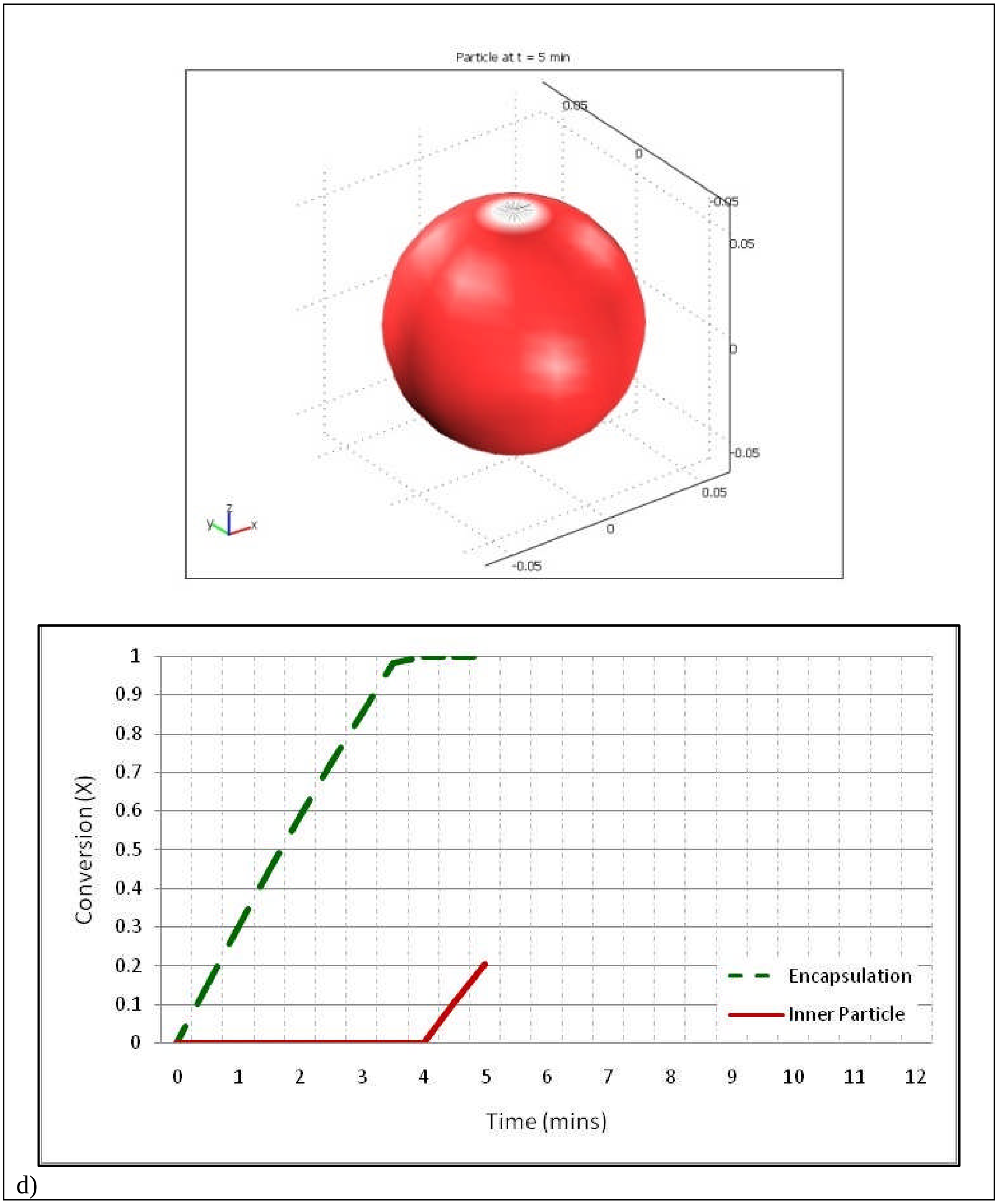
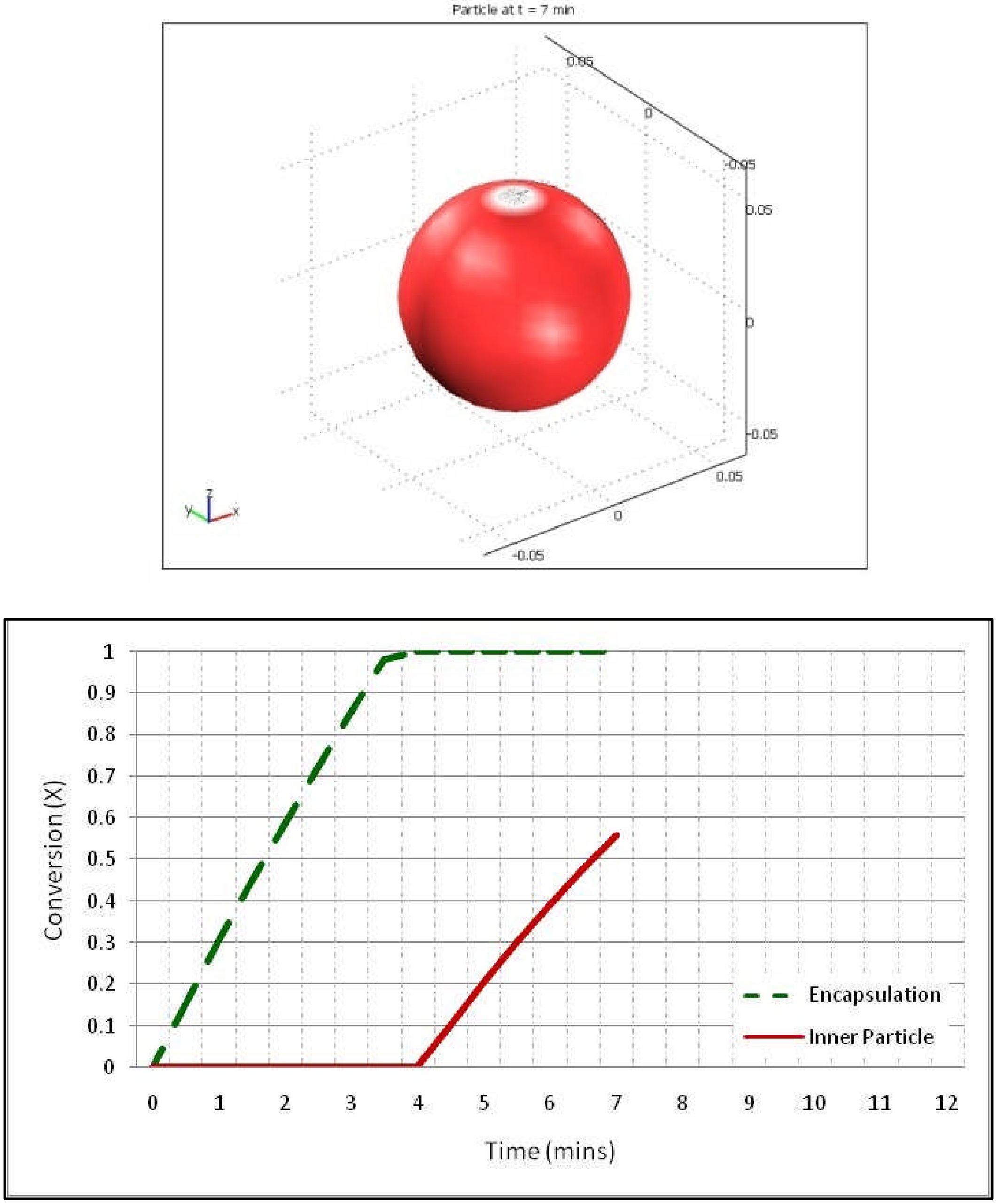


Figure 5.9 (Continued).



e)

Figure 5.9 (Continued).

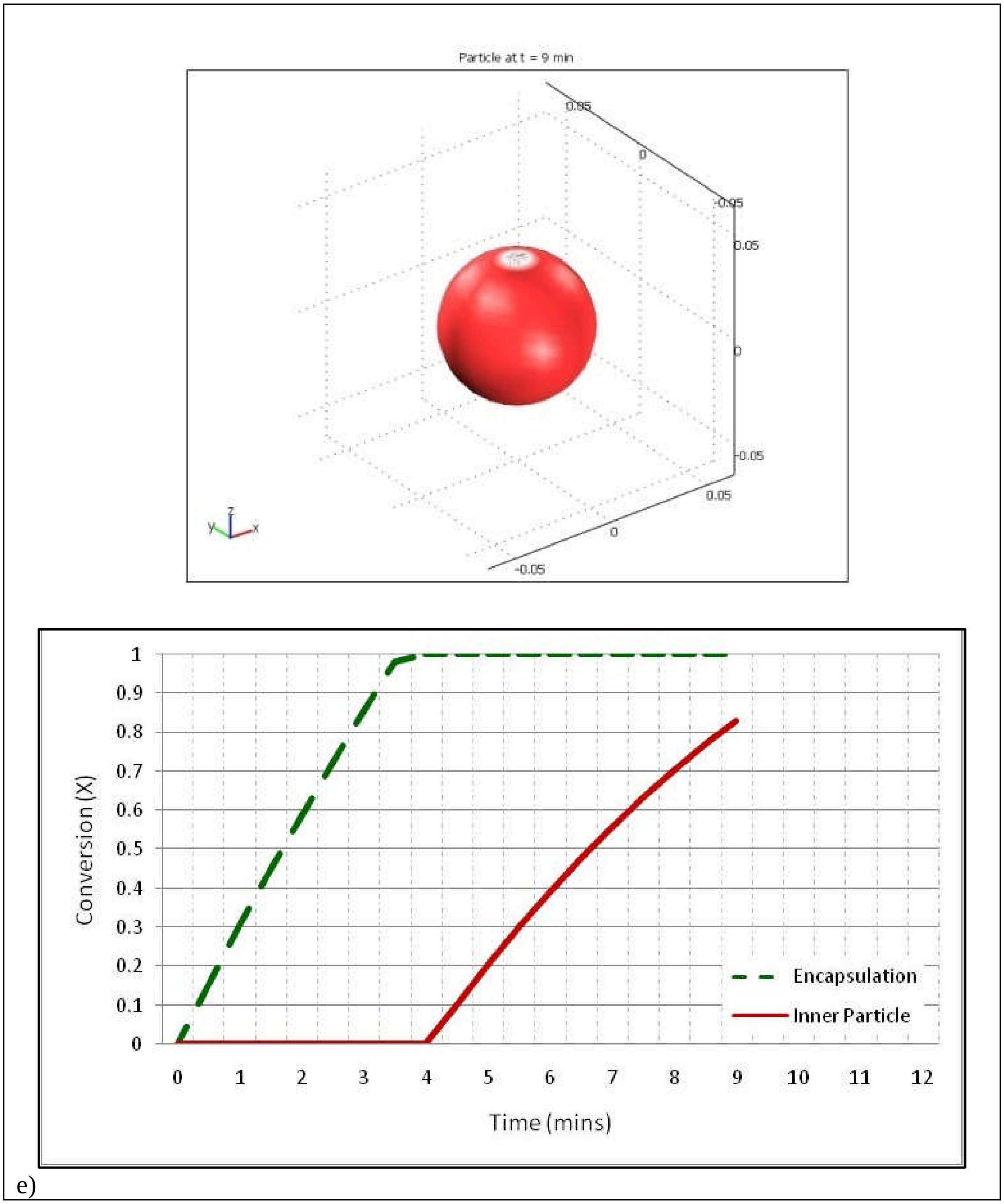


Figure 5.9 (Continued).

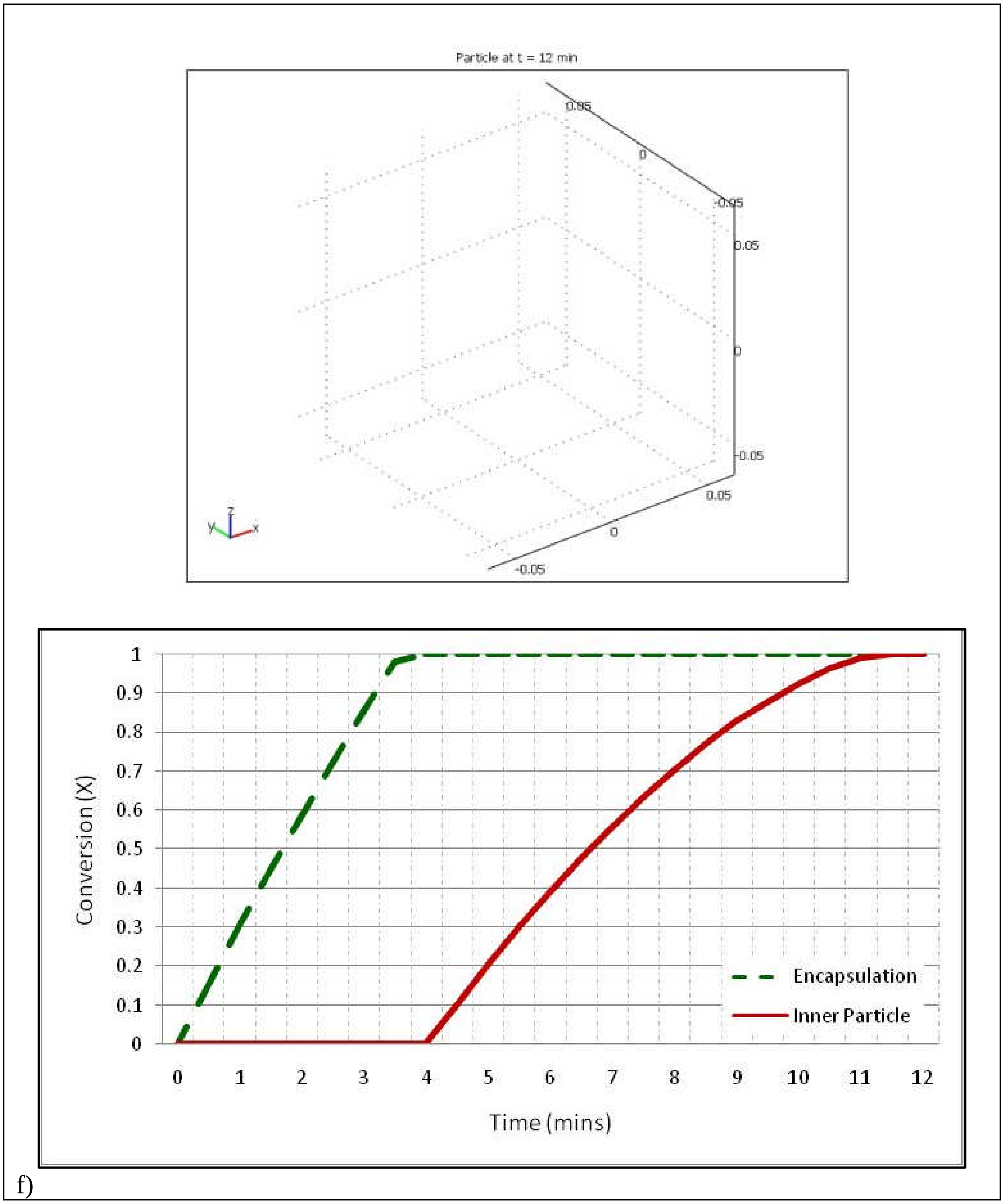


Figure 5.9 (Continued).

5.2.4 Encapsulation Thickness Effect

The thickness of the encapsulation can be modified to fit a desired release profile. To study the effect of the glucose thickness on the delay of citric acid release, five thickness between 0.0025-0.02 cm were tested. The amount of glucose required for each test was also calculated and could be used in determining the material demands for a specific project, perhaps aiding the decision of the most economical design.

Table 5.2: Effect of encapsulation thickness on glucose amount and dissolution time.

h (cm)	Glucose (g)	Delay in Dissolution Time of Citric Acid (mins)
0.0025	0.0133	1.5
0.0050	0.0278	2.5
0.0100	0.0602	4.5
0.0150	0.0977	6.5
0.0200	0.1407	9.0

As predicted, the increase in thickness of the encapsulation layer can significantly delay the release of the citric acid. A thin layer of glucose, 0.0025 cm or 25 µm, delays the release of citric acid by 1.5 minutes, while a thicker layer of 0.0200 cm, or 200 µm, delays the release by almost twice that amount, giving a total dissolution time of 9 minutes. The amount of glucose required for this increased thickness is amount 0.13 grams. The cost of this additional material could be used to determine whether the functional design is also an economically favorable choice.

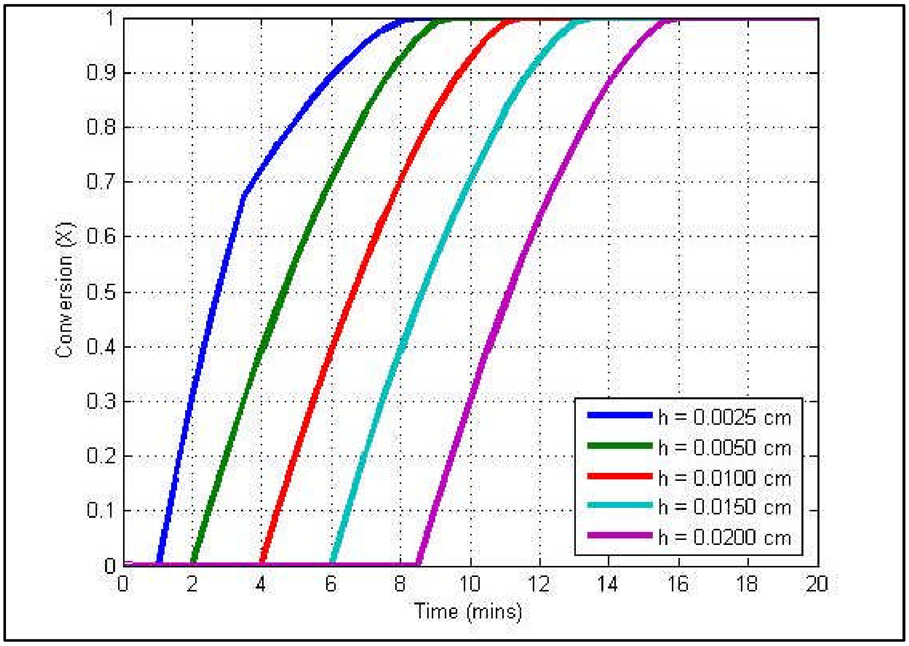


Figure 5.10: Effect of encapsulation thickness on the delayed conversion of citric acid.

Chapter 6 : Encapsulated Polysdisperse Particle Dissolution

As discussed previously, substantial errors can occur when the dissolution of particles is assumed to be monodispersed. The program developed in Chapter 5 for the encapsulated particles can be further improved when multisized encapsulated particles are considered. This was accomplished by combing the technique developed in Chapter 4 for polydispersed particles with the modeling method for encapsulated particle dissolution found in Chapter 5.

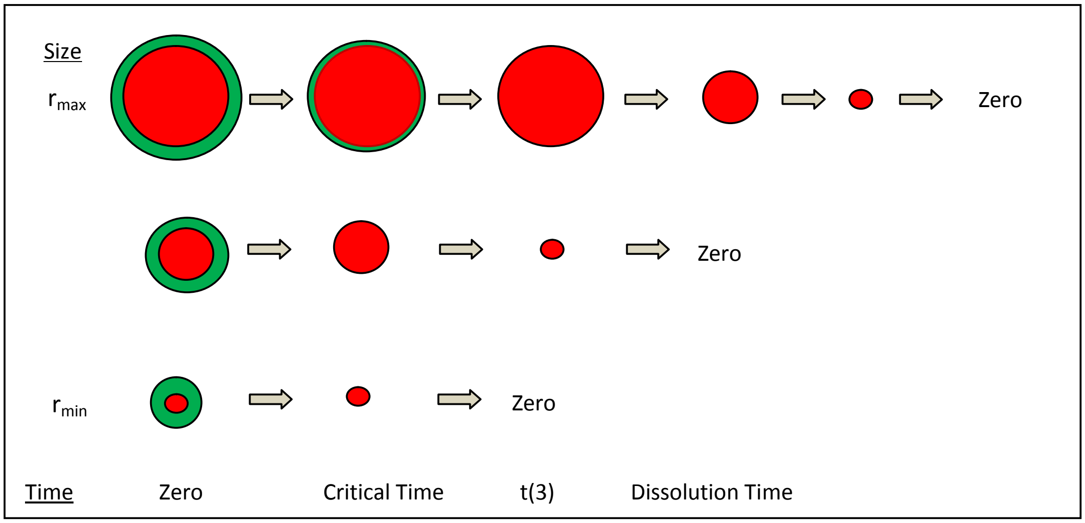


Figure 6.1: Schematic illustration for encapsulated polydisperse particles.

6.1 Program Build Up

The methods for solving multisized particles and encapsulated particles were combined in this design. The program would first be initialized and the diffusion coefficients would be solved for the encapsulation and the inner particle material, following the code developed from Eq. (3.4) and Eq. (5.2). Next, the number of particles in each size group would be determined based on the given percents in each size distribution, similar to those found in Eqs. (4.1 - 4.7). The radius for the encapsulated layer was calculated until all of the coating had eroded. This calculation was

was repeated until all of the size groups. When the encapsulation layer completely eroded, the program would begin calculations for the inner particle. Again, this was repeated for all of the size groups present. Finally, the total concentration and conversion for the inner particle and encapsulation was calculated for all size groups. The concentration and conversion data for encapsulation layer and inner particle were plotted. The program algorithm for this model is shown in Figure 6.2.

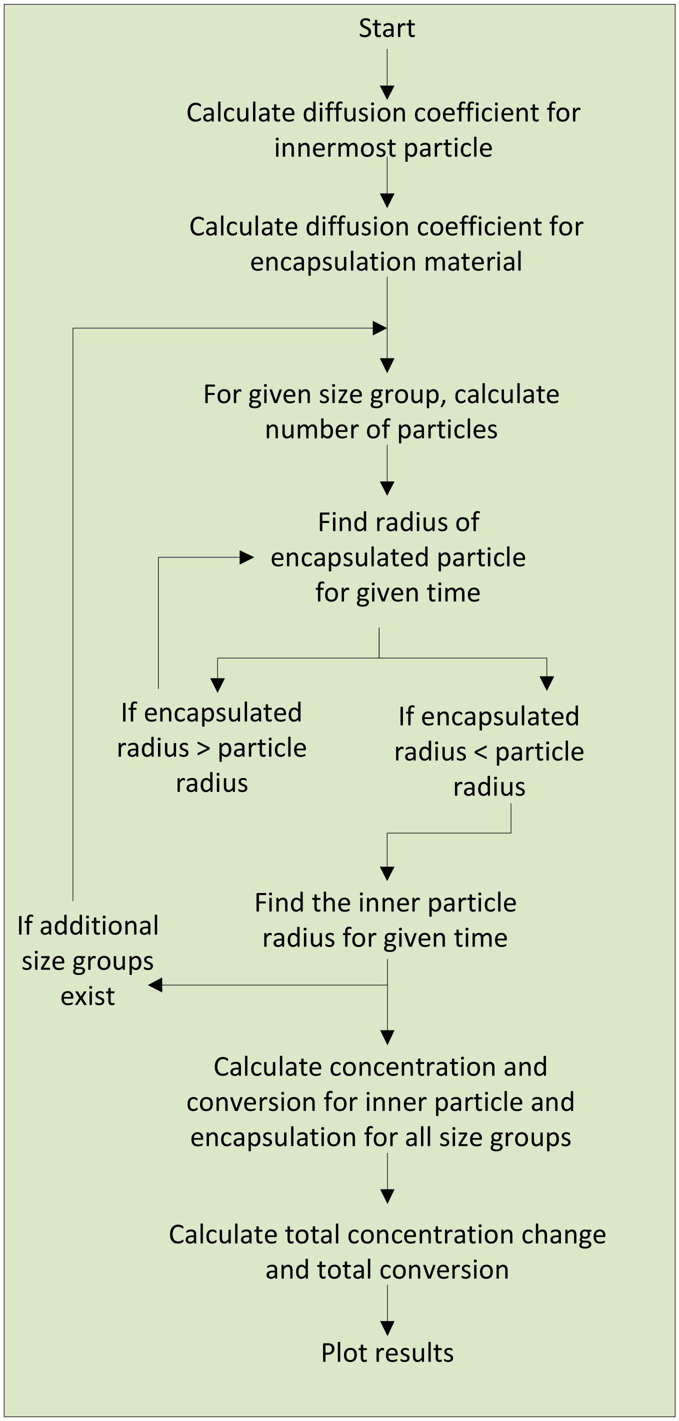

Start
Calculate diffusion coefficient for innermost particle
Calculate diffusion coefficient for encapsulation material
For given size group, calculate number of particles
Find radius of encapsulated particle for given time
If encapsulated radius > particle radius
If encapsulated radius < particle radius
Find the inner particle radius for given time
If additional size groups exist
Calculate concentration and conversion for inner particle and encapsulation for all size groups
Calculate total concentration change and total conversion
Plot results

Figure 6.2: Program algorithm for encapsulated polydisperse particles model.

6.1.1 Calculating Equivalent Number of Particles

The rate of diffusion is dependent on the concentration gradient between the solute and solvent. A technique similar to the one used in Chapter 4, for polydisperse particles, was developed to calculate the equivalent number of particles in each size distribution for a specified concentration. This equivalent number of particles was used to calculate the rate of dissolution based on the total concentration gradient of the system, and then was adjusted to match the actual size distribution.

6.2 Encapsulated Polydisperse Particle Example

For comparison purposes, a simulation for polydisperse citric acid encapsulated with glucose was tested. The parameters for citric acid and glucose were the same as provided in Table 3.1 and Table 5.1, respectively. The size distribution found experimentally from SEM techniques, Figure 4.8, was used in this test run. In addition, the encapsulation layer thickness was set to 0.0010 cm, the same as the test run completed in Chapter 5. The first graph, Figure 6.3, produced by this program shows the concentration changes for each of the size groups. The dashed line represents the encapsulation layer and the solid line represents the inner particle. The encapsulation results can be viewed better by zooming in on the area between 0-30 seconds, shown in Figure 6.4. Since the encapsulation layer is small with respect to the total size of the particle, it has virtually no effect on delaying the release of citric acid.

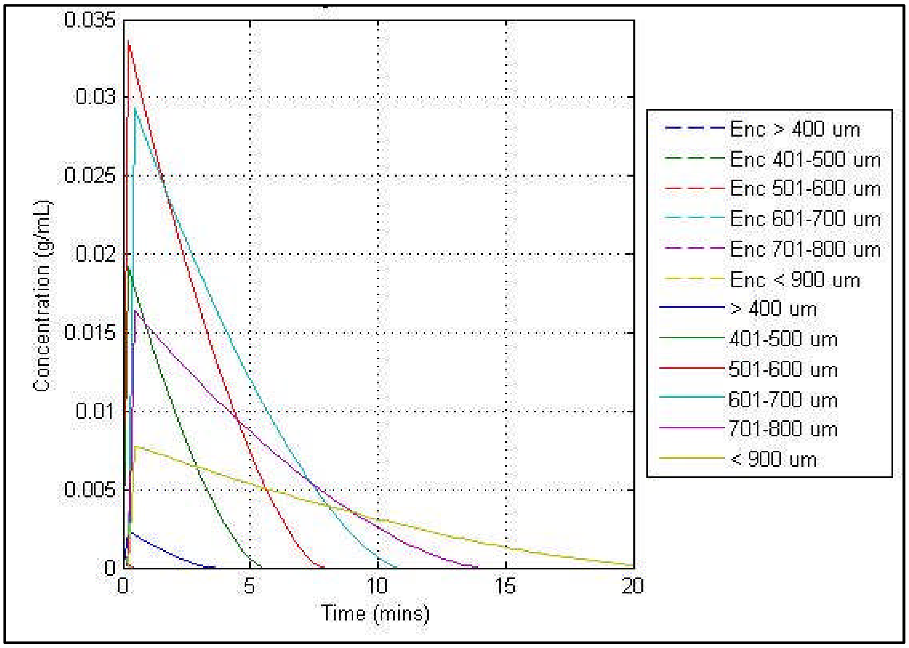


Figure 6.3: Concentration changes for encapsulation and inner particle for all size distributions.

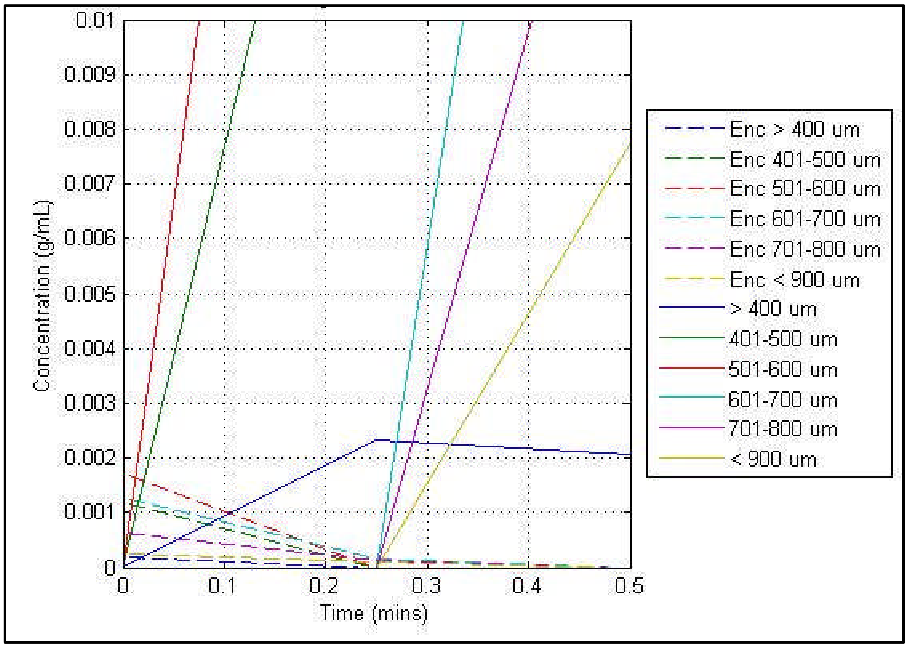


Figure 6.4: Concentration change for various size distributions.

The main difference with the polydisperse encapsulated model can be seen in the time for citric acid to complete dissolution. In the model developed in Chapter 5, total dissolution of citric acid occurred in about 8 minutes. For the polydisperse model, the citric acid took twice the amount of time, about 16 minutes to complete. As would be expected based on the results found in Chapter 4, the dissolution curve for the polydisperse model was less steep and had a much smoother curve.

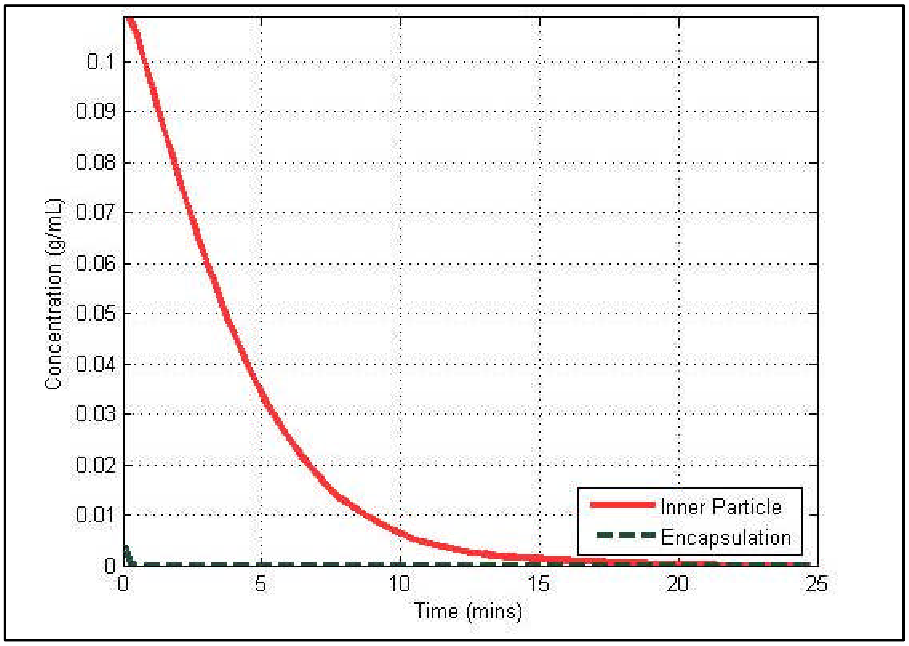


Figure 6.5: Total concentration versus time for polydisperse encapsulated particles.

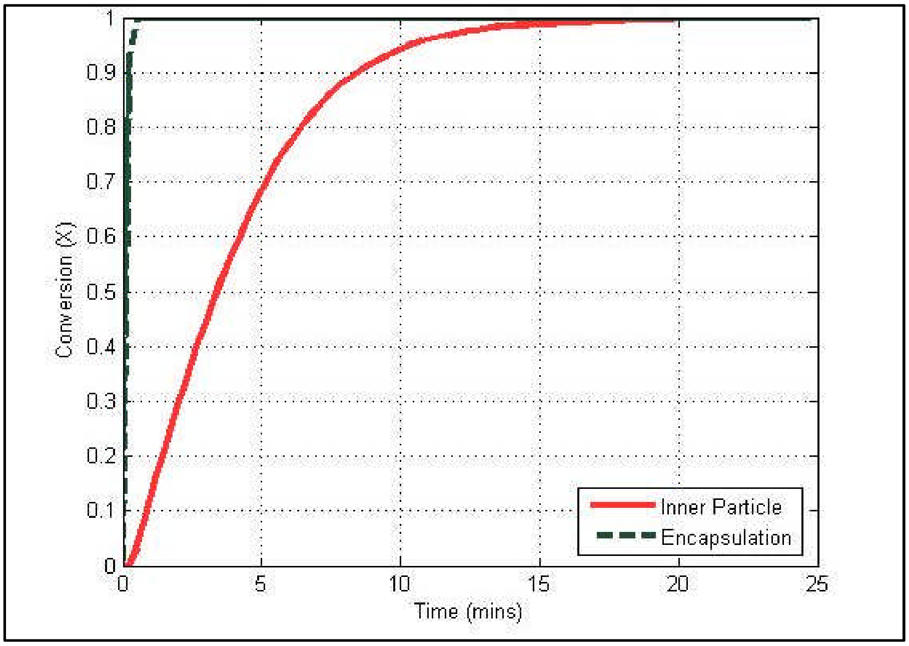


Figure 6.6: Total conversion versus time for polydisperse encapsulated particles.

6.3 Effect of Encapsulation Thickness

For comparison purposes, an increased encapsulation layer thickness was tested to see its effect on the release rate of the citric acid. The glucose layer was increased from 0.0010 cm to 0.01 cm. The graphs are shown in Figure 6.7 and Figure 6.8.

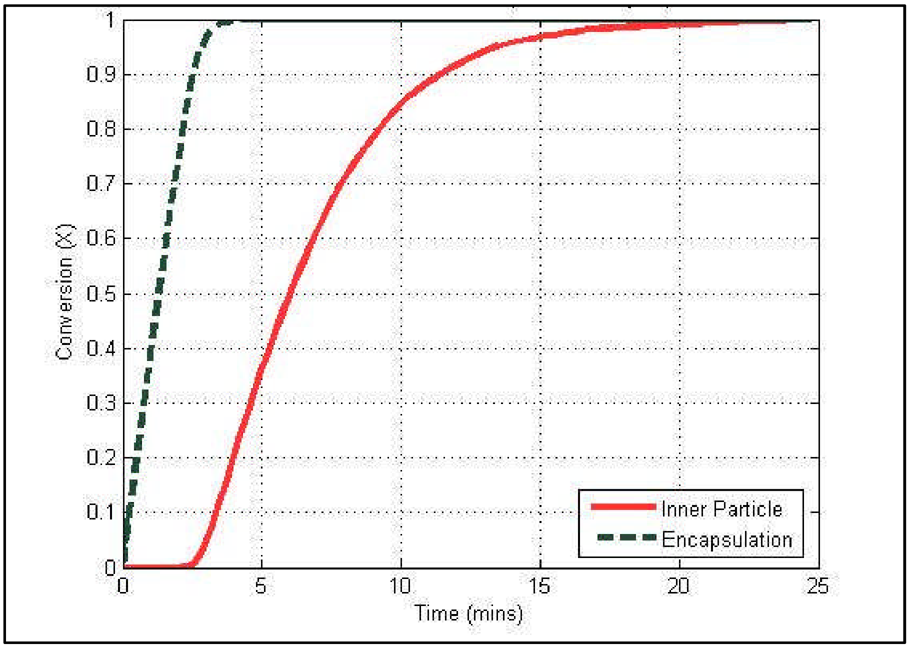


Figure 6.7: Increased encapsulation layer thickness for polydisperse model.

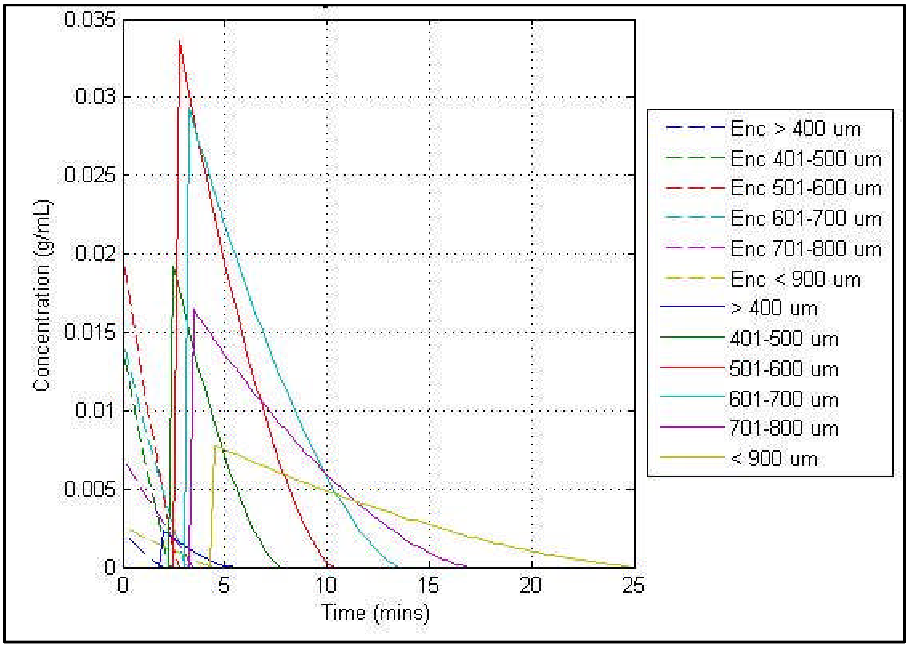


Figure 6.8: Concentration change versus time for all size distributions with increased coating thickness.

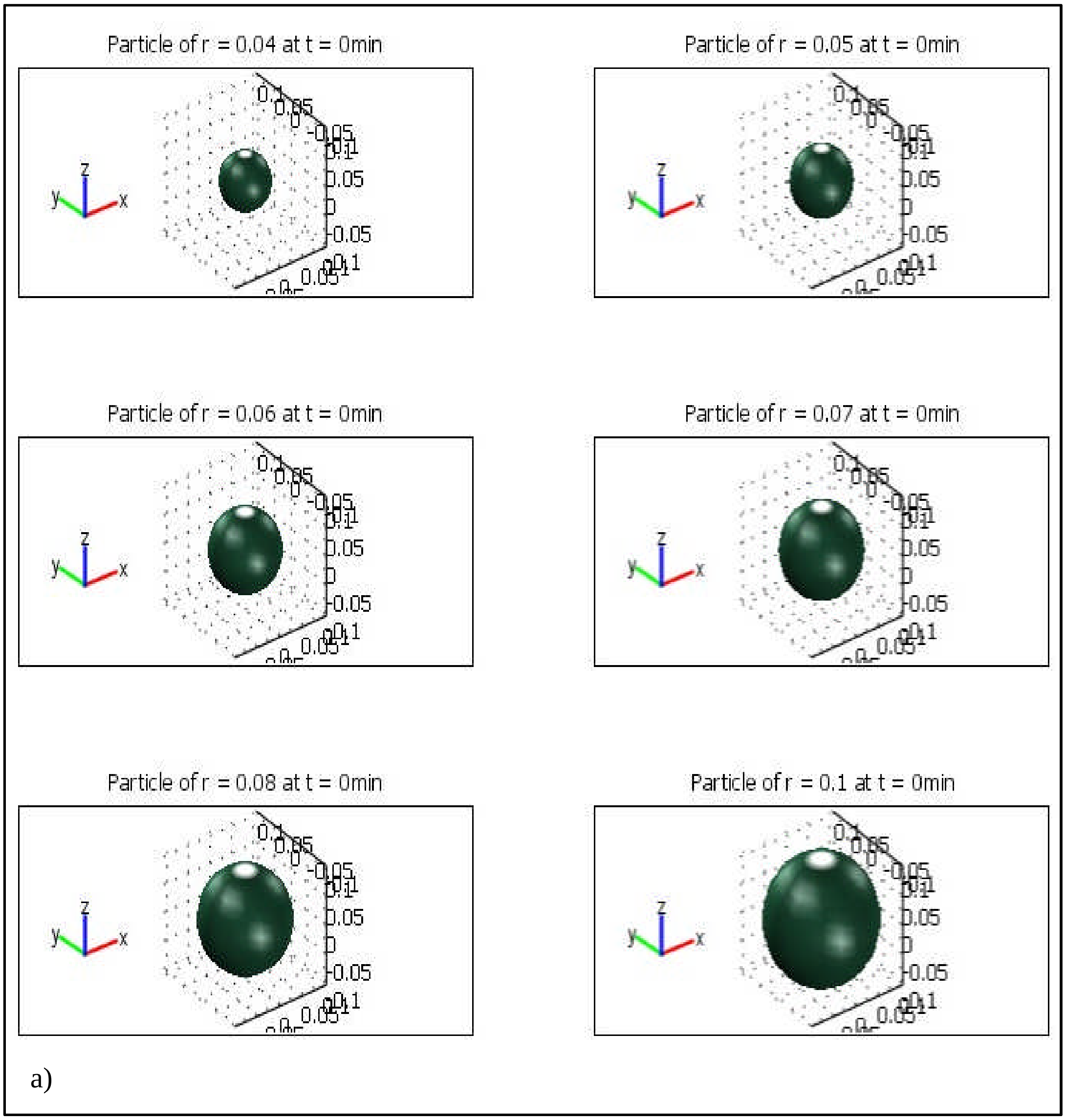

Particle of r = 0.04 at t = 0min
Particle of r = 0.05 at t = 0min
Particle of r = 0.06 at t = 0min
Particle of r = 0.07 at t = 0min
Particle of r = 0.08 at t = 0min
Particle of r = 0.1 at t = 0min
x
y
z
0.1
0.05
0
-0.05
-0.1
a)

Figure 6.9: Encapsulated polydisperse particle dissolution for a) t = 0 mins, b) t = 2 mins, c) t = 4 mins, d) t = 6 mins, e) t = 8 mins, f) t = 10 mins and g) t = 12 mins.

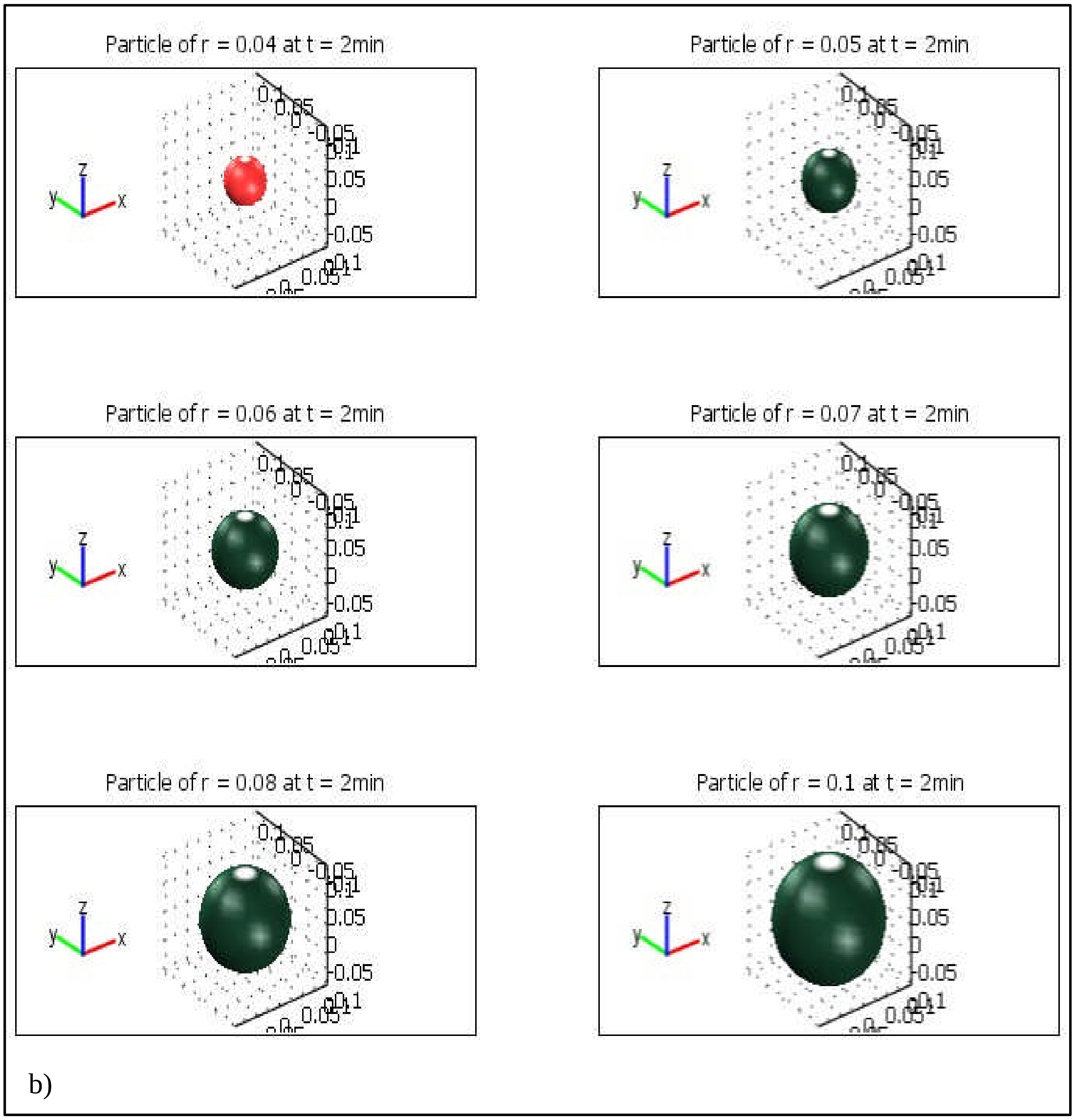
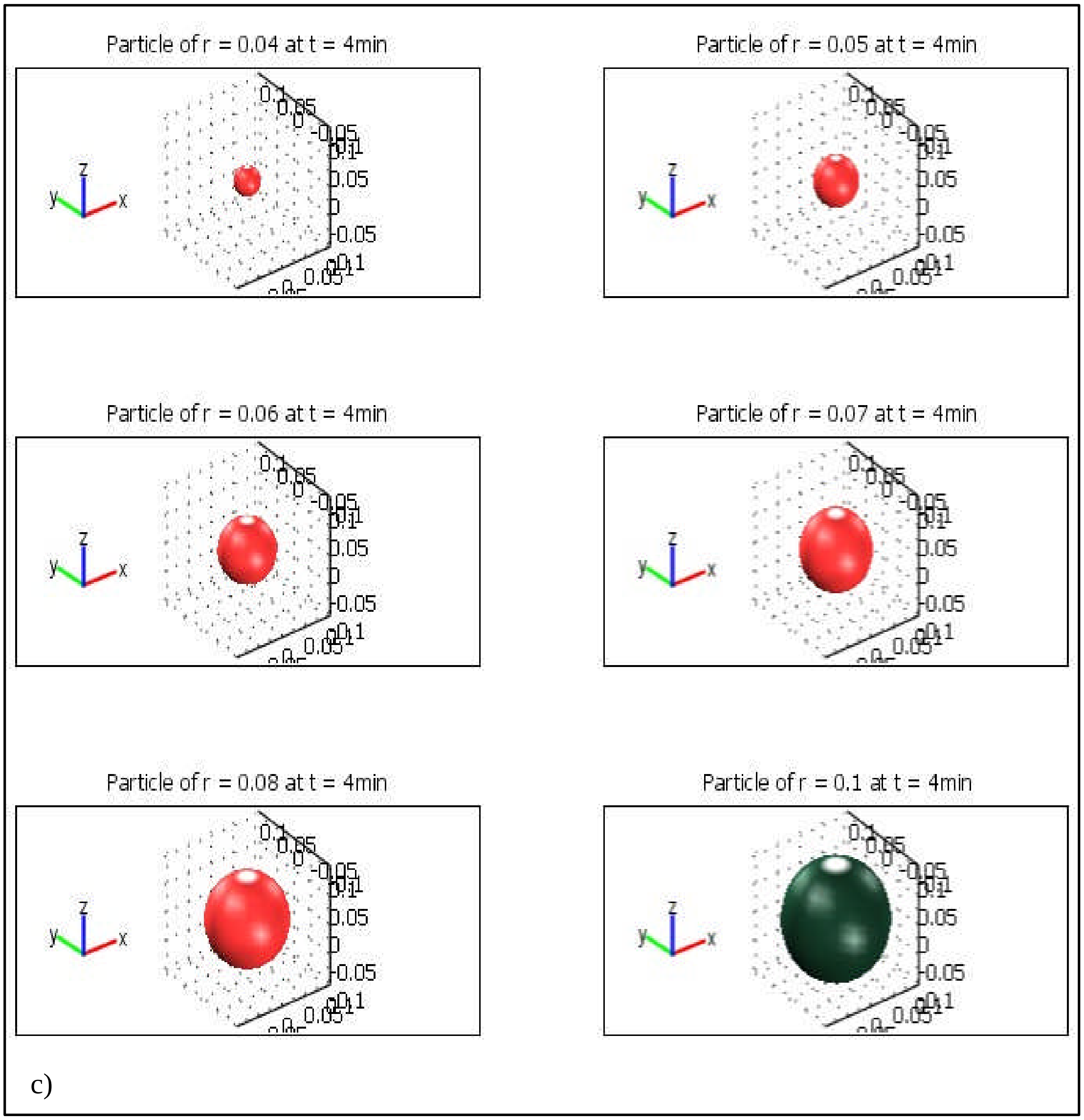

Particle of r = 0.04 at t = 2min
Particle of r = 0.05 at t = 2min
Particle of r = 0.06 at t = 2min
Particle of r = 0.07 at t = 2min
Particle of r = 0.08 at t = 2min
Particle of r = 0.1 at t = 2min
z
y
x
0.1
0.05
0
-0.05
-0.1
b)

Figure 6.9 (Continued).


Particle of r = 0.04 at t = 4min

Particle of r = 0.05 at t = 4min

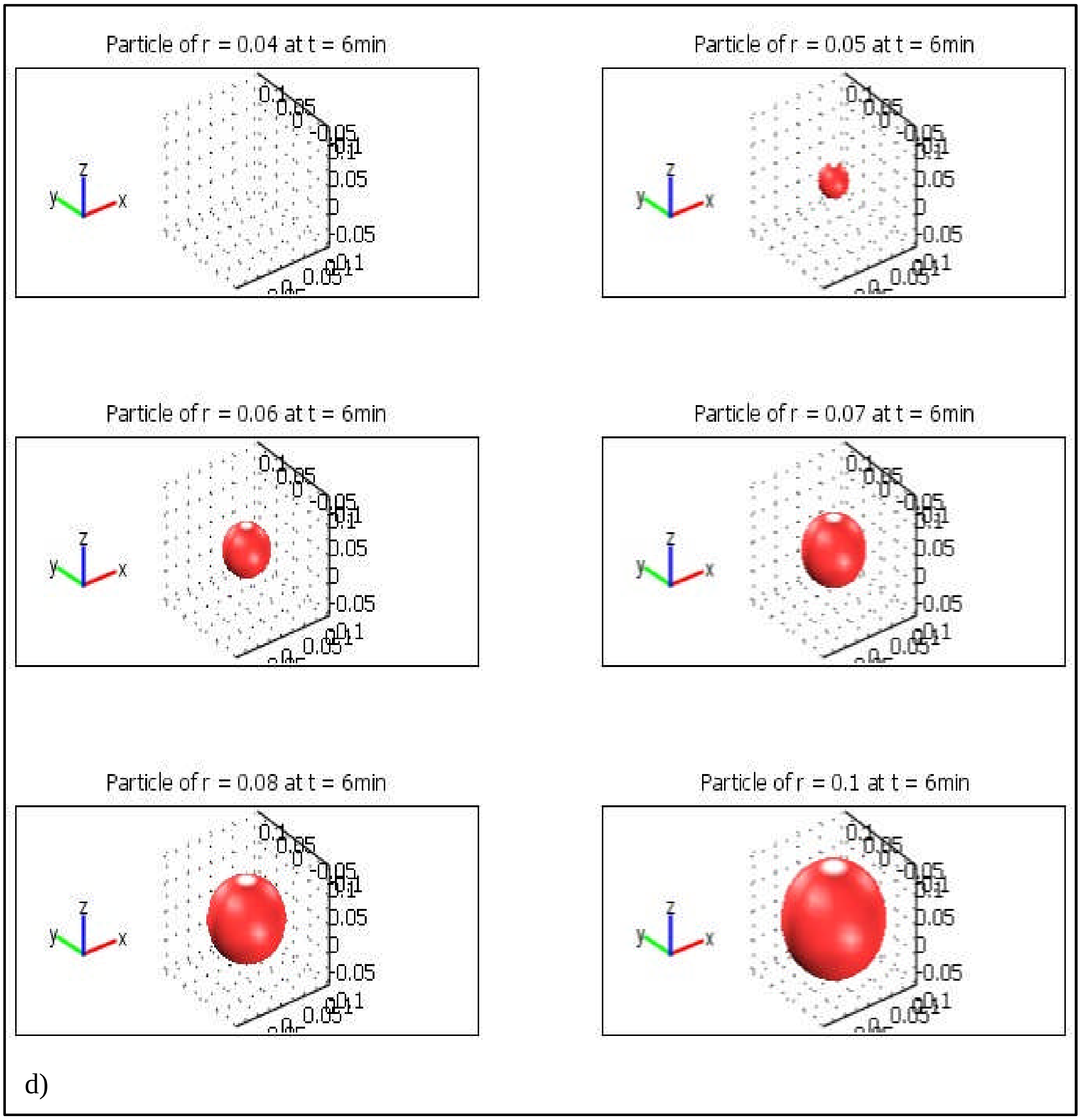
Particle of r = 0.06 at t = 4min

Particle of r = 0.07 at t = 4min

Particle of r = 0.08 at t = 4min

Particle of r = 0.1 at t = 4min

Figure 6.9 (Continued).



Particle of r = 0.04 at t = 6min

Particle of r = 0.05 at t = 6min

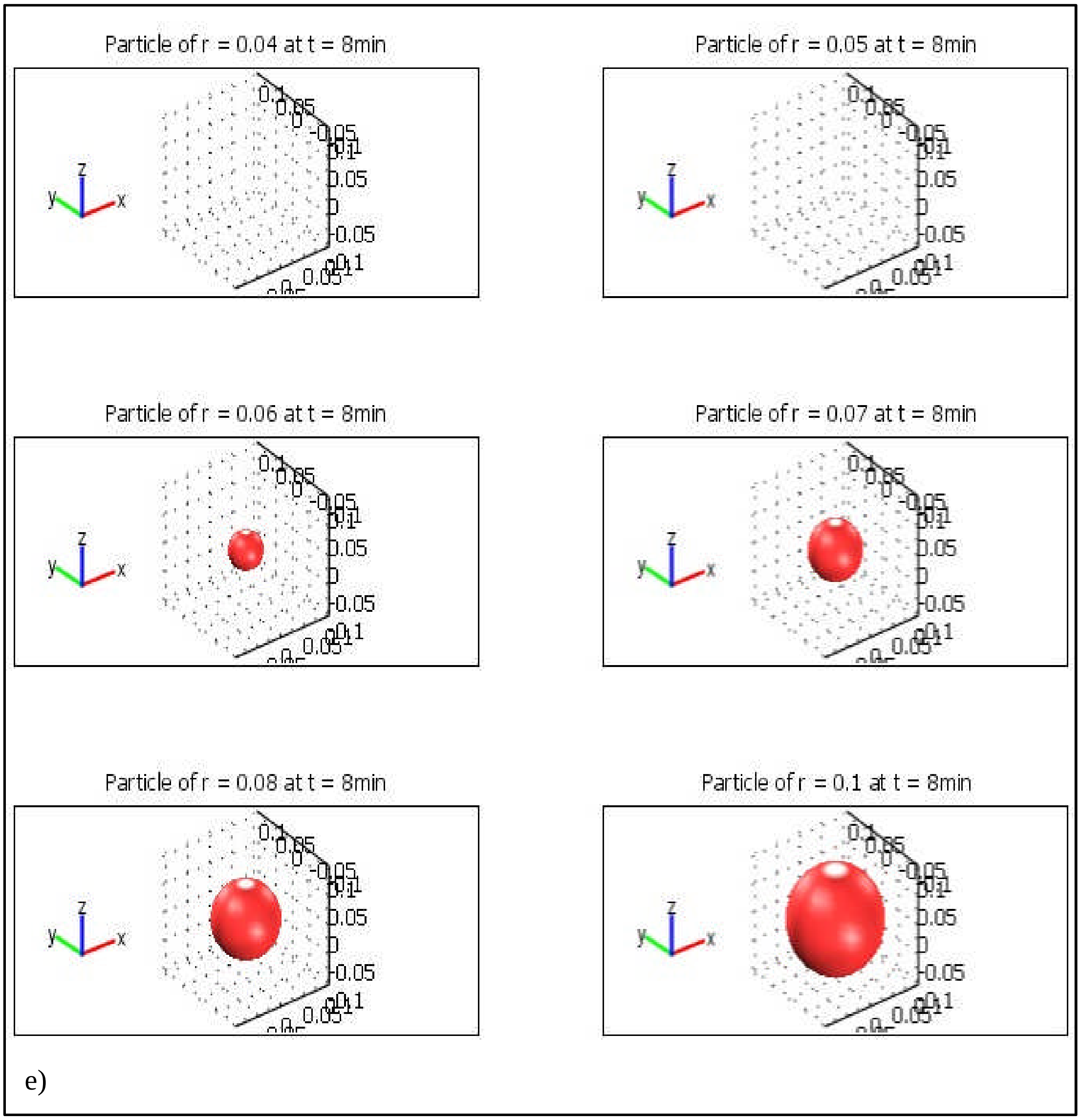
Particle of r = 0.06 at t = 6min

Particle of r = 0.07 at t = 6min

Particle of r = 0.08 at t = 6min

Particle of r = 0.1 at t = 6min

Figure 6.9 (Continued).


Particle of r = 0.04 at t = 8min

Particle of r = 0.05 at t = 8min

Particle of r = 0.06 at t = 8min

Particle of r = 0.07 at t = 8min

Particle of r = 0.08 at t = 8min

Particle of r = 0.1 at t = 8min

Figure 6.9 (Continued).

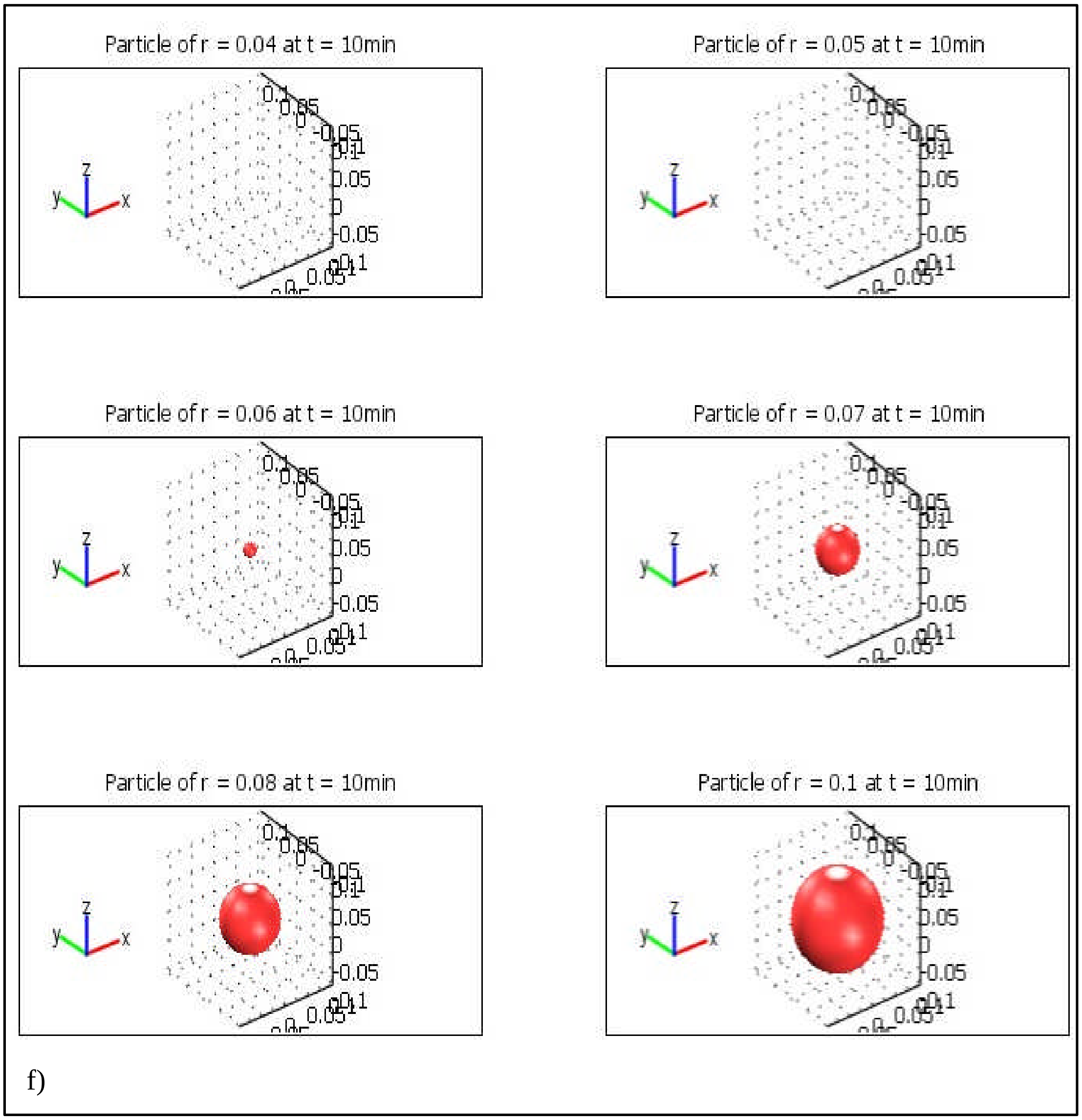
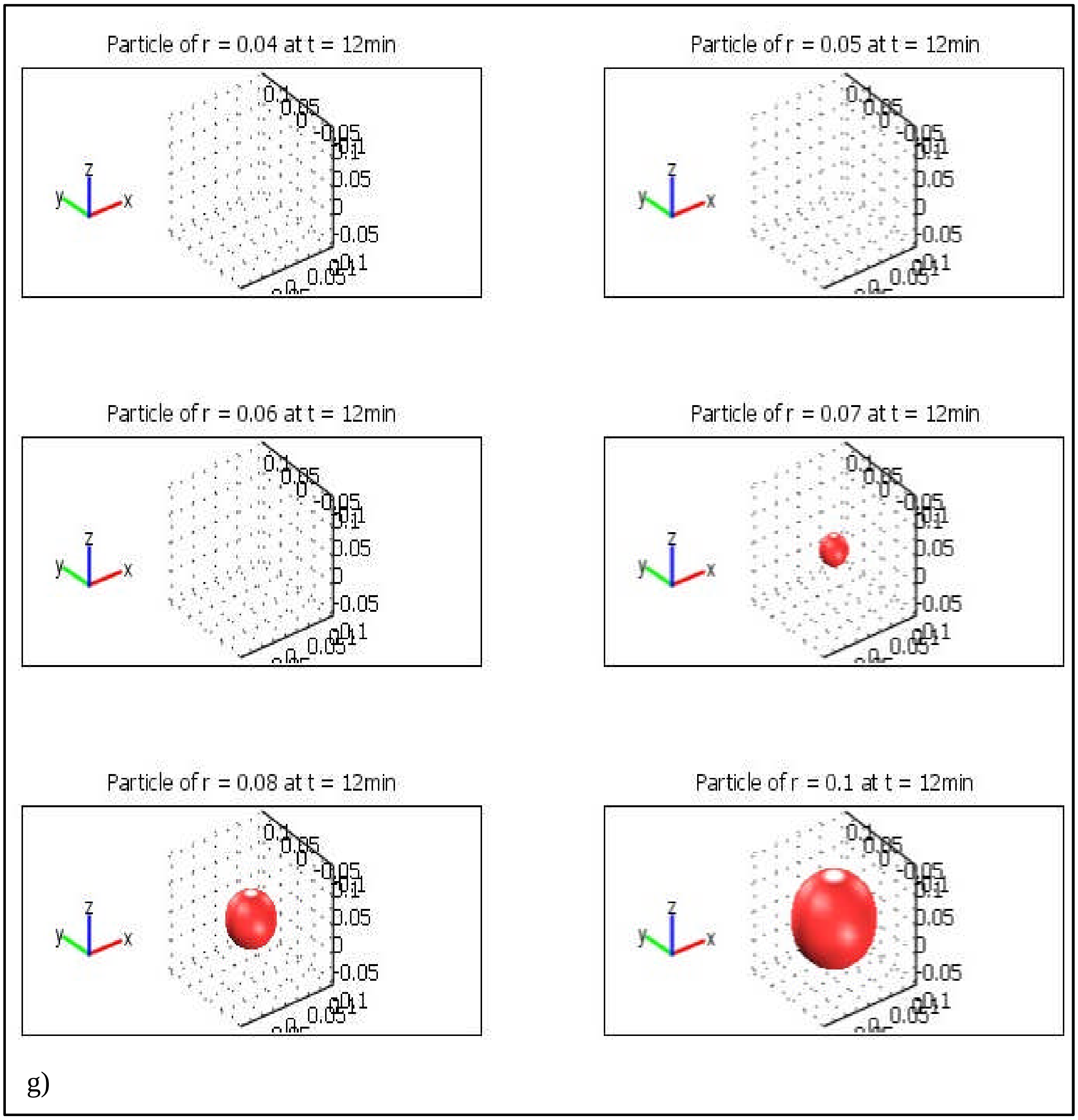

Particle of r = 0.04 at t = 10min
Particle of r = 0.05 at t = 10min
Particle of r = 0.06 at t = 10min
Particle of r = 0.07 at t = 10min
Particle of r = 0.08 at t = 10min
Particle of r = 0.1 at t = 10min
x
y
z
0.1
0.05
0
-0.05
-0.1
f)

Figure 6.9 (Continued).



Particle of r = 0.04 at t = 12min

Particle of r = 0.05 at t = 12min

Particle of r = 0.06 at t = 12min

Particle of r = 0.07 at t = 12min

Particle of r = 0.08 at t = 12min

Particle of r = 0.1 at t = 12min

Figure 6.9 (Continued).

Chapter 7 : Results and Discussion

The previous chapters outlined the development of a comprehensive computer code to aid in estimating the dissolution rates of substances for controlled release applications. While the long term goals of such a program would be to eliminate unnecessary experimental testing, the initial validation of the models must be completed by comparing the model results to those found in experiments. This chapter is dedicated to examining the results found experimentally with the simulated results found from Chapters 3-6.

Dissolution tests of citric acid in water were performed experimentally to validate the results found in the simulation. An OmniCal Technologies SuperCRC 20-305-2.4 reaction microcalorimeter was used was used to measure the conversion versus time. The process for experimentally measuring the conversion with the use of a reaction microcalorimeter is outlined in the following section.

7.1 Experimental Validation

Reaction heat flow calorimetery has been used in many technical fields from drug discovery to process development [50-55]. Microcalorimeters are used to monitor the change in heat which accompanies the chemical and physical processes as they undergo mechanisms such as adsorption, dilution, dissolution, mixing or chemical reactions. Microcalorimeters measure the heat quantity of a chemical process along with its time derivative, heat flow. Heat flow is also proportional to the reaction rate and can be used to calculate a other reaction kinetic parameters.

The heat flow, q, measured during an experiment is proportional to the reaction rate. r. where ΔH_{rxn} is the heat of the reaction and V is the volume.

$$q = \Delta H_{rxn} \cdot V \cdot r \tag{11.1}$$

The integration of the observed heat flow versus time curves yields the heat of reaction. The fractional conversion can alo be obtained by using the calculated fractional area under the temporal heat flow curve. The following equation can be used to calculate the fractional conversion

$$X = \frac{\int_0^t q \cdot dt}{\int_0^f q \cdot dt} \tag{11.2}$$

The numerator is found by calculating the are under the heat flow curve to any time point t and the denominator is the total area under the heat flow curve. The heat evolved and the calorimetric response can be found by monitoring the reaction progress and measuring the enthalpic changes of the chemical reactions.

7.1.1 Reaction Microcalorimeter Experiments

Experiments were conducted in a OmniCal Technologies SuperCRC 20-305-2.4 reaction microcalorimeter, as shown in Figure 7.1 [51]. The reaction microcalorimeter uses a differential scanning calorimeter (DSC) technique to measure the heat released or consumed in a sample vessel compared to an empty reference vessel.

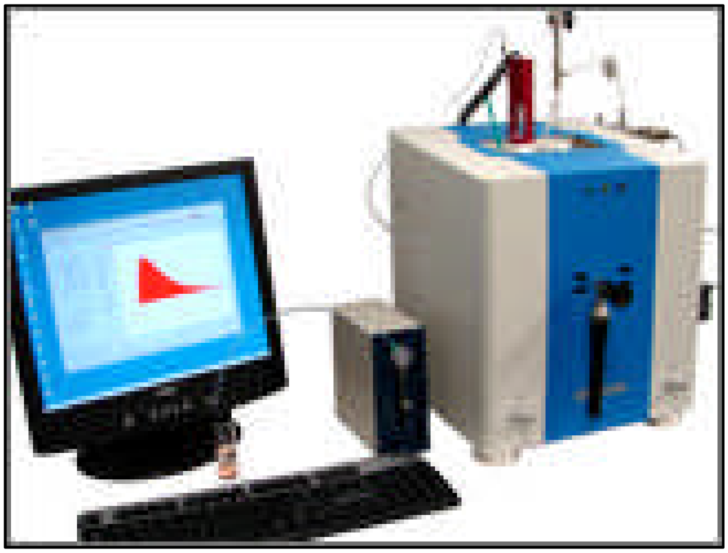
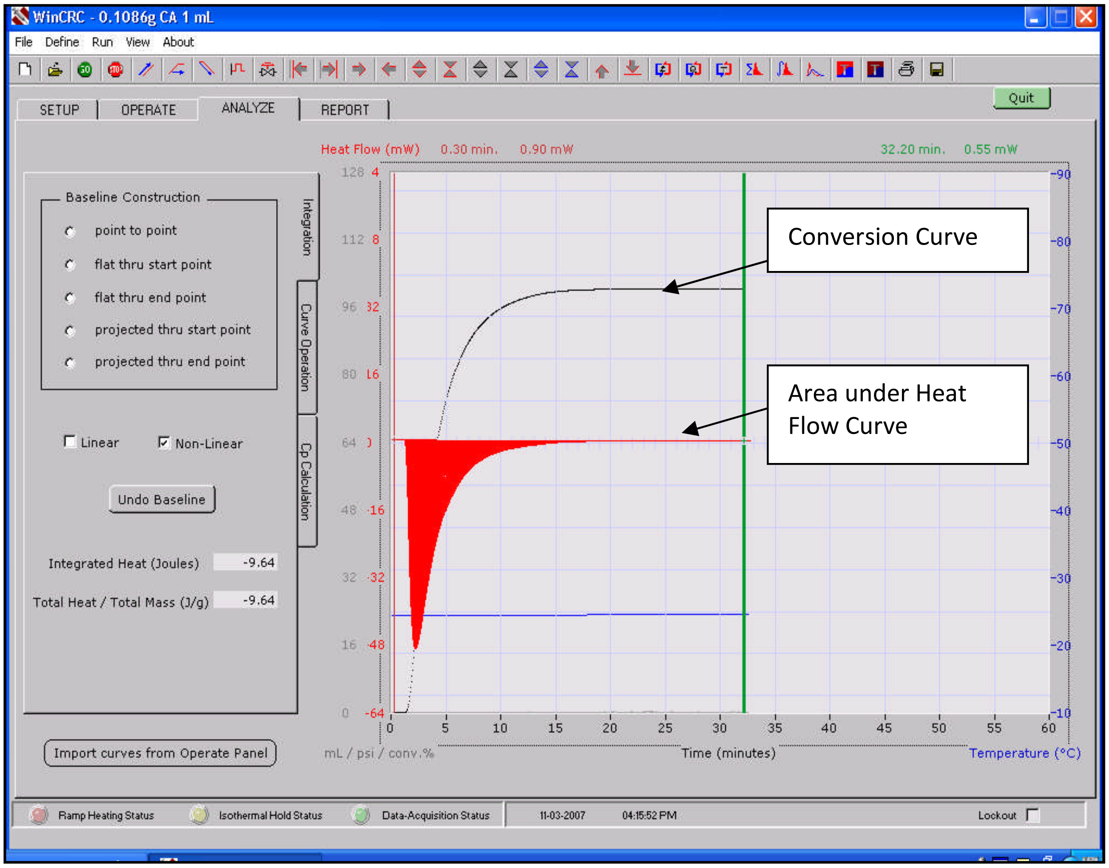


Figure 7.1: OmniCal SuperCRC reaction microcalorimeter.

Reaction vessels were 16 mL screw-fit thread glass vials fit with Teflon-lined screw-caps. The desired amount of anhydrous citric acid, obtained from Fisher Chemicals was measured into the glass vial and was sealed. Both the vessel containing the citric acid and the empty reference vessel were placed in the microcalorimeter for 35 minutes, allowing the system to reach thermal equilibrium. Simultaneously, two liquid samples of 1 mL of water were measured into two 5 mL syringes and placed into the sample barrels in the calorimeter and were allowed to thermally equilibrate.

The microcalorimeter unit is turned on and the samples and syringe barrels were allowed to come to thermal equilibrium with the calorimeter heat sink temperature. The microcalorimeter operates on-line with the program software WinCRC Turbo, which is executed after the samples have been inserted into the unit and the unit is turned on. WinCRC Turbo was used for both data acquisition and conversion calculations. Samples were monitored until a smooth baseline in the heat flow was maintained. The reaction was initiated by injecting the water into the vessels while the temperature of the microcalorimeter was held at room temperature, approximately 25°C. The internal thermal controller in the microcalorimeter held the temperature constant throughout the process ensuring that the experiment was run under isothermal conditions. The detection and collection of data was maintained until no changes were seen in the heat flow for at least 30 minutes. A heat flow curve was produced by measuring the heat flow from the reaction vessel in increments of 3 samples per second, as seen in Figure 7.2. The data collected would then need to be calibrated to account for the delay between the instantaneous heat flow evolved and the time the thermopile sensor is able to detect the heat flow.

The process of calibration involved passing a known quantity of heat, produced by passing a known current through a resistor, into the sample chamber of the calorimeter. The WinCRC software then records the heat response curve. This curve is then transformed into a square wave using the software, which allows for the response time of the microcalorimeter to be calculated. Using the calculated response time, the software can then readjust the heat flow curve data from the experimental reaction to account for the delays from the sensors. The new graph is saved as a tau corrected graph. The tau corrected graph is used to calculate the fractional conversion using the Eq. (7.2).



Baseline Construction

- ○ point to point
- ○ flat thru start point
- ○ flat thru end point
- ○ projected thru start point
- ○ projected thru end point

□ Linear ☑ Non-Linear

Integrated Heat (Joules) -9.64

Total Heat / Total Mass (J/g) -9.64

11-03-2007 04:15:52 PM Lockout □

Figure 7.2: Reaction microcalorimeter heat flow and conversion graph.

The first experiment for citric acid was conducted at the same conditions as listed in Table 3.1. The results of the simulation are shown in Figure 7.3. The experimental results do not seem to fit any of the models for monodisperse particles with one given average radius. Instead, the experimental data seem to be in between the results found for the average radius sizes of range 0.05 to 0.10 cm. During the initial minutes of dissolution, the experiment closely follows the model for the radius size of 0.05 and 0.06 cm. However, as time progresses, the time for total dissolution is delayed and the curve appears to match the results for large particle radius sizes, greater than 0.08 cm. This suggests that the particles used in the experiment include a range of particle sizes, which is indeed the case, as was discovered using the SEM.

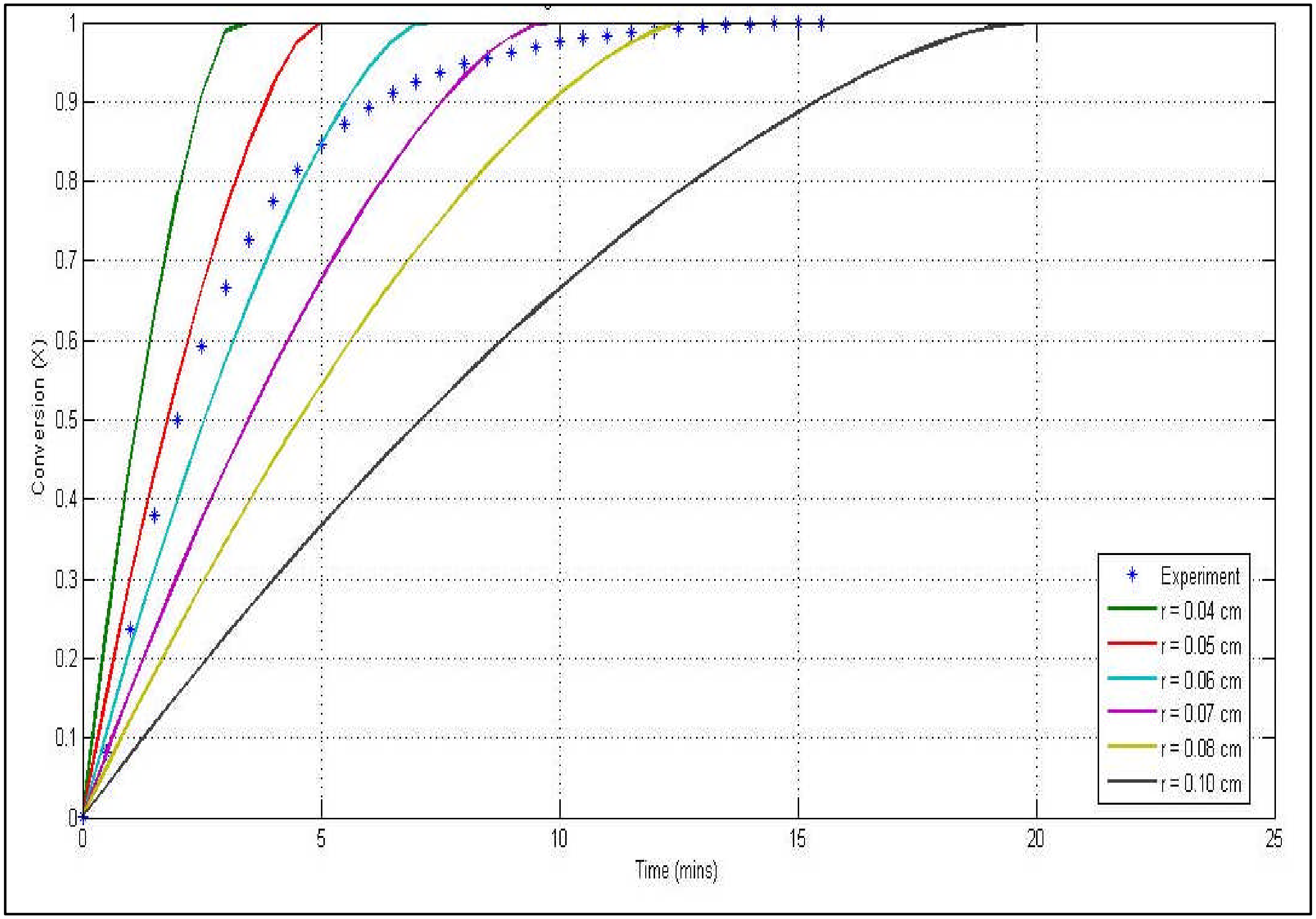


Figure 7.3: Experimental data versus monodisperse model for varying initial radii.

After establishing that the particles included a distribution of radius sizes, the experimental results were compared to the model developed for polydisperse particles. The distribution found through SEM, Figure 4.8, was inputted into the program for polydisperse particles and the results were graphed along with the experimental data in Figure 7.4. For this concentration, 0.10g citric acid in 1 mL water, the results from the polydisperse model provided a good estimate of the dissolution time. Both the experiment and polydisperse model had an approximate dissolution time of 15 minutes. The polydisperse model appeared to be a better fit for the experimental data provided. For a more detailed examination, the experimental data was plotted against both models in Figure 7.5. The noticeable difference between the models is the final dissolution time. The polydisperse model is twice as long as predicted with the monodisperse model and appears to have the same curve shape as found experimentally.

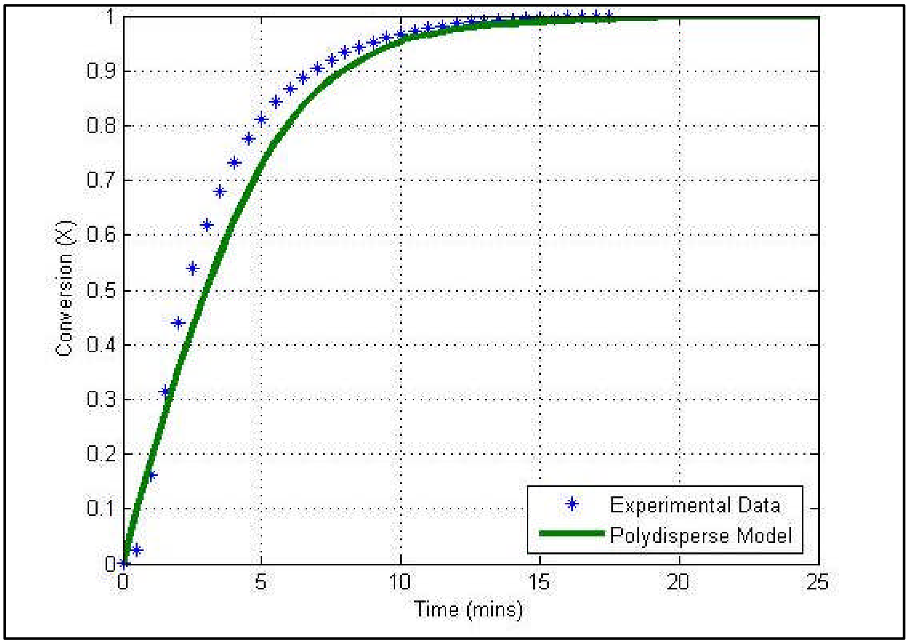


Figure 7.4: Experimental data versus polydisperse model for 0.10 grams of citric acid.

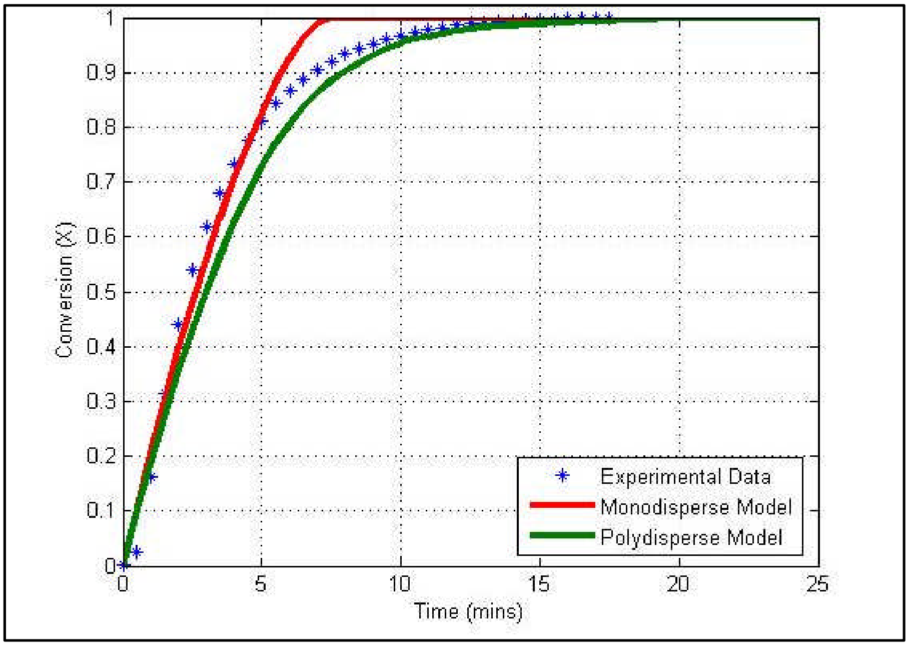


Figure 7.5: Comparison of experimental data for 0.10 grams citric acid with both models.

7.1.2 Additional Experimental Tests

Following the sample run of 0.01 grams of citric acid in 1 mL of water, several additional tests were done for concentrations below and above this amount. The results showed the limits on the accuracy of the models for concentrations in the extreme ranges, from dilute to more concentrated solutions. The possible explanations for the discrepancies of the models are investigated later in this chapter.

A total of nine experiments were conducted for citric acid weights of 0.02, 0.04, 0.08, 0.10, 0.20, 0.30, 0.40 and 0.50 grams in 1 mL of water at approximately 298 Kelvin. The experimental results were plotted in Figure 7.6. In general, increasing in the amount of citric acid added increased the total dissolution time of the particles.

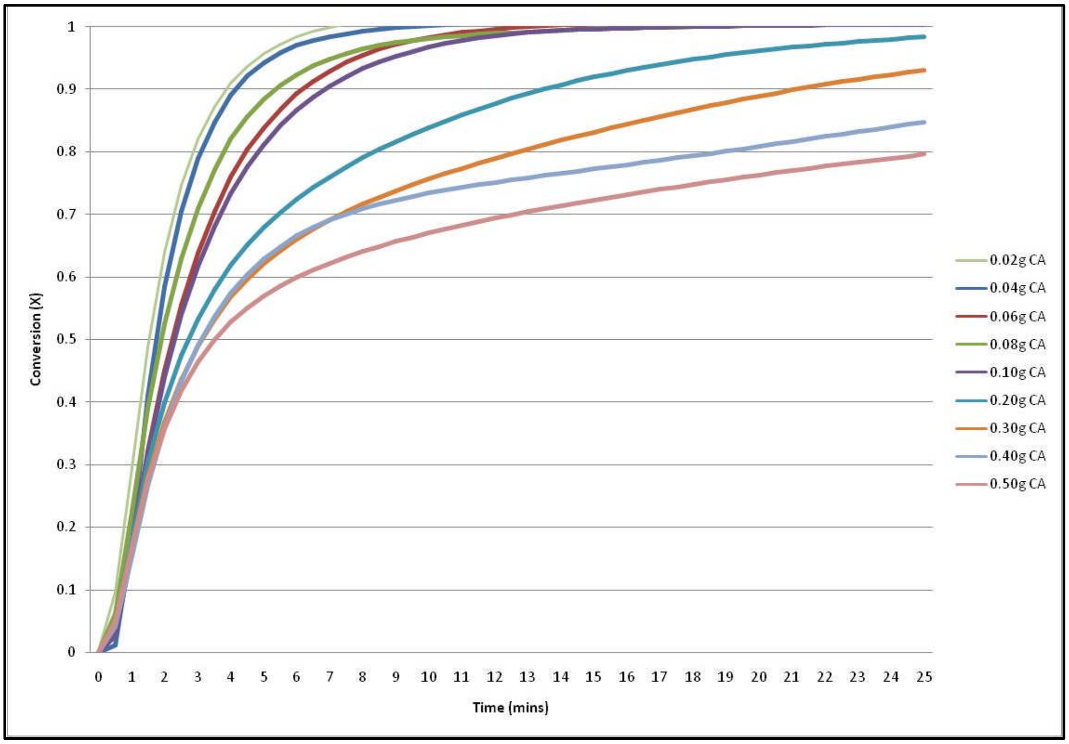


Figure 7.6: Experimental results for different amounts of citric acid.

Next, graphs comparing each of the experimental concentrations with the corresponding results found from the monodisperse and polydisperse models were plotted. These graph were produced in a similar manner to the one produced for the first sample of 0.10 grams of citric acid. The values for the experimental data for six sample data sets for concentrations of 0.2-0.20 grams of citric acid are given in Appendix E. The different concentrations were also tested using the monodisperse model and polydisperse model. The residuals between the experimental data and the two models showed that the monodisperse model was more accurate for low concentrations while the polydisperse model more accurately described the dissolution behavior at higher concentrations.

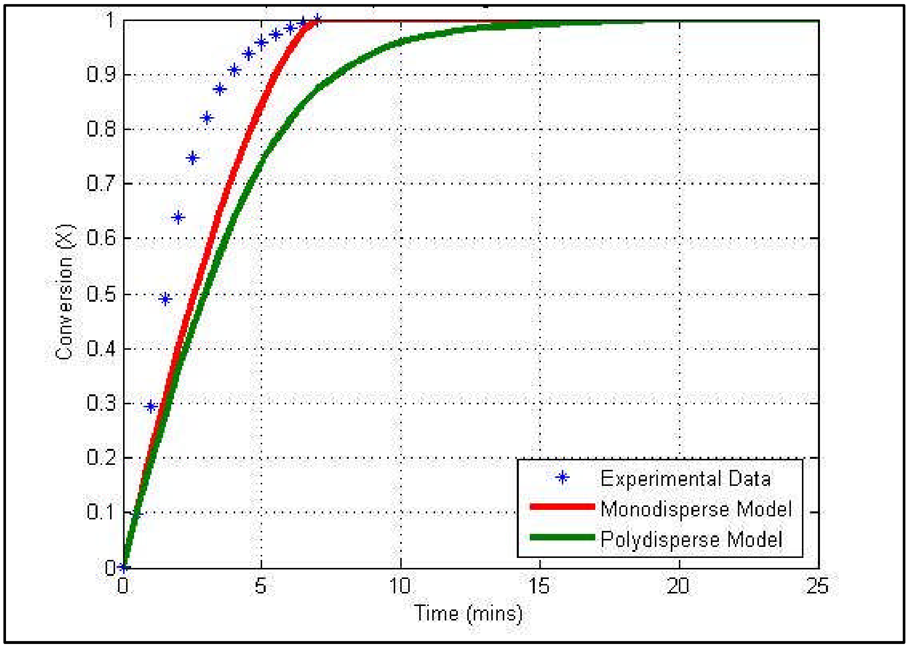


Figure 7.7: Comparison of experimental data for 0.02 grams citric acid with both models.

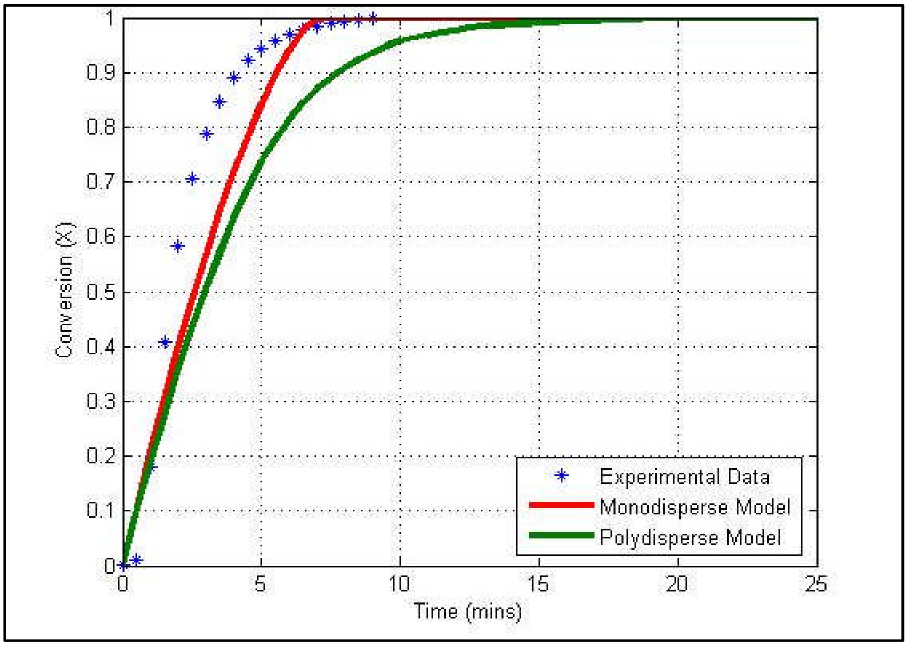


Figure 7.8: Comparison of experimental data for 0.04 grams citric acid with both models.

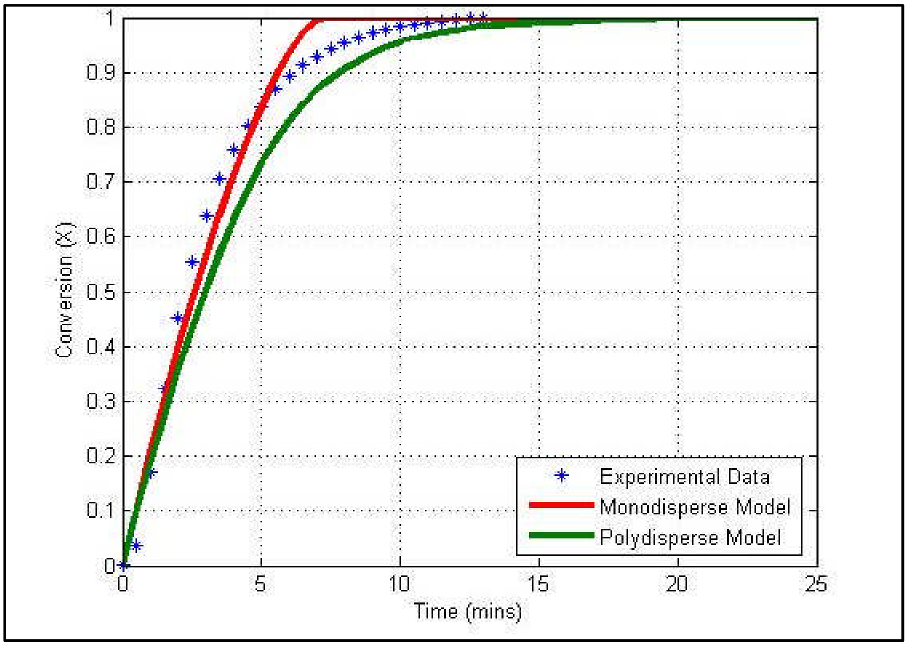


Figure 7.9: Comparison of experimental data for 0.06 grams citric acid with both models.

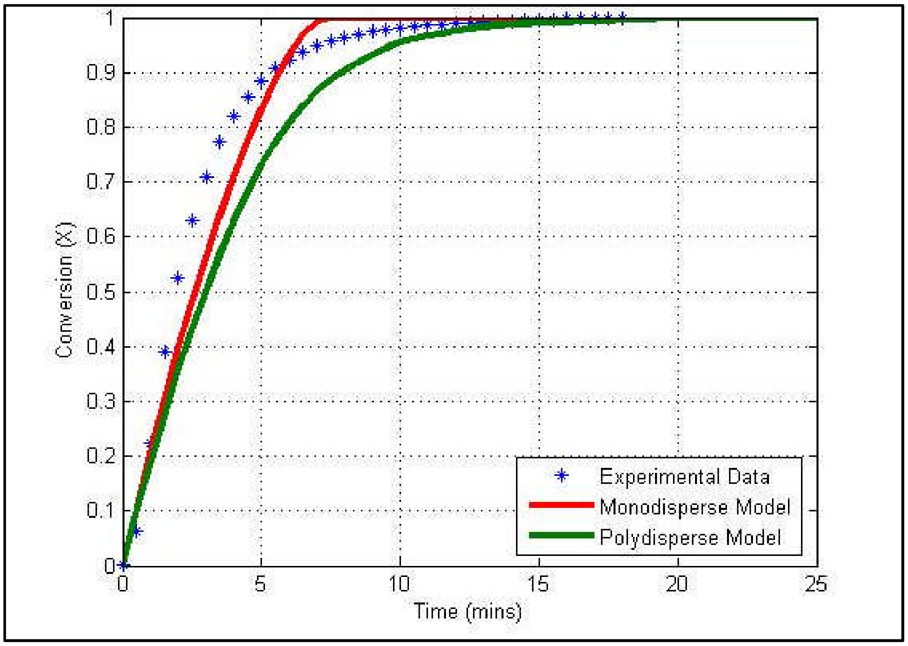


Figure 7.10: Comparison of experimental data for 0.08 grams citric acid with both models.

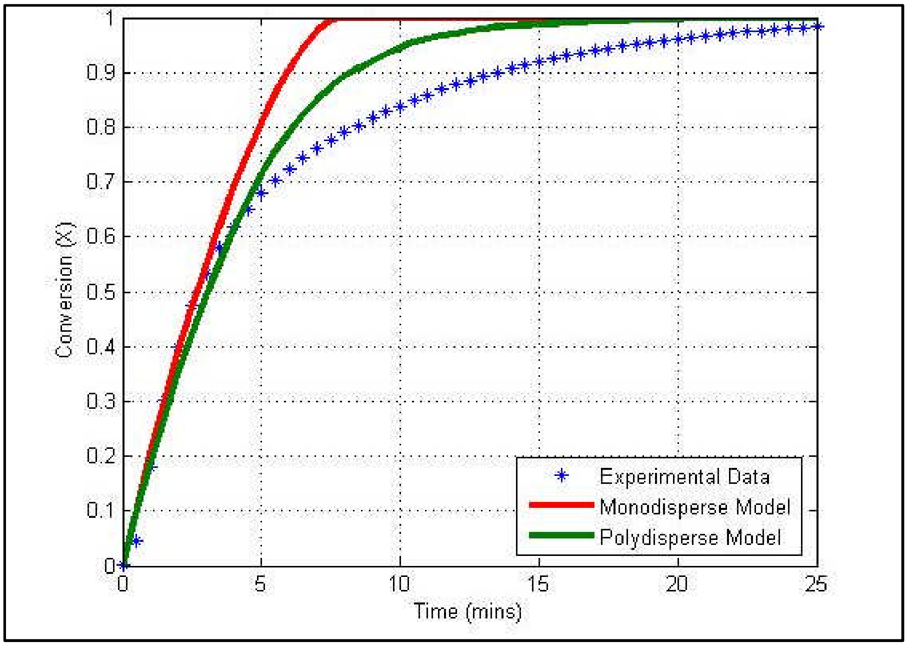


Figure 7.11: Comparison of experimental data for 0.20 grams citric acid with both models.

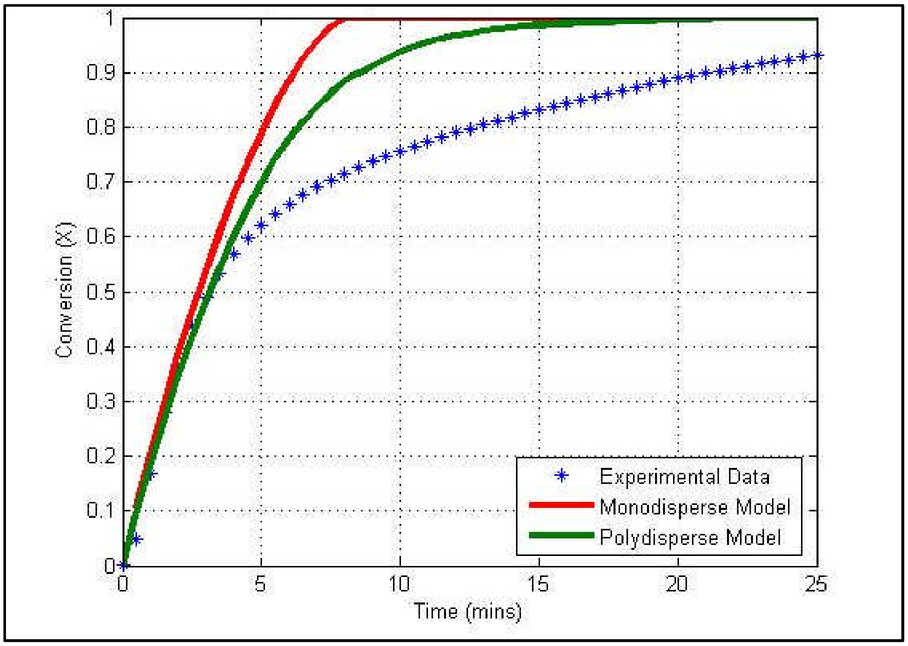


Figure 7.12: Comparison of experimental data for 0.30 grams citric acid with both models.

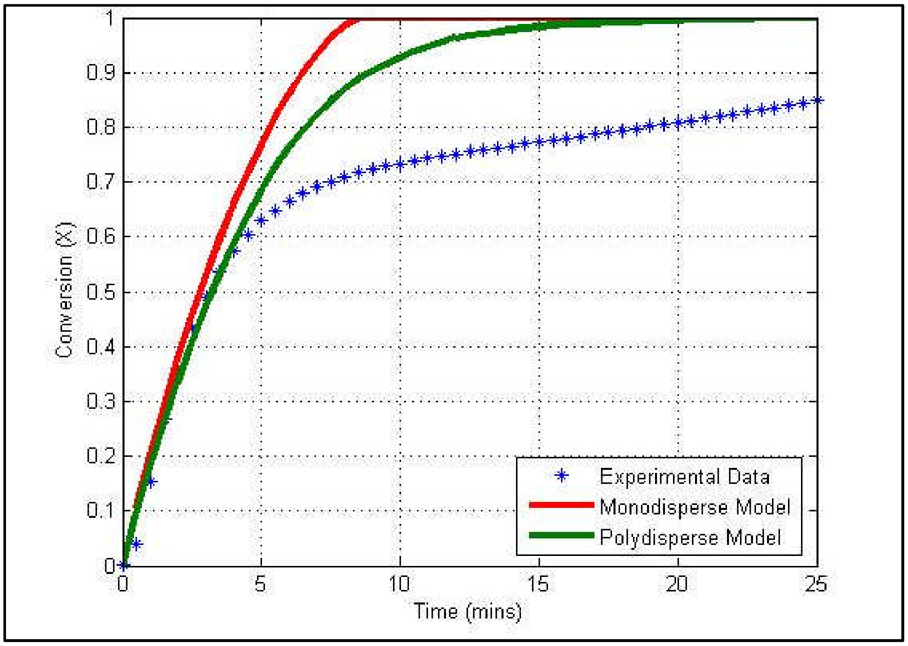


Figure 7.13: Comparison of experimental data for 0.40 grams citric acid with both models.

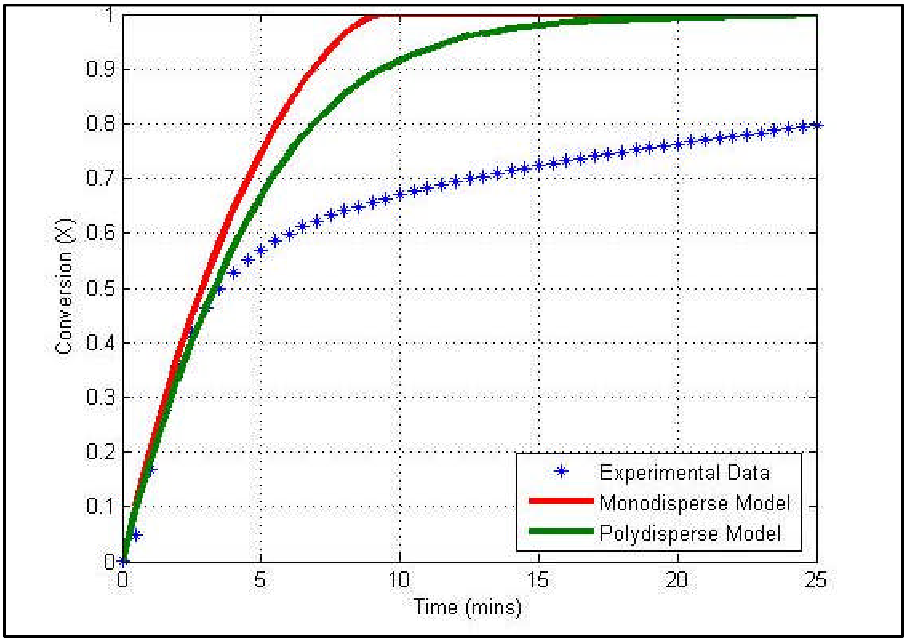


Figure 7.14: Comparison of experimental data for 0.50 grams citric acid with both models.

7.2 Interpreting Discrepancies

For concentrations of citric acid between 0.06-0.10 grams, the models for dissolution provide a good estimate for release rate behavior. For concentrations below 0.06 grams, the experimental data shows a faster dissolution time than predicted. For concentrations above 0.10 grams, the experimental data shows a much longer dissolution time than the models predict. This section will look at the two main causes for these discrepancies: modeling assumptions and experimental inaccuracies.

7.2.1 Modeling Assumptions

Numerous assumptions were made in the analytical derivation of the original model for calculating the dissolution time of monodisperse particles, as listed in Chapter 2. Many of the assumptions made conflict with the actual experimental setup. For instance, the particle shape was assumed to be perfectly spherical. SEM results show that the actual citric acid particles are not spherical and are not uniform in shape, as can be seen in Figure 7.15. Particle shape has a significant effect on the release rate characteristics, since the rate of dissolution is dependent on the surface area of solute exposed to the solvent. As discussed in Chapter 4, the polydisperse size distribution was approximated using the equivalent spherical area of the particles. While a spherical shape provides a good initial estimate, it does not accurately describe the dissolution of particles with cylindrical, cubic or rectangular geometries.

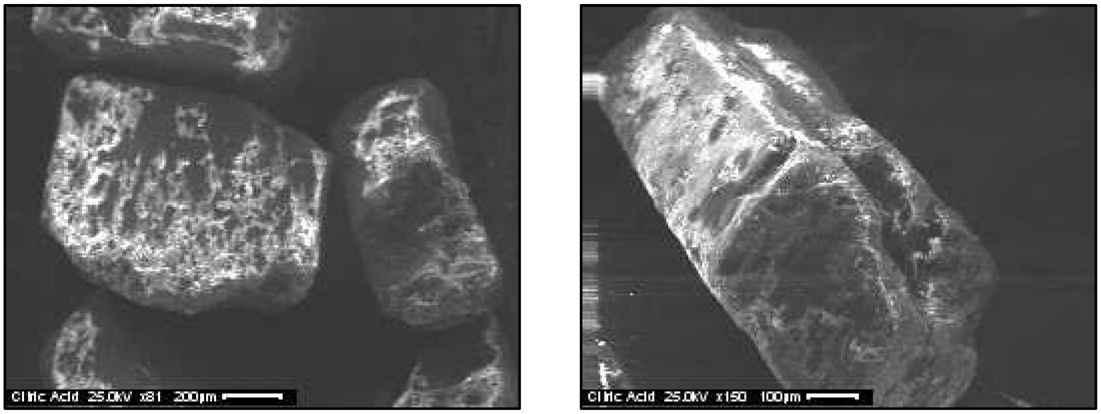


Figure 7.15: SEM photos of non-spherical citric acid particles.

Further assumptions reguarding the size distribution also have an effect on the modeling results. The mass percent of each of the size distributions is inputted into the MATLAB program and is used to back calculate the number of particles in each size group. The assumption is that every citric acid sample tested has the same size distribution. However, it was seen in the SEM results, in Figure 4.6, that the size distribution varied for each sample. For small sample sizes, such as 0.02 grams, there are only a few particles and inaccuracies in predicting the number of particles with a given radius size dramatically change the results produced.

The model for monodisperse particles utilized the concentration gradient between the solubility of the solute, C_s, and the concentration at a given time, C, as the driving force for dissolution. The assumption that the concentration of C around the particle is representative of the concentration of the entire surrounding solution further implies that the system is well stirred. However, the additional effects of stirring are not incorporated into the model. This includes considering the agitation a stirrer would have on the system and additional convective mass transfer effects resulting from the moving liquid. To monitor the effects of stirring, five samples of citric acid were tested experimentally. The dissolution of the concentrations of 0.02, 0.06, 0.10, 0.30 and 0.50 grams in 1 mL of water were tested with the addition of a an electromagnetic stir bar. The stirred dissolution experiments completed much faster than the unstirred tests and can be seen in Figure 7.16. The results for the stirred dissolution experiment of 0.10 grams of citric acid were plotted along with the non-stirred experimental results, monodisperse and polydisperse model in Figure 7.17. The data for the stirred experiment showed a steeper slope but had a final dissolution time similar to the monodisperse model. The models appeared to be a combination of the results of stirred and unstirred experiments, which is also suggested by research [52].

Finally, the parameters for the chemical substances are assumed to be constant which can lead to erroneous calculations. Viscosity and temperature are assumed to be constant, which causes the diffusion coefficient to be calculated as a constant as well. In actuality, the temperature of the system changes because the reaction of citric acid and water is endothermic. The decrease in temperature would have an effect on the rate of the diffusion, however that was not considered in these models. Another influential parameter, C_s, the solubility of the solute, was considered to be constant throughout the dissolution process. Although this was helpful in simplifying the mathematics involved, it does not fairly represent the system, since solubility is never constant and is highly dependent on temperature [53-54]. Simplifications made in the derivation of the dissolution model explain some of the discrepancies seen between experimental data and the results model predicted.

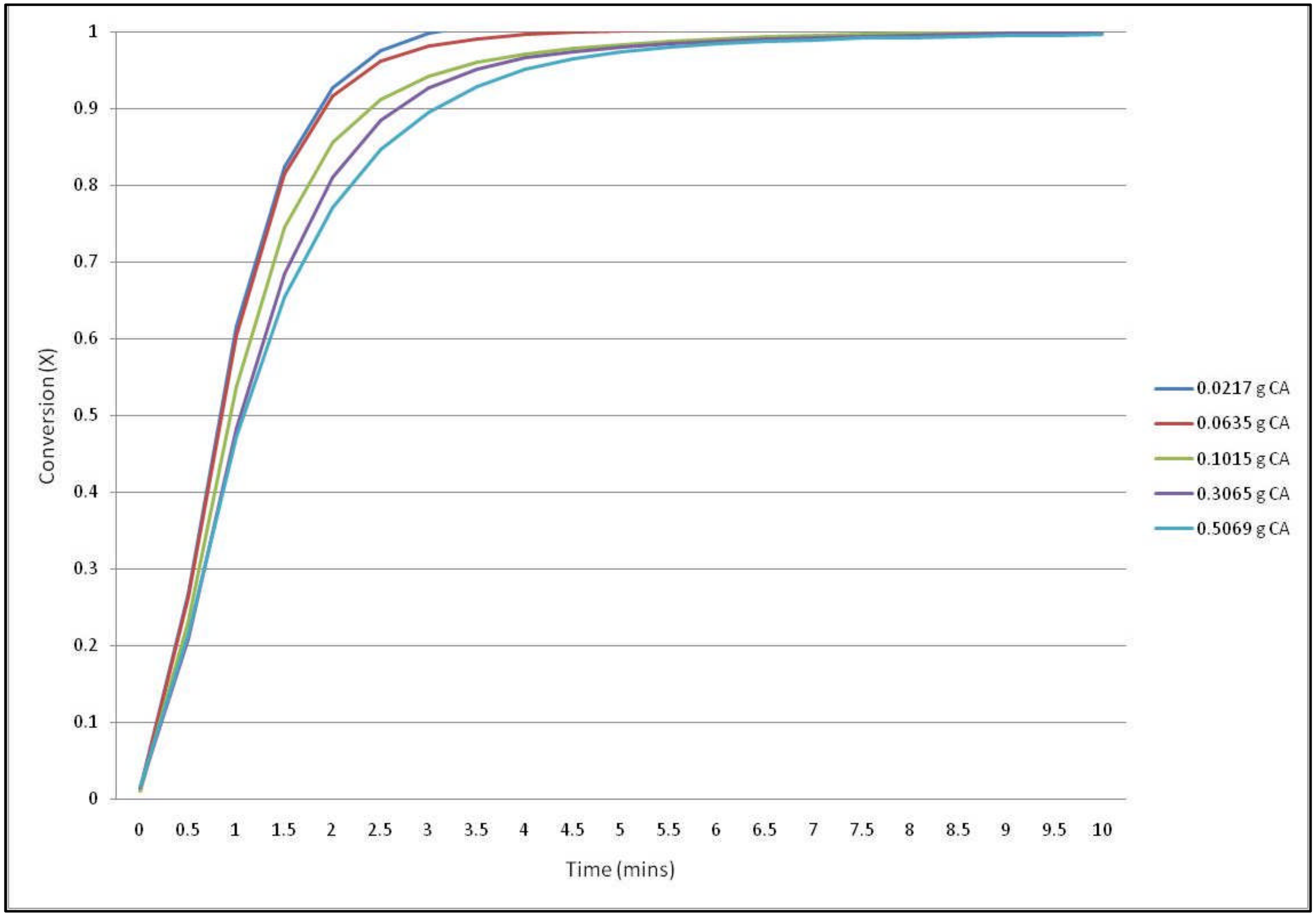


Figure 7.16: Stirred experimental data for citric acid for different initial amounts.

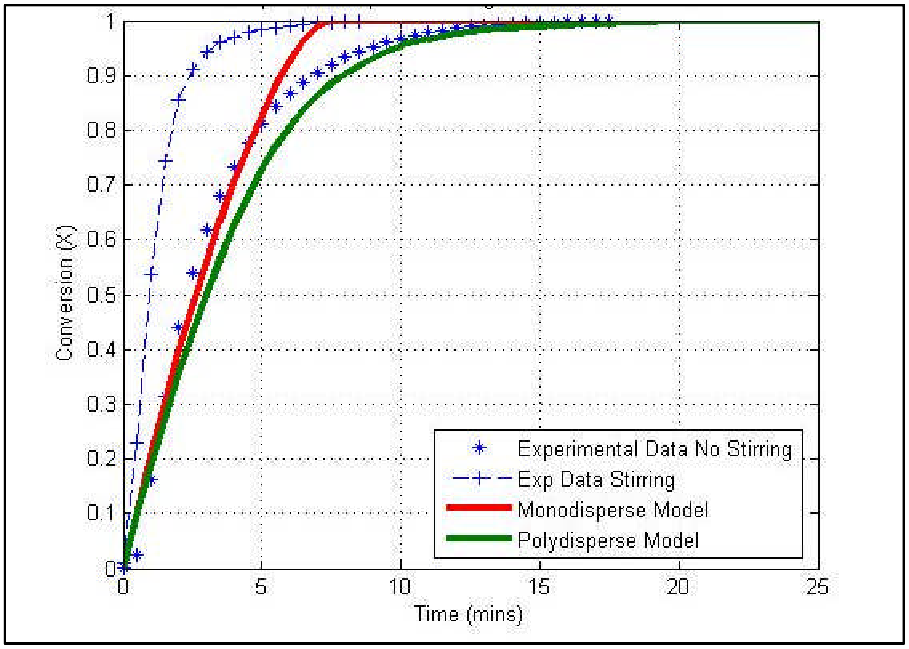


Figure 7.17: Comparison plot for stirred data using 0.10 grams of citric acid.

7.2.2 Experimental Inaccuracies

The experimental procedure and data analysis should also be considered while relating the results to the accuracy of the models. First, the calorimetric setup used was an OmniCal Technologies SuperCRC 20-305-2.4 reaction microcalorimeter. The reaction took place on a micro scale, involving less than 5 mL of solvent. The small sample size made experimental measurements difficult, especially for dilute solutions. At low concentrations, the amount of citric acid present was minuscule, and the change in heat flow measured was very small. With such low measurements, the limitations of the equipment could have affected the results found.

While the experiment was conducted using the isothermal setting in the calorimeter, the initial temperatures of each of the experiments had slight variations. All experiments were conducted at room temperature, but room temperature varied depending on the temperature of the day the experiment was run. The tau correction, which is used to establish the corrected time of the calorimeter, required raising the temperature of the system for an extended time, a period of about 30 minutes. Following the tau correction procedure, the system could never return exactly to the initial temperature, it always remained a few percentages higher than the starting temperature. It was shown in Chapter 3, in Figure 3.4, that temperature can has a slight affect on the dissolution time. Therefore, the variations in the initial temperatures of each of the experimental runs should be considered.

Reaction microcalorimeters have been used in dissolution testing by providing the heat flow produced by a reaction with respect to time. The integration of the heat curve allows for the determination of the conversion as a function of time. The total conversion represents the integration of the complete heat curve generated by the reaction. Errors from this calculation can be found at high concentrations near the solubility limit. Even in instances where the particles have not completely dissolved, such as in cases near the solubility limit, the integration of the complete curve would suggest 100% conversion. This implies that the DSC results always need to be validated at high concentrations to ensure that complete dissolution is has indeed taken place.

7.3 Encapsulated Models

At the time this project was completed, glucose encapsulated citric acid particles were not available for testing. For that reason, it was not possible to test the encapsulated particles experimentally to the simulated models. It was, however, possible to estimate the effects of the glucose encapsulation layer by testing glucose concentration in water and then using the glucose water solution to dissolve citric acid. The two step process provided an initial estimate to the dissolution behavior an encapsulated particle might have.

For the monodisperse model an encapsulation thickness of 0.001 cm was used, which is approximately 0.01 grams of glucose. For the polydisperse model, an encapsulation thickness of 0.016 cm was used, around 0.10 grams of glucose. The dashed lines represent the encapsulated layer while the solid lines are the inner particle. The amounts of 0.01 grams and 0.10 grams of glucose are graphed along with the inner particle data. The polydisperse model assumed the same radius size distribution as found in Chapter 4. All particles were assumed to have equal encapsulation layer thicknesses.

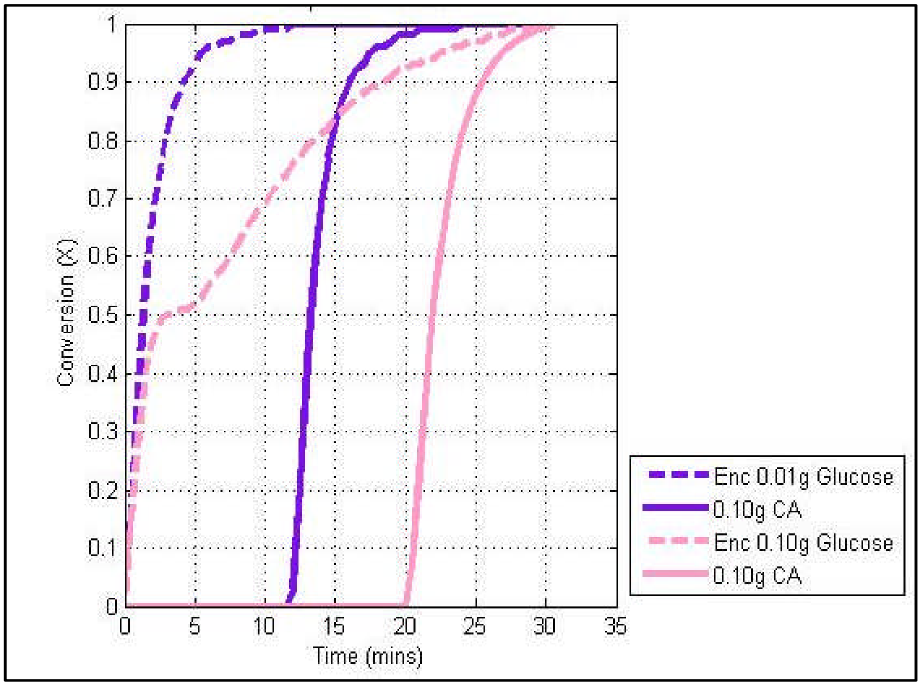


Figure 7.18: Experimental results for citric acid encapsulated with 0.01 and 0.10 grams of glucose.

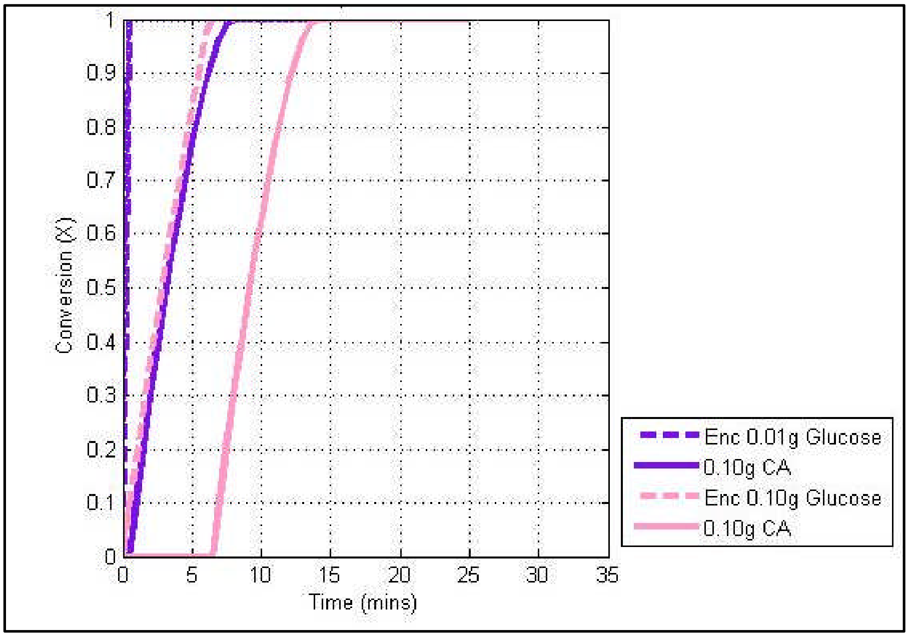


Figure 7.19: Monodisperse model for citric acid encapsulated with 0.01 and 0.10 grams of glucose.

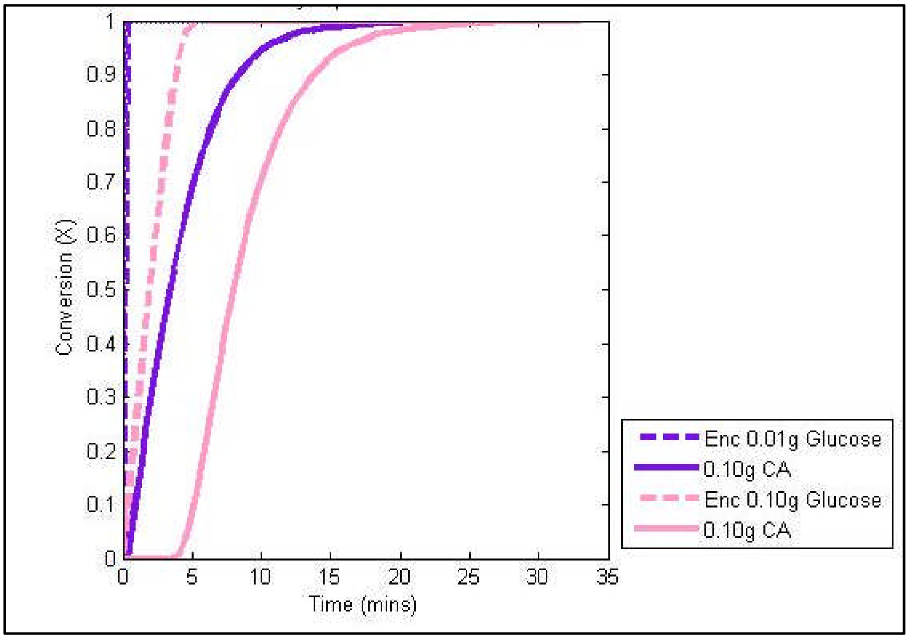


Figure 7.20: Polydisperse model for citric acid encapsulated with 0.01 and 0.10 grams of glucose.

7.3.1 Sources of Error

Comparisons of the experimental data, monodisperse encapsulated model and polydisperse encapsulated model show similar discrepancies to the ones outlined in Section 7.1. It also appears that for 0.10 grams of glucose, the dissolution behavior is much different than expected. One explanation for the discrepancies is that both the monodisperse and polydisperse model assume the glucose is a thin layer surrounding each particle. The reaction microcalorimetry experiments conducted with 0.01 and 0.10 grams of solid particle glucose, which has completely different surface area characteristics. The particle size distribution of the glucose was not taken into account and therefore does not represent the size of an encapsulated particle as assumed.

As mentioned previously, the parameters associated with the glucose-water solution and citric acid particles were not known for the experimental conditions. Once further experimentation is done to establish that the diffusion coefficients calculated are appropriate for the system, more confidence can be given to the models developed.

7.4 Comparison of Methods

The models developed in this work were primarily based chemical engineering transport phenomena principles. The main difference between the models developed in this work with the dissolution models reviewed in Chapter 1 is that chemical properties alone can be used to determine the dissolution behavior of the system. For the classical Nernst-Brunner equation, Eq. (1.2), and Hixon-Crowell cube root equation, Eq. (1.5), several constants k_1 and k_3 are needed to evaluate the model. These constants are generally determined by best fitting experimental data.

$$\frac{\mathrm{dC}}{\mathrm{dt}} = k_1 \cdot (C_s - C) \tag{1.2}$$

$$w_0^{1/3} - w^{1/3} = k_3 t \tag{1.5}$$

Another difference in the model used is that it incorporates the radius of the particle in the dissolution model, rather than only total concentration. This can be exceedingly useful in the design of particles with radius size distributions, such as in polydisperse systems, and can not be examined with models which consider concentration or particle weight alone.

Chapter 8 : Conclusion

In this work, the analytical derivation for the dissolution of solid particles in liquid solvents was formulated in mathematical terms. The structure of the program consisted of developing a simple core structure which allowed for the particle radius size to be calculated as a function of time. Following the design, additional levels of complexity were added to the computational model including accounting for polydispersity of particles and encapsulation. The comprehensive computer program from the four programs developed is shown in Figure 8.1. A parametric study of the effects of particle size, concentration, and composition was performed to be utilized in the development of specially design particles for controlled release.

The initial time delay in the release of the active core ingredient was observed when the particles were encapsulated with a water-soluble compound. The dissolution rate was found to be dependent on the particle size. Parameters associated with the composition mainly affected the calculation of the diffusion coefficient, while concentration determined the number of particles and also impacted the release rate characteristics. Accounting for polydispersity produced more natural curves, which more accurately resembled the data found experimentally.

8.1 Implications

The methodology for computer-aided design of specialized particles has been developed in this work. The program can be utilized in several design approaches. First, the simulation can be used to predict the results of a given particle. The program can be modified to see the affect of various parameter changes on the given particles, such as particle size or concentration. Additionally, encapsulation options can be investigated if a delay in the release rate is desired. The second approach, which is the inverse of the first approach, involves starting with a desired release rate profile and manipulating the parameters and particle characteristics to determine the type of particle which can produce the desired effect.

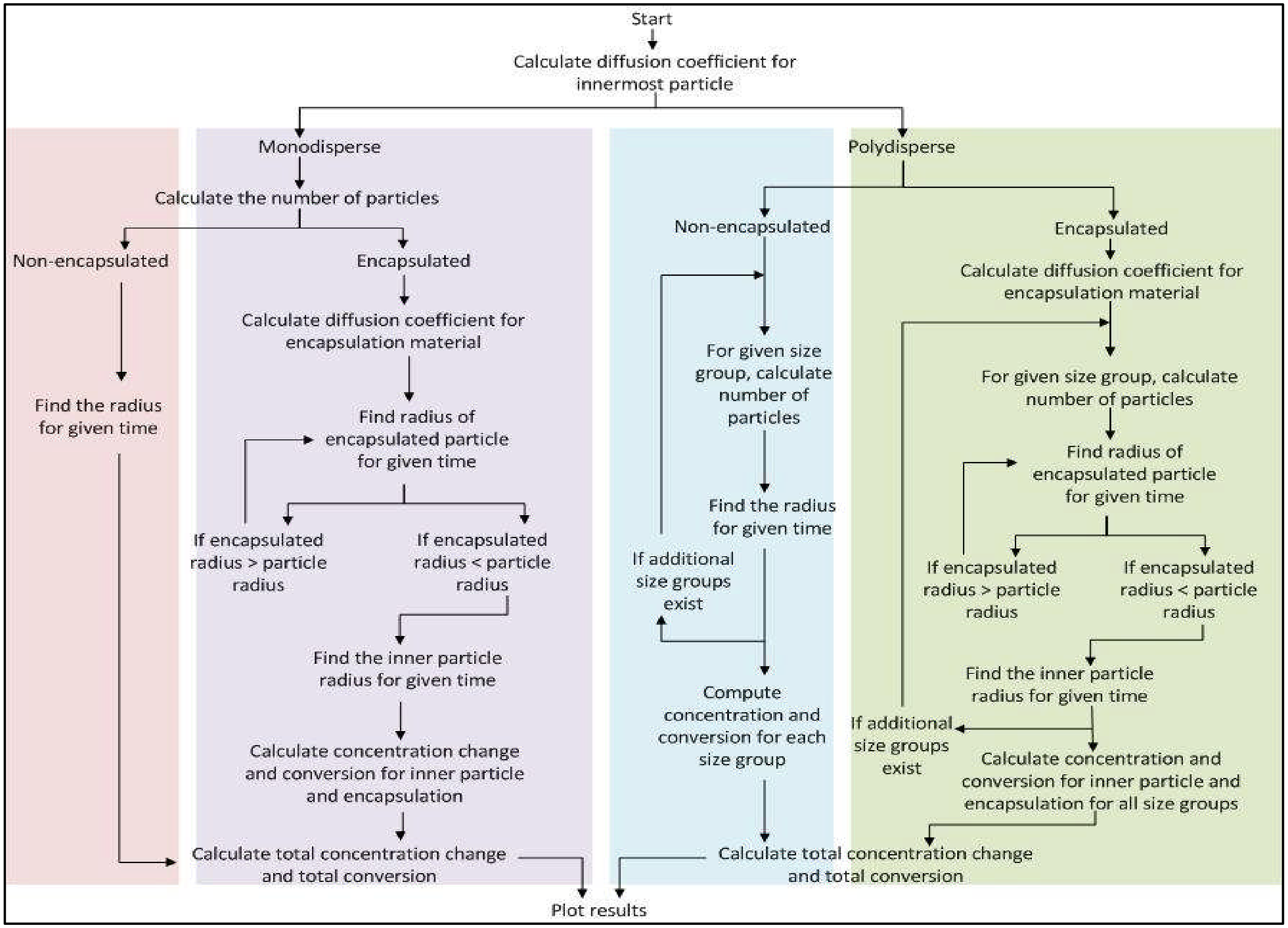

Start
Calculate diffusion coefficient for innermost particle
Monodisperse
Polydisperse
Calculate the number of particles
Non-encapsulated
Encapsulated
Find the radius for given time
Calculate diffusion coefficient for encapsulation material
Find radius of encapsulated particle for given time
If encapsulated radius > particle radius
If encapsulated radius < particle radius
Find the inner particle radius for given time
Calculate concentration change and conversion for inner particle and encapsulation
Calculate total concentration change and total conversion
Non-encapsulated
For given size group, calculate number of particles
Find the radius for given time
If additional size groups exist
Compute concentration and conversion for each size group
Encapsulated
Calculate diffusion coefficient for encapsulation material
For given size group, calculate number of particles
Find radius of encapsulated particle for given time
If encapsulated radius > particle radius
If encapsulated radius < particle radius
Find the inner particle radius for given time
If additional size groups exist
Calculate concentration and conversion for inner particle and encapsulation for all size groups
Calculate total concentration change and total conversion
Plot results

Figure 8.1: Comprehensive flowsheet for all four programs developed in this work.

Since all the computations are carried out numerically, the virtual experiments can be carried out numerous times before the appropriate particle formulation is discovered. A variety of particle options which meet a specified dissolution time can be compiled, and theses can then be compared to determine the most feasible alternative based on material and manufacturing factors.

8.2 Future Work

While the main framework for the computer program was developed, potential improvements to the code exist. Several options can be added which expand versatility of program and would ultimately increase the accuracy of the computations.

In the analytical derivation of the time required for dissolution, the diffusion boundary layer thickness, *h*, was approximated by the particle radius. This is applicable for large particles, where the diffusion boundary layer thickness decreases with decreasing particle radius. If the diffusion boundary layer is constant, then the time for particles to dissolve can be modeled using the derivation from Sertsou [42] in Eq. (8.1),

$$t = \frac{\left[\frac{1}{6c^2}\ln\frac{(c-r_0)^3\cdot(c^3-r^3)}{(c^3-r_0)^3\cdot(c^3-r^3)} + \frac{1}{\sqrt{3}c^2}\left[\tan^{-1}\frac{2r+c}{\sqrt{3}c} - \tan^{-1}\frac{2r_0+c}{\sqrt{3}c}\right]\right]\cdot h\cdot 3\cdot V_m}{D\cdot N\cdot 4\cdot\pi} \qquad (8.1)$$

where *r* is the constant diffusion layer thickness. This equation is used to describe the transition of a constant diffusion boundary layer to a boundary layer which changes with the radius, as derived in Chapter 2. Based on the dissolving material and dissolution conditions, both of the expressions may be required to find the best fit to experimental data.

The program code can be further improved by adjusting the derivations to include descriptions of non-spherical shapes, non-uniform coating, and multiple layered coating. Some possible non-spherical geometric shapes for particles are shown in Figure 8.2. While the assumption of perfectly spherical particles was made in the models used in this work, incorporating the derivation additional shapes can dramatically influence the dissolution behavior. For instance, the derivation of controlled drug release for cylindrical binary system similar to was derived by Marentette and Grosser in [56]. Recent research conducted by Ansari and Stepanek, [57], focused on the dissolution of granules, which are agglomerations of primary solid particles. Incorporation of increasingly advanced geometries can expand the applications of the program.

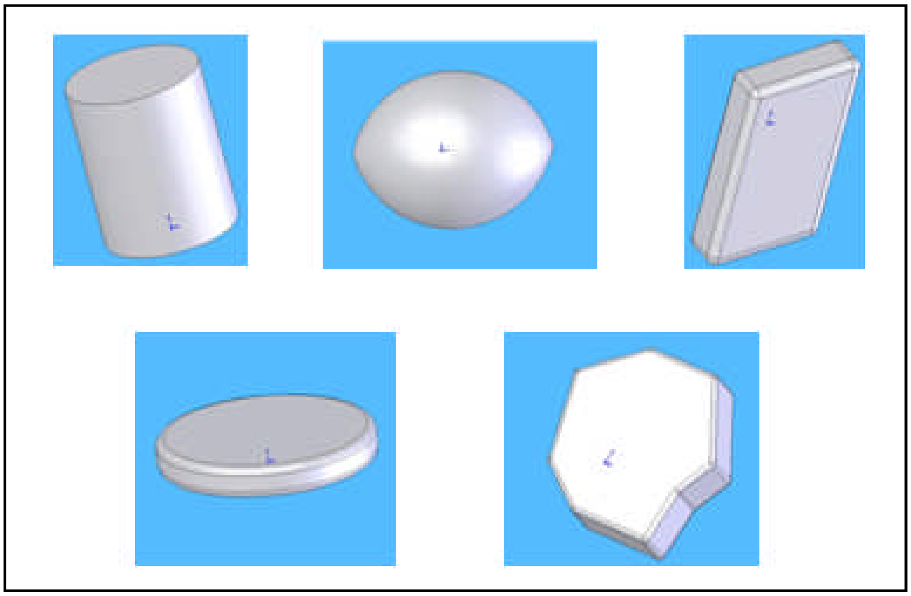


Figure 8.2: Other non-spherical geometric shapes of particles.

Further improvements to the model can be made by accounting for contact of the particles with other particles and the system container. The more contact a particle has with its surroundings, the less surface area is exposed which affects the dissolution time, as shown in Figure 8.3. As was seen in Chapter 7, the models for monodisperse and polydisperse particles fall between the experimental results for unstirred and stirred solutions.   If the unstirred case is considered, then modifications on the model must be made to account for concentration of the stagnant film surrounding the particle, which do not represent the total concentration of the system. If the stirred case is considered, then the model must be adjusted to account for the agitation the stirring produces, such as an increase in convective mass transfer.

The incorporation of additional equations to model the chemical parameters of the system could also improve the results. Many of the parameters which were assumed to be constant change throughout the dissolution process and should be more accurately described using a function. For instance, temperature is assumed to be constant throughout the process, but the reaction is endothermic, meaning the temperature will decrease as the reaction proceeds. Incorporating the change in temperature would affect the calculation of the diffusion coefficient.

The solubility of the solute is also taken to be constant, but is actually a function of temperature, and should be written as an equation rather than inputted as a single value.

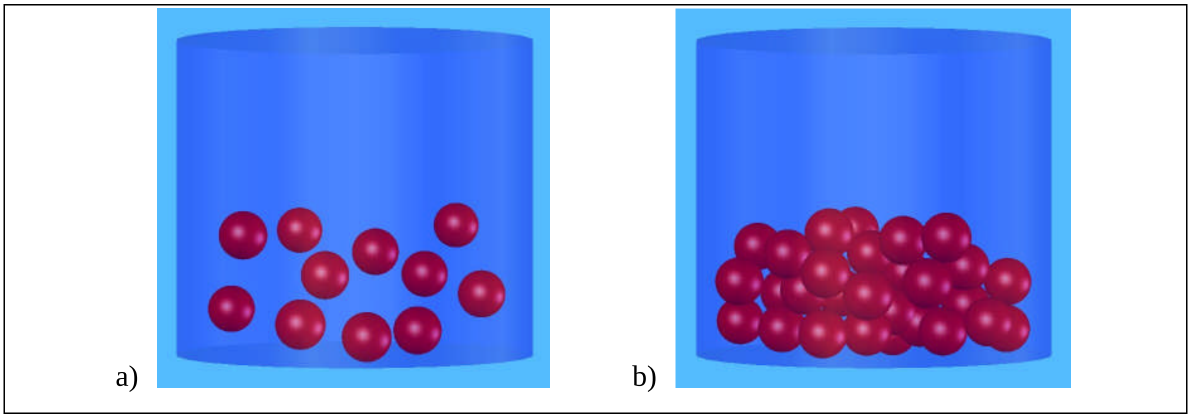


Figure 8.3: Surface area changes between a) low concentration and b) high concentration of solid particles.

Further advancements to the existing framework of the code could potentially be developed to simulate dissolution testing of the release of solid dosage forms for pharmaceuticals. According to the U.S. Department of Health and Human Services Food and Drug Administration [58], dissolution tests are usually conducted in dissolution mediums of 500, 900, or 1000 mL. The aqueous medium used usually has a pH range between 1.2 to 6.8, with a pH of 6.8 used to simulate intestinal fluid. The computer model developed should be tested under these conditions to determine if the results match those found experimentally. The program should then be adjusted to match the methods approved by the U.S. Pharmacopeia (USP) for solid dosage forms testing. The improved program could be a valuable tool in drug development and design of the controlled release of pharmaceutical agents. Current research suggests that controlled-release dosage forms can significantly enhance clinical efficacy while decreasing treatment costs, providing additional economic value over immediate-release dosage forms [59]. The initial challenge of controlled release is the determination of the formulation for a desired release profiles. Computer-aided design, based on the fundamental understanding of the dissolution process, could reduce the trial and error associated with experimental prototype testing. In the future, computer simulations could be used to optimize drug delivery systems by aiding in the design of specialized controlled release formulations.

References

[1] Business Communications Company, Inc. 2004. RP-370 Active, “Controlled Release Technologies: Estabilished and Emerging Markets”, [Online]. Available: BBC Research, http://www.bccresearch.com/RepTemplate.cfm?reportID=370&RepDet=HLT&cat=mst&target=repdetail.cfm [Accessed Sep. 9, 2007].

[2] A.N. Bhaskarwar, “General Population Balance Model of Dissolution of Polydisperse Particles”, *AIChE Journal*, vol. 35, no. 4, 1989, pp. 658-661.

[3] J. Wang and D.R. Flanagan, “General Solution for Diffusion-Controlled Dissolution of Spherical Particles. 1. Theory”, *American Chemical Society and American Pharmaceutical Association*, vol. 88, no.7, 1999, pp. 731-738.

[4] D.O. Cooney, “Effect of Geometry on the Dissolution of Pharmaceutical Tablets and Other Solids: Surface Detachment Kinetics Controlling”, *AIChE Journal*, vol. 18, no.2, 1972, pp. 446-449.

[5] A. Madene, M. Jacquot, J. Scher and S. Desobry, “Flavour encapsulation and controlled release – a review”, *International Journal of Food Science and Technology*, vol. 46, 2006, pp. 1-21.

[6] N. Muro-Sune, R. Gani, G. Bell, and I. Shirley, “Model-based computer-aided design for controlled release of pesticides”, *Computers and Chemical Engineering*, vol. 30, 2005, pp. 28-41.

[7] F. Stepanek, “Computer-Aided Product Design Granule Dissolution”, *Chemical Engineering Research and Design*, vol. 82, no.11, 2004, pp. 1458-1466.

[8] R.W. Baker and H.K. Lonsdale, “Controlled Release: Mechanisms and Rates”, in *Controlled Release of Biologically Active Agents*, New York: Plenum Press, 1974, pp. 15-72.

[9] K. Das, *Controlled-Release Technology: Bioengineering Aspects*, New York: John Wiley & Sons, Inc., 1983.

[10] D.R. Paul, “Polymers in Controlled Release Technology”, in *Controlled Release Polymeric Formulations*, Washington D.C.: American Chemical Society, 1976.

[11] J.T. Cartensen, *Advanced Pharmaceutical Solids*, vol. 110, New York: Marcel Dekker, Inc., 2001.

[12] J. Siepmann, F. Siepmann and A.T. Florence, "Local controlled drug delivery to the brain: Mathematical modeling of the underlying mass transport mechanisms", *International Journal of Pharmaceutics,* vol. 314, 2006, pp. 101-119.

[13] W.I. Higuchi, E.P. Parrott, D.E. Wurster and T. Higuchi, "Investigation of Drug Release from Solids II: Theoretical and Experiemtnal Study of Influences of Bases and Buffers on Rates of Dissolution of Acidic Solids", *Journal of the American Pharmaceutical Association,* vol. 47, no. 5, 1957, pp. 376-383.

[14] R. Ito, B. Golman and K. Shinohara, "Controlled release with coating layer of permeable particles", *Journal of Controlled Release,* vol. 92, 2003, pp. 361-368.

[15] H.B. Hopfenberg, "Controlled Release from Erodible Slabs, Cylinders, and Spheres" in *Controlled Release Polymeric Formulations,* Washington D.C.: American Chemical Society, 1976.

[16] D.R. Paul, "Polymers in Controlled Release Technology", in *Controlled Release Polymeric Formulations,* Washington D.C.: American Chemical Society, 1976, pp. 1-14.

[17] E. Ron and R. Langer, "Erodible Systems", in *Treatise on Controlled Drug Delivery,* New York: Marcel Dekker, Inc., 1992, pp. 199-224.

[18] D. Hsieh, "Controlled Release Systems: Past, Present, and Future", in "*Controlled Release Systems: Fabrication Technology* ", Boca Raton: CRC Press, Inc., 1988, pp. 1-16.

[19] D. Manca and M. Rovaglio, "Modeling the controlled release of microencapsulated drugs: theory and experimental validation", *Chemical Engineering Science*, vol. 58, 2003, pp. 1337-1351.

[20] M.Z. Zhang, C. Li-Ling and C.-H. Wang, "Simulation of Drug Release from Biodegradable Polymeric Microspheres with Bulk and Surface Erosions", *Journal of Pharmaceutical Sciences,* vol. 92, no. 10, 2003, pp. 2040-2056.

[21] U.-C. A. Edlund, "Degradable Polymer Microspheres for Controlled Drug Delivery", *Advances in Polymer Science,* vol. 157, 2002, pp. 67-112.

[22] B. F. Gibbs, S. Kermasha, I. Alli and C.N. Mulligan, “Encapsulation in the food industry: a review”, *International Journal of Food Sciences and Nutrition*, vol. 50, 1999, pp. 213-224.

[23] M. Mellema, W. Van Benthum, B. Boer, J. Von Harris and A Visser, “Wax Encapsulation of water-soluble compounds for application in foods”, *Journal of Microencapsulation*, vol. 23, no.7, 2006, pp.729-740.

[24] M. McCoy, “Special Delivery: Getting fragrance onto clothes presents a challenge for detergent companies and their suppliers”, *Chemical and Engineering News*, vol. 85, no.5, 2007, pp. 21-23.

[25] G. L. Flynn, S. H. Yalkowsky and T. J. Roseman, Mass Transport Phenomena and Models: Theoretical Concepts Review. *Journal of Pharmaceutical Sciences*, vol. 63, no.4, 1974, pp. 479-510.

[26] A.A. Noyes and W.R. Whitney, “The rate of solution of solid substances in their own solutions”, *Journal of the American Chemical Society*, vol. 19, 1897, pp. 930-934.

[27] L. Brunner and S. Tolloczko, “Uber die Auflosungsgeschwindigkeit Fester Korper”, *Zeitschrift fur Physikalische Chemie,* vol. 35, 1900, pp. 283-290.

[28] E. Brunner, “Reaktionsgeschwindigkeit in heterogenen Systemen”, *Zeitschrift fur Physikalische Chemie,* vol. 43, 1904, pp. 56-102.

[29] W. Nerst, “Theorie der Reaktionsgeschwindigkeit in heterogenen Systemen”, *Zeitschrift fur Physikalische Chemie,* vol.47, 1904, pp. 52-55.

[30] A.W. Hixson and C.J. H, “Dependence of reaction velocity upon surface and agitation”, *Industrial & Engineering Chemistry Research* , vol. 23, 1931, pp. 923-931.

[31] J. Wang and D.R. Flanagan, “General Solution for Diffusion-Controlled Dissolution of Spherical Particles. 2. Evaluation of Experimental Data”, *Journal of Pharmaceutical Sciences*, vol. 91, no. 2, 2002, pp. 534-542.

[32] P.J. Niebergall, G. Milosovich and J.E. Goyan, “Dissolution Rate Studies II. Dissolution of Particles Under Conditions of Rapid Agitation”, *Journal of Pharmaceutical Sciences*, vol. 52, 1963, pp. 236-241.

[33] T. Higuchi, “Rate of release of medicaments from ointment bases containing drugs in suspension”, *Journal of Pharmaceutical Sciences,* vol. 50, 1961, pp. 874-875.

[34] A. Dokoumetzidis and P. Macheras, “A century of dissolution research: From Noyes and Whitney to the Biopharmaceutics Classification System”, *International Journal of Pharmaceutics,* vol. 321, 2006, pp. 1-11.

[35] J. Mauger and S. Howard, “Model Systems for Dissolution of Finely Divided (Multisized) Drug Powders”, *Journal of Pharmaceutical Sciences,* vol. 65, no.7, 1976, pp. 1042-1045.

[36] G. Frenning, “Theoretical analysis of the release of slowly dissolving drugs from spherical matrix systems”, *Journal of Controlled Release,* vol. 95, 2004, pp. 109-117.

[37] G. Frenning, “Theoretical investigation of drug release from planar matrix systems: effects of a finite dissolution rate”, *Journal of Controlled Release,* vol. 92, 2003, pp. 331-339.

[38] F.C. Hoppensteadt and C.S. Peskin, *Modeling and Simulation in Medicine and the Life Sciences.* New York: Springer, 2002.

[39] M. Polakovic, T. Gorner, R.Gref and E. Dellacherie, “Lidocaine loaded biodegradable nanospheres II. Modelling of drug release”, *Journal of Controlled Release* , vol. 60, 1999, pp. 169-177.

[40] S. LeBlanc and H. Fogler, “Population Balance Modeling of the Dissolution of Polydisperse Solids: Rate Limiting Regimes”, *AIChE Journal* , vol. 33, no.1, 1987, pp. 54-63.

[41] A. Kondo, *Microcapsule Processing and Technoolgy,* New York: Marcel Dekker, Inc., 1979.

[42] G. Sertsou, “Analytical Derivation of Time Required for Dissolution of Monodisperse Drug Particles”, *Journal of Pharmaceutical Sciences,* vol. 93, no. 8, 2004, pp. 1941-1944.

[43] *“MATLAB- The Language of Technical Computing”*, 1994-2007, [Online]. Available: The MathWorks Web site, http://www.mathworks.com [Accessed July 2007].

[44] W.J. Palm III, *Introduction to MATLAB 7 for Engineers,* New York: McCraw Hill, 2005.

[45] R.C. Reid and T.K. Sherwood, *The Properties of Gases and Liquids,* New York: McGraw-Hill Book Company, 1958.

[46] COMSOL Script User's Guide, 1994-2006.

[47] B.R. Jennings and K. Parslow, “Particle size measurement: the equivalent spherical diameter”, *Proceedings of the Royal Society London A,* vol. 419, 1988, pp. 137-149.

[48] G. Zuccari, R. Carosio, A. Fini, P. Montaldo and I. Orienti, “Modified polyvinylalchol for encapsulation of all-trans-retinoic acid in polymeric micelles”, *Journal of Controlled Release,* vol. 103, 2005, pp. 369-380.

[49] W.B. Zimmerman, *Process Modelling and Simulation with Finite Element Methods,* London: World Scientific Publishing Co., 2004.

[50] T. Yamauchi and A. Hasegawa, “Determination of PEG Concentration in Its Aqueous Solution Using Differential Scanning Calorimetry”, *Journal of Applied Polymer Science,* vol. 49, 1993, pp. 1653-1658.

[51] Omnical Inc. Reaction Calorimeter Instrucments, 2007. Active. [Online]. Available: Omnical Technologies, http://www.omnicaltech.com [Accessed Sep. 9, 2007].

[52] A.P. Simonelli, D.R. Flanagan and W.I. Higuchi, “Apparent Discrepancy Between Theory and Experimental Data for Dissolution from the Rotating Disk Under Stirred and Unstirred Conditions”, *Journal of Pharmaceutical Sciences,* vol. 57, no. 9, 1968, pp. 1629-1631.

[53] J. Siepmann and A. Gopferich, “Mathematical modeling of bioerodible, polymeric drug delivery systems”, *Advanced Drug Delivery Reviews,* vol. 41, 2001, pp. 229-247.

[54] J.B. Schwartz, A.P. Simonelli and W.I. Higuchi, “Drug Release from Wax Matrices I: Analysis of Data with First-Order Kinetics and with the Diffusion-Controlled Model”, *Journal of Pharmaceutical Sciences,* vol. 57, no. 2, 1968, pp. 274-277.

[55] I. Ryusei, B. Golman and K. Shinohara, “Controlled release with coating layer of permeable particles”, *Journal of Controlled Release,* vol. 92, 2003, pp. 361-368.

[56] J. Marentette and A. Grosser, “Modeling of the Kinetics of Drug Release from a Binary System”, *Journal of Pharmaceutical Sciences,* vol. 81, no. 4, 1992, pp. 318-320.

[57] M. Ansari and F. Stapanek, “Design of Granule Structure: Computational Methods and Experimental Realization”, *AIChE Journal,* vol. 52, no. 11, 2006, pp. 3762-3774.

[58] U.S. Department of Health and Human Services Food and Drug Administration Center for Drug Evaluation and Research, “*Guidance for Industry: Dissolution Testing of Immediate Release Solid Oral Dosage Forms*”, Maryland: 1997.

[59] S. Saks and L. Gardner, “The pharmacoeconomic value of controlled-release dosage forms”, *Journal of Controlled Release*, vol. 48, 1997, pp. 237-242.

Appendices

Appendix A: Derivation of Dissolution of Solid Particles in a Liquid

The following mathematical model was developed by Sertsou [42] for the dissolution of solid particles in a liquid. The paper omitted several mathematical steps for deriving the needed equations, which are shown in this section. The equations were imperative to developing the MATLAB code used throughout this thesis.

Dissolution of solid particles in a liquid as described by Nernst-Brunner type kinetics:

$$\frac{\mathrm{dM}}{\mathrm{dt}} = -\frac{D}{h} \cdot A \cdot (C_s - C) \tag{A.1}$$

where M = mass of solid material at time t, k=dissolution rate constant, A = area available for mass transfer, D = diffusion coefficient of the dissolving material, and h = diffusion boundary layer thickness. For a spherical particle, the surface area is

$$A = 4\pi r^2 \tag{A.2}$$

The volume is

$$V = 4\pi r^3 / 3 \tag{A.3}$$

Therefore the change in volume

$$dV = A \cdot dr = 4\pi r^2 dr \tag{A.4}$$

Substitution of (4) into (1) and substituting r for h yields

$$\frac{dM}{dt} = \frac{N \cdot \rho \cdot dV}{dt} = \frac{N \cdot \rho \cdot 4 \cdot \pi \cdot r^2 \cdot dr}{dt} = -\frac{D}{r} \cdot N \cdot 4 \cdot \pi \cdot r^2 \cdot (C_s - C) \tag{A.5}$$

where the total mass of the particles is $M = N \cdot \rho \cdot V$, where ρ = density, N=number of particles of radius r. Canceling like terms gives:

$$\frac{\rho \cdot dr}{dt} = -\frac{D}{r}(C_s - C) \tag{A.6}$$

The mass dissolved at any time, M_d, is given as

$$M_d = M_0 - M \tag{A.7}$$

where M_0 is the initial mass of the particles, and M = mass of the particles remaining.

Appendix A: (Continued)

A mass balance of M_d divided by the dissolution medium, $V_{m,}$ gives the concentration, C

$$C = \frac{M_d}{V_m} = \frac{M_0 - M}{V_m} = \frac{N\cdot\rho\cdot\frac{4}{3}\cdot\pi\cdot r_0{}^3 - N\cdot\rho\cdot\frac{4}{3}\cdot\pi\cdot r^3}{V_m} \tag{A.8}$$

where r_0 = initial particle radius. Substituting (8) into (6) yields

$$\frac{\rho\cdot dr}{dt} = -\frac{D}{r}\cdot(C_s - \frac{N\cdot\rho\cdot\frac{4}{3}\cdot\pi\cdot r_0{}^3 - N\cdot\rho\cdot\frac{4}{3}\cdot\pi\cdot r^3}{V_m}) \tag{A.9}$$

Expanding

$$\frac{\rho\cdot dr}{dt} = -\frac{D\cdot C_s}{r} + \frac{N\cdot\rho\cdot\frac{4}{3}\cdot\pi\cdot r_0{}^3}{r\cdot V_m} - \frac{N\cdot\rho\cdot\frac{4}{3}\cdot\pi\cdot r^2}{V_m} \tag{A.10}$$

Dividing by ρ

$$\frac{dr}{dt} = \frac{-\frac{D\cdot C_s}{r} + \frac{N\cdot\rho\cdot\frac{4}{3}\cdot\pi\cdot r_0{}^3}{r\cdot V_m} - \frac{N\cdot\rho\cdot\frac{4}{3}\cdot\pi\cdot r^2}{V_m}}{\rho} \tag{A.11}$$

$$\frac{dr}{dt} = \frac{1}{3}\cdot\frac{D\cdot(-3\cdot C_s\cdot V_m + N\cdot\rho\cdot4\cdot\pi\cdot r_0{}^3 - 4\cdot N\cdot\rho\cdot\pi\cdot r^3)}{r\cdot V_m\cdot\rho} \tag{A.12}$$

Multiply by r

$$\frac{r\cdot dr}{dt} = \frac{1}{3}\cdot\frac{D\cdot(-3\cdot C_s\cdot V_m + N\cdot\rho\cdot4\cdot\pi\cdot r_0{}^3 - 4\cdot N\cdot\rho\cdot\pi\cdot r^3)}{V_m\cdot\rho} \tag{A.13}$$

Multiply by dt

$$r\cdot dr = \frac{1}{3}\cdot\frac{D\cdot(-3\cdot C_s\cdot V_m + N\cdot\rho\cdot4\cdot\pi\cdot r_0{}^3 - 4\cdot N\cdot\rho\cdot\pi\cdot r^3)}{V_m\cdot\rho}\cdot dt \tag{A.14}$$

$$r\cdot dr = \left(\frac{-3\cdot C_s\cdot V_m + N\cdot\rho\cdot4\cdot\pi\cdot r_0{}^3 - 4\cdot N\cdot\rho\cdot\pi\cdot r^3}{N\cdot4\cdot\pi\cdot\rho}\right)\cdot\left(\frac{D\cdot N\cdot4\cdot\pi}{3\cdot V_m}\right)\cdot dt \tag{A.15}$$

$$\frac{r\cdot dr}{\left(\frac{-3\cdot C_s\cdot V_m+N\cdot\rho\cdot 4\cdot\pi\cdot r_0^{\ 3}-4\cdot N\cdot\rho\cdot\pi\cdot r^3}{N\cdot 4\cdot\pi\cdot\rho}\right)}=\frac{D\cdot N\cdot 4\cdot\pi}{3\cdot V_m}\cdot dt \tag{A.16}$$

$$\frac{r\cdot dr}{\left(\frac{-3\cdot C_s\cdot V_m}{N\cdot 4\cdot\pi\cdot\rho}+r_0^{\ 3}-r^3\right)}=\frac{D\cdot N\cdot 4\cdot\pi}{3\cdot V_m}\cdot dt \tag{A.17}$$

$$\frac{r\cdot dr}{\left(r_0^{\ 3}-\frac{3\cdot C_s\cdot V_m}{N\cdot 4\cdot\pi\cdot\rho}-r^3\right)}=\frac{D\cdot N\cdot 4\cdot\pi}{3\cdot V_m}\cdot dt \tag{A.18}$$

$$\frac{r\cdot dr}{\left[\left(r_0^{\ 3}-\frac{3\cdot C_s\cdot V_m}{N\cdot 4\cdot\pi\cdot\rho}\right)^{\frac{1}{3}}\right]^3-r^3}=\frac{D\cdot N\cdot 4\cdot\pi}{3\cdot V_m}\cdot dt \tag{A.19}$$

Let

$$c=\left(r_0^{\ 3}-\frac{3\cdot C_s\cdot V_m}{N\cdot 4\cdot\pi\cdot\rho}\right)^{\frac{1}{3}} \tag{A.20}$$

Then

$$\frac{r\cdot dr}{c^3-r^3}=\frac{D\cdot N\cdot 4\cdot\pi}{3\cdot V_m}\cdot dt \tag{A.21}$$

rearranging gives

$$\frac{r\cdot dr}{c^3-r^3}=\frac{D\cdot N\cdot 4\cdot\pi}{3\cdot V_m}\cdot dt \tag{A.22}$$

where

$$c=\left(r_0^{\ 3}-\frac{3\cdot C_s\cdot V_m}{N\cdot\rho\cdot 4\cdot\pi}\right)^{\frac{1}{3}} \tag{A.23}$$

c is a constant with respect to time. Eq. (A.21) describes the rate of change of the particle's radius with respect to time. Integration of the left hand side yields

$$\int \frac{r \cdot dr}{c^3 - r^3} = \left[\begin{array}{l} -\frac{1}{3} \cdot \frac{\log(r-c)}{c} + \frac{1}{6} \cdot \frac{\log(r^2 + r \cdot c + c^2)}{c} \\ -\frac{1}{3} \cdot \frac{\sqrt{3} \tan^{-1}\left(\frac{1}{3} \cdot \frac{(2 \cdot r + c) \cdot \sqrt{3}}{c} \right)}{c} \end{array} \right] + k \qquad \text{(A.24)}$$

where k = integration constant. Simplifying

$$\int \frac{r \cdot dr}{c^3 - r^3} =$$

$$\left[-\frac{1}{6} \cdot \frac{2 \cdot \log(r-c) - \log(r^2 + r \cdot c + c^2) + 2\sqrt{3} \cdot \tan^{-1}\left(\frac{1}{3} \cdot \frac{(2 \cdot r + c) \cdot \sqrt{3}}{c} \right)}{c} \right] + k \qquad \text{(A.25)}$$

Expand and rearrange

$$\int \frac{r \cdot dr}{c^3 - r^3} = \left[\begin{array}{l} -\frac{1}{3} \cdot \frac{\log(r-c)}{c} + \frac{1}{6} \cdot \frac{\log(r^2 + r \cdot c + c^2)}{c} - \\ \frac{1}{3} \cdot \frac{\sqrt{3} \tan^{-1}\left(\frac{2\sqrt{3} \cdot r}{3c} + \frac{\sqrt{3}}{3} \right)}{c} \end{array} \right] + k \qquad \text{(A.26)}$$

For initial conditions where r = r_0 at t =0

$$\int \frac{D \cdot N \cdot 4 \cdot \pi}{3 \cdot V_m} \cdot dt = \int \frac{r \cdot dr}{c^3 - r^3} \qquad \text{(A.27)}$$

$$\frac{D \cdot N \cdot 4 \cdot \pi}{3 \cdot V_m} \cdot t = \left[\begin{array}{l} -\frac{1}{3} \cdot \frac{\log(r-c)}{c} + \frac{1}{6} \cdot \frac{\log(r^2 + r \cdot c + c^2)}{c} - \\ \frac{1}{3} \cdot \frac{\sqrt{3} \tan^{-1}\left(\frac{2\sqrt{3} \cdot r}{3c} + \frac{\sqrt{3}}{3} \right)}{c} \end{array} \right] + k \qquad \text{(A.28)}$$

Then since r = r_0 at t =0

$$k=\frac{1}{3}\cdot\frac{\log(r_0-c)}{c}-\frac{1}{6}\cdot\frac{\log(r_0{}^2+r_0\cdot c+c^2)}{c}+\frac{1}{3}\cdot\frac{\sqrt{3}\tan^{-1}\left(\frac{2\sqrt{3}\cdot r_0}{3c}+\frac{\sqrt{3}}{3}\right)}{c} \qquad \text{(A.29)}$$

To solve for time,

$$t=\frac{3\cdot V_m}{D\cdot N\cdot 4\cdot\pi}\left[\begin{array}{l}-\frac{1}{3}\cdot\frac{\log(r-c)}{c}+\frac{1}{6}\cdot\frac{\log(r^2+r\cdot c+c^2)}{c}-\\ \frac{1}{3}\cdot\frac{\sqrt{3}\tan^{-1}\left(\frac{2\sqrt{3}\cdot r}{3c}+\frac{\sqrt{3}}{3}\right)}{c}\\ +\frac{1}{3}\cdot\frac{\log(r_0-c)}{c}-\frac{1}{6}\cdot\frac{\log(r_0{}^2+r_0\cdot c+c^2)}{c}\\ +\frac{1}{3}\cdot\frac{\sqrt{3}\tan^{-1}\left(\frac{2\sqrt{3}\cdot r_0}{3c}+\frac{\sqrt{3}}{3}\right)}{c}\end{array}\right] \qquad \text{(A.30)}$$

$$\int\frac{r\cdot dr}{c^3-r^3}=\frac{1}{6c}\ln\frac{c^3-r^3}{(c-r)^3}-\frac{1}{\sqrt{3}c}\tan^{-1}\frac{2r+c}{\sqrt{3}c}-k=\frac{D\cdot N\cdot 4\cdot\pi}{3\cdot V_m}\cdot t \qquad \text{(A.31)}$$

where k = constant of integration. Applying the initial conditions of r = r_0 at t= 0

$$k=\frac{1}{6c}\ln\frac{c^3-r_0{}^3}{(c-r_0)^3}-\frac{1}{\sqrt{3}c}\tan^{-1}\frac{2r_0+c}{\sqrt{3}c} \qquad \text{(A.32)}$$

For the dissolution of large particles until the time when diffusion boundary layer thickness begins to decrease with decreasing particle radius, the following differential equation is applicable

$$\frac{dr}{c^3-r^3}=\frac{D\cdot N\cdot 4\cdot\pi}{h\cdot 3\cdot V_m}\cdot dt \qquad \text{(A.33)}$$

where h = an approximate constant diffusion boundary layer thickness. Integration of (14) yields

$$\int \frac{dr}{c^3-r^3} = -\frac{1}{6c^2}\ln\frac{c^3-r^3}{(c-r)^3}+\frac{1}{\sqrt{3}c^2}\tan^{-1}\frac{2r+c}{\sqrt{3}c}-k_1=\frac{D\cdot N\cdot 4\cdot \pi}{h\cdot 3\cdot V_m}\cdot t \qquad \text{(A.34)}$$

where k_1 = the constant of integration. Applying initial conditions of $r = r_0$ at $t= 0$

$$k_1=\frac{1}{6c^2}\ln\frac{c^3-{r_0}^3}{(c-r_0)^3}+\frac{1}{\sqrt{3}c^2}\tan^{-1}\frac{2r_0+c}{\sqrt{3}c} \qquad \text{(A.35)}$$

For cases when the diffusion boundary layer thickness is decreasing and approximated by the particle radius,

$$t=\frac{\left[\frac{1}{6c^2}\ln\frac{(c-r_0)^3}{(c^3-r_0)^3}+\frac{1}{\sqrt{3}c}\left[\tan^{-1}\frac{2r_0+c}{\sqrt{3}c}-\tan^{-1}\frac{1}{\sqrt{3}}\right]\right]\cdot 3\cdot V_m}{D\cdot N\cdot 4\cdot \pi} \qquad \text{(A.36)}$$

this corresponds to finding the root of Eq. (A.32). For cases when the diffusion boundary layer thickness is constant, the time for particles to dissolve corresponds to

$$t=\frac{\left[\begin{array}{l}\frac{1}{6c^2}\ln\frac{(c-r_0)^3\cdot(c^3-r^3)}{(c^3-r_0)^3\cdot(c^3-r^3)}+\\ \frac{1}{\sqrt{3}c^2}\left[\tan^{-1}\frac{2r+c}{\sqrt{3}c}-\tan^{-1}\frac{2r_0+c}{\sqrt{3}c}\right]\end{array}\right]\cdot h\cdot 3\cdot V_m}{D\cdot N\cdot 4\cdot \pi} \qquad \text{(A.37)}$$

Appendix B: MATLAB Sample Source Code

The following is the MATLAB source code for the monodisperse particle model. The function *Start.m* initiates the program to run from the command window. The resulting plots seen after the program has completed calculations is shown at the end of this section.

```
% MONDISPERSE PARTICLE DISSOLUTION
% START.m 2007
%          START calls the three subrountines in this program
%               1) NUMPART calculates the total number of particles
%               2) DIFFPART finds the diffusion coefficient
%               3) FINDRADIUS calculates the radius at each given time
%               4) CONCCONV calculates the concentration and conversion
%                   based on the radius
%               5) PRINTRESULTS creates table including time, conversion
%                   and concentration
% Plots graph of conversion and concentration versus time

global Cs D12 N r0 rho t Vm

% NUMPART calculates the total number of particles
NumPart
% DIFFPART finds the diffusion coefficient
DiffPart
% FINDRADIUS calculates the radius at each given time
FindRadius
% CONCCONV calculates the concentration and conversion based on the radius
ConcConv
% PRINTRESULTS creates table including time, conversion and concentration
PrintResults
```

```
% MONDISPERSE PARTICLE DISSOLUTION
% NUMPART.m 2007
%          NUMPART calculates the number of particles

% Given parameters
%Initial Radius of One Particle
r0 = [0.06];    % cm
%Total Grams of Citric Acid, Mt
Mt = [0.1086];       % grams
%Density of particle, rho
rho = 1.665;        % g/cm^3

% Calculations
%Total Volume of Citric Acid, Volume_CA
Vt = Mt*(1/rho);      % cm^3
%Volume of One Particle, Particle_Volume
Vp = (4/3)*pi*r0.^3;      % cm^3
%Number of particles, N
N = Vt./Vp;    % particles of Citric Acid
```

Appendix B: (Continued)

```
% MONDISPERSE PARTICLE DISSOLUTION
% DIFFPART.m 2007
%          DIFFPART calculates the diffusion coefficient for the particle
%          in solution

global Cs N r0 rho t Vm

% ESTIMATION OF DIFFUSION COEFFIECENT
% (Properties of Gases and Liquids Reid, R.C. and Sherwood T.K)
% Wilke and Chang
% Given Parameters
% Molecular weight of solvent, M2
M2 = 18.015;            % grams
% Temperature, T
Temp = 318.15;                % Kelvin
% Viscosity of solution (solvent), v2
v2 = 0.91;              % centipoises
% Molecular weight of solute, M1
M1 = 192.12;            % grams
% Molal volume of the solute at its normal boiling point, V1
V1 = rho*M1;             % cm^3/g mole
% Association parameter of solvent, phi
phi = 2.6;              % recommended by Wilke and Chang for water
% Calculation of diffusion coefficients
% Mutual diffusion of solute 1 in solvent 2 at very low solute
% concentration (cm^2/sec), D12
D12 = 7.4e-8*(((phi*M2)^(1/2)*Temp)/(v2*V1^0.6));
```

```
% MONDISPERSE PARTICLE DISSOLUTION
% FINDRADIUS.m 2007
%          FINDRADIUS creates a loop between 1-10000 seconds in increments
%          of 100 and calculates the radius as each time.  The radius
%          is calculated using the fzero function for the equation in
%          FinalDiss.

global Cs D12 N r0 rho t Vm

% Initial time to start loop
t = 0;
% Loop set from 1-100
for i = 1:100
% Assign value for time
Time(i)= t;
% Solve function for radius at each given time
CalR = fzero(@Function,r0);

% Check that calculated radius is not negative, and if it is, assign
% the particle radius to equal zero.
if CalR < 0
    r(i) = 0;
else
    r(i) = CalR;
end
% Update increment t
t = t + 100; end
```

Appendix B: (Continued)

```
% MONDISPERSE PARTICLE DISSOLUTION
% FUNCTION.m 2007
%          FUNCTION calculates the constant, c, and the integration
%          constant, k, for the particle and the final eqaution is written as
%          function F.

function F = Function(r)

global Cs D12 N r0 rho t Vm

% CALCULATION OF CONSTANT, c
%Solubility of the solute, Cs
Cs = 1.33;          % g/mL
%Dissolution Medium Volume, Vm
Vm = 1;             % 1 mL H2O
%Constant with respect to time, c
c =nthroot((r0.^3.-(3.*Cs*Vm)./(N.*rho*4*pi)),3);

% CONSTANT OF INTEGRATION, k
k = (1./(6*c)).*log((c.^3-r0.^3.)/(c-r0).^3.)-
(1./(sqrt(3)*c)).*(atan((2*r0+c)./(sqrt(3)*c)));

% FINAL FUNCTION RELATING TIME AND RADIUS
F=(((1./(6*c)).*log((c.^3-r.^3.)./(c-r).^3.)
(1./(sqrt(3)*c))*(atan((2*r+c)./(sqrt(3)*c))))-k)-(((D12*N*4*pi)*t)/(3*Vm));

end
```

```
% MONDISPERSE PARTICLE DISSOLUTION
% CONCCONV.m 2007
%          CONCCONV calculates the concentration and conversion using the
%          radius found in FINDRADIUS.

global Cs D12 N r0 rho t Vm

%Initial Mass of Particles, Mo
Mo = (r0.^3)*(4/3)*rho.*N*pi;
%Mass of Particles Remaining, M
M = N*rho*(4/3)*pi*r.^3;
%Concentration of solid, C1
C1 = (M)/Vm;          % g/mL
%Conversion, X
Md=(Mo-M);
if (Md==0)
X=1;
else
X = ((Mo-M)/Vm)/(Mo/Vm);
end
```

```
% MONDISPERSE PARTICLE DISSOLUTION
% PRINTRESULTS.m 2007
%          PRINTRESULTS converts time into minutes then displays
%          concentration and conversion in a table.  Then plots of the
%          concentration and conversion of citric acid are displayed.
```

```
global Cs D12 N r0 rho t Vm

%Time Converted to Minutes
TimeMin=Time/60;

PRINT RESULTS
plot(TimeMin, C1,'-r','LineWidth',3)
xlabel('Time(min)'); ylabel('Concentration of Citric Acid (g/mL)');title(‘Monodisperse
Model Concentration versus Time ');grid on
hold on;
figure;
plot(TimeMin, X,'Color',[1 0.2 0.2],'LineWidth',3)
xlabel('Time(min)'); ylabel('Conversion of Citric Acid (X)');title('Monodisperse Model
Conversion versus Time');grid on
ylim([0 1.1]);
xlim([0 20]);
hold off;
```

Two graphs are produced by this program which are displayed in the command window. The first graph shows the concentration of citric acid as a function of time:

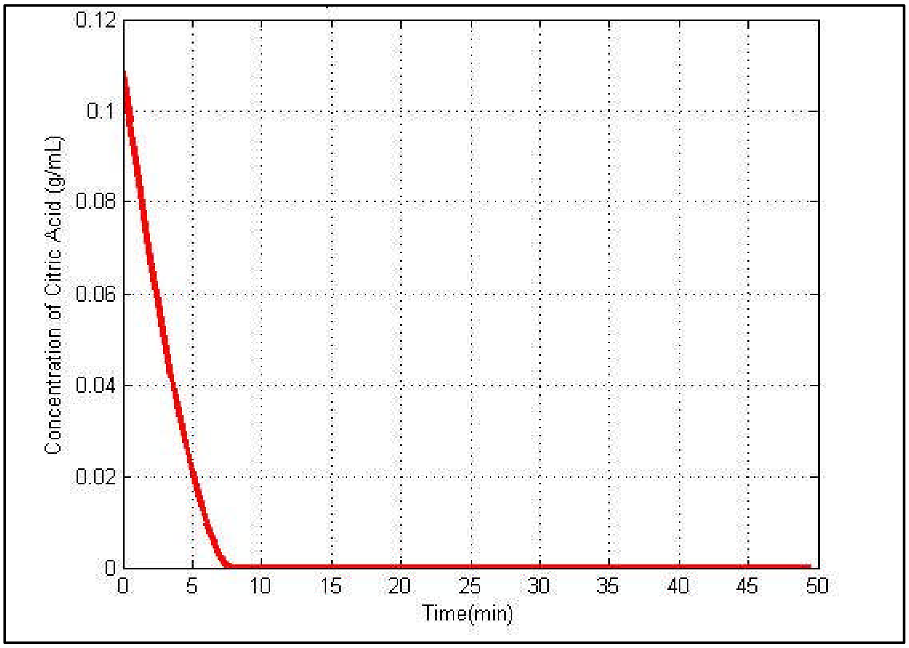


Figure B.1: MATLAB resulting plot for concentration for monodisperse model.

The second graph shows the conversion of citric acid as a function of time:

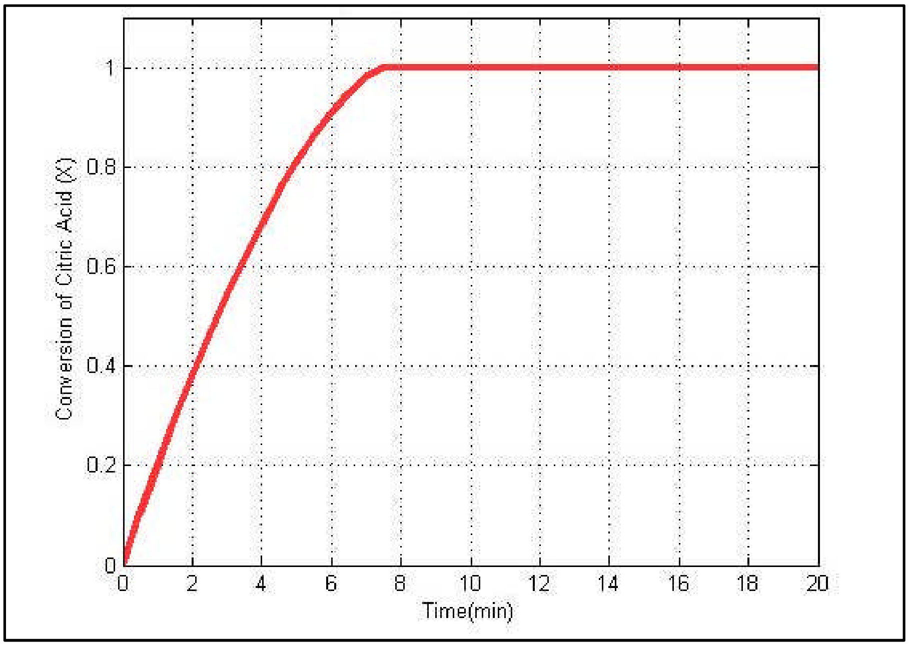


Figure B.2: MATLAB resulting plot for conversion for monodisperse model.

The total time for conversion was under 8 minutes for the monodisperse model.

Appendix C: COMSOL Sample Source Code

This section describes how to import the data results calculated by MATLAB into a predesigned function in COMSOL Script which creates three-dimensional visualizations of the shrinking spherical particle.

The first step involves copying the data vectors for variables *r0*and *Pr* from the Workspace window in MATLAB and inputting the data into the Command Prompt window in COMSOL. The following figure demonstrates this technique.

```
---------------------------------------------------------
     COMSOL Script 1.1.0.511
     Copyright (c) COMSOL AB 1994-2007
     Type 'help' to see available functions.
---------------------------------------------------------
C» ParticleR= [0.06     0.057891        0.055721        0.053481        0.051161
        0.048748        0.046228        0.04358 0.040778        0.037787        0.034557
        0.031009        0.027018        0.022345        0.016411        0.0063142       0
        0       0       0       0       0       0       0       0       0       0       0
        0       0       0       0       0       0       0       0       0       0       0
        0       0       0       0       0       0       0       0       0       0       0
        0       0       0       0       0       0       0       0       0       0       0
        0       0       0       0       0       0       0       0       0       0       0
        0       0       0       0       0       0       0       0       0       0       0
        0       0       0       0       0       0       0       0       0       0       0
        0       0       0       0       0       0];
C» r0=0.06;
```

Figure C.1: Command prompt window in COMSOL Script.

After the data values are inputted, the function *MonodisperseMOVIE* is called. This function parametrically converts the radius data inputted in the *ParticleR* variable vector. It also displays the three-dimensional sphere and corresponding plot for a give time. Animation capabilities are also available, since each frame is saved in avi format and can be played using Windows Media Player.

The following is a sample source code for the function which generates three-dimensional spherical animations of the particles for a given time range:

```
% MONODISPERSE MODEL MOVIE
global r0 ParticleR
phi=0:pi/20:pi;
theta=0:pi/10:2*pi;
[Phi,Theta]=meshgrid(phi,theta);
%Equations used in parametrization of sphere
X=sin(Phi).*cos(Theta);
Y=sin(Phi).*sin(Theta);
Z=cos(Phi);
```

```
%Plot the surface, and scale the axes so that it looks like a sphere
h = surf(X,Y,Z,'EdgeColor','interp','FaceColor',[1 0.2 0.2]);
axis([-r0 r0 -r0 r0 -r0 r0]);
% Save the frames to create a movie
m=movie('width',800,'height',600);
for i=1:60
delete(h)
a=ParticleR(i);
X1=a*X;
Y1=a*Y;
Z1=a*Z;
h = surf(X1,Y1,Z1,'EdgeColor','interp','FaceColor',[1 0.2 0.2]);
axis([-r0 r0 -r0 r0 -r0 r0]);
title(['Particle at t = ', num2str(i), 'sec']);
grid on;
m.addFrame;
% Creation of animation file
m.generate('Monodisperse Model Movie');
end
```

The resulting plot and animation will be represented graphically on the a grid resembling the one seen below.

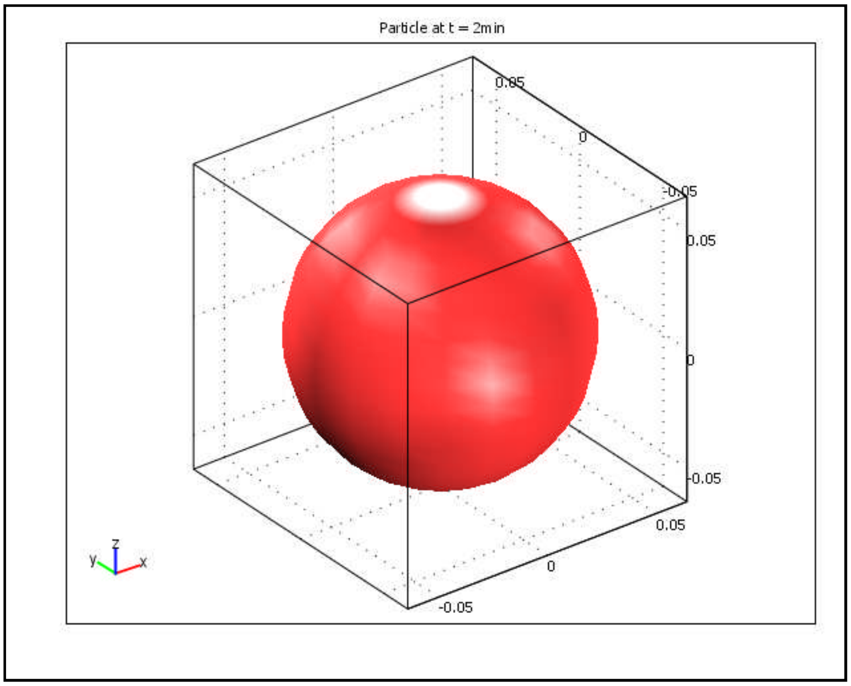


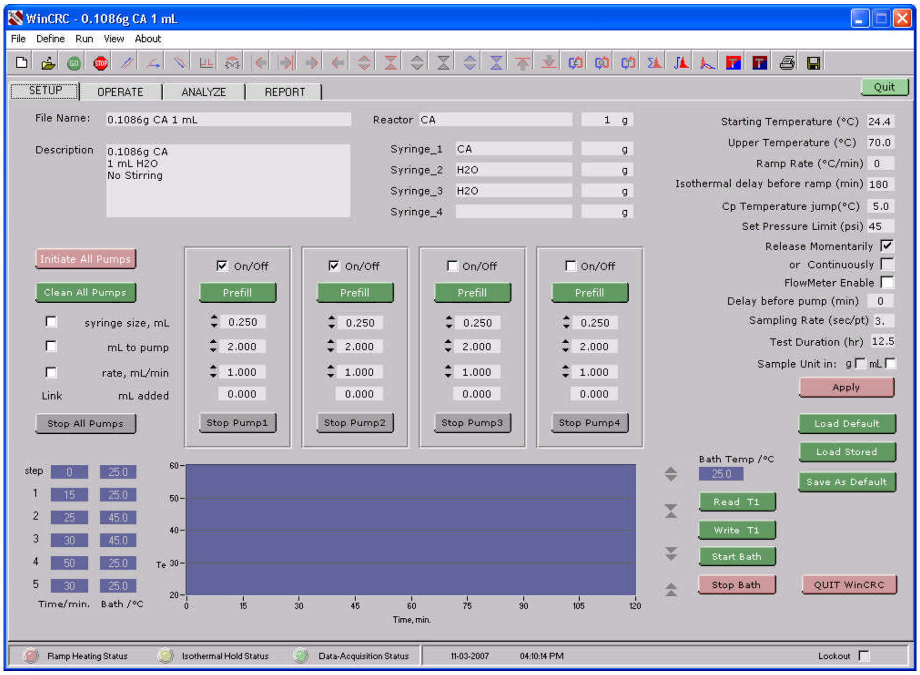
Figure C.2: COMSOL Script visualization of spherical particle.

Appendix D: Experimental Procedure using Reaction Microcalorimeter

For the dissolution studies of sold particles in liquid, an OmniCal Technologies SuperCRC 20-305-2.4 reaction microcalorimeter was used. Individual samples of citric acid were weighed directly into the 16 mL glass vials. Anhydrous citric acid, $C_6H_8O_{7,}$ was obtained from Fisher Chemicals. A Teflon-lined screw-top was used to seal the vials and used to avoid any loss of material during the experiment. Vial-picking needles were used to place the vials into the CRC unit. The liquid samples, 1 mL of water, were measured using into two syringes and placed into the syringe sample barrels. The CRC unit is turned on and the samples and syringe barrels were allowed to come to thermal equilibrium with the calorimeter heat sink temperature, usually at least 30 minutes were given for this step.

The CRC calorimeter operates on-line with the program software WinCRC Turbo, which is executed after the samples have been inserted into the unit and the unit is turned on. The operational program WinCRC Turbo is the microcalorimeter operational program. The setup window is shown in Figure D.1. WinCRC Turbo was used for both data acquisition and conversion calculations. Samples were monitored until a smooth baseline in the heat flow was maintained. Then the liquid was injected into both the sample and reference vial, and the detection and collection of data was maintained until no changes were seen in the heat flow for at least 30 minutes. An example of a resulting heat curve is shown in Figure D.2.

Appendix D: (Continued)



File Name: 0.1086g CA 1 mL

Description: 0.1086g CA
1 mL H2O
No Stirring

Reactor CA 1 g

Syringe_1 CA g
Syringe_2 H2O g
Syringe_3 H2O g
Syringe_4 g

	Pump 1	Pump 2	Pump 3	Pump 4
	☑ On/Off	☑ On/Off	□ On/Off	□ On/Off
□ syringe size, mL	0.250	0.250	0.250	0.250
□ mL to pump	2.000	2.000	2.000	2.000
□ rate, mL/min	1.000	1.000	1.000	1.000
Link mL added	0.000	0.000	0.000	0.000

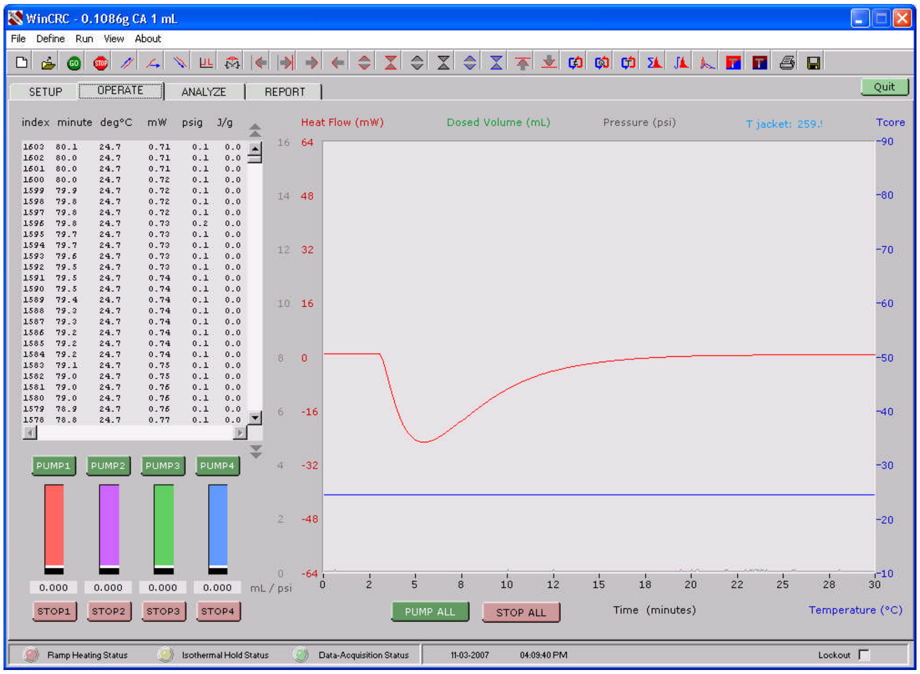
Starting Temperature (°C) 24.4
Upper Temperature (°C) 70.0
Ramp Rate (°C/min) 0
Isothermal delay before ramp (min) 180
Cp Temperature jump(°C) 5.0
Set Pressure Limit (psi) 45
Release Momentarily ☑
or Continuously □
FlowMeter Enable □
Delay before pump (min) 0
Sampling Rate (sec/pt) 3.
Test Duration (hr) 12.5
Sample Unit in: g □ mL □

step	Time/min.	Bath /°C
	0	25.0
1	15	25.0
2	25	45.0
3	30	45.0
4	50	25.0
5	30	25.0

Bath Temp /°C 25.0

11-03-2007 04:10:14 PM

Figure D.1: WinCRC Turbo microcalorimeter program setup window.



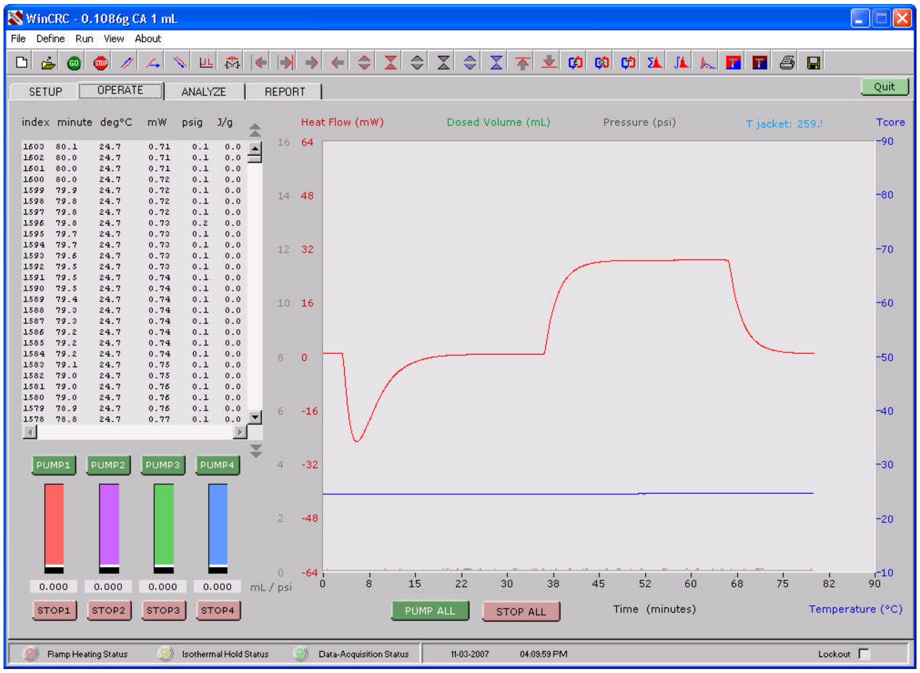
```
index  minute  deg°C   mW    psig   J/g
1603   80.1    24.7    0.71   0.1   0.0
1602   80.0    24.7    0.71   0.1   0.0
1601   80.0    24.7    0.71   0.1   0.0
1600   80.0    24.7    0.72   0.1   0.0
1599   79.9    24.7    0.72   0.1   0.0
1598   79.8    24.7    0.72   0.1   0.0
1597   79.8    24.7    0.72   0.1   0.0
1596   79.8    24.7    0.73   0.2   0.0
1595   79.7    24.7    0.73   0.1   0.0
1594   79.7    24.7    0.73   0.1   0.0
1593   79.6    24.7    0.73   0.1   0.0
1592   79.5    24.7    0.73   0.1   0.0
1591   79.5    24.7    0.74   0.1   0.0
1590   79.5    24.7    0.74   0.1   0.0
1589   79.4    24.7    0.74   0.1   0.0
1588   79.3    24.7    0.74   0.1   0.0
1587   79.3    24.7    0.74   0.1   0.0
1586   79.2    24.7    0.74   0.1   0.0
1585   79.2    24.7    0.74   0.1   0.0
1584   79.2    24.7    0.74   0.1   0.0
1583   79.1    24.7    0.75   0.1   0.0
1582   79.0    24.7    0.75   0.1   0.0
1581   79.0    24.7    0.76   0.1   0.0
1580   79.0    24.7    0.76   0.1   0.0
1579   78.9    24.7    0.76   0.1   0.0
1578   78.8    24.7    0.77   0.1   0.0
```

0.000 0.000 0.000 0.000

11-03-2007 04:09:40 PM

Figure D.2: Heat flow curve from reaction.

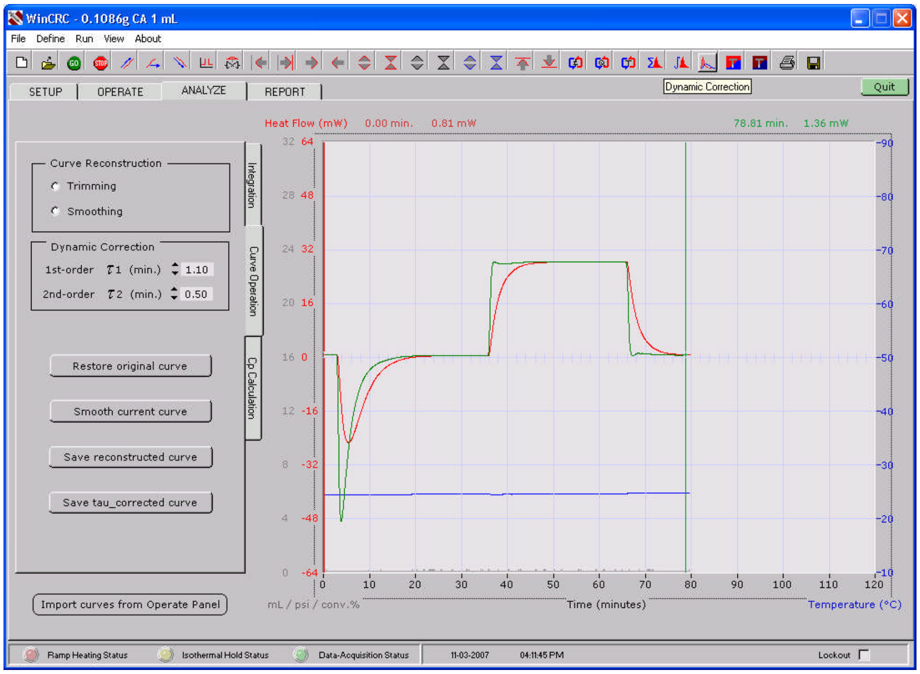
The following steps were preformed so that a dynamic correction could later be calculated for the data. Since the heat rate curves were to be used for the reaction kinetics, it was important to have the most accurate heat flow peak values and curve fitting of the heat flow data. Therefore, the thermal lag, or thermal inertia of the heat conduction reactor station system, was measured and accounted for[51]. After the reaction heat flow signal remains at the baseline for 30 minutes, as the previous step mentioned, the calibration heater was turned on for 30 minutes and the heat flow was allowed to stabilized. Then the heater was switched off and 10 minutes are allotted for the calibration heat flow signal to return to the baseline. The data collection is then stopped and saved. Figure D.3 shows an example of the resulting graph.



index	minute	deg°C	mW	psig	J/g
1603	80.1	24.7	0.71	0.1	0.0
1602	80.0	24.7	0.71	0.1	0.0
1601	80.0	24.7	0.71	0.1	0.0
1600	80.0	24.7	0.72	0.1	0.0
1599	79.9	24.7	0.72	0.1	0.0
1598	79.8	24.7	0.72	0.1	0.0
1597	79.8	24.7	0.72	0.1	0.0
1596	79.8	24.7	0.73	0.2	0.0
1595	79.7	24.7	0.73	0.1	0.0
1594	79.7	24.7	0.73	0.1	0.0
1593	79.6	24.7	0.73	0.1	0.0
1592	79.5	24.7	0.73	0.1	0.0
1591	79.5	24.7	0.74	0.1	0.0
1590	79.5	24.7	0.74	0.1	0.0
1589	79.4	24.7	0.74	0.1	0.0
1588	79.3	24.7	0.74	0.1	0.0
1587	79.3	24.7	0.74	0.1	0.0
1586	79.2	24.7	0.74	0.1	0.0
1585	79.2	24.7	0.74	0.1	0.0
1584	79.2	24.7	0.74	0.1	0.0
1583	79.1	24.7	0.75	0.1	0.0
1582	79.0	24.7	0.75	0.1	0.0
1581	79.0	24.7	0.76	0.1	0.0
1580	79.0	24.7	0.76	0.1	0.0
1579	78.9	24.7	0.76	0.1	0.0
1578	78.8	24.7	0.77	0.1	0.0

Figure D.3: Dynamic correction using heat curve option.

The data was analyzed in the WinCRC Turbo and the curve was imported from the operational panel. In the curve operation menu, the red curser was dragged to the position 5 minutes in front of the heating peak and the green cursor was moved to 5 minutes after the peak. In the dynamic corrections section, values for the 1st order τ_1 and 2nd order τ_2 time constants are manipulated until a square curve results. The tau corrected curve is then saved. To apply the determined correction to the actual heat flow data, the red cursor is moved to 5 minutes in front of the actual reaction heat peak and green cursor is moved to 5 minutes after the peak. Using the same correction constants, the dynamic correction icon in the toolbar is clicked and a corrected heat rate curve will be generated, as shown in Figure D.4. The tau corrected heat flow data should then be saved.

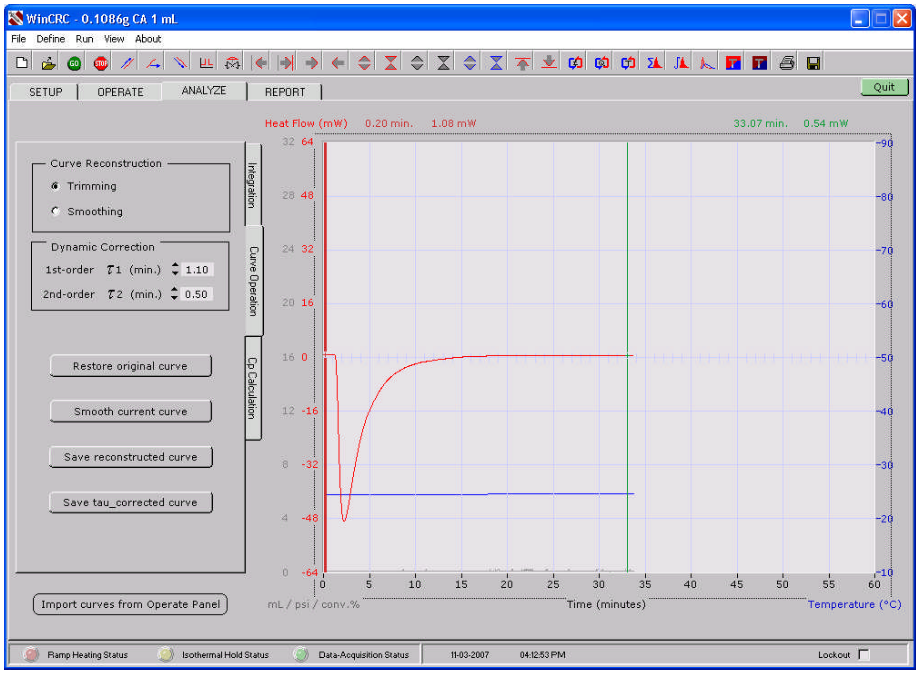


1st-order τ1 (min.) 1.10

2nd-order τ2 (min.) 0.50

Figure D.4: Applying tau correction to reaction heat flow curve.

In another program menu, open the saved data and import the curve. The start and end points of the curve should be selected. In the integration menu, the baseline construction selection should be point-to-point nonlinear. The integration should be calculated and the total should be recorded. The data should be trimmed, starting from the point of injection and ending about 30 minutes after the heat peak, using the curve operation as shown in Figure D.5. The trimmed curve will have to be saved as a reconstructed curve.


WinCRC - 0.1086g CA 1 mL

Curve Reconstruction

● Trimming
○ Smoothing

Dynamic Correction

1st-order τ1 (min.) 1.10
2nd-order τ2 (min.) 0.50

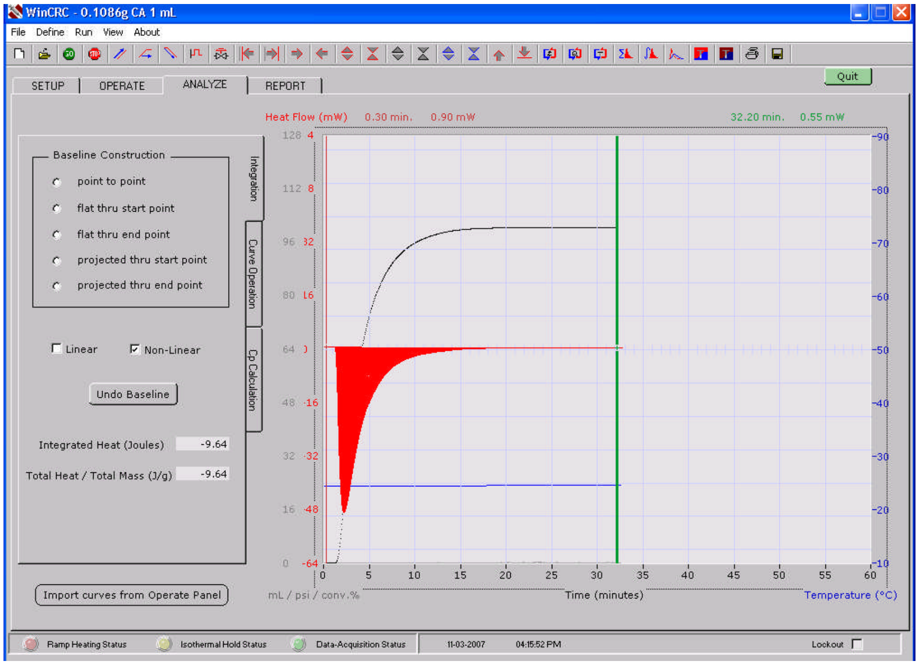
Heat Flow (mW) 0.20 min. 1.08 mW

33.07 min. 0.54 mW

11-03-2007 04:12:53 PM

Figure D.5: Trimmed corrected heat flow curve.

Another program menu must be opened and the curve should be reopened. Using the red cursor select the starting point and with the green cursor select the end point. The process of integration should be repeated, with the construction of a baseline point-to-point nonlinear.



☐ Linear ☑ Non-Linear

Integrated Heat (Joules) -9.64

Total Heat / Total Mass (J/g) -9.64

Figure D.6: Integrated heat flow curve and conversion.

The integration should be recalculated and compared to the total integration previously found, so that the value can be validated. If the value calculated is reasonable, the File menu is opened and the save conversion data option is selected. The conversion data can also be seen in the program window as a black line, as shown in Figure D.6.

Appendix E: Sample Data

This section includes sample data obtained experimentally compared with values calculated from the computer models. Data for citric acid amounts of 0.02-0.20 grams in 1 mL H2O at 298K with no stirring is given in the following tables. Data for 0.02 grams of citric acid:

Time(mins)	Experiment	Model 1: Monodisperse	Model 2: Polydisperse	Residual 1	Residual 2
0	0.0001	0	0	0.0001	0.0001
0.5	0.0108	0.10545	0.094921	0.09465	0.084121
1	0.179	0.2066	0.18576	0.0276	0.00676
1.5	0.4091	0.30334	0.27239	0.10576	0.13671
2	0.5841	0.39554	0.35461	0.18856	0.22949
2.5	0.7054	0.48303	0.43218	0.22237	0.27322
3	0.789	0.56563	0.50471	0.22337	0.28429
3.5	0.8478	0.64311	0.57132	0.20469	0.27648
4	0.8901	0.71518	0.63255	0.17492	0.25755
4.5	0.9211	0.78148	0.68829	0.13962	0.23281
5	0.943	0.84153	0.73724	0.10147	0.20576
5.5	0.9587	0.8947	0.777	0.064	0.1817
6	0.97	0.94	0.81281	0.03	0.15719
6.5	0.9778	0.97582	0.84459	0.00198	0.13321
7	0.9841	0.99834	0.8711	0.01424	0.113
7.5	0.9889	1	0.88992	0.0111	0.09898
8	0.9926	1	0.90692	0.0074	0.08568
8.5	0.9956	1	0.92246	0.0044	0.07314
9	0.9983	1	0.93637	0.0017	0.06193
9.5	1.0003	1	0.94833	0.0003	0.05197
10	1.0019	1	0.95745	0.0019	0.04445
10.5	1.0037	1	0.96325	0.0037	0.04045
11	1.0049	1	0.96863	0.0049	0.03627
11.5	1.0058	1	0.97355	0.0058	0.03225
12	1.0067	1	0.97793	0.0067	0.02877
12.5	1.0072	1	0.98169	0.0072	0.02551
13	1.0079	1	0.98458	0.0079	0.02332
13.5	1.0086	1	0.98612	0.0086	0.02248
14	1.0091	1	0.98761	0.0091	0.02149
14.5	1.0097	1	0.98904	0.0097	0.02066
15	1.01	1	0.9904	0.01	0.0196
			Sum of Residuals	1.69373	3.259341

Appendix E: (Continued)

Data for 0.04 grams of citric acid

Time(mins)	Experiment	Model 1: Monodisperse	Model 2: Polydisperse	Residual 1	Residual 2
0	0.0001	0	0	0.0001	0.0001
0.5	0.0108	0.10554	0.094988	0.09474	0.084188
1	0.179	0.20692	0.18601	0.02792	0.00701
1.5	0.4091	0.30401	0.27291	0.10509	0.13619
2	0.5841	0.39665	0.35547	0.18745	0.22863
2.5	0.7054	0.48465	0.43341	0.22075	0.27199
3	0.789	0.5678	0.5063	0.2212	0.2827
3.5	0.8478	0.64583	0.5732	0.20197	0.2746
4	0.8901	0.71842	0.63474	0.17168	0.25536
4.5	0.9211	0.78517	0.69067	0.13593	0.23043
5	0.943	0.84555	0.73953	0.09745	0.20347
5.5	0.9587	0.89885	0.77921	0.05985	0.17949
6	0.97	0.944	0.81511	0.026	0.15489
6.5	0.9778	0.97917	0.84683	0.00137	0.13097
7	0.9841	1	0.87305	0.0159	0.11105
7.5	0.9889	1	0.89144	0.0111	0.09746
8	0.9926	1	0.90847	0.0074	0.08413
8.5	0.9956	1	0.92402	0.0044	0.07158
9	0.9983	1	0.93787	0.0017	0.06043
9.5	1.0003	1	0.94966	0.0003	0.05064
10	1.0019	1	0.95816	0.0019	0.04374
10.5	1.0037	1	0.96397	0.0037	0.03973
11	1.0049	1	0.96935	0.0049	0.03555
11.5	1.0058	1	0.97424	0.0058	0.03156
12	1.0067	1	0.97858	0.0067	0.02812
12.5	1.0072	1	0.98224	0.0072	0.02496
13	1.0079	1	0.98481	0.0079	0.02309
13.5	1.0086	1	0.98636	0.0086	0.02224
14	1.0091	1	0.98786	0.0091	0.02124
14.5	1.0097	1	0.98929	0.0097	0.02041
15	1.01	1	0.99066	0.01	0.01934
			Sum of Residuals	1.6678	3.225288

Appendix E: (Continued)

Data for 0.06 grams of citric acid

Time(mins)	Experiment	Model 1: Monodisperse	Model 2: Polydisperse	Residual 1	Residual 2
0	0.0001	0	0	0.0001	0.0001
0.5	0.0377	0.10536	0.094838	0.06766	0.057138
1	0.1721	0.20627	0.18545	0.03417	0.01335
1.5	0.3222	0.30263	0.27174	0.01957	0.05046
2	0.4506	0.39435	0.35354	0.05625	0.09706
2.5	0.5546	0.48129	0.43066	0.07331	0.12394
3	0.6382	0.5633	0.50276	0.0749	0.13544
3.5	0.7054	0.64019	0.569	0.06521	0.1364
4	0.7598	0.7117	0.62985	0.0481	0.12995
4.5	0.8036	0.77752	0.68533	0.02608	0.11827
5	0.839	0.83721	0.73435	0.00179	0.10465
5.5	0.8685	0.89021	0.77428	0.02171	0.09422
6	0.8928	0.93564	0.80996	0.04284	0.08284
6.5	0.9128	0.97207	0.84179	0.05927	0.07101
7	0.9293	0.99637	0.86872	0.06707	0.06058
7.5	0.9433	1	0.88805	0.0567	0.05525
8	0.9546	1	0.90498	0.0454	0.04962
8.5	0.9641	1	0.92052	0.0359	0.04358
9	0.9719	1	0.93449	0.0281	0.03741
9.5	0.9781	1	0.94663	0.0219	0.03147
10	0.9832	1	0.95632	0.0168	0.02688
10.5	0.9873	1	0.96235	0.0127	0.02495
11	0.9908	1	0.96773	0.0092	0.02307
11.5	0.9935	1	0.97267	0.0065	0.02083
12	0.996	1	0.97711	0.004	0.01889
12.5	0.9981	1	0.98097	0.0019	0.01713
13	0.9996	1	0.98405	0.0004	0.01555
13.5	1.0013	1	0.98583	0.0013	0.01547
14	1.0026	1	0.98731	0.0026	0.01529
14.5	1.0037	1	0.98873	0.0037	0.01497
15	1.0047	1	0.99009	0.0047	0.01461
			Sum of Residuals	0.90983	1.700378

Appendix E: (Continued)

Data for 0.08 grams of citric acid

Time(mins)	Experiment	Model 1: Monodisperse	Model 2: Polydisperse	Residual 1	Residual 2
0	0.0002	0	0	0.0002	0.0002
0.5	0.0617	0.10527	0.094755	0.04357	0.033055
1	0.2249	0.20593	0.18514	0.01897	0.03976
1.5	0.3901	0.30191	0.27109	0.08819	0.11901
2	0.5247	0.39315	0.35248	0.13155	0.17222
2.5	0.6293	0.47954	0.42914	0.14976	0.20016
3	0.71	0.56097	0.50078	0.14903	0.20922
3.5	0.7721	0.63726	0.56667	0.13484	0.20543
4	0.82	0.7082	0.62713	0.1118	0.19287
4.5	0.8566	0.77353	0.68234	0.08307	0.17426
5	0.8847	0.83285	0.73136	0.05185	0.15334
5.5	0.9065	0.88566	0.77152	0.02084	0.13498
6	0.9238	0.93117	0.80707	0.00737	0.11673
6.5	0.9379	0.96814	0.83891	0.03024	0.09899
7	0.949	0.99395	0.86618	0.04495	0.08282
7.5	0.9579	1	0.88616	0.0421	0.07174
8	0.9649	1	0.90302	0.0351	0.06188
8.5	0.9705	1	0.91854	0.0295	0.05196
9	0.9748	1	0.93256	0.0252	0.04224
9.5	0.9784	1	0.94486	0.0216	0.03354
10	0.9812	1	0.95495	0.0188	0.02625
10.5	0.9837	1	0.96144	0.0163	0.02226
11	0.9858	1	0.96682	0.0142	0.01898
11.5	0.9877	1	0.97178	0.0123	0.01592
12	0.9892	1	0.97627	0.0108	0.01293
12.5	0.9906	1	0.98021	0.0094	0.01039
13	0.9919	1	0.98346	0.0081	0.00844
13.5	0.9932	1	0.98553	0.0068	0.00767
14	0.9941	1	0.987	0.0059	0.0071
14.5	0.9952	1	0.98841	0.0048	0.00679
15	0.996	1	0.98977	0.004	0.00623
			Sum of Residuals	1.33113	2.337365

Appendix E: (Continued)

Data for 0.10 grams of citric acid

Time(mins)	Experiment	Model 1: Monodisperse	Model 2: Polydisperse	Residual 1	Residual 2
0	0.0001	0.11214	0.094682	0.11204	0.094582
0.5	0.0541	0.21885	0.18487	0.16475	0.13077
1	0.2059	0.3201	0.27052	0.1142	0.06462
1.5	0.3586	0.41583	0.35155	0.05723	0.00705
2	0.4831	0.50595	0.42781	0.02285	0.05529
2.5	0.5797	0.59031	0.49907	0.01061	0.08063
3	0.6543	0.66874	0.56464	0.01444	0.08966
3.5	0.7123	0.74098	0.62476	0.02868	0.08754
4	0.7576	0.80669	0.67973	0.04909	0.07787
4.5	0.7932	0.86537	0.72871	0.07217	0.06449
5	0.821	0.91629	0.76912	0.09529	0.05188
5.5	0.8429	0.95827	0.80454	0.11537	0.03836
6	0.8602	0.98899	0.83638	0.12879	0.02382
6.5	0.8742	1	0.86389	0.1258	0.01031
7	0.8859	1	0.88451	0.1141	0.00139
7.5	0.8956	1	0.90131	0.1044	0.00571
8	0.9036	1	0.9168	0.0964	0.0132
8.5	0.9103	1	0.93085	0.0897	0.02055
9	0.916	1	0.94327	0.084	0.02727
9.5	0.921	1	0.95365	0.079	0.03265
10	0.9253	1	0.96065	0.0747	0.03535
10.5	0.9291	1	0.96602	0.0709	0.03692
11	0.9323	1	0.97099	0.0677	0.03869
11.5	0.935	1	0.97551	0.065	0.04051
12	0.9373	1	0.97952	0.0627	0.04222
12.5	0.9394	1	0.98289	0.0606	0.04349
13	0.9412	1	0.98527	0.0588	0.04407
13.5	0.9429	1	0.98673	0.0571	0.04383
14	0.9444	1	0.98814	0.0556	0.04374
14.5	0.9459	1	0.98949	0.0541	0.04359
15	0.9473	1	0.99078	0.0527	0.04348
			Sum of Residuals	2.35881	1.433532

Appendix E: (Continued)

Data for 0.20 grams of citric acid

Time(mins)	Experiment	Model 1: Monodisperse	Model 2: Polydisperse	Residual 1	Residual 2
0	0.0001	1.36E-16	0	1E-04	0.0001
0.5	0.0465	0.11166	0.094292	0.06516	0.047792
1	0.1784	0.21707	0.18342	0.03867	0.00502
1.5	0.3029	0.31635	0.26749	0.01345	0.03541
2	0.3997	0.40962	0.34659	0.00992	0.05311
2.5	0.4743	0.49695	0.42074	0.02265	0.05356
3	0.5328	0.57838	0.48989	0.04558	0.04291
3.5	0.5799	0.65389	0.55378	0.07399	0.02612
4	0.6186	0.72344	0.61203	0.10484	0.00657
4.5	0.6512	0.78688	0.66555	0.13568	0.01435
5	0.679	0.84398	0.71395	0.16498	0.03495
5.5	0.703	0.89436	0.75617	0.19136	0.05317
6	0.7243	0.93739	0.79078	0.21309	0.06648
6.5	0.7434	0.97195	0.82233	0.22855	0.07893
7	0.7605	0.99548	0.85048	0.23498	0.08998
7.5	0.7761	1	0.87447	0.2239	0.09837
8	0.7905	1	0.89202	0.2095	0.10152
8.5	0.8039	1	0.90728	0.1961	0.10338
9	0.8164	1	0.92135	0.1836	0.10495
9.5	0.8281	1	0.93412	0.1719	0.10602
10	0.8391	1	0.94544	0.1609	0.10634
10.5	0.8495	1	0.95498	0.1505	0.10548
11	0.8593	1	0.96163	0.1407	0.10233
11.5	0.8685	1	0.9666	0.1315	0.0981
12	0.8772	1	0.97122	0.1228	0.09402
12.5	0.8853	1	0.97545	0.1147	0.09015
13	0.893	1	0.97924	0.107	0.08624
13.5	0.9003	1	0.98253	0.0997	0.08223
14	0.9072	1	0.98514	0.0928	0.07794
14.5	0.9136	1	0.98666	0.0864	0.07306
15	0.9196	1	0.98798	0.0804	0.06838
			Sum of Residuals	3.8154	2.106962