Model-based Computer Simulation of Froth Flotation

Kyle Robert Kelley

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Mining and Minerals Engineering

Roe-Hoan Yoon, Chair
Gregory T. Adel
Gerald H. Luttrell

January 2011
Blacksburg, Virginia

Keywords: froth flotation, modeling, simulation

Model-based Computer Simulation of Froth Flotation

Kyle Robert Kelley

Abstract

Froth flotation is a separation process by which particles are selectively attached to air bubbles. It is one of the most dynamically complex industrial processes in use today. This complexity has steered research towards understanding the fundamental principles of the process. Relatively few researchers have successfully attempted to create a flotation simulator based on first principles. This thesis presents the development and testing of a simulator called SimuFloat, which is based on the flotation model developed at Virginia Tech. Flotation of chalcopyrite, coal, and phosphate are simulated. These simulations show the effects of changing the input parameters of the flotation circuit. The accuracy of SimuFloat is validated by comparing the predictions with the experimentally obtained flotation results.

Acknowledgements

I would like to express my sincerest thanks to my advisor Dr. Roe-Hoan Yoon for his guidance and support throughout my research. I would also like to thank Dr. Gerald Luttrell and Dr. Gregory Adel for serving on my graduate committee, Hyunsun Do for all his help and suggestions in developing SimuFloat, and Aaron Noble for providing me with his experimental results for use in running the simulations. I wish to express my appreciation to the U.S. Department of Energy, the National Energy Technology Laboratory, and the Mining and Minerals Engineering Department at Virginia Tech for providing me with financial support for this research. Lastly, I would like to thank my wife Ashley for all her encouragement and support.

Table of Contents

Abstract ii

Acknowledgements iii

Nomenclature and Symbols v

List of Figures vii

List of Tables ix

1 Introduction 1

  1.1 General Introduction to Flotation 1

  1.2 Literature Review 5

  1.3 Flotation Model 8

  1.4 Research Objective 14

2 Simulator Development 15

  2.1 Introduction 15

  2.2 Physical Parameters 15

  2.3 Chemical Parameters 15

  2.4 Simulator Overview 17

3 Simulation 22

  3.1 Introduction 22

  3.2 Single Component Feed 22

  3.3 Multiple Component Feed 40

  3.4 Model Validation 47

  3.5 Conclusion 49

4 Conclusion 51

  4.1 General Conclusion 51

  4.2 Recommendations for Future Work 51

Works Cited 52

Appendix A 57

Appendix B 83

Appendix C 89

Vita 90

Nomenclature and Symbols

CFD – Computational Fluid Dynamics

DLVO – Derjaguin and Landau, Verwey and Overbeek

MIBC – Methyl Isobutyl Carbinol

PPG 400 – Polypropylene Glycol 400

DOL – Degree of Liberation

QEM*SEM – Quantitative Evaluation of Minerals by Scanning Electron Microscopy

d_1 – Particle diameter

d_2 – Bubble diameter

d_{12} – Collision diameter

$d_{2\text{-}0}$ – Diameter of bubbles entering the froth phase

$d_{2\text{-}f}$ – Diameter of bubbles at the top

E_k – Kinetic energy of attachment

E'_k – Kinetic energy of detachment

h_f – Height of the froth

K_{132} – Hydrophobic force constant between the bubble and particle

K_{131} – Hydrophobic force constant between two particles

K_{232} – Hydrophobic force constant between two bubbles

m_1 – Mass of the paticle

m_2 – Mass of the bubble

n – Number of cells in the bank

N – Number of particles attached to each bubble

P_a – Probability of attachment

P_c – Probability of collision

P_d – Probability of detachment

P_f – Probability of bubble-particle aggregates transferring from the pulp to the froth

r_1 – Radius of the particle

r_2 – Radius of the bubble

R – Bank recovery

R_c – Pulp zone recovery

R_f – Froth zone recovery

R_w – Maximum theoretical water recovery

Re – Reynolds number

S_b – Superficial gas velocity, rate of gas addition

t – Retention time per flotation cell

$\bar{u}_1$ – Particle RMS velocity

$\bar{u}_2$ – Bubble RMS velocity

U_{Hc} – Velocity of a particle approaching a bubble at the critical rupture distance

V_E – Electrostatic interaction energy

V_D – van der Waals dispersion force

V_H – Hydrophobic force

W_a – Work of adhesion

Z_{12} – Collision frequency between particles and bubbles

ε – Energy dissipation rate

ε_0 – Maximum liquid fraction for closely-packed spherical bubbles

γ_{lv} – Surface tension

ρ_1 – Particle density

ρ_2 – Bubble density

ρ_3 – Medium density

θ – Contact angle

ν – Kinematic viscosity of the pulp

ζ_1 – Particle ζ-potential

ζ_2 – Bubble ζ-potential

List of Figures

Figure 1: Mechanical Flotation Cell 2
Figure 2: Column Flotation Cell (Solo Resources Ltd.) 3
Figure 3: ζ-potential vs. pH 17
Figure 4: User Input Form for SimuFloat 18
Figure 5: Particle Size Distribution Input Form 19
Figure 6: Contact Angle Distribution Input Form 20
Figure 7: Feed Grade Input Form 21
Figure 8: Chalcopyrite Recovery at Standard Conditions, Logarithmic 23
Figure 9: Chalcopyrite Recovery at Standard Conditions, Linear 24
Figure 10: Chalcopyrite Recovery, Decreased Specific Power 25
Figure 11: Chalcopyrite Recovery, Increased Froth Height 26
Figure 12: Chalcopyrite Froth Recovery, Increased Froth Height. 27
Figure 13: Chalcopyrite Recovery, Bubble Size Distribution 28
Figure 14: Chalcopyrite Recovery, Increased Gas Rate 29
Figure 15: Chalcopyrite Recovery, Increased Frother Concentration 30
Figure 16: Chalcopyrite Recovery, Standard Conditions vs. Contact Angle Distribution 32
Figure 17: Chalcopyrite Recovery, Effect of Liberation 33
Figure 18: Coal Recovery, Standard Conditions 34
Figure 19: Coal Recovery, Decreased Froth Height. 35
Figure 20: Coal Recovery, 1 cm Froth Height. 36
Figure 21: Phosphate Recovery, Standard Conditions 37
Figure 22: Chalcopyrite vs. Coal vs. Phosphate, Standard Conditions 38
Figure 23: Probability of Attachment, Chalcopyrite vs. Coal vs. Phosphate 39
Figure 24: Probability of Remaining Attached, Chalcopyrite vs. Coal vs. Phosphate 39
Figure 25: Phosphate Recovery, Effect of ζ-Potential. 40
Figure 26: Chalcopyrite Recovery, 3 Component Feed 41
Figure 27: Chalcopyrite, 3 Component Feed, Increased Froth Height. 42
Figure 28: Chalcopyrite Recovery vs. Grade, Contact Angles 40° and 60°. 43
Figure 29: Coal Recovery, 3 Component Feed. 44
Figure 30: Coal Recovery, 3 Component Feed, Increased Number of Cells 45

Figure 31: Phosphate Recovery, 3 Component Feed. 46
Figure 32: Phosphate Recovery, 3 Component Feed, Increased Retention Time 47
Figure 33: Experimental vs. Simulated Silica Flotation Recovery 49

List of Tables

Table 1: Fitting Parameters for K_{131} 11
Table 2: Langmuir- Szyszkowski Equation Constants 16
Table 3: Simulation Standard Operating Conditions 22
Table 4: Liberation Data for a Batch Flotation Feed (Sutherland, 1989) 31
Table 5: Liberation Data for Flotation Concentrate (Sutherland, 1989) 31
Table 6: Multiple Component Feed Parameters 41
Table 7: Silica Flotation Test Parameters 48
Table 8: Silica Flotation Recoveries 48

1 Introduction

1.1 General Introduction to Flotation

Froth flotation, often referred to simply as ‘flotation,’ was first used commercially in 1877 for treating graphite ore in Germany. Widespread use of the technology did not occur until the turn of the twentieth century, and mineral production began to expand rapidly around mid-century (Fuerstenau, Jameson, & Yoon, 2007). Froth flotation is a technique used to separate materials based on differences in their surface properties. Originally developed as a mineral processing technique, there are numerous other uses of flotation. Today it is used to treat billions of tons of materials yearly in the mining, recycling, and wastewater treatment industries (Rubio, Souza, & Smith, 2002).

In froth flotation, particles are selectively attached to air bubbles. These bubble-particle aggregates rise to the surface of the flotation cell and are removed from the system. Particles are selected to attach according to their level of hydrophobicity, or fear of water. In theory, flotation occurs only when hydrophobic particles attach to air bubbles, while hydrophilic particles stay in the system. In reality there are three mechanisms by which flotation occurs: attachment, entrainment, and agglomeration. Bubble-particle attachment is the most important of the three mechanisms, and it is the only mode of recovery that is selective. Entrainment occurs when particles are recovered by entrapment in the water films formed between bubbles. Agglomeration, sometimes called coagulation, takes place when small particles attach to each other and act as a single larger particle. These agglomerates have the potential to trap hydrophilic particles within them. Recovery by agglomeration is sometimes referred to as entrapment

In the past, the flotation process was often modeled as a first-order process with a single rate constant for the recovery processes occurring in both the pulp and froth phases of a flotation cell. In effect, flotation was viewed as a single-phase process. However, the cell consists of two distinctly different phases, each having different mechanisms of particle recovery and roles in the production of a concentrate. More recently, flotation is modeled by considering the differences between and determining the rate constants for the pulp and froth zones.

Flotation cells are nearly always arranged in a series called a bank. The product of one cell becomes the feed to the next cell. This setup helps to give the particles not recovered in the first few cells additional opportunities for recovery in rougher and scavenger cells and reduces the effects of entrainment and entrapment, to achieve higher quality products at maximum recoveries. The effect of entrainment can also be reduced through the use of froth wash water.

There are two basic types of flotation cells: column and mechanical. Mechanical flotation is performed under turbulent conditions in a stirred tank. Column flotation is performed under relatively quiescent conditions in a tall narrow cell. Mechanical flotation cells are more common than column cells because they can process high tonnages and are more flexible. Column flotation cells can achieve more efficient separations, but are limited by relatively low capacity due to their smaller cross-sectional area.

As the name would suggest, mechanical cells are mechanically agitated by a spinning rotor. The rotor generates turbulence, which serves as a mechanism for bubble-particle collisions, particle suspension, and air dispersion. As can be seen in Figure 1, mechanical cells generally have a low height to diameter ratio and hence a large cross-sectional area to volume ratio.

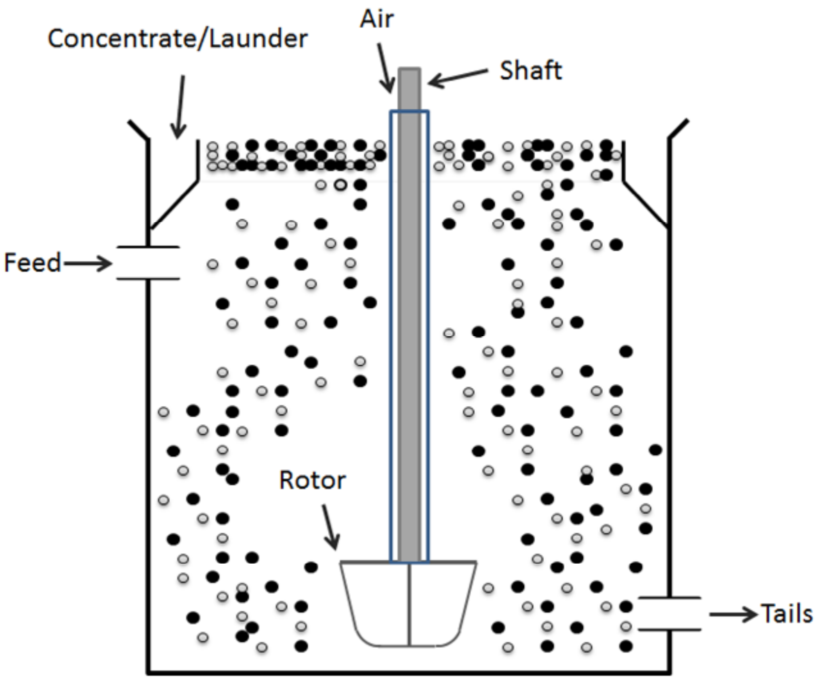


Figure 1: Mechanical Flotation Cell

Slurry is generally fed into the tank near the middle, and air is injected through the center of the shaft that spins the rotor. Hydrophobic particles are recovered by a launder at the top of the cell, while the tails exit the cell through a pipe at the bottom. In addition to the rotor, a stator ring is often used to induce additional turbulence, and prevent the formation of a vortex within the cell. Many cells also employ a beveled edge along the bottom of the cell. This forces particles towards the rotor, and prevents the buildup of sediment on the cell floor.

Column flotation cells have a high height-to-diameter ratio, as shown in Figure 2 on the following page. Air is injected at the bottom of the cell, making use of either in-line mixers or spargers to reduce the bubble size. The feed, introduced at the top of the cell, flows downward while the air bubbles rise. This creates a mixing action and eliminates the need for a rotor to mix the pulp. As with mechanicals cells, the froth overflows into a launder and the tails exit through the bottom of the cell. Column cells typically use wash water to reduce the effects of entrainment and improve the product grade (Finch, 1994).

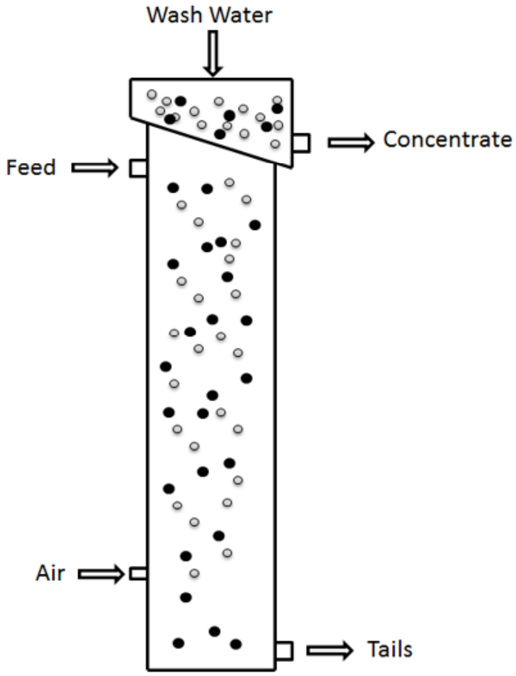


Figure 2: Column Flotation Cell

Both mechanical and column cells employ a series of chemical reagents to improve flotation performance. The three main groups of reagents are frothers, collectors, and modifiers. Surfactants lower the surface tension of water and promote the buildup of a stable froth.

Collectors render the target minerals hydrophobic, enabling them to attach to air bubbles. There are numerous varieties of collectors, which are broken into two main groups; thiol-type and non-thiol-type reagents. Each collector is tailored to treat a specific mineral or group of minerals. Modifiers are categorized into activators, depressants, and pH regulators. These particular additives are used either to enhance the adsorption of collectors or reduce the floatability of undesirable minerals.

The mining industry utilizes flotation to upgrade the finest fraction of run-of mine (ROM) ores. Particles that are larger than 100 mesh can often be recovered by other methods, making flotation most useful in the 10 to 150 µm range. In the U.S. coal industry, many companies discard -44 µm materials due to the difficulty in floating and dewatering finer particles. Now this fraction can be treated using flotation with advanced dewatering techniques, such as hyperbaric centrifugation in the coal industry (Keles, 2010).

The importance of improving flotation performance has become more pronounced as the industry has expanded. An increase in performance as small as a fraction of a percent can have a vast financial and environmental impact when billions of tons of material are treated each year. The present work focuses on the simulation of flotation, with the hope of improving the general understanding of flotation and aiding in the advancement of flotation technology. These simulations will focus on three important industrial minerals: chalcopyrite, coal, and phosphate.

Copper is predominantly found in porphyry deposits containing < 1% copper. The principal mineral being mined for copper is chalcopyrite ($CuFeS_2$), which contains 34.5% copper when pure. As the grade of the ore is so low, it is often necessary to grind the ore to finer sizes in order to achieve a high degree of liberation (DOL). Greater than 50% DOL is achieved at particle sizes less than 100 mesh (Subrahmanyam & Forssberg, 1995). For such small particles, flotation is regarded as the most efficient beneficiation method. Short chain collectors, such as xanthates, are frequently used for the flotation of copper minerals.

Coal processing, also known as coal preparation, is used to increase heating value and reduce transportation costs. Flotation is often used for the beneficiation of fine coal. Coal is naturally hydrophobic; therefore, it can often be floated without using a collector. In general, the hydrophobicity of coal increases with rank and vitrinite content within the coal (Ding, 2009).

Much of the world's phosphate reserves are found in sedimentary deposits. Nearly half of the world's phosphate is cleaned using froth flotation to remove silicates, carbonates, and clays from the ore. There are two methods of flotation commonly used in the phosphate industry: direct and reverse flotation. Direct flotation is a one-step process in which the phosphate is directly floated. Reverse flotation is a two-step process, in which gangue materials are removed. Long-chain fatty acids are ordinarily used as collectors for phosphates (Sis & Chander, 2003).

1.2 Literature Review

Although extensive research is being conducted on flotation, both in industry and academia, most investigators agree that flotation is the least understood of all mineral processing techniques. This lack of understanding stems from the large number of variables in flotation, and the fact that it is a three-phase process which is difficult to model mathematically. It is widely believed that flotation may be described as a first order process in the pulp, and often in the froth as well (Fichera & Chudacek, 1992). Flotation models generally consist of a bubble-particle collision rate term and a probability of flotation term.

The theoretical basis for the collision term in the majority of current flotation models is Abrahamson's model for the collision rate of small particles in a turbulent fluid:

$$Z_{12} = 5N_1N_2d_{12}^2\sqrt{\overline{U}_1^2 + \overline{U}_2^2} \qquad [1]$$

where Z_{12} is the collision rate between two types of particles, N_1 and N_2 are the number concentrations of two types of particles, d_{12} is the sum of colliding particle radii, and $\overline{U}_1$ and $\overline{U}_2$ are the root-mean square (RMS) velocities of the particles and bubbles. This model assumes that particles move randomly under infinite Stokes number conditions as if no other particles are present and that their velocities are fully independent of each other (Abrahamson, 1975).

The advances made in computing energy dissipation rates during the past decade have made it feasible for researchers to develop models of flotation cells using computational fluid dynamics (CFD). As it applies to flotation, CFD is used to determine local turbulence by breaking the flotation cell into many small, finite volume elements. These local turbulence values are then utilized to determine collision, attachment, and detachment rates in the cell by using a flotation model (Koh & Schwarz, 2006). Koh and Schwarz found that the volume neighboring the impeller and stator is more turbulent than in the bulk of the pulp, the collision probability decreases as particle size decreases due to streamlining effects, and the attachment rate decreases as particle size decreases (Koh & Schwarz, 2003).

Schubert found that it is impossible to attain optimum hydrodynamics for all particle sizes simultaneously. The specific power input must be minimized while maintaining particle suspension for optimum flotation of coarse particles, while a much higher power input is required for optimum flotation of fine particles (Schubert, 1999). Yang and Aldrich found that the water recovery increases with aeration rate and power input. It was also found that solids entrainment in the froth can be linearly related to the water recovery, independent of aeration rate and power input (Yang & Aldrich, 2006).

The mineralogical characteristics of particles also play a role in flotation performance. Lastra (2007), Savassi (2006), and Sutherland (1989) all found that mineral liberation has an effect on particle recovery. Through the use of quantitative evaluation of minerals by scanning electron microscopy (QEM*SEM), these researchers were able to determine that particles with a higher degree of liberation had a greater probability of ending up in the flotation concentrate. QEM*SEM and other SEM techniques have become an important tool, allowing mineral processors to better characterize their flotation feeds rapidly.

The complexity of the flotation process has severely limited the availability of workable flotation simulators. While a vast amount of work has gone into understanding the fundamental principles of flotation, relatively few researchers have taken on the task of developing a flotation simulator, and even fewer have accomplished the goal of producing a useful product. The simulators that are available in industry and from the literature include JKSimFloat, USIM PAC, MODSIM, and a simulator developed at University of Petrosani, Romania.

JKSimFloat is a commercially available flotation simulation program that is under continuing development. The simulator is a collaborative effort between the Julius Kruttschnitt Mineral Research Centre (JKMRC) at the University of Queensland, the University of Cape Town, and McGill University (JKTech Pty Ltd, 2010). JKSimFloat allows users to input data for each object on the flow sheet, and solves the flow sheet using a mass balance. The basis for the model behind JKSimFloat is the following equation:

$$k = P \cdot Sb \cdot Rf \qquad [2]$$

where P is the ore floatability, Sb is the bubble surface area flux, and Rf is the loss in recovery due to the froth phase (Harris, Runge, Whiten, & Morrison, 2002). The ore floatability is determined by either the Distributed Property Floatability Component Model (DPFC-Model) or the Empirical Floatability Component Model (EFC-Model). These approaches involve determining the floatability of a mineral experimentally, and using the results as an input to the simulator. The drawback of this model is that it is dependent on the collection of good experimental data from operating plants. The floatability here refers to size-by-size and class-by-class flotation recoveries. The term class refers to degree of liberation, which increases with decreasing particle size. Many researchers using models of this form also include a ¼ term in Eq. [2].

USIM PAC$^{©}$ is a steady state simulator that can model over 100 unit operations, including both column and mechanical flotation machines. It is currently used in industry for data reconciliation, plant simulation and design, flow sheet development, and cost estimation (Metso Minerals). The USIM PAC$^{©}$ simulator contains two separate models for flotation. The first model takes an approach that uses three sub-populations for each mineral, i.e., non-floating, fast-floating, and slow-floating components, each have their own rate constants. The second model uses a distribution of rate constants that are dependent on particle size:

$$k_{ij} = \frac{a_j}{x_i^{0.5}}\left[1 - \left(\frac{x_i}{xl_j}\right)^{1.5}\right] exp\left| - \left(\frac{xe_j}{2x_i}\right)^2\right] \qquad [3]$$

Where x_i is the average size in fraction i, a_j is an adjustment parameter, xl_j is the largest floating size for mineral j, and xe_j is the easiest floating particle size for mineral j (Villeneuve, Guillaneau, & Durance, 1995). The model also accounts for the effects of entrainment of particles in the froth based on water recovery. Like JKSimFloat, this simulator requires the collection of large amounts of accurate experimental data.

The MODSIM mineral processing simulator can simulate crushing, classification, flotation and several other processes. MODSIM is based on a population balance model and can account for changes in size and mineral liberation. The model used for flotation simulation is a form of the distributed rate constant model. The simulator provides integrated flow sheets, allowing the user to easily simulate an entire processing plant (Mineral Technologies).

The simulator developed at the University of Petrosani is used to simulate flotation circuits. It is apparent that the simulator uses both population and mass balances to design flow sheets. However, the usefulness and accuracy of this tool cannot be verified because the details of the model that drives the simulator are not revealed in the literature (Samoila & Marcu, 2010).

1.3 Flotation Model

A comprehensive flotation model was developed by Do. The model is derived from first principles and is used as the basis for the flotation simulator described in Chapter 2. The model accounts for both the surface chemistry and hydrodynamic properties of the system, which allows it to predict real world flotation results (Do, 2010). Unlike the models used in the flotation simulators described in Section 1.2, this flotation model does not require the input of experimental data. The key advantage of this model is that it can predict flotation results from contact angle, particle diameter, and ζ-potential which are the key parameters affection flotation. This section presents the key analytical equations of the model.

In order for flotation to occur, a bubble must collide with a particle, attach to it, enter the froth phase, and then overflow to a launder without becoming detached. The rate of flotation, *k*, can be given as follows,

$$k = -Z_{12}P \qquad [4]$$

in which Z_{12} is the collision frequency given in units of s^{-1} and P is a probability of flotation which has no unit. Thus, k has a unit of s^{-1}.

The collision frequency is determined by using the Abrahamson's equation [1] for random collisions:

$$Z_{12} = 2^{3/2}\pi^{1/2}N_1N_2d_{12}^2\sqrt{\bar{u}_1^2 + \bar{u}_2^2} \qquad [5]$$

where Z_{12} is the collision frequency between particles and bubbles, N_1 is the number of particles, N_2 is the number of bubbles, d_{12} is the sum of radii of one bubble and one particle which is referred to as collision diameter, and $\bar{u}_1$ and $\bar{u}_2$ are the RMS velocities of the particles and bubbles, respectively.

The diameter of the bubbles generated in the cell is calculated using the following relationship derived by Schulze (1984):

$$d_2 = \left(\frac{2.11\gamma_{lv}}{\rho_3 \varepsilon_b^{0.66}}\right)^{0.6} \qquad [6]$$

where γ_{lv} is the surface tension of the water in a flotation cell, ρ_3 is the density of the water, and ε_b is the energy dissipation rate in the bubble generation zone. It has been reported that the high energy zone around the impeller and stator typically has a dissipation rate of 5-30 times larger than the mean (Schulze, 1984). In the present work it is assumed that the energy dissipation rate in the bubble generation zone is approximately 15 times larger than the mean energy dissipation rate in the cell.

The RMS velocity of the particles is calculated using the following relationship:

$$\bar{u}_1 = 0.4 \frac{\varepsilon^{4/9} d_1^{7/9}}{\upsilon^{1/3}} \left(\frac{\rho_1 - \rho_3}{\rho_3}\right)^{2/3} \qquad [7]$$

where ε is the energy dissipation rate, d_1 is the particle diameter, ν is the kinematic viscosity of water, ρ_1 is the density of the particle, and ρ_3 is the density of water (Schubert, 1999).

The bubble RMS velocity is calculated using the equation derived by Lee and Erickson:

$$\bar{u}_2 = \left(C_0(\varepsilon d_2)^{2/3}\right)^{1/2} \qquad [8]$$

where C_0 is a constant given as 2 and d_2 is the bubble diameter (Lee & Erickson, 1987).

The total probability of flotation, P, is given by

$$P = P_a P_c (1 - P_d) P_f \qquad [9]$$

where P_a is the probability of attachment, P_c is the probability of collision, P_d is the probability of detachment, and P_f is the probability of bubble-particle aggregates transferring from the pulp phase to the froth phase.

The probability of attachment is calculated as follows,

$$P_a = \exp\left(\frac{-E_1}{E_k}\right) \qquad [10]$$

where E_1 is the energy barrier as calculated using the extended DLVO theory, and E_k is the kinetic energy of attachment (Yoon & Mao, 1996).

The extended DLVO theory states that

$$E_1 = V_E + V_D + V_H \qquad [11]$$

where V_E is the electrostatic interaction energy, V_D is the van der Waals dispersion force, and V_H is the hydrophobic force.

The electrostatic interaction energy can be obtained from the following relation,

$$V_E = \frac{\pi\epsilon_0\epsilon r_1 r_2\left(\zeta_1^2+\zeta_2^2\right)}{r_1+r_2}\left[\frac{\zeta_1^2\zeta_2^2}{\zeta_1^2+\zeta_2^2}\ln\left(\frac{1+e^{-\kappa H}}{1-e^{-\kappa H}}\right)+\ln\left(1+e^{-2\kappa H}\right)\right] \qquad [12]$$

where ϵ_0 is the permittivity in a vacuum, ϵ is the dielectric constant of the medium, ζ_1 is the ζ-potential of the particle, ζ_2 is ζ-potential of the bubble, κ is the inverse Debye length, and H is the separation distance between the bubble and particle (Hogg, Healy, & Fuerstenau, 1966) and (Do, 2010).

The van der Waals dispersion energy can be calculated using the following relationship,

$$V_D = -\frac{A_{132}r_1 r_2}{6H(r_1+r_2)}\left[1-\frac{1+2bl}{1+bc/H}\right] \qquad [13]$$

where A_{132} is the Hamaker constant for the bubble-particle interaction in the medium, b and l are characterization parameters for the materials involved, and c is the speed of light (Rabinovich & Churaev, 1979).

The hydrophobic force can be expressed as:

$$V_H = -\frac{K_{132}r_1 r_2}{6H(r_1+r_2)} \qquad [14]$$

where K_{132} is the hydrophobic force constant between the bubble and particle (Rabinovich & Churaev, 1979), which can be obtained using the following relationship

$$K_{132} = \sqrt{K_{131}K_{232}} \quad [15]$$

where K_{131} is the hydrophobic force constant between two particles and K_{232} is the hydrophobic force constant between two bubbles (Yoon, Flinn, & Rabinovich, 1997). This relationship, which is referred to as geometric mean combining rule has recently been proven in wetting film studies (Pan and Yoon, 2010).

The hydrophobic force constant between two particles may be found by:

$$K_{131} = ae^{b_k\theta} \quad [16]$$

where a and b_k are fitting parameters shown in Table 1 (Pazhianur & Yoon, 2003). The hydrophobic force constant between two bubbles is $2.5\text{x}10^{-18}$ (Do, 2010).

Table 1: Fitting Parameters for K_{131}

Θ	a	b_k
> 92.28°	$6.327\text{x}10^{-27}$	0.2127
92.28° > θ > 86.89°	$4.888\text{x}10^{-44}$	0.6441
< 86.89°	$2.732\text{x}10^{-21}$	0.04136

The kinetic energy of attachment is calculated using the following relation,

$$E_k = 0.5m_1U_{H_c} \quad [17]$$

where m_1 is the mass of the particle, and U_{Hc} is the velocity of a particle approaching a bubble at the critical rupture distance. This velocity may be found by the following equation:

$$U_{H_c} = \frac{\overline{u}_1}{\beta} \quad [18]$$

where β is the drag coefficient in the boundary layer of the bubble (Goren & O'Neill, 1971).

The drag coefficient may be expressed as (Luttrell & Yoon, 1992):

$$\beta = 0.37\left(\frac{r_1}{H}\right)^{0.83} \quad [19]$$

which has been derived from the Reynolds lubrication theory.

The probability of collision equation used in this model was derived by Luttrell and Yoon, and modified to ensure the probability may not be greater than 1 (Do, 2010),

$$P_c = \tanh^2\left(\sqrt{\frac{3}{2}\left(1+\frac{\frac{3}{16}Re}{1+0.249Re^{0.56}}\right)}\left(\frac{d_1}{d_2}\right)\right) \quad [20]$$

where *Re* is the Reynolds number (Weber & Paddock, 1983).

Probability of detachment, as suggested by Yoon and Mao, is given as follows:

$$P_d = \exp\left(\frac{-W_a+E_1}{E_k'}\right) \quad [21]$$

where W_a is the work of adhesion (a function of contact angle), and E'_k is the kinetic energy of detachment (Yoon & Mao, 1996). The work of adhesion can be obtained by:

$$W_a = \gamma_{lv}\pi r_1^2(1-\cos\theta)^2 \quad [22]$$

where γ_{lv} is the surface tension of water, r_1 is the radius of the particle, and θ is the angle of contact between water and the particle (Yoon & Mao, 1996). The kinetic energy for detachment is calculated using the following equation (Do, 2010):

$$E'_k = 0.5m_1\left((d_1+d_2)\sqrt{\varepsilon/\nu}\right)^2 \quad [23]$$

where ε is the energy dissipation rate and ν is the kinematic viscosity.

The probability of bubble particle aggregates transferring to the froth phase accounts for instances in which the aggregates may bounce off the pulp-froth interface:

$$P_f = P_i(1-P_r) \quad [24]$$

where P_i is the probability that the aggregate will remain at the interface after bouncing n times, and P_r is the probability of aggregate rupture. The first term, P_i, is represented as:

$$P_i = 13\sqrt{\frac{9\mu^2}{d_2\gamma_{lv}\rho_2}} \quad [25]$$

where μ is the dynamic viscosity of water. The second term, P_r is calculated by:

$$P_r = exp\left(-\frac{E_{iw}}{E_{ka}}\right) \quad [26]$$

where E_{iw} is the kinetic energy transferred to the bubble-particle aggregate by the motion of the pulp-froth interface, and E_{ka} is the kinetic energy of the aggregate after bouncing off the interface (Do, 2010).

The kinetic energy that is transferred to the bubble-particle aggregate can be expressed as:

$$E_{ie} = \frac{g}{4\pi}\left(\frac{\nu^3}{\varepsilon_b}\right)^{1/4} \quad [27]$$

where g is the acceleration of gravity and v is the kinematic viscosity of water (Sanada, Watanabe, & Fukano, 2005).

The kinetic energy of the bubble-particle aggregate after bouncing is determined using the following equation:

$$E_{ie} = \frac{\left(m_2\sqrt{\bar{u}_2^2}-2\left(\frac{d_2}{d_1}\right)^2 m_1\sqrt{\bar{u}_1^2}\right)^2}{100\left(m_2+2\left(\frac{d_2}{d_1}\right)^2 m_1\right)^2} \quad [28]$$

where m_2 is the mass of the bubble (Do, 2010).

Fractional recovery (R_c) of particles in the pulp phase is given by

$$R_c = 1 - (1 + kt)^{-1} \quad [29]$$

where k is the flotation rate constant in the pulp phase, as determined from the preceding equations, and t is the retention time of the particles within the pulp.

As discussed in the foregoing section, the fractional recovery (R_f) in the froth phase is the sum of the recovery by attachment and the recovery by entrainment,

$$R_f = \frac{d_{2-0}}{d_{2-f}} e^{-N\frac{6h_f}{d_{2-0}}\left(1-\frac{d_{2-0}}{d_{2-f}}\right)\left(\frac{d_1}{d_{2-0}}\right)^2} + R_w e^{-0.0325\left(\frac{\rho_3}{\rho_1}-1\right)-63000d_1} \qquad [30]$$

where $d_{2\text{-}0}$ is the diameter of the bubbles entering the froth phase, $d_{2\text{-}f}$ is the diameter of the bubbles at the top, N is the number of particles attached to each bubble, h_f is the froth height, R_w is the maximum theoretical water recovery, ρ_3 is the density of water, and ρ_1 is the particle density. The first term of Eq. [30] represents the recovery due to attachment, while the second term represents the recovery due to entrainment (Do, 2010).

Entrainment is closely related to the water recovery, which can be calculated using the following relation,

$$R_w = \frac{Q_{out}^{air}/Q_{in}^{liq}}{{}^{1}/_{E_l}-1} \qquad [31]$$

where Q_{out}^{air} is the volumetric flow rate of air leaving the cell and is equal to the superficial gas velocity, Q_{in}^{liq} is the volume flow rate of pulp entering the cell, and E_l is the fraction of water in the froth phase (Kelley, Do, Keles, Luttrell, & Yoon, 2011).

Assuming that the flotation rate is constant over all the cells in a bank, recovery for the bank can be expressed by:

$$R = 1 - \left(1 - \left(\frac{R_c R_f}{R_c R_f + 1 - R_c}\right)\right)^n \qquad [32]$$

where R_c is the pulp or collection zone recovery, R_f is the froth zone recovery, and n is the number of cells in the bank (Finch & Dobby, 1990).

1.4 Research Objective

The objective of the research presented in this communication is to develop a software tool that can easily and accurately simulate froth flotation based on the model presented in Section 1.3. This tool will be used to improve upon the understanding of the fundamental principles of flotation, allowing researchers to improve flotation processes and design more efficient flotation machines.

2 Simulator Development

2.1 Introduction

The aim of this research is to develop a user-friendly simulation tool for predicting flotation recovery. This was accomplished using the programming language Visual Basic in conjunction with Microsoft Visual Studio 2008. Since the model is based on first principles, the simulator has predictive and diagnostic capabilities, differentiating it from other currently available flotation simulators that are based on empirical models. The following sections will detail the development of the flotation simulator called SimuFloat and provide an overview of how the program functions.

2.2 Physical Parameters

The simulator requires the input of a number of hydrodynamic operating parameters, including specific power, superficial gas rate, particle specific gravity, particle size distribution, air fraction (air holdup), slurry fraction (% solids), and froth height. The user may also elect to manually input a bubble size distribution if it is known. Use of a bubble size distribution can be beneficial in achieving effective simulations. Other physical parameters that effect recovery such as number of cells and retention time per cell are also needed to perform the simulation. These parameters, along with the chemical parameters discussed below, represent the majority of the operator-controlled flotation parameters in a flotation plant.

2.3 Chemical Parameters

SimuFloat requires user input of contact angle, type of frother (surfactant), frother concentration, and particle zeta (ζ)-potential. Bubble ζ-potential, permittivity of air, and dielectric constant of the medium are given in the simulator by default but may be changed if necessary. The user may also elect not to select a frother, in which case the surface tension of pure water will be used in the calculations.

Contact angle determines the strength of the bubble-particle attachment, as defined in [14] and [20]. A greater concentration of collector in the system will render the particles more hydrophobic, thus increasing the probability of attachment and the work of adhesion.

SimuFloat accepts the input of four common frothers: methyl isobutyl carbinol (MIBC), polypropylene glycol 400 (PPG 400), pentanol, and octanol. Frother concentrations are converted into surface tension using the Langmuir-Szyszkowski equation:

$$\gamma = \gamma_0 - RT\Gamma_m \ln(1 + K_L c) \qquad [33]$$

where γ_0 is the surface tension of pure water, c is the frother concentration, R is the universal gas constant, T is the temperature (assumed to be 23 °C), Γ_m is the maximum adsorption density, and K_L is the equilibrium adsorption constant. Equation [33] is used in conjunction with a series of empirical constants, shown in Table 2, to determine the surface tension within the flotation cell (Wang & Yoon, 2007).

Table 2: Langmuir- Szyszkowski Equation Constants

Frother	K_L (M^{-1})	Γ_m (μmol/m^2)
MIBC	230	5
PPG 400	1.7×10^6	1
Pentanol	55	6
Octanol	2200	8

The key factor affecting ζ-potential of the system is the pH (Malvern Instruments Ltd, 2004). One may roughly equate pH to ζ-potential by means of a plot showing the effect of pH on ζ-potential such as the plot in Figure 3.

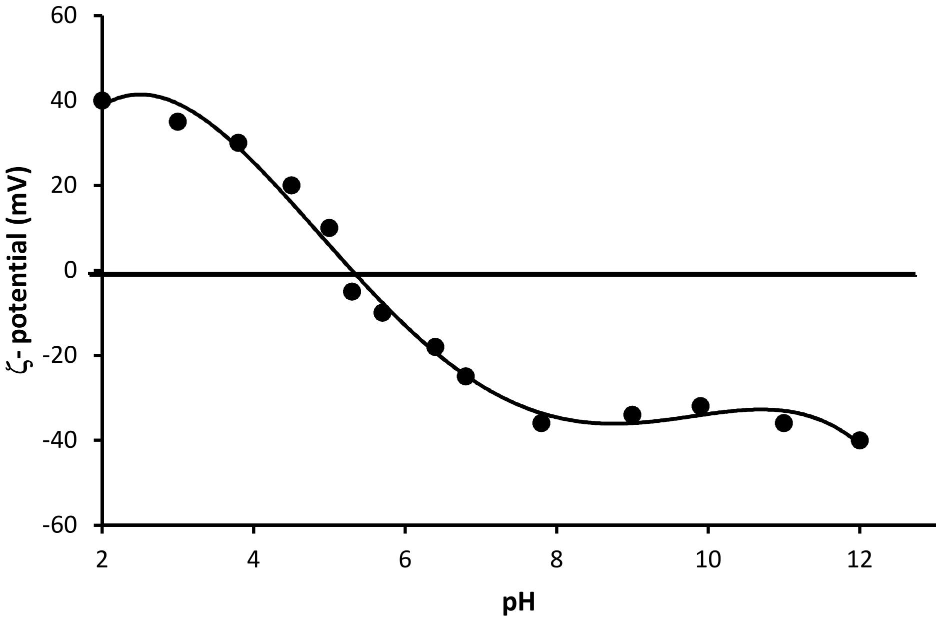
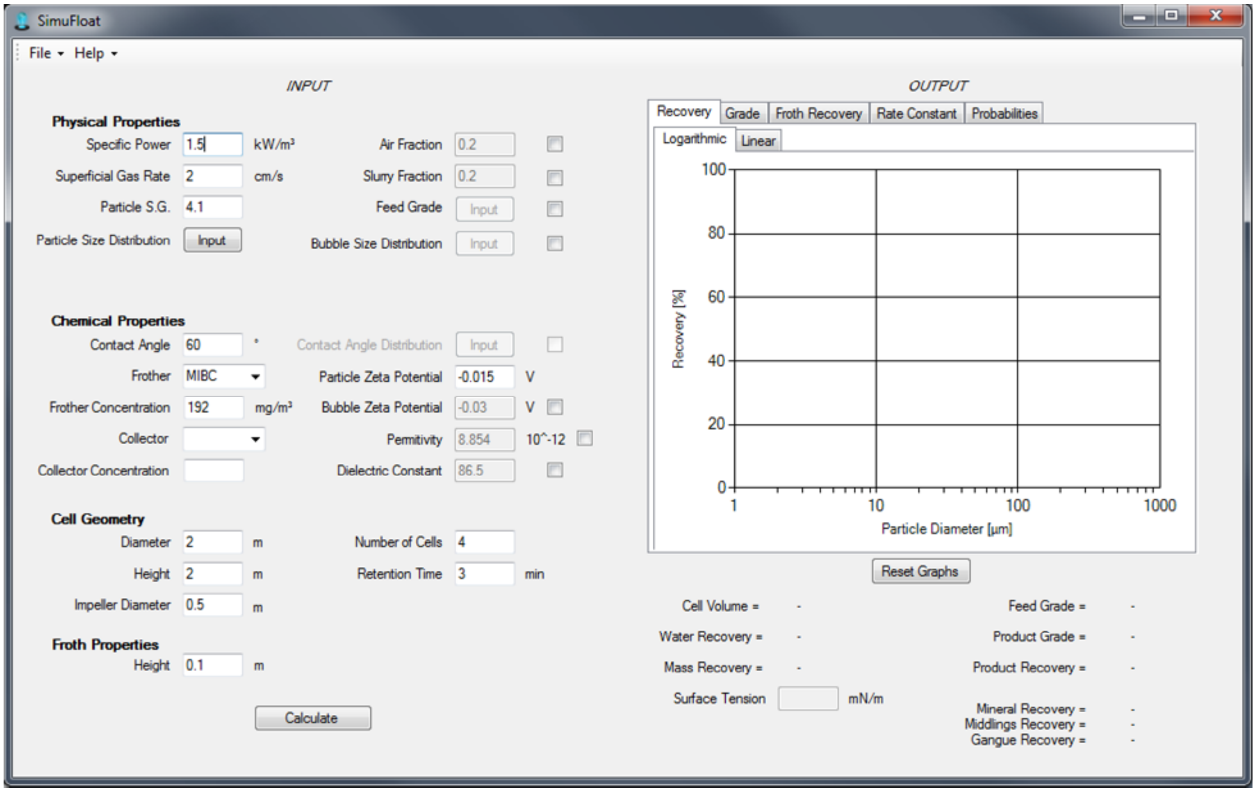


Figure 3: ζ-potential vs. pH. This figure shows the general effect of pH on the ζ-potential but is not specific to any one mineral.

Plots showing the ζ-potential versus pH for common minerals treated by flotation may be available in the literature. However, discussion of such plots is beyond the scope of this work, as SimuFloat does not yet allow for input of pH.

2.4 Simulator Overview

The main input form for SimuFloat is shown in Figure 4. The user may input the desired parameters on the left hand side of the form, and all simulation outputs are shown on the right hand side of the form. Greyed input fields and buttons may be enabled by placing a check in the adjacent checkbox. For the user's convenience, some input fields contain preset values for properties of water or air at or near 23 °C. These values can change with large changes in temperature, and may be modified by the user. The simulator allows for the input of both single component and multi-component feeds.



Physical Properties
Specific Power 1.5 kW/m³
Superficial Gas Rate 2 cm/s
Particle S.G. 4.1
Air Fraction 0.2 ☐
Slurry Fraction 0.2 ☐
Feed Grade Input ☐
Bubble Size Distribution Input ☐

Chemical Properties
Contact Angle 60 °
Frother MIBC
Frother Concentration 192 mg/m³
Collector
Collector Concentration
Contact Angle Distribution Input ☐
Particle Zeta Potential -0.015 V
Bubble Zeta Potential -0.03 V ☐
Permitivity 8.854 10^-12 ☐
Dielectric Constant 86.5 ☐

Cell Geometry
Diameter 2 m
Height 2 m
Impeller Diameter 0.5 m
Number of Cells 4
Retention Time 3 min

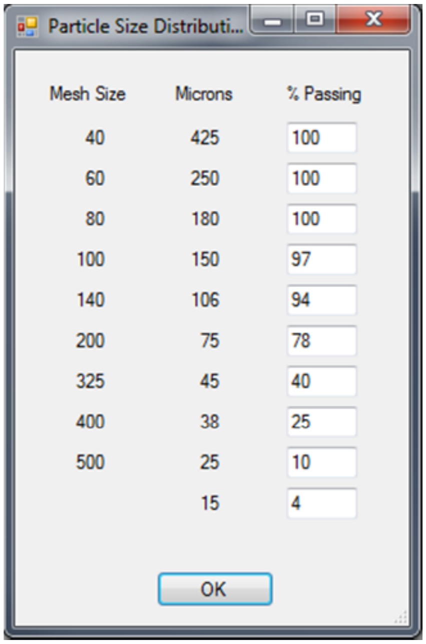
Froth Properties
Height 0.1 m

Cell Volume = -
Water Recovery = -
Mass Recovery = -
Surface Tension mN/m
Feed Grade = -
Product Grade = -
Product Recovery = -
Mineral Recovery = -
Middlings Recovery = -
Gangue Recovery = -

Figure 4: User Input Form for SimuFloat. Input values are entered on the left hand side, output is displayed on the right hand side.

Text input is limited to numerical values, including negative numbers. Input error checking has been implemented to ensure that all required inputs have been entered, and that values are within reasonable ranges for real-world flotation conditions, as model accuracy may deteriorate with extreme values (e.g. - particle specific gravity must be greater than that of water, otherwise particles naturally float).

SimuFloat determines recovery curves based on the model that is briefly described in Section 1.3. The overall recovery for each size class of particles is determined and plotted both linearly and logarithmically. The linear plot is included to illustrate the difficulty in floating coarse particles that is experienced in industry. When utilizing the particle size distribution feature, the program calculates the recovery for each size class and sums them up to obtain the total recovery. The user input form for particle size distribution is shown in Figure 5.



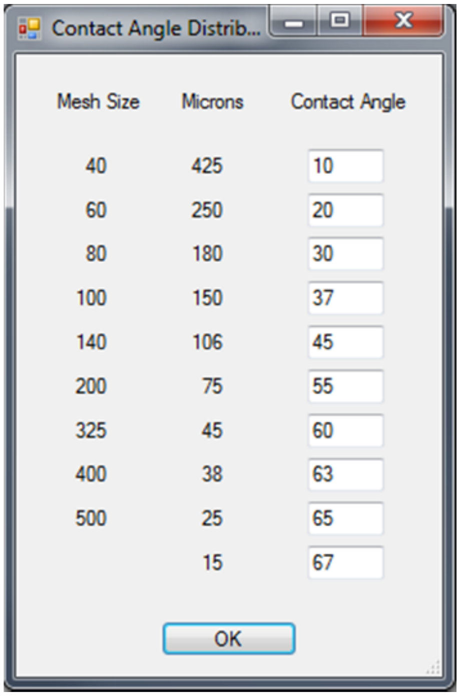
Mesh Size	Microns	% Passing
40	425	100
60	250	100
80	180	100
100	150	97
140	106	94
200	75	78
325	45	40
400	38	25
500	25	10
	15	4

Figure 5: Particle Size Distribution Input Form. Particle Sizes are displayed in both mesh and microns. Input is entered on a percent passing basis

Particle sizes are shown in both mesh and micron sizes, and should be entered as the weight percent passing the given size. The particle size distribution must be input if values for recovery are desired, otherwise SimuFloat will simply output the recovery curve.

The results are output in the form of plots for overall recovery, grade, froth phase recovery, flotation rate constant, and probabilities of collision, attachment, and detachment. The independent variable is particle diameter in microns for all of the graphs. Graphs are viewed through tab selection on the main form. In addition to graphical output, cell volume and calculated surface tension are output as text. Additionally, when using multiple feed components, feed grade, product grade, and water, mass, product, mineral, middlings, and gangue recoveries are output as text.

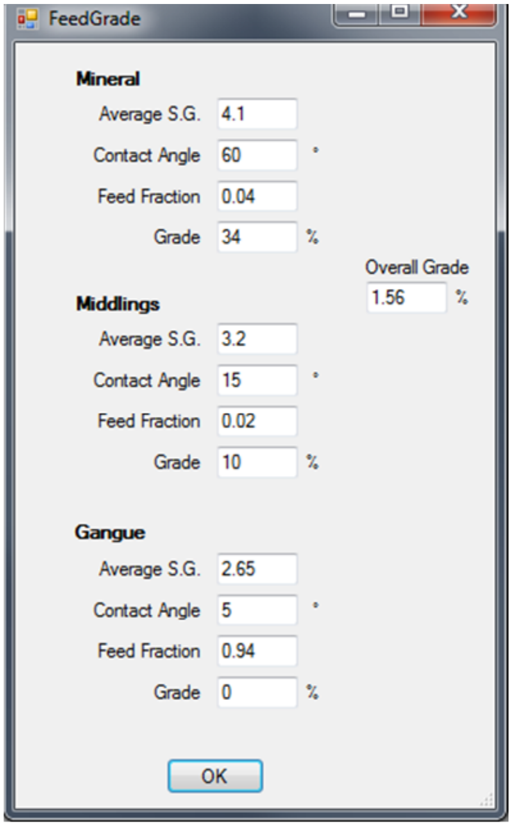
Single component input is straight forward; the user must enter all the values on the main form. SimuFloat also allows for the input of a distribution of contact angles for a single component feed based on particle size. This function can be used to simulate variance in liberation characteristics due to particle size. As particle size increases, the number of locked particles tends to increase. It has been shown that this tendency can have a significant effect on flotation performance (Sutherland, 1989). Figure 6 shows the input form for contact angle distribution.



Mesh Size	Microns	Contact Angle
40	425	10
60	250	20
80	180	30
100	150	37
140	106	45
200	75	55
325	45	60
400	38	63
500	25	65
	15	67

Figure 6: Contact Angle Distribution Input Form. Contact angles for particles within each size class are interpolated based on the user input values to produce a smooth recovery curve.

Multiple components may be entered by clicking the “Feed Grade” button, which is greyed by default. A popup form is displayed allowing the user to input contact angle, grade, feed fraction, and specific gravity for the mineral, middlings, and gangue, as shown in Figure 7.



Mineral

Average S.G. 4.1

Contact Angle 60 °

Feed Fraction 0.04

Grade 34 %

Overall Grade 1.56 %

Middlings

Average S.G. 3.2

Contact Angle 15 °

Feed Fraction 0.02

Grade 10 %

Gangue

Average S.G. 2.65

Contact Angle 5 °

Feed Fraction 0.94

Grade 0 %

Figure 7: Feed Grade Input Form. The overall grade of the feed is calculated real-time as the user inputs values for feed fraction and grade for each component.

When using this function, each component is plotted separately on the graphs.

The majority of the most recent source code for SimuFloat, as of this writing, can be found in Appendix A. Due to the nature of software development, the code may have changed drastically in a short period of time. However, the source may be useful in understanding the fundamental model behind the program.

3 Simulation

3.1 Introduction

The following sections consist of a series of simulations designed to emulate flotation of chalcopyrite, coal, and phosphate. Each simulation was conducted to imitate typical industry flotation conditions. The results show the effect of changing the operating conditions of the system. Single and multiple component simulations were run for each of the minerals, and direct output from SimuFloat is used here to display the results.

Unless otherwise noted, the operating conditions for both single and multiple component flotation are as shown in Table 3.

Table 3: Simulation Standard Operating Conditions

Mineral	Power (kW/m^3)	S_b (cm/s)	S.G.	θ (°)	Frother	Frother Conc. (mg/m^3)	ζ_1 (-V)	# Cells	t (min)	h_f (cm)
Chalcopyrite	1.5	2	4.1	60	MIBC	192	0.15	4	3	10
Coal	1.5	2	1.3	55	MIBC	192	0.15	4	3	10
Phosphate	1.5	2	2.3	55	MIBC	192	0.15	4	3	10

The frother concentration used in these simulations corresponds to a surface tension of 68 mN/m. A ζ-potential of -0.15V corresponds to a near neutral pH, which is generally favorable for flotation (Mitchell, Nguyen, & Evans, 2005). A full description of all the input parameters used in the following simulations can be found in Appendix B.

3.2 Single Component Feed

Simulations run using a single component feed assume a 100% feed grade of the given mineral. Chalcopyrite was first simulated under the standard conditions listed in Table 3. Figure 8 shows the overall recovery resulting from this simulation.

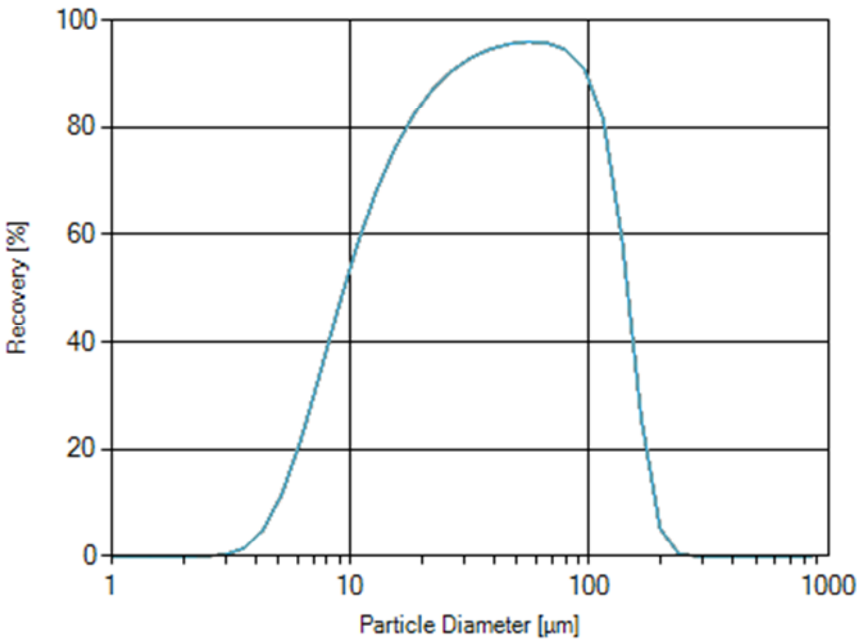


Figure 8: Chalcopyrite Recovery at Standard Conditions, Logarithmic. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 4.1, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = -15 mV, 4 cells, Retention time = 3 min, Froth Height =10 cm. This plot shows the typical bell-shaped curve obtained in flotation practice.

Note that there is almost no recovery of particles larger than 200 µm, which is accentuated in Figure 9.

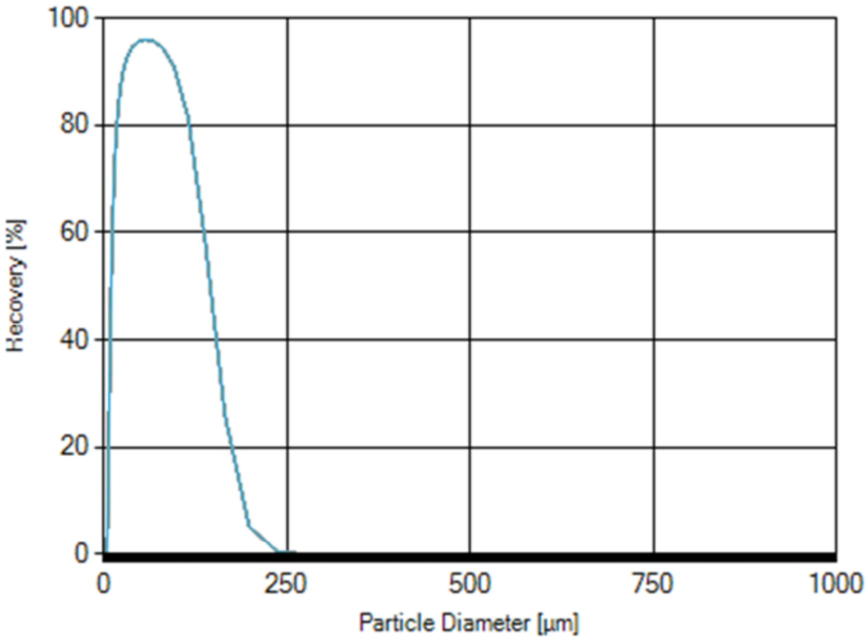


Figure 9: Chalcopyrite Recovery at Standard Conditions, Linear. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 4.1, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = -15 mV, 4 cells, Retention time = 3 min, Froth Height =10 cm. This plot illustrates the difficulty of coarse particle flotation that is observed in industry.

When the energy put into the flotation cell is reduced, a decrease in fine particle recovery and an increase in coarse particle recovery are observed, as shown in Figure 10:

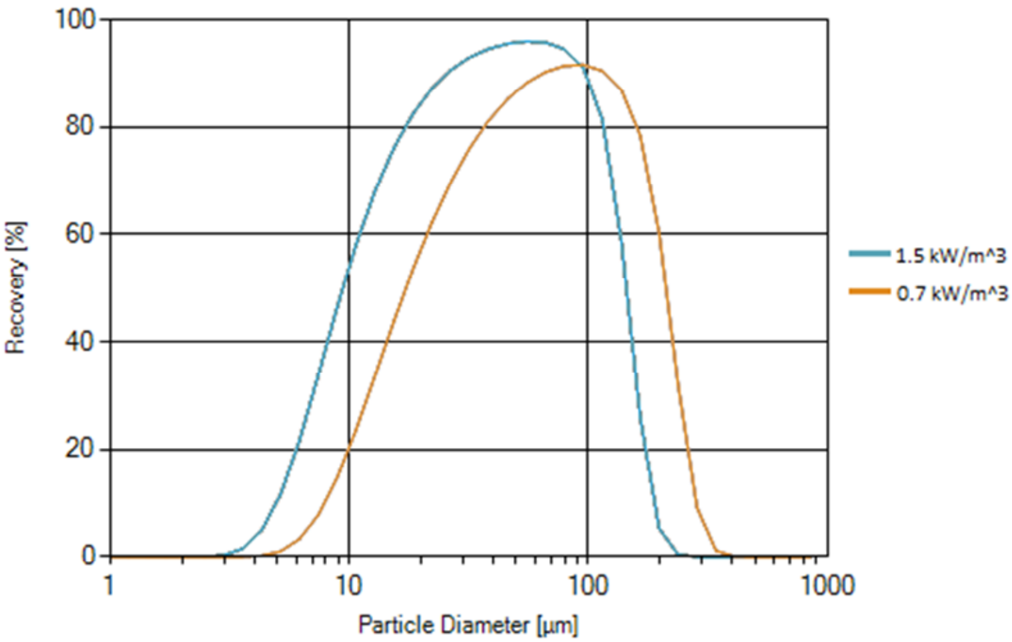


Figure 10: Chalcopyrite Recovery, Decreased Specific Power. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 4.1, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = -15 mV, 4 cells, Retention time = 3 min, Froth Height =10 cm. The blue line represents flotation at standard conditions and the orange line represents flotation with the specific power input decreased to 0.7 kW/m^3. A reduction in specific power aids coarse particle flotation, but harms fine particle flotation.

When the power input is reduced the kinetic energy of attachment, Eq. [17], is reduced, lowering the probability of fine particles attaching to bubbles, Eq [10]. Fines recovery deteriorates because the small particles no longer have the kinetic energy to rupture the wetting film of a bubble, causing them to bounce off the bubble when a collision occurs. Inversely, coarse particle recovery improves because the kinetic energy of detachment, Eq [23], is reduced at a lower power input. This results in a lower probability of detachment, Eq [21].

The effect of increasing froth height is shown in Figure 11:

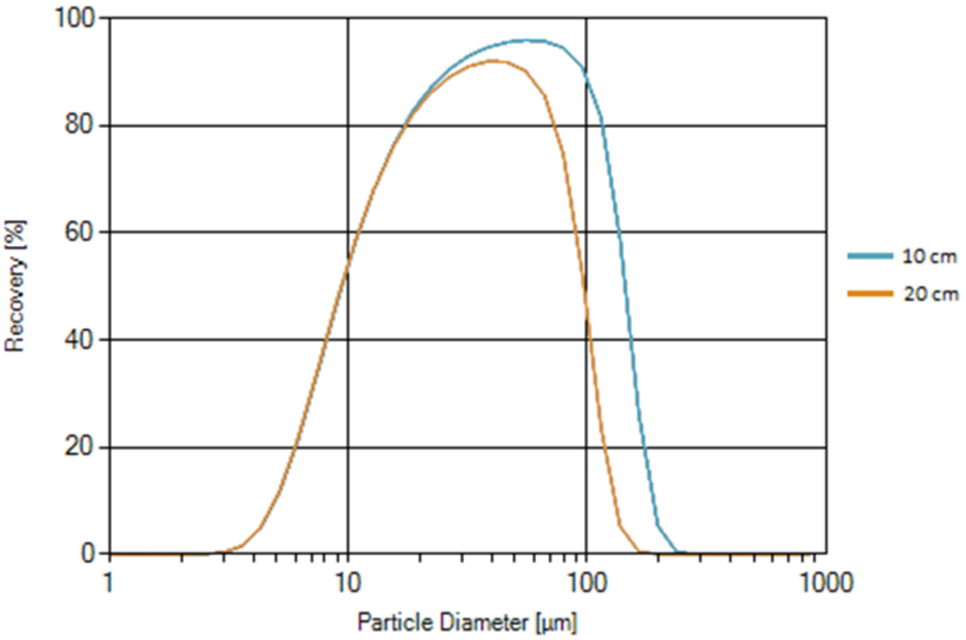


Figure 11: Chalcopyrite Recovery, Increased Froth Height. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 4.1, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = -15 mV, 4 cells, Retention time = 3 min, Froth Height =10 cm. The blue line represents standard conditions and the orange line represents recovery with the froth height increased to 20 cm. Increasing froth height harms coarse particle recoveries.

As froth height increases, bubble size in the froth becomes larger. This reduces the total surface area of the bubbles, which reduces their particle carrying capacity. Thus, the larger and less hydrophobic particles fall back to the pulp. This improves flotation selectivity, but lowers the recovery, especially that of coarse particles. This effect on the overall recovery from the flotation cell is not caused by any mechanism in the pulp phase; it is caused exclusively by the froth phase.

The froth recovery curve for flotation at standard conditions, as well as a froth height of 20 cm is displayed in Figure 12.

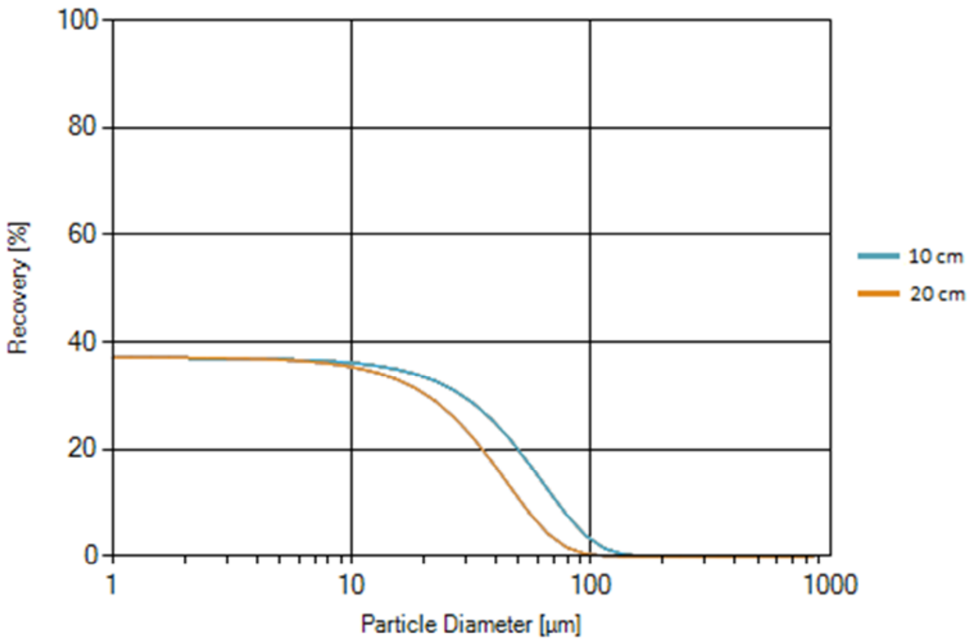


Figure 12: Chalcopyrite Froth Recovery, Increased Froth Height. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 4.1, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = -15 mV, 4 cells, Retention time = 3 min, Froth Height =10 cm**.** The blue line represents standard conditions, and the orange line represents a froth height increased to 20 cm. Froth recovery begins to deteriorate for particles larger than 10 µm with the increased froth height.

This plot emphasizes the fact that increasing the froth height is detrimental to the froth phase recovery of coarse particles. It also shows that the froth phase recovery is significantly lower than the overall recovery. This is in agreement with results reported in the literature (Laplante, Toguri, & Smith, 1983).

Figure 13 shows the effect of using a distribution of bubble sizes in simulations.

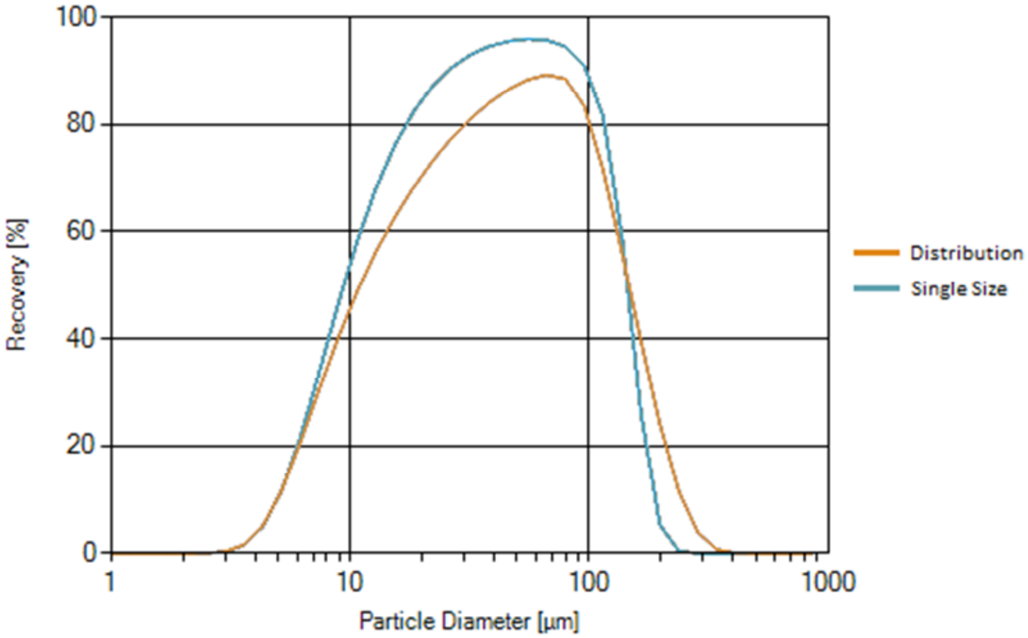


Figure 13: Chalcopyrite Recovery, Bubble Size Distribution. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 4.1, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = -15 mV, 4 cells, Retention time = 3 min, Froth Height =10 cm. The blue line represents flotation at standard conditions, while the orange line represents flotation using 6 bubble sizes. Coarse particle recovery improves while fine and medium particle recovery deteriorates.

The bubble size calculated under standard conditions is 1.7 mm. The distributed bubble sizes range from 1.25 to 2.5 mm in increments of 0.25 mm. These results are in line with expectations from results found by Schubert. Larger bubbles are able to float coarser particles due to a higher buoyant force (Schubert, 1999). The inverse is also observed in simulations and in practice; small bubbles are better for flotation of fine particles.

Figure 14 shows the effect of increasing the rate of air addition to the flotation cell.

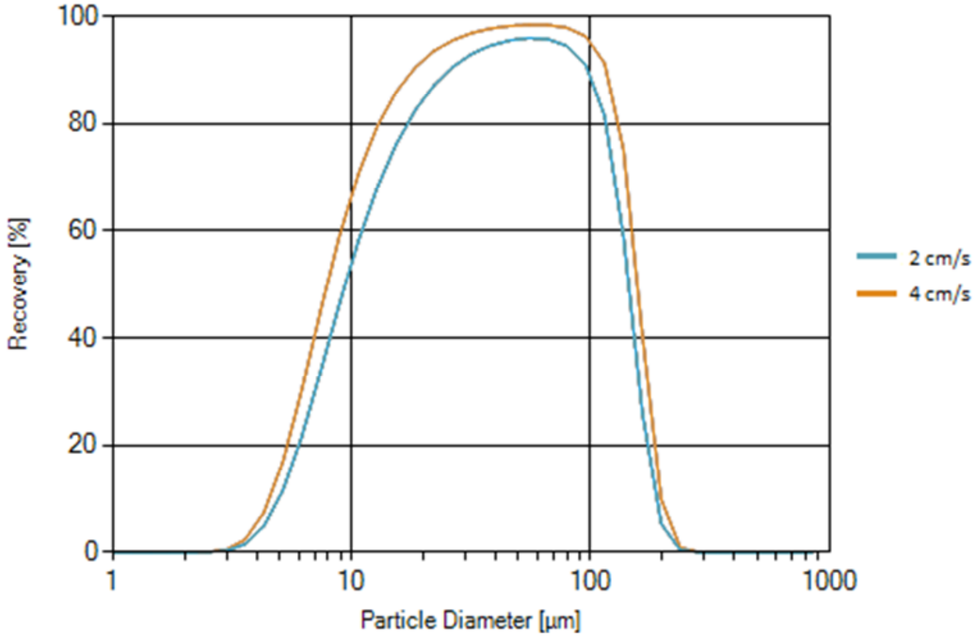


Figure 14: Chalcopyrite Recovery, Increased Gas Rate. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 4.1, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = -15 mV, 4 cells, Retention time = 3 min, Froth Height =10 cm. The blue line represents flotation at standard conditions, while the orange line represents flotation with a gas rate of 4 cm/s. Increasing the gas rate aids in overall chalcopyrite recovery.

The recovery of chalcopyrite increased from 88.7% to 93.4%. These results are qualitatively supported by results obtained through batch flotation tests reported in the literature, which suggest that an increase in the superficial gas velocity will aid in flotation (Yang & Aldrich, 2006; Laplante, Toguri, & Smith, 1983).

The effect of frother concentration on flotation performance is shown in Figure 15.

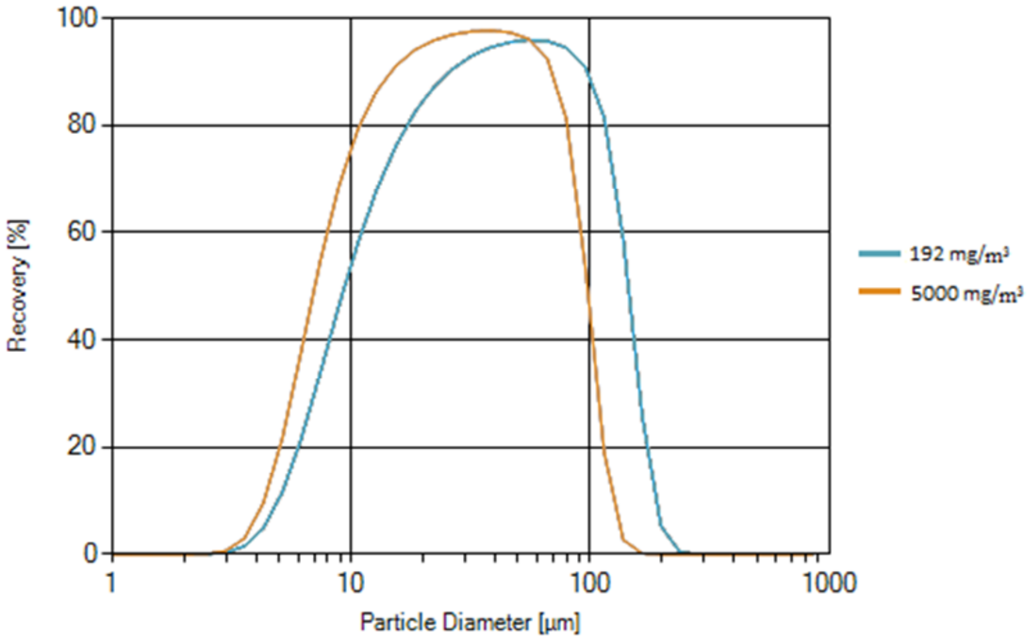


Figure 15: Chalcopyrite Recovery, Increased Frother Concentration. Input Parameters other = MIBC, Frother Concentration = 192 mg/m3, ζ-potential: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 4.1, θ = 60°, ζ-potential = -15 mV, 4 cells, Retention time = 3 min, Froth Height =10 cm. The blue line represents flotation at standard conditions, and the orange line represents flotation at a frother concentration of 5000 mg/m^3. The addition of too much surfactant is detrimental to flotation performance.

An increase in surfactant proves to be detrimental to flotation; the overall recovery for the flotation bank drops by 10%. Galvin, Nicol, and Waters found that the addition of too much surfactant becomes harmful to flotation, however a moderate concentration often aids in flotation (1992). The negative effect on coarse particle flotation is seen because an increase in surfactant concentration lowers the surface tension of the medium. This lowers the work of adhesion, Eq [22], thus increasing the probability of detachment, Eq [21]. An increase in fine particle recovery is observed because lowering the surface tension enables the creation of smaller bubbles which aid in flotation of fines.

The advent of QEM*SEM enabled an easier determination of particle liberation characteristics for laboratory flotation feeds. A QEM*SEM liberation data set given by Sutherland is summarized in Table 4 and Table 5.

Table 4: Liberation Data for a Batch Flotation Feed (Sutherland, 1989)

Size (microns)		Wt. %	Wt.% Cu by QEM*SEM
	425	2.126	1.00
-425	300	0.768	0.43
-300	212	5.144	0.45
-212	150	12.807	0.90
-150	106	16.592	1.20
-106	75	10.49	1.67
-75	53	9.151	2.03
-53	38	5.937	8.46
-38	24	2.963	3.30
-24	17	5.398	2.13
-17		28.623	2.06
		100.0	

Note: Size functions less than 38 microns were separated using a Cyclosizer.
The sizes indicated here represent free chalcopyrite.

After flotation of the feed shown above, Sutherland found that the percentage of liberated (90-100% liberation) chalcopyrite particles in the concentrate were as shown in Table 5:

Table 5: Liberation Data for Flotation Concentrate (Sutherland, 1989)

Size (microns)		% Liberated Particles	Average DOL%
	150	54	70
-150	106	67	78
-106	75	76	84
-75	53	87	89
-53	38	90	91
-38	24	95	93
-24	17	96	93
-17	12	97	94
-12		97	94

The average DOLs were used to determine the average contact angle for each size class. These contact angles were then used to simulate flotation. Figure 16 shows the recovery of chalcopyrite when using a distribution of contact angles.

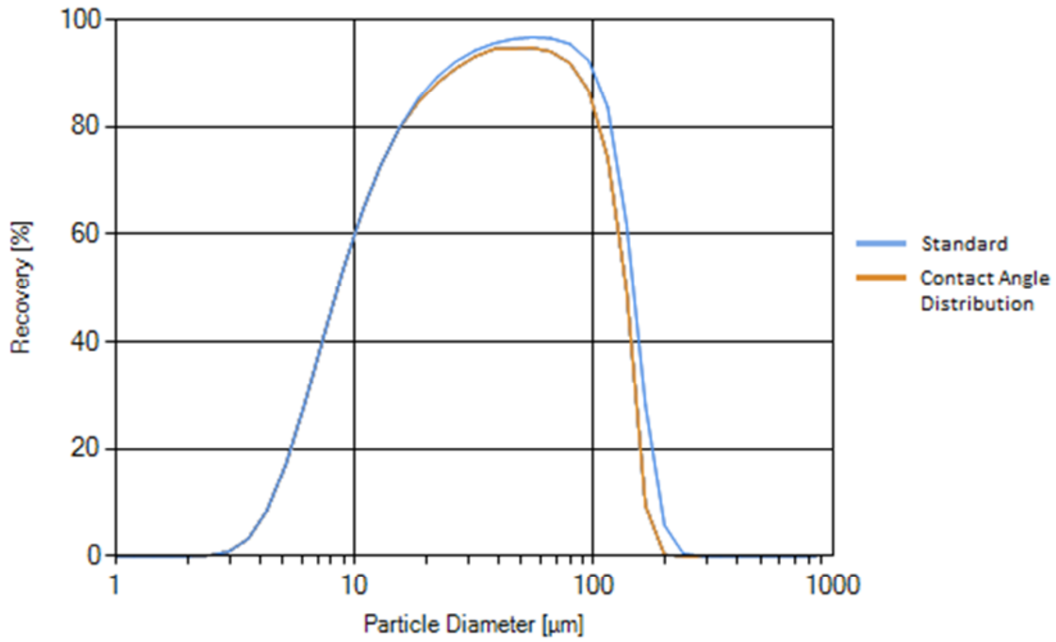


Figure 16: Chalcopyrite Recovery, Standard Conditions vs. Contact Angle Distribution. The simulated effect of liberation is only observed for coarse particles because the fine particles are almost all fully liberated.

With this data set only a small effect on flotation performance is seen because the simulated feed is a flotation concentrate. The absence of gangue particles, especially at fine sizes where the degree of liberation is high, produces an inclination for the two curves to be alike. Lastra, Sutherland and Savassi all observed the trend of increasing flotation recovery with increasing DOL (Sutherland, 1989; Savassi, 2006; Lastra, 2007).

The effect of liberation on recovery with respect to particle size is shown in Figure 17.

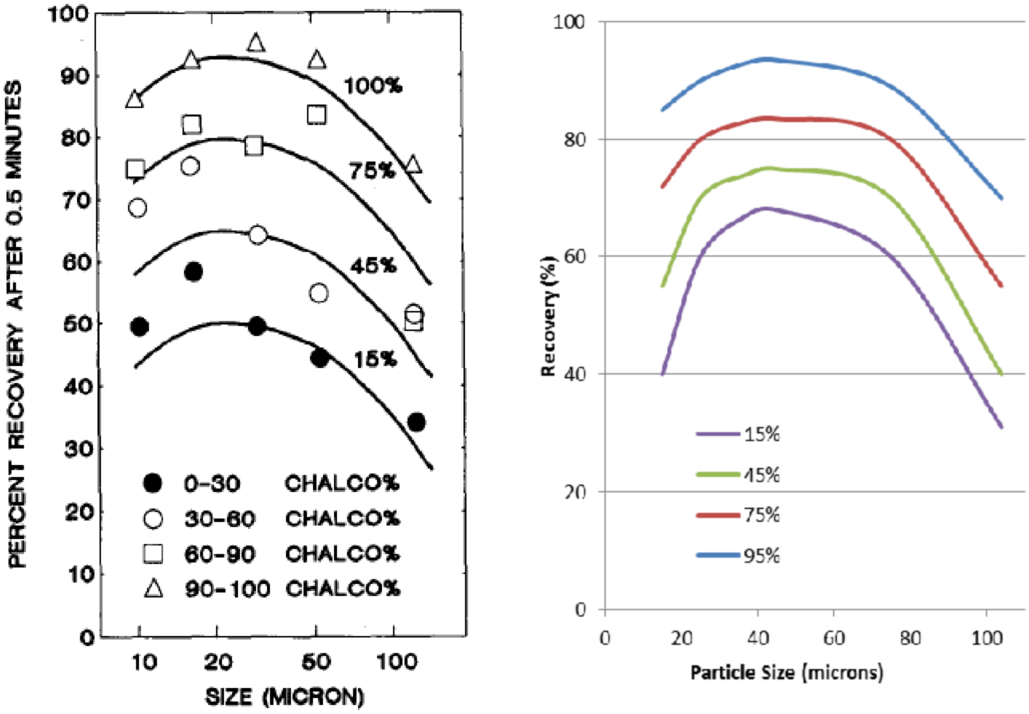


Figure 17: Chalcopyrite Recovery, Effect of Liberation. Input parameters 15% θ: 25°, 45% θ: 33°, 75% θ: 40°, 95% θ: 60°, Power = 2.5 kW/m^3, Gas rate = 2 cm/s, Froth height = 10 cm, Frother concentration = 192 mg/m^3, ζ-poential = -15 mV, Cells = 4, and Retention time = 3 min. The left hand plot shows Sutherland's results and the plot on the right displays results from SimuFloat. The same general trend is seen in both plots.

The operating parameters were not given for Sutherland's plot, shown in Figure 17, left (Sutherland, 1989). Simulations, the results of which are shown in Figure 17, right, were run to approximate Sutherland's results as closely as possible. As degree of liberation increases, the contact angle should increase, therefore the only variable between the "liberation" classes in these simulations is the contact angle.

Next, simulations were performed to predict the performance of coal flotation under the standard conditions. The overall recovery curve is shown in Figure 18.

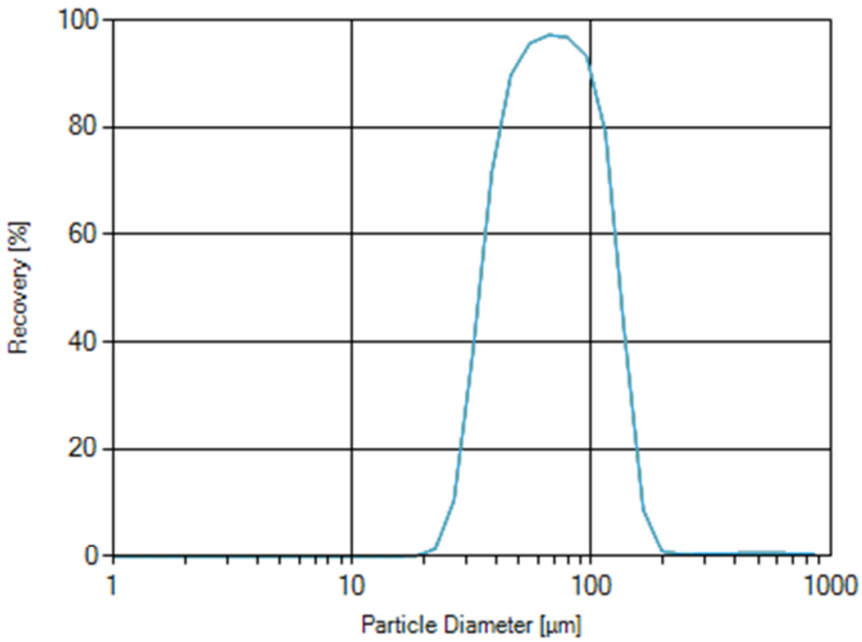


Figure 18: Coal Recovery, Standard Conditions. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 1.3, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = 0.15-V, 4 cells, Retention time = 3 min, Froth Height =10 cm. The recovery of coal is effective over a narrower size range than that of chalcopyrite. This is due to differences in both specific gravity and contact angle.

As particle size decreases below 325 mesh the recovery drops off dramatically. Above 100 mesh there is almost no particle recovery, but this can be combatted by reducing the froth height as shown in Figure 19.

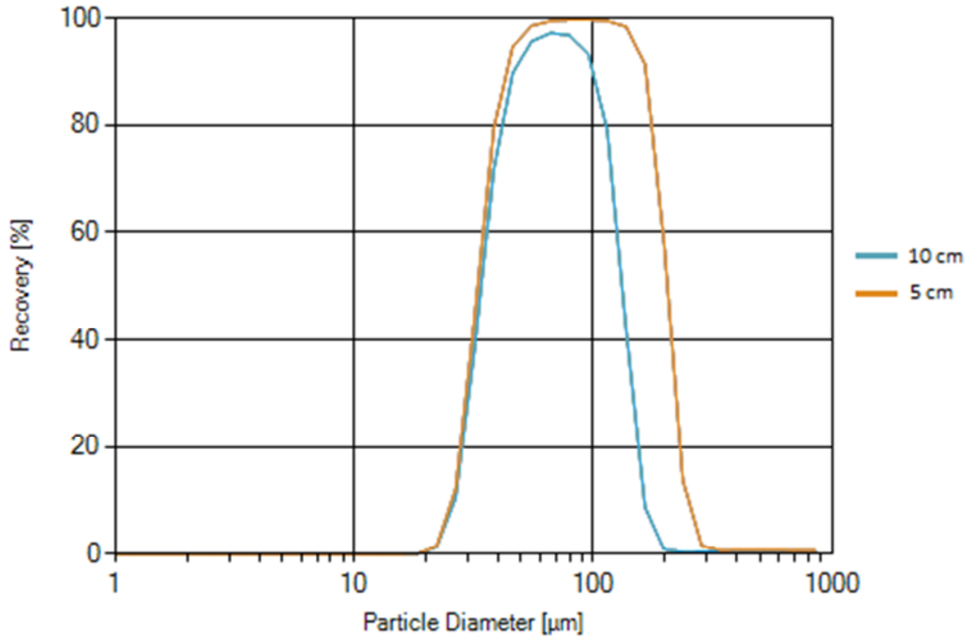


Figure 19: Coal Recovery, Decreased Froth Height. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 1.3, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = 0.15-V, 4 cells, Retention time = 3 min, Froth Height =10 cm. The blue line represents coal flotation under standard conditions, while the orange line represents flotation with a froth height of 5 cm.

This reduction in froth height has only a slight effect on fine particles, but coarse particle recovery improves more significantly. To further illustrate this effect, Figure 20 shows the effect of further reducing the froth height to 1 cm.

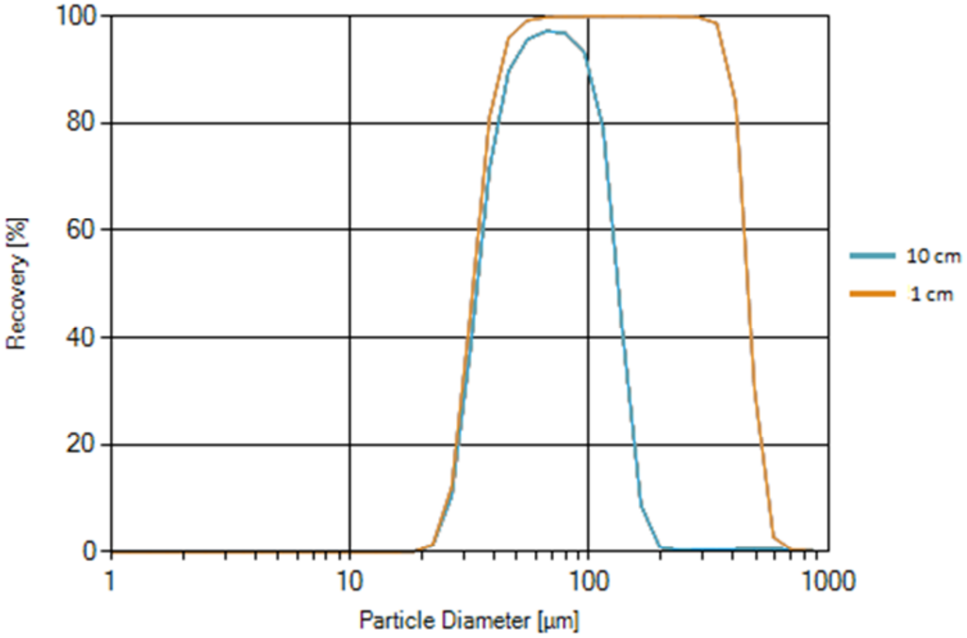


Figure 20: Coal Recovery, 1 cm Froth Height. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 1.3, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = 0.15-V, 4 cells, Retention time = 3 min, Froth Height =10 cm. The reduction in froth height allows for flotation of coarser particles.

When the froth height is reduced to 1 cm, particles of nearly 800 µm are floated. It has been shown that flotation with a froth height of zero, or "frothless" flotation will result in an increased recovery of coarse particles (Rubinstein & Melik-Gaikazyan, 1998). However, the increased recovery comes at the expense of selectivity due to an increase in recovery by entrainment, Eq [30].

The last mineral to be simulated was phosphate, for which the flotation recovery curve is shown in Figure 21.

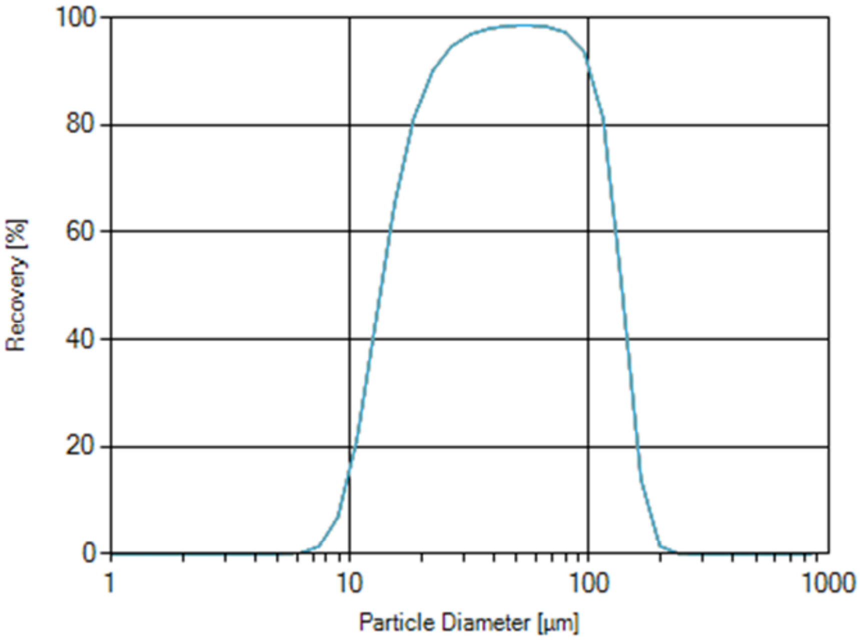


Figure 21: Phosphate Recovery, Standard Conditions. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 2.3, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, ζ-potential = 0.15-V, 4 cells, Retention time = 3 min, Froth Height =10 cm. Phosphate floats over a wider size range than coal under standard conditions because it has a higher specific gravity.

To show the effect of differing specific gravities, Figure 22 compares the recovery of chalcopyrite, coal, and phosphate when floated under standard conditions.

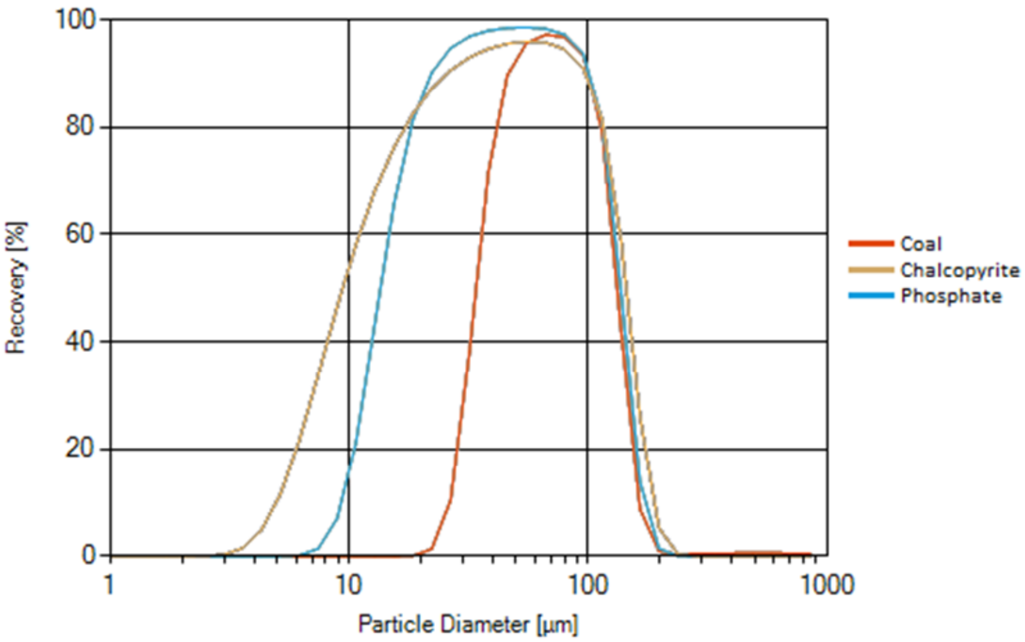


Figure 22: Chalcopyrite vs. Coal vs. Phosphate, Standard Conditions. Input parameters shown in Table 3. The maximum particle size floated is virtually the same for each mineral.

The differences in specific gravity between the three minerals affects fine particle recoveries, but the three lines nearly converge around a particle size of 100 µm. This effect is observed due to the convergence of P_a [11] and P_d [12], shown in Figure 23 and Figure 24.

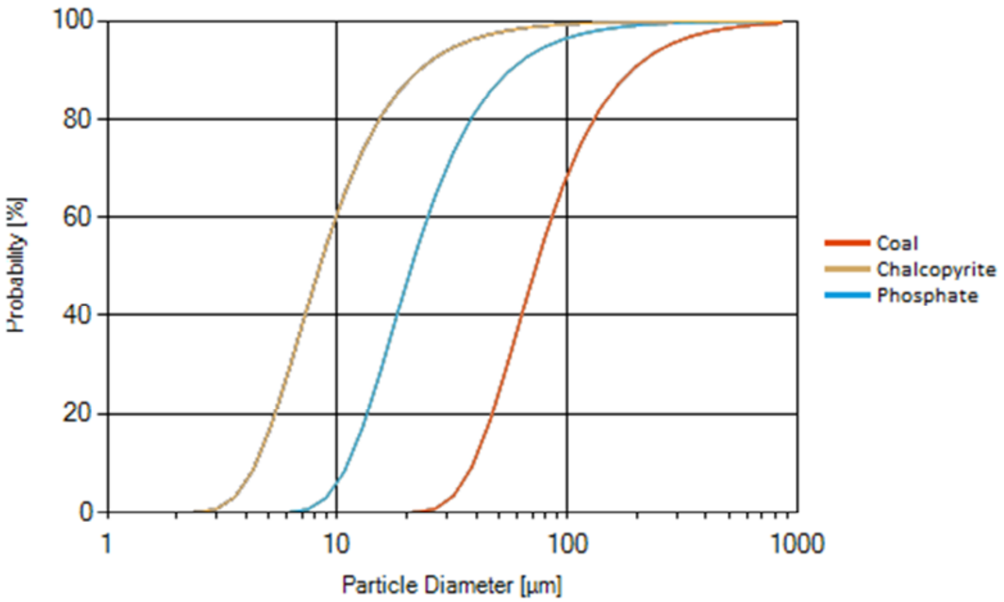


Figure 23: Probability of Attachment, Chalcopyrite vs. Coal vs. Phosphate. Input parameters shown in Table 3. The convergence of P_a is seen with an increase in the E_k due to increasing particle diameter.

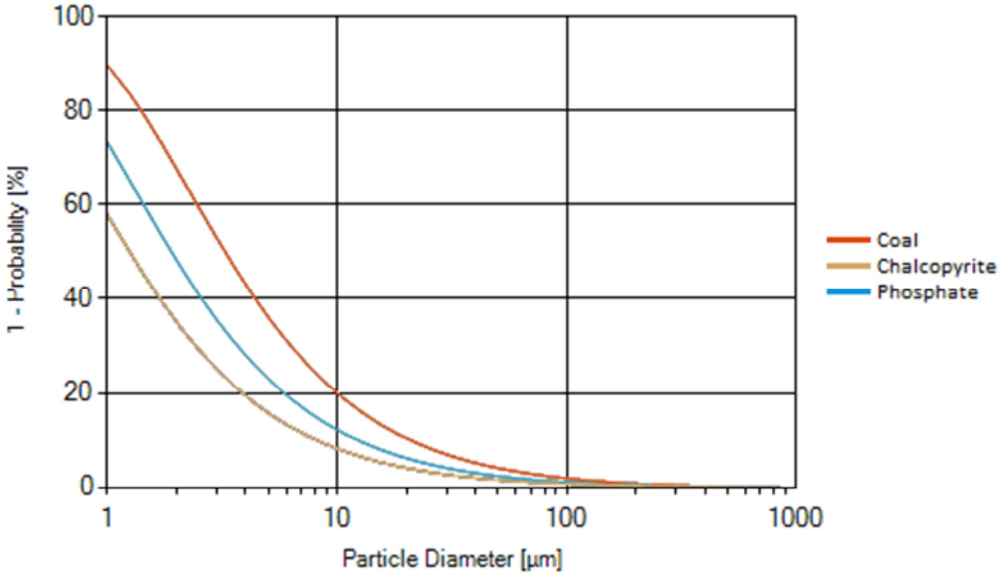


Figure 24: Probability of Remaining Attached, Chalcopyrite vs. Coal vs. Phosphate. Input parameters shown in Table 3. The probability of remaining attached decreases with increasing particle size due to an increase in E'_k.

Increasing particle diameter increases the mass of the particle. This increases both the probability of attachment and the probability of detachment because the kinetic energy of attachment [17] and detachment [23] are dependent on mass. As seen from the plots, large particles almost always attach to a bubble, but they have a very low probability of remaining attached to the bubble.

As shown in Figure 25, the particle ζ-potential has an effect on fine particle flotation.

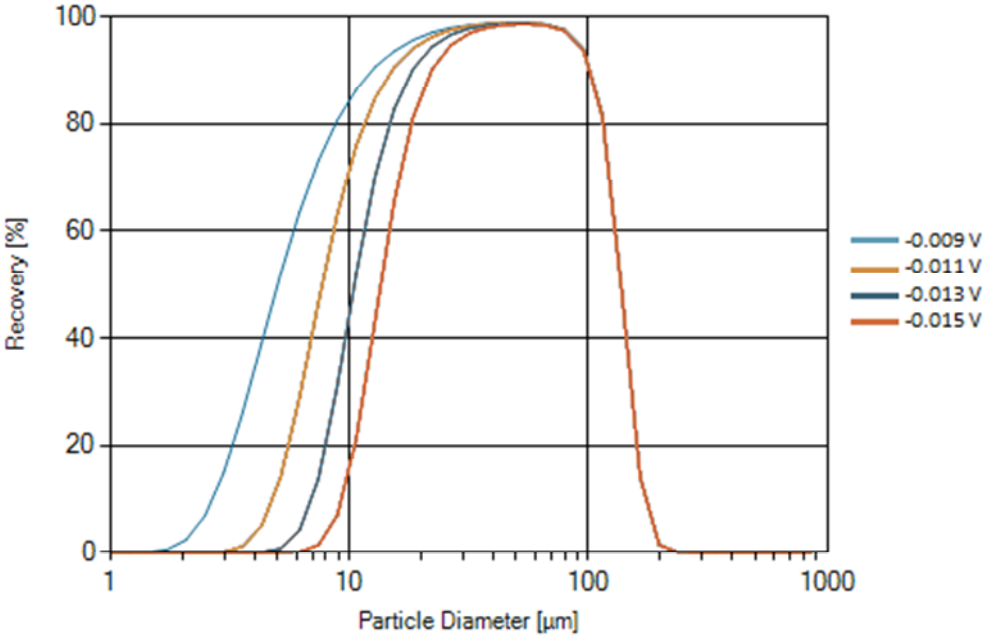


Figure 25: Phosphate Recovery, Effect of ζ-Potential. Input Parameters: Power = 1.5 kW/m3, Gas Rate = 2 cm/s, S.G. = 1.3, θ = 60°, Frother = MIBC, Frother Concentration = 192 mg/m3, 4 cells, Retention time = 3 min, Froth Height =10 cm. Fine particle flotation benefits as the negative ζ-potential approaches zero.

The ζ-potentials of the plots from left to right are -0.009, -0.011, -0.013, and -0.015. As the negative ζ-potential decreases, the energy barrier [11] increases, decreasing the probability of attachment for small particles.

3.3 Multiple Component Feed

Input of multiple component feeds requires the use of the Feed Grade form, previously shown in Figure 7. The input parameters used in this simulation for multiple component feeds are shown in Table 6. The particle size distribution used roughly approximates a Gaudin-Schuhmann distribution.

Table 6: Multiple Component Feed Parameters

	Mineral				Middlings				Gangue			
Ore	SG	% Feed	% Grade	θ	SG	% Feed	% Grade	θ	SG	% Feed	% Grade	θ
Chalcopyrite	4.1	4	34	60	3.2	2	10	15	2.7	94	0	5
Coal	1.3	50	100	55	2.0	30	50	15	2.7	20	0	5
Phosphate	2.3	15	100	55	2.5	30	30	15	2.7	55	0	5

The values given for chalcopyrite simulate a flotation feed grade of 1.56% copper. The coal feed is 35% ash and the phosphate feed is 24% grade.

The plots for multiple component feeds become difficult to read if more than one simulation is shown on each. For this reason, each plot in this section will show a single simulation. Figure 26 shows the recovery curves for chalcopyrite using standard conditions and the component parameters from Table 6.

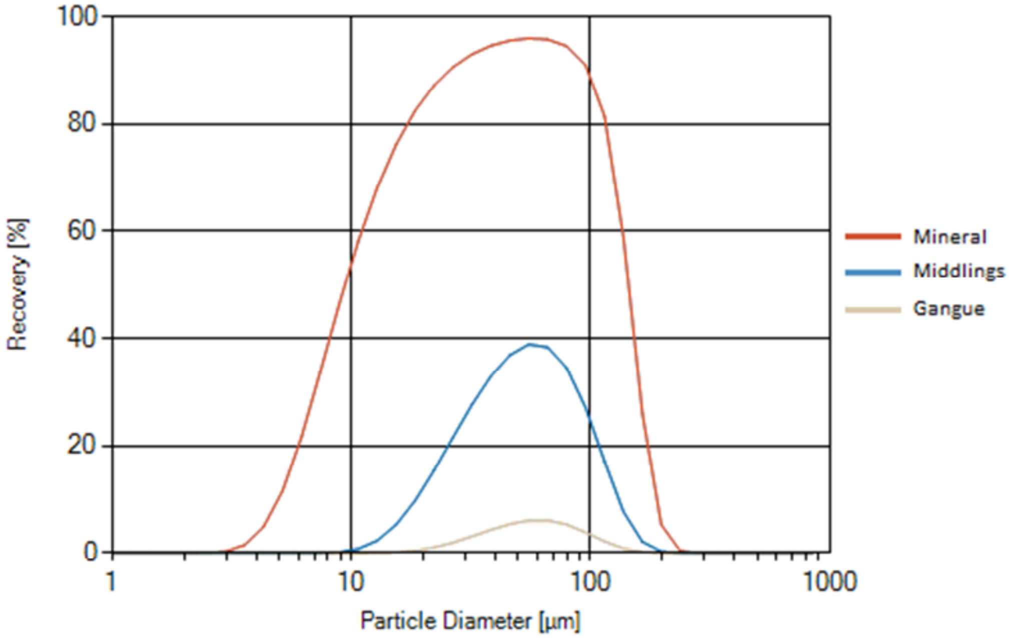


Figure 26: Chalcopyrite Recovery, 3 Component Feed. Input parameters shown in Table 6. Each line represents a different component of the feed. Recoveries vary due to differences in the contact angle and specific gravity.

The red line represents the mineral, the blue line represents the middlings, and the tan line represents gangue. Overall mass recovery to the product is 10.6%, copper recovery is 86.7%, and the product grade is 12.8%. SimuFloat also reports a mineral recovery of 93.6%, a middlings recovery of 39.7% and a gangue recovery of 6.4%.

It is widely reported that flotation selectivity is improved by increasing the froth height. Figure 27 shows the effects of changing the froth height on the three feed components.

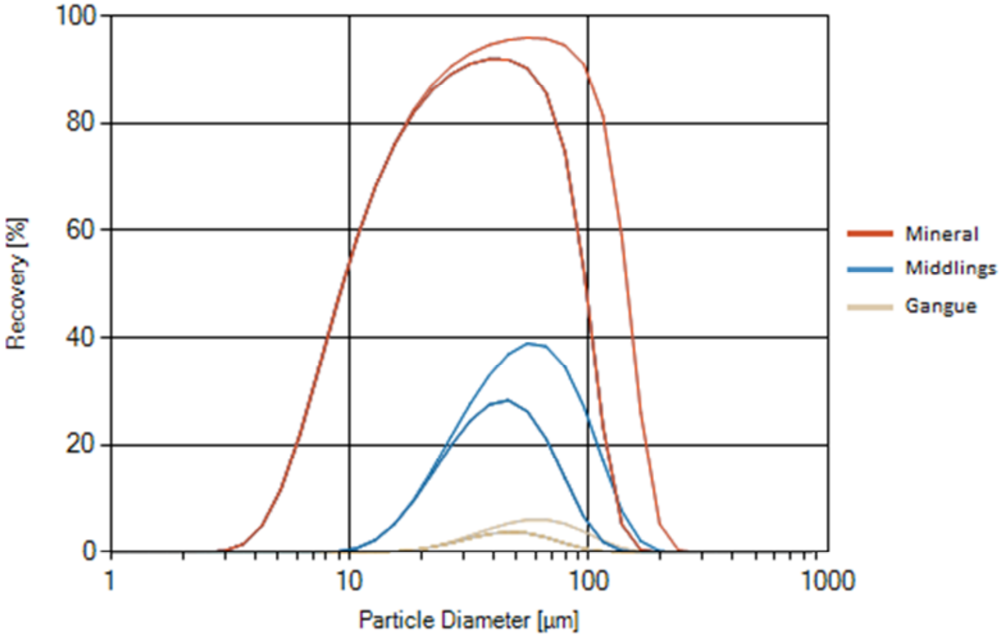


Figure 27: Chalcopyrite, 3 Component Feed, Increased Froth Height. Input parameters shown in Table 6. The increase in froth height from 10 cm to 20 cm lowered the overall copper recovery, but increased the grade of the product.

Each of the colors represents the same feed stream as those in the previous figure. An increase in the froth height causes each curve to shift down and to the left on the coarse end. Initially, at a froth height of 10 cm, the product grade is 15.6% copper. After increasing the froth height to 20 cm, the product grade increases to 19.6%. The increased froth height reduces recovery by entrainment [30]. This trend is supported by many researchers who found that increasing the froth height provided better drainage of entrained particles and of less hydrophobic coarse particles (Ekmekci, Bradshaw, Allison, & Harris, 2003; Hanumanth & Williams, 1990).

While no model for cleaning stages in flotation yet exists in SimuFloat, cleaning may be simulated by substituting the results back into the simulation. This facilitates the generation of grade-recovery curves for the flotation bank. The recovery versus product grade is plotted for the values given in Table 6, as well as for contact angles reduced by 1/3, in Figure 28.

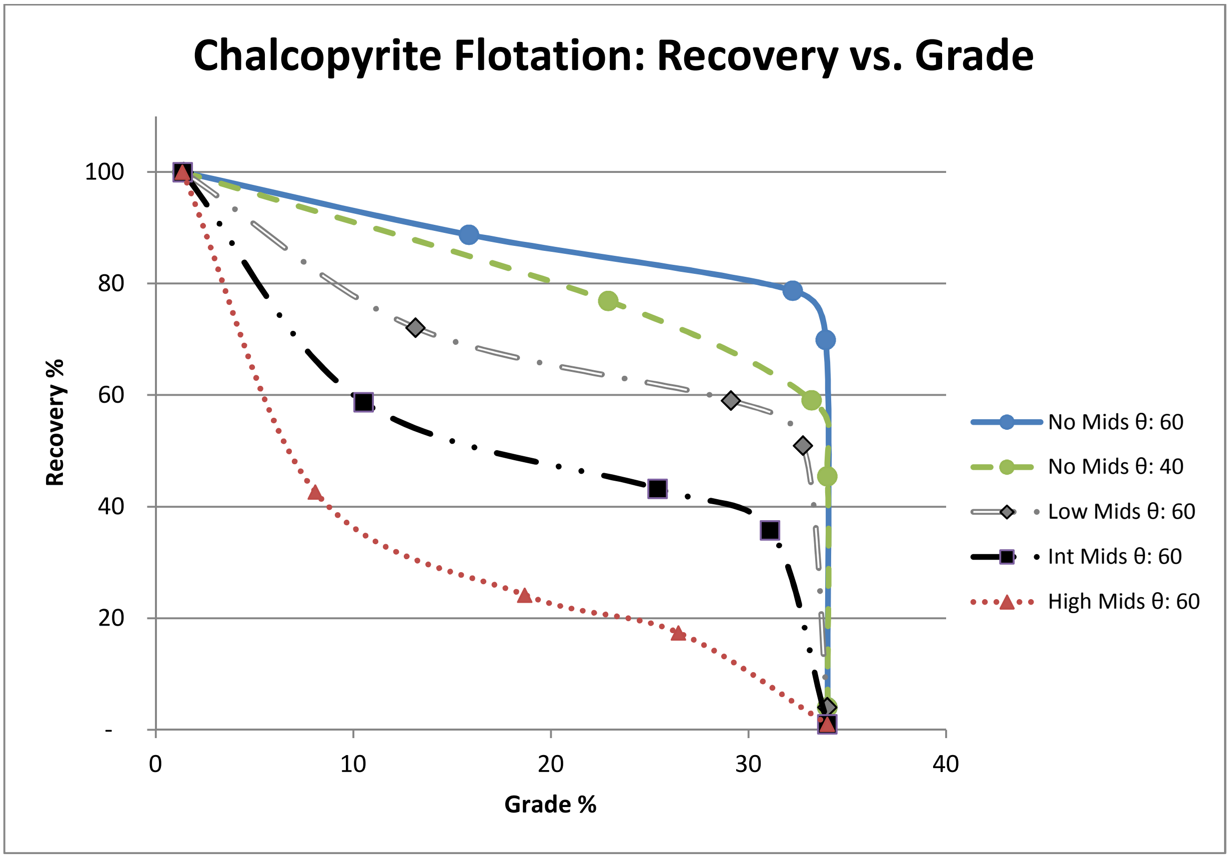


Figure 28: Chalcopyrite Recovery vs. Grade. The solid line and dashed lines represent feeds with no middlings at 60° and 40° contact angle, respectively; all other simulations have a 60° contact angle. The dash-dot line represents a low middlings feed, the dash-dot-dot line represents an intermediate middlings feed and dotted line represents a high middlings feed. As shown by the two no middlings feeds, flotation at the lower contact angle produces a slight higher grade product at the expense of copper recovery. Flotation performance deteriorates as the concentration of non-liberated particles increases.

Two simulations were run with the same feed characteristics, but at different contact angles. Three simulations were run with a increasing concentrations of middlings in the feed. All five simulated feeds contained 1.36% copper by weight. As expected, the simulation with a lower contact angle produces a steeper grade-recovery curve. After the first flotation stage, copper recovery for the 40° simulation is nearly 10% lower than the 60° simulation, but it produces a product with a 7% higher grade. At the end of three stages, the 40° simulation recovers 18% less copper, while achieving only 1% higher product grade. The non-liberated feed returns lackluster flotation results recovering only 42% of the copper in the feed after the first stage.

Figure 29 shows the recovery of coal for a three component feed.

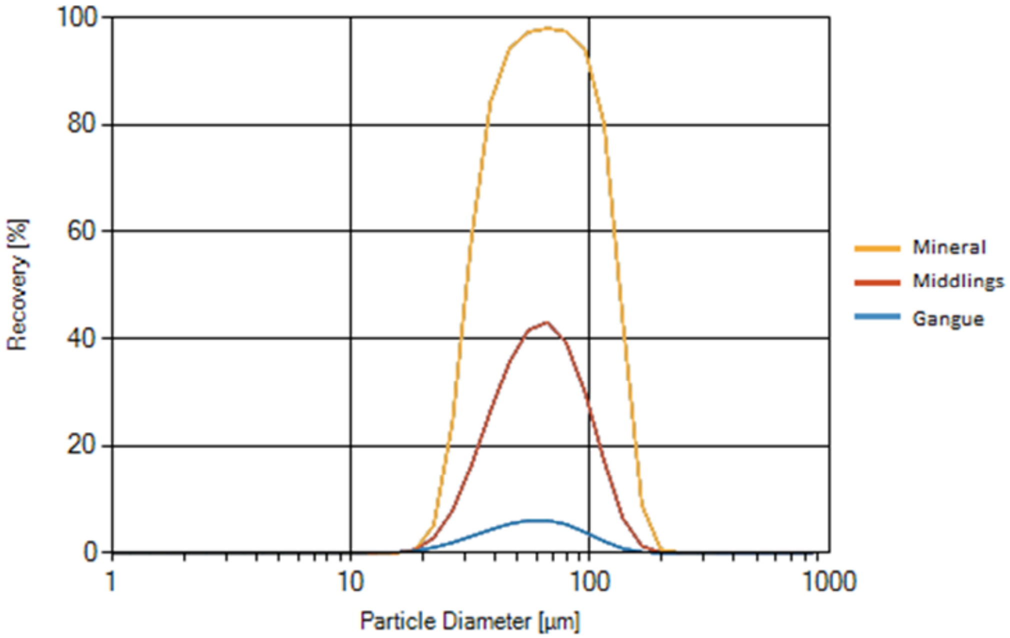


Figure 29: Coal Recovery, 3 Component Feed. Input parameters shown in Table 6. Each line represents a different component of the feed. As expected, recoveries vary due to differences in the contact angle and specific gravity.

The orange line represents the mineral, the red line represents the middlings, and the blue line represents gangue. Overall mass recovery to the product is 49.8%, coal recovery is 68.5%, and the product grade is 89.4%. SimuFloat also reports a mineral recovery of 80.3%, a middlings recovery of 29.4% and a gangue recovery of 4.2%.

The effect of increasing the number of cells in the flotation bank is seen in Figure 30.

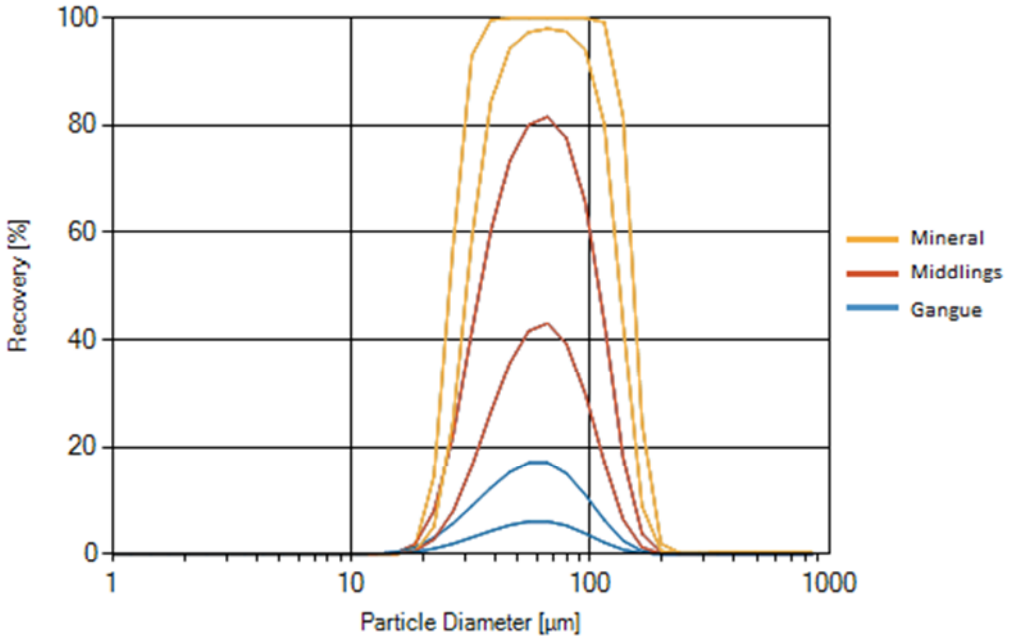


Figure 30: Coal Recovery, 3 Component Feed, Increased Number of Cells. Input parameters shown in Table 6. An increase in the number of flotation cells to twelve caused each curve to shift directly upwards.

Each of the colors represents the same feed stream as those in the previous figure. With four cells the product grade was 89.4% with 68.5% recovery. After increasing the number of cells to twelve, the product grade decreases to 82.4%, but recovery increases to 82.7%. Changing the number of cells directly affects the overall flotation bank recovery Eq. [32].

Figure 31 shows the recovery of phosphate for a three component feed.

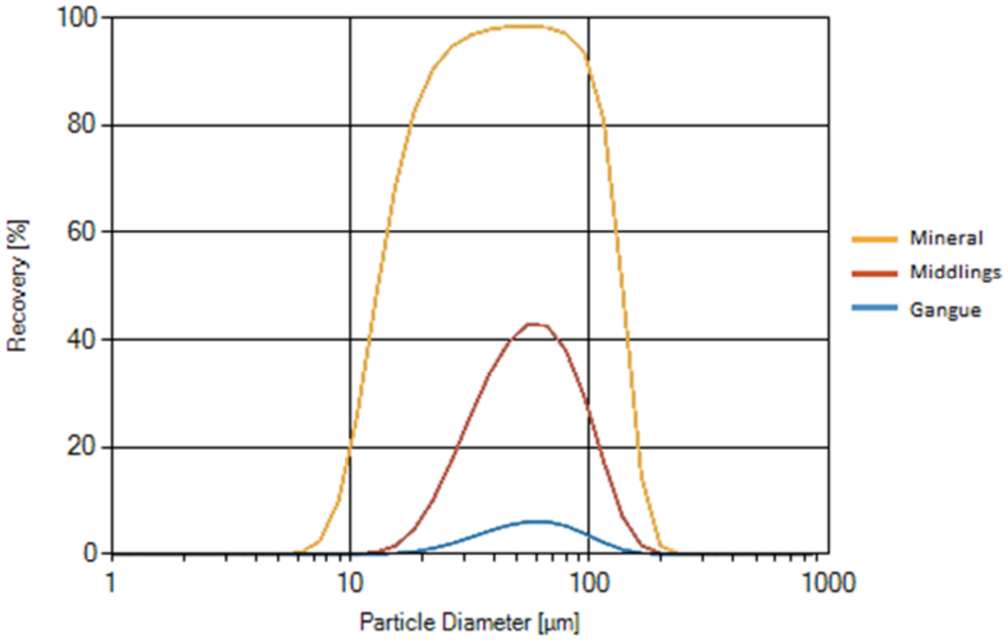


Figure 31: Phosphate Recovery, 3 Component Feed. Input parameters shown in Table 6. Each line represents a different component of the feed. As expected, recoveries vary due to differences in the contact angle and specific gravity. Much like with the single component feed, the recovery curves for phosphate fall in between those of chalcopyrite and coal.

The orange line represents the mineral, the red line represents the middlings, and the blue line represents gangue. Overall mass recovery to the product is 25.3%, phosphate recovery is 68.1%, and the product grade is 64.7%. SimuFloat also reports a mineral recovery of 90.2%, a middlings recovery of 31.4% and a gangue recovery of 4.2%.

The effect of increasing the number of cells in the flotation bank is seen in Figure 32.

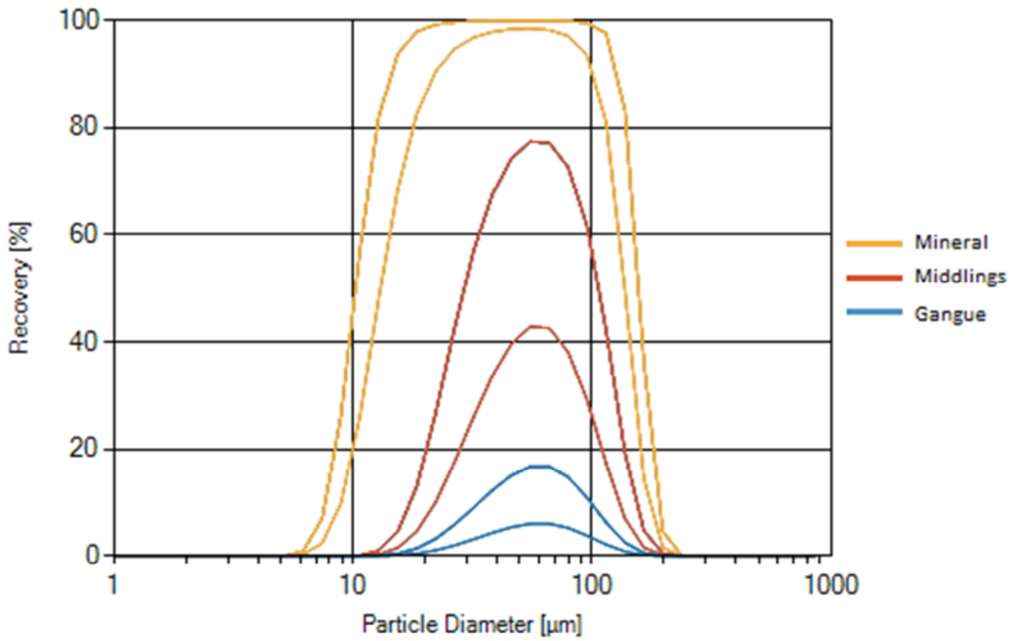


Figure 32: Phosphate Recovery, 3 Component Feed, Increased Retention Time. Input parameters shown in Table 6. By increasing the retention time, more phosphate, middlings, and gangue were recovered. The net effect is a decrease in phosphate lost to the tailings, but a lower product grade.

Each of the colors represents the same feed stream as those in the previous figure. An increase in the retention time to nine minutes per cell causes each curve to shift directly upwards. With a retention time of three minutes per cell the product grade is 64.7% with 68.1% recovery. After increasing the retention time to nine minutes per cell, the product grade decreases to 50.6%, but recovery increases to 82.9%.

3.4 Model Validation

Simulation results were compared to the results of batch flotation tests performed by Aaron Noble at Virginia Tech. The test parameters for the batch flotation tests are shown in Table 7.

Table 7: Silica Flotation Test Parameters

Test Parameter	Value
Weight % Solids	5
Particle Specific Gravity	2.65
Froth Height	4 cm
Superficial Gas Velocity	2 cm/s
Frother	MIBC
Frother Concentration	10 ppm

The flotation tests were performed on 35 µm nominally mono-size silica particles, the true size distribution can be found in Appendix C. A 9.75” diameter cell was used in conjunction with a 2.75” rotor and corresponding stator. The slurry was treated with 15 g/tonne of dodecylamine.

Flotation tests were conducted at three specific power inputs. The measured bubble sizes were 1.61, 1.35, and 1.28 mm for the 0.08, 0.266, and 0.488 kW/m^3 tests, respectively. However, it was determined that the bubble sizes were measured with a high degree of uncertainty. For this reason, the bubble sizes were adjusted in the simulation to fit the experimental recoveries. The actual bubble sizes used in simulation were 1.15, 1.05, and 1.00 mm for the 0.08, 0.266, and 0.488 kW/m^3 tests, respectively. Table 8 shows the recoveries obtained in the lab tests and in simulation.

Table 8: Silica Flotation Recoveries

Specific Power	0.08 (kw/m^3)		0.266 (kW/m^3)		0.488 (kW/m^3)	
Time (min)	Sim.	Exp.	Sim.	Exp.	Sim.	Exp.
0.5	44.7	47.2	60.9	61.6	62.5	68.7
1.0	59.1	65.3	74.8	75.6	76.4	75.0
2.0	71.5	78.3	84.4	85.3	86.3	84.8
4.0	81.0	85.5	91.2	89.8	92.4	89.0

The average difference between the experimental and simulated results is ± 3.0%. All simulated recoveries are within 7% of the actual recovery. These results are also shown graphically in Figure 33.

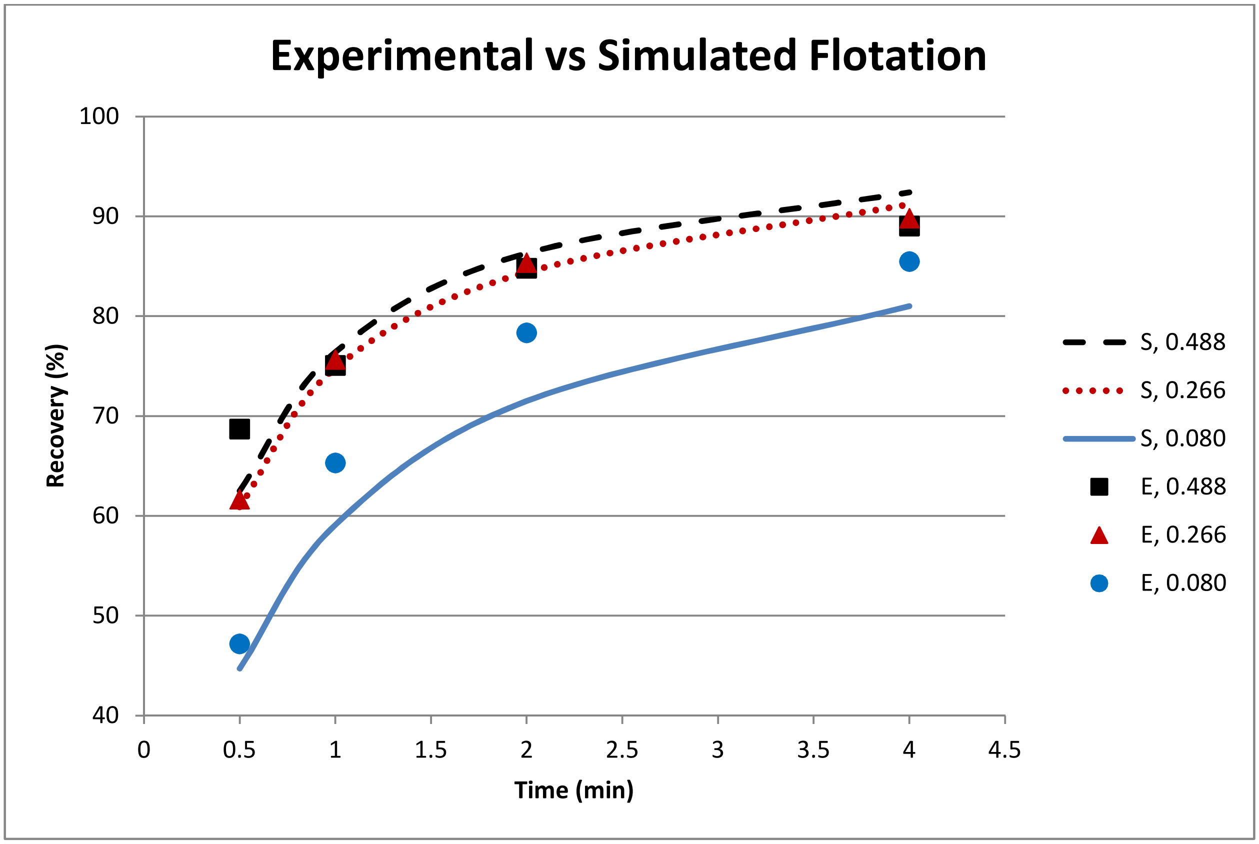


Figure 33: Experimental vs. Simulated Silica Flotation Recovery. Input parameters shown in Table 7. Black lines and markers represent a specific power input of 0.488 kW/m^{3,} red lines and markers represent a specific power input of 0.266 kW/m^3, and blue lines and markers represent a specific power input of 0.080 kW/m^3. For the simulation of silica flotation, recoveries matched well with experimental results.

The markers represent experimental results, the lines represent simulated results, and the quantities in the legend correspond to the specific power input. This figure makes it clear SimuFloat has predictive capabilities. The shapes of the simulated curves closely approximate those for the flotation recoveries obtained in the lab.

3.5 Conclusion

Froth flotation simulations have been performed using a predictive model derived from first principles. Unlike many of the current flotation models, the model used in SimuFloat does not require the input of a floatability constant determined from plant data. This gives the simulator predictive capabilities without the need for extensive in-plant flotation studies. Detailed simulations were run for chalcopyrite, coal, and phosphate. These simulations show the effect of changing the physical and chemical parameters of flotation, the outcomes of which are supported by results reported in the literature. A successful first attempt at validating the accuracy of both the model and the simulator by comparisons to batch silica flotation tests was made. Simulated silica recovery was found to be within 3% of the experimental results on average.

4 Conclusion

4.1 General Conclusion

Modeling of flotation is a vital task for improving the flotation process. It allows the researcher to learn much about the mechanics of a flotation cell without the cost or time requirements of lab or pilot scale testing. The flotation simulator developed in the present work is a useful tool for simulating flotation, while accounting for both hydrodynamics and surface chemistry. SimuFloat was found to be relatively accurate at predicting flotation under a variety of conditions, and has been validated through comparison with experimental data.

4.2 Recommendations for Future Work

While SimuFloat marks a step forward in the process of developing a comprehensive flotation simulator, it is not complete. The following are areas of the simulator that could be improved through further research.

1. Introduce user defined, integrated flow sheets that may be solved using mass balances.
2. Include a relationship between concentration and contact angle for collectors used in flotation. This would make SimuFloat more industry friendly, as contact angle is commonly not measured in flotation practice.
3. Incorporate a relationship between ζ-potential and pH. The pH is not the only factor that affects ζ-potential. Like the contact angle, ζ-potential is not generally measured in the field. Replacement of ζ-potential with pH as a simulation input would make the program more industry friendly.
4. Small interface tweaks, such as the ability to input parameters in different units, and the ability to retain input values upon closing the program would make SimuFloat more user friendly.
5. Account for the effects of hydrophobic coagulation. This will improve the observed recovery of fine particles and bring the simulation predictions more in line with results observed in flotation practice.
6. Develop an equation relating air holdup, bubble size, and gas rate. In actual flotation systems, these three variables are interdependent.
7. Include a model to calculate contact angle based on liberation class.

Works Cited

Abrahamson, J. (1975). Collision rates of small particles in a vigorously turbulent fluid. *Chemical Engineering Science, 30*, 1371-1379.

Anderson, J. D. (1995). *Computational Fluid Dynamics: The Basics With Applications.* McGraw-Hill Book Co.

Ding, L. P. (2009). Investigation of Bituminous Coal Hydrophobicity and its Influence on Flotation. *Energy Fuels, 23*, 5536-5543.

Do, H. (2010). *Development of a Turbulent Flotation Model from First Principles.* PhD Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA.

Ekmekci, Z., Bradshaw, D. J., Allison, S. A., & Harris, P. J. (2003). Effects of frother type and froth height on the flotation behaviour of chromite in UG2 ore. *Minerals Engineering, 16*, 941-949.

Fichera, M. A., & Chudacek, M. W. (1992). Batch Cell Flotation Models - A Review. *Minerals Engineering, 5*, 41-55.

Finch, J. A. (1994). Column Flotation: A Selected Review - Part IV: Novel Flotation Devices. *Minerals Engineering, 8*, 587-602.

Finch, J. A., & Dobby, G. S. (1990). *Column Flotation.* Pergamon Press.

Fuerstenau, M. C., Jameson, G. J., & Yoon, R.-H. (2007). *Froth Flotation: A Century of Innovation.* Society for Mining, Metallurgy, and Exploration, Inc.

Galvin, K. P., Nicol, S. K., & Waters, A. G. (1992). Selective ion flotation of gold. *Colloids and Surfaces, 64*, 21-23.

Goren, S. L., & O'Neill, M. E. (1971). On the hydrodynamic resistance to a particle of a dilute suspension when in the neighbourhood of a large obstacle. *Chemical Engineering Science, 26*, 325-338.

Hanumanth, G. S., & Williams, D. J. (1990). An Experimental Study of the Effects of Froth Height on Flotation of China Clay. *Powder Technology, 60*, 131-144.

Harris, M. C., Runge, K. C., Whiten, W. J., & Morrison, R. D. (2002). JKSimFloat as a practical tool for flotation process design and optimization. *Proceedings of the SME Mineral Processing Plant Design, Practice and Control Conference* (pp. 461-478). Vancouver: SME.

Hogg, R., Healy, T. W., & Fuerstenau, D. W. (1966). Mutual coagulation of colloidal dispersions. *Transactions of Faraday Society, 62,* 1638-1651.

JKTech Pty Ltd. (2010). *JKSimFloat*. Retrieved August 2010, from http://www.jktech.com.au/Products_Services/Software-Products/JKSimFloat/index.htm

Keles, S. (2010). *Fine Coal Dewatering Using Hyperbaric Centrifugation.* PhD Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA.

Kelley, K., Do, H., Keles, S., Luttrell, G., & Yoon, R.-H. (2011). Development of a Model-Based Computer Simulator for Flotation. *Submitted to Proceedings for 2011 SME Annual Meeting & Exhibit.* Denver, CO.

Koh, P. T., & Schwarz, M. P. (2003). CFD modelling of bubble–particle collision rates. *Minerals Engineering, 16,* 1055-1059.

Koh, P. T., & Schwarz, M. P. (2006). CFD modelling of bubble–particle attachments in flotation cells. *Minerals Engineering, 19,* 619–626.

Laplante, A. R., Toguri, J. M., & Smith, H. W. (1983). The effect of air flow rate on the kinetics of flotation. Part 2: The transfer of material from the froth over the cell lip. *International Journal of Mineral Processing, 11*, 221-234.

Lastra, R. (2007). Seven practical application cases of liberation analysis. *International Journal of Mineral Processing, 84*, 337-347.

Lee, C. A., & Erickson, L. E. (1987). Bubble breakup and coalescence in turbulent gas-liquid. *Chemical Engineering Communications, 59*, 65-84.

Luttrell, G. H., & Yoon, R.-H. (1992). A Hydrodynamic Model for Bubble-Particle Attachment. *Journal of Colloid and Interface Science, 154*, 129-137.

Malvern Instruments Ltd. (2004). Zetasizer Nano Series User Manual. England.

Metso Minerals. (n.d.). *USIM PAC simulator*. Retrieved August 2010, from http://www.metso.com/miningandconstruction/mm_proc.nsf/WebWID/WTB-041208-2256F-99C61

Mineral Technologies. (n.d.). *MODSIM*. Retrieved August 2010, from http://www.mineraltech.com/MODSIM/

Mitchell, T. K., Nguyen, A. V., & Evans, G. M. (2005). Heterocoagulation of chalcopyrite and pyrite minerals in flotation separation. *Advances in Colloid Interface Science, 114-115*, 227-237.

Pazhianur, R., & Yoon, R.-H. (2003). Model for the origin of hydrophobic force. *Minerals and Metallurgical Processing, 20*, 178-184.

Rabinovich, Y., & Churaev, N. V. (1979). Effect of electromagnetic delay on the forces of molecular attraction. *Kolloidnyi Zhurnal,41*, 468-474.

Rubinstein, J. B., & Melik-Gaikazyan, V. I. (1998). Characterization of Flotation Froth. In J. S. Laskowski, & E. T. Woodburn, *Frothing in Flotation II* (pp. 145-177). CRC Press.

Rubio, J., Souza, M. L., & Smith, R. W. (2002). Overview of flotation as a wastewater treatment technique. *Minerals Engineering, 15*, 139-155.

Samoila, B. L., & Marcu, M. D. (2010). Contributions in Mineral Flotation Modeling and Simulation. In K. Elleithy, *Advanced Techniques in Computing Sciences and Software Engineering* (pp. 289-294).

Sanada, T., Watanabe, M., & Fukano, T. (2005). Effects ofviscosity on coalescence ofa bubble upon impact with a free surface. *Chemical Engineering Science, 60*, 5372-5384.

Savassi, O. N. (2006). Estimating the recovery of size-liberation classes in industrial flotation cells: A simple technique for minimizing the propagation of the experimental error. *International Journal of Mineral Processing, 78*, 85-92.

Schubert, H. (1999). On the turbulence-controlled microprocesses in flotation machines. *International Journal of Mineral Processing, 56*, 257-276.

Schulze, H. J. (1984). *Physico-chemical Elementary Processes in Flotation.* New York: Elsevier.

Sis, H., & Chander, S. (2003). Reagents used in the flotation of phosphate ores: a critical review. *Minerals Engineering, 16*, 577-585.

Subrahmanyam, T. V., & Forssberg, K. S. (1995). Grinding and Flotation Pulp Chemistry of a Low Grade Copper Ore. *Minerals Engineering, 8*, 913-921.

Sutherland, D. N. (1989). Batch Flotation Behaviour of Composite Particles. *Minerals Engineering, 2*, 351-367.

Villeneuve, J., Guillaneau, J. C., & Durance, M.-V. (1995). Flotation Modelling: A Wide Range of Solutions for Solving Industrial Problems. *Minerals Engineering, 8*, 409-420.

Wang, L., & Yoon, R.-H. (2007). Effects of surface forces and film elasticity on foam stability. *International Journal of Mineral Processing,85*, 101-110.

Weaire, D., & Hutzler, S. (1999). *The Physics of Foams.* New York: Oxford University Press.

Weber, M., & Paddock, D. (1983). Interceptional and gravitational collision efficiencies for single collectors at intermediate Reynolds numbers. *Journal of Colloid and Interface*, 328-335.

Wills, B. A., & Napier-Munn, T. (2006). *Wills' Mineral Processing Technology, Seventh Edition: An Introduction to the Practical Aspects of Ore Treatment and Mineral Recovery.* Butterworth-Heinemann.

Yang, X.-S., & Aldrich, C. (2006). Effects of impeller speed and aeration rate on flotation performance of sulphide ore. *Transactions of Nonferrous Metals Society of China, 16*, 185-190.

Yoon, R.-H., & Mao, L. (1996). Application of Extended DLVO Theory, IV: Derivation of Flotation Rate Equation from First Principles. *Journal of Colloid and Interface Science, 181*, 613-626.

Yoon, R.-H., Flinn, D. H., & Rabinovich, Y. I. (1997). Hydrophobic Interactions between Dissimilar Surfaces. *Journal of Colloid and Interface Science, 185*, 363-370.

Appendix A

```
Imports System.Math 'imports math class to use math functions without qualification
Imports System.Windows.Forms.DataVisualization.Charting

Public Class MainWindow

Const pi As Double = 3.141592
Const waterDensity As Double = 1000 'kg/m³
Const airDensity As Double = 1.2 'kg/m³
Const waterViscosity As Double = 0.001 'Ns/m²
Const c As Double = 2.988 * 10 ^ 8 'm/s... speed of light
Const gravity As Double = 9.813 'm/s²

Dim dblSpPower, dblSGasRate As Double
Dim dblBubbleDiam, dblParticleDiam, dblImpellerDiam As Double
Dim dblParticleDens, dblTotalDens, dblFeedGrade As Double
Dim dblCellDiam, dblCellHeight, dblNumCells, dblRetTime As Double
Dim dblSurfaceTension, dblContactAngle, dblFrothHeight As Double
Dim dblParticleZ, dblBubbleZ, dblDielectric, dblPermitivity As Double
Dim dblVolCell, dblAirFraction, dblSlurryFraction As Double
Dim dblEnergyBarrier, dblFrotherConc As Double
Dim dblRateConst, dblRecovery As Double
Dim dblDragBeta As Double 'drag coefficient (Goren & O'Niell)
Dim dblH_c_Factor As Double = 5 'adjustable fitting parameter for dragbeta (higher floats smaller particles)
Dim dblGrowthFactor As Double 'ratio of bubble size in froth
Dim dblR_Water_max, dblR_Water_avg As Double 'max. water rec. in froth (approximation)
Dim ri As Integer = 1 'counter for recovery plot marker color
Dim g As Integer 'counter for grade
Dim dblOvrRecovery As Double
Dim blnCheck As Boolean 'are all inputs valid?
Public arrRecovery(37), arrPDiam(37), arrRateK(37) As Double 'storage arrays
Public arrPa(37), arrPc(37), arrPd(37), arrFR(37) As Double
Public arrGrMineral(37), arrGrMiddling(37), arrGrGangue(37) As Double 'store recoveries
Public arrSizeGrade(37) As Double




Private Sub CalcButton_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CalcButton.Click

If Button_FeedGrade.Enabled = True Then
Call FeedGradeCalc()
Else
Call MainCalculation()
End If

End Sub
Private Sub Button_Clear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button_Clear.Click

Dim i As Integer = 1
Dim xx As Integer
```

```
For i = 1 To (ri - 1)
ChartK.Series(i).Points.Clear()
Chart_Rec2.Series(i).Points.Clear()
Chart_RecLinear.Series(i).Points.Clear()
Chart_Grade.Series(i).Points.Clear()
Chart_Pa.Series(i).Points.Clear()
Chart_Pc.Series(i).Points.Clear()
Chart_Pd.Series(i).Points.Clear()
Chart_FRec.Series(i).Points.Clear()
Next i

End Sub
Private Sub Button_ContactDistrib_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button_ContactDistrib.Click
ContactDist.ShowDialog()
End Sub
Private Sub Button_FeedGrade_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button_FeedGrade.Click
FeedGrade.ShowDialog()
End Sub
Private Sub Button_SizeDist_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button_SizeDist.Click
SizeDist.ShowDialog()
End Sub


Public Function AddToolTip()

'this function adds a tooltip to each series on each graph
Dim myToolTip As String
myToolTip = "Specific Power = " & dblSpPower & ControlChars.NewLine & _
"Gas Rate = " & dblSGasRate & ControlChars.NewLine & _
"Bubble Diameter = " & Format(dblBubbleDiam, "#.####") & ControlChars.NewLine & _
"Particle S.G. = " & dblParticleDens / 1000 & ControlChars.NewLine & _
"Air Fraction = " & dblAirFraction & ControlChars.NewLine & _
"Slurry Fraction = " & dblSlurryFraction & ControlChars.NewLine & _
"Frother = " & ComboBox_Frother.Text & ControlChars.NewLine & _
"Frother Concentration = " & dblFrotherConc & ControlChars.NewLine & _
"Contact Angle = " & dblContactAngle & ControlChars.NewLine & _
"Dielectric Constant = " & dblDielectric & ControlChars.NewLine & _
"Particle Zeta Potential = " & dblParticleZ & ControlChars.NewLine & _
"Bubble Zeta Potential = " & dblBubbleZ & ControlChars.NewLine & _
"Permitivity = " & dblPermitivity & ControlChars.NewLine & _
"Cell Diameter = " & dblCellDiam & ControlChars.NewLine & _
"Cell Height = " & dblCellHeight & ControlChars.NewLine & _
"Impeller Diameter = " & dblImpellerDiam & ControlChars.NewLine & _
"Number of Cells = " & dblNumCells & ControlChars.NewLine & _
"Retention Time = " & dblRetTime & ControlChars.NewLine & _
"Froth Height = " & dblFrothHeight & ControlChars.NewLine & _
"Growth Factor = " & dblGrowthFactor & ControlChars.NewLine & _
"Max. Water Rec. = " & Format(dblR_Water_max, "#.####")

```

```
Chart_Rec2.Series(ri).ToolTip = myToolTip
Chart_RecLinear.Series(ri).ToolTip = myToolTip
ChartK.Series(ri).ToolTip = myToolTip
Chart_Pa.Series(ri).ToolTip = myToolTip
Chart_Pc.Series(ri).ToolTip = myToolTip
Chart_Pd.Series(ri).ToolTip = myToolTip
Chart_FRec.Series(ri).ToolTip = myToolTip

End Function
Public Function CheckInputs()
blnCheck = False

'===== Check for all Inputs, then Output =====
If TextBox_SpecificPower.Text = "" Then
MsgBox("Please enter a value for Specific Power")
ElseIf TextBox_SpecificAir.Text = "" Then
MsgBox("Please enter a value for Superficial Gas Rate")
ElseIf TextBox_AirFraction.Text = "" Then
MsgBox("Please enter a value for Air Fraction")
ElseIf Val(TextBox_AirFraction.Text) > 0.6 Then
MsgBox("Slurry Air cannot be greater than 0.6")
ElseIf TextBox_SlurryFraction.Text = "" Then
MsgBox("Please enter a value for Slurry Fraction")
ElseIf Val(TextBox_SlurryFraction.Text) > 0.5 Then
MsgBox("Slurry Fraction cannot be greater than 0.5")
ElseIf TextBox_FrotherConc.Text = "" And ComboBox_Frother.Text <> "" Then
MsgBox("Please enter a value for Frother Concentration")
ElseIf TextBox_DielectricConst.Text = "" Then
MsgBox("Please enter a value for Dielectric Constant")
ElseIf TextBox_BubbleZ.Text = "" Then
MsgBox("Please enter a value for Bubble Zeta Potential")
ElseIf TextBox_ParticleZ.Text = "" Then
MsgBox("Please enter a value for Particle Zeta Potential")
ElseIf TextBox_Permitivity.Text = "" Then
MsgBox("Please enter a value for Permitivity")
ElseIf TextBox_CellHeight.Text = "" Then
MsgBox("Please enter a value for Cell Height")
ElseIf TextBox_CellDiameter.Text = "" Then
MsgBox("Please enter a value for Cell Diameter")
ElseIf TextBox_NumCells.Text = "" Then
MsgBox("Please enter a value for Number of Cells")
ElseIf TextBox_RetentionTime.Text = "" Then
MsgBox("Please enter a value for Retention Time")
ElseIf TextBox_FrothHeight.Text = "" Then
MsgBox("Please enter a value for Froth Height")
ElseIf Button_FeedGrade.Enabled = False Then 'if using feedgrade dont need following inputs
If TextBox_ParticleDensity.Text = "" Then
MsgBox("Please enter a value for Particle Specific Gravity")
ElseIf TextBox_ContactAngle.Text = "" Then
MsgBox("Please enter a value for Contact Angle")
ElseIf Val(TextBox_ContactAngle.Text) > 88.7 Then
MsgBox("Contact Angle must be 88.7 or less") 'cant have negative cosine
ElseIf Val(TextBox_ParticleDensity.Text) <= 1 Then
MsgBox("Particle Specific Gravity Must be Greater than 1") 'SG must be greater than 1 otherwise it floats itself
Else
blnCheck = True
End If
```

```
ElseIf Button_FeedGrade.Enabled = True Then
If FeedGrade.TextBox_MineralDensity.Text = "" Then
MsgBox("Please enter a value for Mineral Specific Gravity")
ElseIf FeedGrade.TextBox_MiddlingsDensity.Text = "" Then
MsgBox("Please enter a value for Middlings Specific Gravity")
ElseIf FeedGrade.TextBox_GangueDensity.Text = "" Then
MsgBox("Please enter a value for Gangue Specific Gravity")
ElseIf FeedGrade.TextBox_MineralContactAngle.Text = "" Then
MsgBox("Please enter a value for Mineral Contact Angle")
ElseIf FeedGrade.TextBox_MiddlingsContactAngle.Text = "" Then
MsgBox("Please enter a value for Middlings Contact Angle")
ElseIf FeedGrade.TextBox_GangueContactAngle.Text = "" Then
MsgBox("Please enter a value for Gangue Contact Angle")
ElseIf Val(FeedGrade.TextBox_MineralContactAngle.Text) >= 90 Then
MsgBox("Contact Angle must be less than 90") 'cant have negative
cosine
ElseIf Val(FeedGrade.TextBox_MineralDensity.Text) <= 1 Then
MsgBox("Ore Specific Gravity Must be Greater than 1") 'SG must be
greater than 1 otherwise it floats itself
Else
blnCheck = True
End If
End If

End Function
Public Function ContactAngleDist()

If CheckBox_ContactDistrib.Checked = True Then
If dblParticleDiam < 0.000015 Then
dblContactAngle = Val(ContactDist.TextBox_CD15.Text)
ElseIf dblParticleDiam < 0.000025 Then
dblContactAngle = Val(ContactDist.TextBox_CD25.Text) - ((Val
(ContactDist.TextBox_CD25.Text) - Val(ContactDist.TextBox_CD15.Text)) _
* ((0.000025 - dblParticleDiam) / 0.00001))
'interpolates contact angle between entered values

ElseIf dblParticleDiam < 0.000038 Then
dblContactAngle = Val(ContactDist.TextBox_CD38.Text) - ((Val
(ContactDist.TextBox_CD38.Text) - Val(ContactDist.TextBox_CD25.Text)) _
* ((0.000038 - dblParticleDiam) / 0.000013))

ElseIf dblParticleDiam < 0.000045 Then
dblContactAngle = Val(ContactDist.TextBox_CD45.Text) - ((Val
(ContactDist.TextBox_CD45.Text) - Val(ContactDist.TextBox_CD38.Text)) _
* ((0.000045 - dblParticleDiam) / 0.000007))

ElseIf dblParticleDiam < 0.000075 Then
dblContactAngle = Val(ContactDist.TextBox_CD75.Text) - ((Val
(ContactDist.TextBox_CD75.Text) - Val(ContactDist.TextBox_CD45.Text)) _
* ((0.000075 - dblParticleDiam) / 0.00003))

ElseIf dblParticleDiam < 0.000106 Then
dblContactAngle = Val(ContactDist.TextBox_CD106.Text) - ((Val
(ContactDist.TextBox_CD106.Text) - Val(ContactDist.TextBox_CD75.Text)) _
* ((0.000106 - dblParticleDiam) / 0.000031))

ElseIf dblParticleDiam < 0.00015 Then
dblContactAngle = Val(ContactDist.TextBox_CD150.Text) - ((Val
(ContactDist.TextBox_CD150.Text) - Val(ContactDist.TextBox_CD106.Text)) _
* ((0.00015 - dblParticleDiam) / 0.000044))
```

```

ElseIf dblParticleDiam < 0.00018 Then
dblContactAngle = Val(ContactDist.TextBox_CD180.Text) - ((Val(ContactDist.TextBox_CD180.Text) - Val(ContactDist.TextBox_CD150.Text)) _
* ((0.00018 - dblParticleDiam) / 0.00003))

ElseIf dblParticleDiam < 0.00025 Then
dblContactAngle = Val(ContactDist.TextBox_CD250.Text) - ((Val(ContactDist.TextBox_CD250.Text) - Val(ContactDist.TextBox_CD180.Text)) _
* ((0.00025 - dblParticleDiam) / 0.00007))

ElseIf dblParticleDiam < 0.000425 Then
dblContactAngle = Val(ContactDist.TextBox_CD425.Text) - ((Val(ContactDist.TextBox_CD425.Text) - Val(ContactDist.TextBox_CD250.Text)) _
* ((0.000425 - dblParticleDiam) / 0.000175))

Else
dblContactAngle = 0
End If

If dblContactAngle >= 88.7 Then
dblContactAngle = 88.7
End If
End If

End Function
Public Function EnergyBarrier()

Dim dblA132_s, dblK132_s, dblK131, a, b_k As Double
Dim dblA11 As Double = 3 * 10 ^ -19 'Hamaker Constants
Dim dblA22 As Double = 0 '
Dim dblA33 As Double = 4.38 * 10 ^ -20 '
Dim dblK232 As Double = 4.07 * 10 ^ -18 'Hydrophobic force constant
Dim dblKappa As Double = 1 / (9.6 * 10 ^ -8) 'inverse Debye Length
Dim dblVT, dblVE, dblVD, dblVH As Double 'total free energy of interaction, electrostatic, van-der waals, hydrophobic force
Dim dblVT1, dblVE1, dblVD1, dblVH1 As Double 'second state of above variables
Dim b As Double = 3 * 10 ^ -17 'correct for retardation effects (most mat'ls)
Dim l As Double = 3.3 * 10 ^ 15 'correction for retardation effects (water, !!change for other media)
Dim dblH0 As Double = 1 * 10 ^ -11 'separation between bubb & part
Dim dblH1 As Double 'for incrementing
Dim dblH As Double = 0
Dim Ce, Cd, Ch As Double 'coefficients of eq 14,15,16 - for efficiency of calculation

'===== Get Textbox Value =====
dblDielectric = Val(TextBox_DielectricConst.Text)
dblPermitivity = Val(TextBox_Permitivity.Text) * 10 ^ -12 '10^12 for right units

'===== Calc Barrier =====
If dblContactAngle < 86.89 Then
a = 2.732 * 10 ^ -21
b_k = 0.04136
ElseIf 86.889 <= dblContactAngle < 92.28 Then
a = 4.888 * 10 ^ -44
b_k = 0.6441
```

```
Else
a = 6.327 * 10 ^ -27
b_k = 0.2172
End If

dblK131 = a * Exp(b_k * dblContactAngle)
dblK132_s = Sqrt(dblK131 * dblK232)
dblA132_s = (Sqrt(dblA11) - Sqrt(dblA33)) * (Sqrt(dblA22) - Sqrt(dblA33))


Ce = pi * (dblPermitivity * dblDielectric * (dblParticleDiam * dblBubbleDiam
_
/ 4) * (dblParticleZ ^ 2 + dblBubbleZ ^ 2)) / (dblParticleDiam / 2 _
+ dblBubbleDiam / 2)
Cd = -(dblA132_s * (dblParticleDiam * dblBubbleDiam / 4)) _
/ (6 * (dblParticleDiam / 2 + dblBubbleDiam / 2))
Ch = -(dblParticleDiam * dblBubbleDiam / 4 * dblK132_s) _
/ (6 * (dblParticleDiam / 2 + dblBubbleDiam / 2))


Dim x As Integer = 0
While x = 0
'equation 14 using H0(Do & Yoon)
dblVE = Ce * (((2 * dblParticleZ * dblBubbleZ) / (dblParticleZ ^ 2 + _
dblBubbleZ ^ 2)) * Log((1 + Exp(-dblKappa * dblH0)) / _
(1 - Exp(-dblKappa * dblH0))) + Log(1 - _
Exp(-2 * dblKappa * dblH0)))

dblVD = Cd / dblH0 * (1 - ((1 + 2 * b * l) / (1 + b * c / dblH0)))
'equation 15 using H0(Do & Yoon)
dblVH = Ch / dblH0 'equation 16 using H0(Do & Yoon)
dblVT = dblVE + dblVD + dblVH 'extended DLVO theory


dblH1 = dblH0 + 1 * 10 ^ -11

'equation 14 using H1(Do & Yoon)
dblVE1 = Ce * (((2 * dblParticleZ * dblBubbleZ) / (dblParticleZ ^ 2 + _
dblBubbleZ ^ 2)) * Log((1 + Exp(-dblKappa * dblH1)) / _
(1 - Exp(-dblKappa * dblH1))) + Log(1 - _
Exp(-2 * dblKappa * dblH1)))

dblVD1 = Cd / dblH1 * (1 - ((1 + 2 * b * l) / (1 + b * c / dblH1)))
'equation 15 using H1(Do & Yoon)
dblVH1 = Ch / dblH1 'equation 16 using H1(Do &
Yoon)
dblVT1 = dblVE1 + dblVD1 + dblVH1 'extended DLVO theory 2nd state

If dblVT > dblVT1 Then
dblH = dblH0
dblDragBeta = 0.37 * (dblParticleDiam / 2 / dblH / dblH_c_Factor) ^
0.83 'h_c_factor is adjustable
dblEnergyBarrier = dblVT
x = 1
Else
dblH0 = dblH1 + 1 * 10 ^ -9 'increments H0 to find correct value
End If

End While

```

```
End Function
Public Function FeedGradeCalc() 'used when user inputs multiple feeds

Dim arrContactAngles(3), arrFeedFractions(3) As Double
Dim arrDensities(3), arrGrades(3) As Double
Dim dblFGRecovery As Double = 0
Dim x As Integer 'counter for 3 feeds
Dim dblMidRec, dblMinRec, dblGangRec As Double

arrContactAngles(1) = Val(FeedGrade.TextBox_MineralContactAngle.Text)
arrContactAngles(2) = Val(FeedGrade.TextBox_MiddlingsContactAngle.Text)
arrContactAngles(3) = Val(FeedGrade.TextBox_GangueContactAngle.Text)
arrFeedFractions(1) = Val(FeedGrade.TextBox_MineralFraction.Text)
arrFeedFractions(2) = Val(FeedGrade.TextBox_MiddlingsFraction.Text)
arrFeedFractions(3) = Val(FeedGrade.TextBox_GangueFraction.Text)
arrDensities(1) = Val(FeedGrade.TextBox_MineralDensity.Text)
arrDensities(2) = Val(FeedGrade.TextBox_MiddlingsDensity.Text)
arrDensities(3) = Val(FeedGrade.TextBox_GangueDensity.Text)
arrGrades(1) = Val(FeedGrade.TextBox_MineralGrade.Text)
arrGrades(2) = Val(FeedGrade.TextBox_MiddlingsGrade.Text)
arrGrades(3) = Val(FeedGrade.TextBox_GangueGrade.Text)

For x = 1 To 3
dblContactAngle = arrContactAngles(x)
dblParticleDens = arrDensities(x) * 1000 'x1000 for kg/m^3


'====begin similar code as maincalc====

Dim dblVolBubble, dblVolParticle, dblVolBP As Double
Dim dblMassBubble, dblMassParticle, dblMassBP, dblMassTotal As Double
Dim dblCollisionDiam, dblNumAttached As Double
Dim dblKinVisc As Double 'kinematic viscosity
Dim dblBulkZone, dblImpellerZone, dblDetach_F As Double '2 compartment model (Lu)
Dim dblVolImpZone As Double = 0.1 'set impeller zone 1/10
Dim dblEMean, dblEBulk, dblEImpeller As Double 'energy dissipations
Dim dblU1Bulk, dblU2Bulk, dblU1Mean, dblU2Mean As Double
Dim dblBeta, dblNParticle, dblNBubble, dblZBubbParticle As Double
Dim dblWorkAdhesion, dblKineticEAttach, dblKineticEDetach As Double
Dim gammaMIBC, gammaPPG400, gammaOctanol, gammaPentanol As Double
Dim kMIBC, kPPG400, kOctanol, kPentanol As Double
Dim dblPAtt, dblPDet, dblPCol, dblRe As Double 'probabilities

Dim arrBRecovery(37), arrBPDiam(37), arrBRateK(37) As Double 'storage arrays for bubble dist
Dim arrBPa(37), arrBPc(37), arrBPd(37), arrBFR(37) As Double

dblBulkZone = 0.5
dblImpellerZone = 15
dblDetach_F = 1 'adjustable parameter for fitting

'===== Froth Parameters =====
Dim dblR_Entrainment, dblR_Attachment, dblFrothRecoveryFactor As Double 'entrainment and attachment recoveries
Dim dblB As Double = 3.3 'fitting parameter
```

```
Dim dblAlpha As Double = 0.01 'fitting parameter
Dim dblCoverage As Double = 0.5 'max particle coverage attached in froth
Dim dblPFTransfer, dblP_i, dblPr As Double
Dim dblEiw, dblEka As Double
Dim dblRmax As Double
'!! see global declarations for more froth parameters !!


'===== Get Textbox Values =====
dblSpPower = Val(TextBox_SpecificPower.Text) * 1000 'x1000 for w/m^3
dblSGasRate = Val(TextBox_SpecificAir.Text) / 100 '/100 for m/s
'dblParticleDens = Val(TextBox_ParticleDensity.Text) * 1000 'x1000 for kg/m^3
dblAirFraction = Val(TextBox_AirFraction.Text)
dblSlurryFraction = Val(TextBox_SlurryFraction.Text)
dblImpellerDiam = Val(TextBox_ImpellerDiameter.Text)
'dblContactAngle = Val(TextBox_ContactAngle.Text)
dblParticleZ = Val(TextBox_ParticleZ.Text)
dblBubbleZ = Val(TextBox_BubbleZ.Text)
dblCellHeight = Val(TextBox_CellHeight.Text)
dblCellDiam = Val(TextBox_CellDiameter.Text)
dblNumCells = Val(TextBox_NumCells.Text)
dblRetTime = Val(TextBox_RetentionTime.Text)
dblFrothHeight = Val(TextBox_FrothHeight.Text)
dblFrotherConc = Val(TextBox_FrotherConc.Text)

gammaMIBC = 0.000005 'mol/m^2
gammaPPG400 = 0.000001 'mol/m^2
gammaOctanol = 0.000008 'mol/m^2
gammaPentanol = 0.000006 'mol/m^2
kMIBC = 230 'M^-1
kPPG400 = 1700000 'M^-1
kOctanol = 2200 'M^-1
kPentanol = 55 'M^-1

If ComboBox_Frother.Text = "MIBC" Then
dblFrotherConc = dblFrotherConc / 102170 'convert ppm to mol/L
dblSurfaceTension = 0.07243 - 8.314 * (273.15 + 23) * gammaMIBC * Log(kMIBC * dblFrotherConc + 1)
ElseIf ComboBox_Frother.Text = "PPG 400" Then
dblFrotherConc = dblFrotherConc / 134170 'convert ppm to mol/L
dblSurfaceTension = 0.07243 - 8.314 * (273.15 + 23) * gammaPPG400 * Log(kPPG400 * dblFrotherConc + 1)
ElseIf ComboBox_Frother.Text = "Octanol" Then
dblFrotherConc = dblFrotherConc / 130230 'convert ppm to mol/L
dblSurfaceTension = 0.07243 - 8.314 * (273.15 + 23) * gammaOctanol * Log(kOctanol * dblFrotherConc + 1)
ElseIf ComboBox_Frother.Text = "Pentanol" Then
dblFrotherConc = dblFrotherConc / 88150 'convert ppm to mol/L
dblSurfaceTension = 0.07243 - 8.314 * (273.15 + 23) * gammaPentanol * Log(kPentanol * dblFrotherConc + 1)
Else
dblSurfaceTension = 0.07243 'pure water @ 23°C
End If

TextBox_SurfaceTension.Text = Format(dblSurfaceTension * 1000, "##.##")

Call CheckInputs() 'call function check for input values
If blnCheck = False Then
```

```
Exit Function 'if inputs missing, exit sub
End If

'===== BEGIN MULTIPARTICLE LOOP =====
dblParticleDiam = 0.000001 '(1 micron)

Dim i As Integer
Dim dblAddRec, dblAddK, dblAddPa, dblAddPc, dblAddPd, dblAddFR As Double

For i = 0 To 37 'particle loop


dblAddRec = 0
dblAddK = 0
dblAddPa = 0
dblAddPc = 0
dblAddPd = 0
dblAddFR = 0
dblR_Water_avg = 0


'===== Energy Dissipation =====

dblTotalDens = dblAirFraction * airDensity + _
(1 - dblAirFraction) * dblSlurryFraction * dblParticleDens _
+ (1 - dblSlurryFraction) * waterDensity
dblEMean = dblSpPower / dblTotalDens
dblEBulk = dblBulkZone * dblEMean
dblEImpeller = dblImpellerZone * dblEMean

If TextBox_BubbleSize.Enabled = False Then
dblBubbleDiam = (2.11 * dblSurfaceTension / (waterDensity * dblEImpeller ^ 0.66)) ^ 0.6
Else
dblBubbleDiam = Val(TextBox_BubbleSize.Text) / 1000
End If


dblNumAttached = dblCoverage * 4 * (dblBubbleDiam / dblParticleDiam) ^ 2 'num of particles attached to one bubble


'===== Cell Calculations =====
dblCollisionDiam = dblParticleDiam + dblBubbleDiam 'avg diam of collision
dblVolParticle = (4 / 3) * pi * (dblParticleDiam / 2) ^ 3 'vol 1 part.
dblVolBubble = (4 / 3) * pi * (dblBubbleDiam / 2) ^ 3 'vol 1 bubb.
dblVolBP = dblVolBubble + dblVolParticle 'vol of 1 BP aggregate
dblVolCell = pi * (dblCellDiam / 2) ^ 2 * dblCellHeight
dblKinVisc = waterViscosity / waterDensity
dblMassParticle = dblParticleDens * dblVolParticle 'mass 1 part.
dblMassBubble = airDensity * dblVolBubble 'mass 1 bubb.
dblMassBP = dblMassBubble + dblMassParticle 'mass of 1 BP aggregate
dblMassTotal = dblVolCell * dblTotalDens
```

```


'===== Velocities by Dissipation =====
dblU1Bulk = (0.4 * (dblEBulk ^ (4 / 9)) * (dblParticleDiam ^ (7 /
9)) _
* (dblKinVisc ^ (-1 / 3)) * (dblParticleDens / _
waterDensity - 1) ^ (2 / 3)) ^ 2 'for
attachment
dblU2Bulk = 2 * (dblEBulk * dblBubbleDiam) ^ (2 / 3)
dblU1Mean = (0.4 * (dblEMean ^ (4 / 9)) * (dblParticleDiam ^ (7 /
9)) _
* (dblKinVisc ^ (-1 / 3)) * (dblParticleDens / _
waterDensity - 1) ^ (2 / 3)) ^ 2
dblU2Mean = 2 * (dblEMean * dblBubbleDiam) ^ (2 / 3)

dblBeta = (2 ^ (3 / 2)) * (pi ^ 0.5) * (dblCollisionDiam ^ 2) * _
Sqrt(dblU1Bulk + dblU2Bulk) 'from Abrahamson model
using bulk dissipation

'===== Calc # Density of Bubbles =====
dblNBubble = dblAirFraction / dblVolBubble
dblNParticle = (1 - dblAirFraction) * dblSlurryFraction /
dblVolParticle
dblZBubbParticle = dblBeta * dblNBubble * dblNParticle

dblWorkAdhesion = dblSurfaceTension * pi * (dblParticleDiam / 2) ^ 2
_
* (1 - Cos(dblContactAngle * (pi / 180)) ^ 2)
'calc work of adhesion for 1 particle

'===== Energy Barrier =====
Call EnergyBarrier() 'calls function to calc energy barrier

If dblEnergyBarrier <= 0 Then
dblEnergyBarrier = 0
End If

'===== Kinetic Energy of Attachment =====
dblKineticEAttach = 0.5 * dblMassParticle * dblU1Bulk / (dblDragBeta
^ 2)
dblKineticEDetach = 0.5 * dblMassParticle * (dblDetach_F *
(dblParticleDiam _
+ dblBubbleDiam) * Sqrt(dblEImpeller /
dblKinVisc)) ^ 2

'===== Probabilities =====
dblPAtt = Exp(-dblEnergyBarrier / dblKineticEAttach) 'prob. of
attachment
dblPDet = Exp(-(dblWorkAdhesion + dblKineticEAttach) _
/ dblKineticEDetach) 'prob. of
detachment
dblRe = Sqrt(dblU2Bulk) * dblBubbleDiam / dblKinVisc 'bubble
Reynold's number

dblPCol = Tanh(Sqrt(3 / 2 * (1 + (3 / 16 * dblRe) / (1 + 0.249 *
dblRe ^ 0.56))) _
* (dblParticleDiam / dblBubbleDiam)) ^ 2 'prob.
collision, modified Luttrell and Yoon

If dblPCol >= 1 Then
```

```
dblPCol = 1
End If

dblEiw = gravity / (4 * pi) * (waterViscosity ^ 3 / dblEBulk) ^ 0.25
dblEka = (dblMassBubble * dblU2Bulk - 2 * (dblBubbleDiam / dblParticleDiam) ^ 2 * dblMassParticle * dblU1Bulk) ^ 2 _
/ (100 * (dblMassBubble + 2 * (dblBubbleDiam / dblParticleDiam) ^ 2 * dblMassParticle))

dblP_i = 13 * Sqrt((9 * waterViscosity ^ 2) / (dblBubbleDiam * dblSurfaceTension * dblTotalDens))
dblPr = Exp(-dblEiw / dblEka)
dblPFTransfer = dblP_i * (1 - dblPr)


'===== Froth Recovery =====
'''===start new froth rec model
Dim dblCoverageFactor = 2
Dim dblAf As Double
Dim dblA0 As Double
Dim dblCoarsenTime As Double
Dim dblFilmThick As Double
Dim dblCoalesceFactor As Double = 2
Dim dblL As Double

dblFilmThick = 3 / 4 * (0.33 / (1 - 0.33)) * dblBubbleDiam
dblCoarsenTime = (4 * waterViscosity * dblFrothHeight) / (waterDensity * gravity * dblFilmThick ^ 2)
dblL = waterViscosity / (airDensity * 0.015) '1.5 cm/s froth velocity


dblA0 = (dblBubbleDiam) ^ 2

dblAf = (Sqrt(dblSGasRate * waterViscosity / (waterDensity * gravity)) * Tan(Atan(Sqrt(airDensity * gravity * dblA0 / (dblSGasRate * waterViscosity))) _
- dblFrothHeight / 2 * Sqrt(waterDensity * gravity * dblSGasRate) / dblSurfaceTension)) ^ 2

dblRmax = Sqrt(Exp(dblCoalesceFactor * Sqrt(dblAf / dblA0) - dblCoalesceFactor))

dblR_Attachment = dblRmax * Exp(-dblCoverageFactor * (6 * dblFrothHeight / (dblBubbleDiam)) _
* (1 - dblRmax) * (dblParticleDiam / dblBubbleDiam) ^ 2)


dblR_Water_max = (0.33 * dblCoarsenTime * (6 * dblSGasRate) / (dblBubbleDiam / dblRmax)) * Exp(-dblFrothHeight / dblL)

If dblR_Water_max > 1 Then
dblR_Water_max = 1
End If

dblR_Entrainment = dblR_Water_max * Exp(-0.0325 * (dblParticleDens - waterDensity) - 0.063 * dblParticleDiam)

dblFrothRecoveryFactor = dblR_Entrainment + dblR_Attachment
```

```


'===== Rate Constant =====
Dim dblRecovery_ci As Double
Dim dblRecovery_I As Double


dblRateConst = dblBeta * dblNBubble * dblPAtt * dblPCol _
* (1 - dblPDet) * 60 'x60 to make 1/min

dblRecovery_ci = 1 - (1 + dblRateConst * dblRetTime) ^ (-1) 'eq
32 (Do & Yoon)

dblRecovery_I = dblRecovery_ci * dblFrothRecoveryFactor /
(dblRecovery_ci * _
dblFrothRecoveryFactor + 1 - dblRecovery_ci)
'eq 6.2 finch & dobby

dblRecovery = 1 - (1 - dblRecovery_I) ^ dblNumCells

''==== grade vs recov ====
'Dim dblGRec_ci, dblGRec_i, dblGRec As Double 'grade v rec
variables
'Dim dblGradeRet As Double = 0.5
'Dim f As Integer
'For f = 1 To 29
' dblGRec_ci = 1 - (1 + dblRateConst * dblGradeRet) ^ (-1)

' dblGRec_i = dblGRec_ci * dblFrothRecoveryFactor / (dblGRec_ci *
_
' dblFrothRecoveryFactor + 1 - dblGRec_ci)

' dblGRec = 1 - (1 - dblGRec_i) ^ dblNumCells
' arrRecGrade(x, i, f) = dblGRec

' dblGradeRet = dblGradeRet * 2 '~~29 iterations
'Next f


'========!!!!!!!! insert condtional code here for hydrophobic
coagulation and water recovery
'if dblParticleDiam < specified size (~ < 15 micron)
'then multiply by some factors or use equations
' changes recovery for that size fraction

dblAddRec = dblRecovery
dblAddK = dblRateConst
dblAddPa = dblPAtt
dblAddPc = dblPCol
dblAddPd = dblPDet
dblAddFR = dblFrothRecoveryFactor
dblR_Water_avg = dblR_Water_max


'===== Output Results =====

arrRecovery(i) = dblAddRec * 100 '100 for percent
arrPDiam(i) = dblParticleDiam * 1000000 '10^6 for microns
arrRateK(i) = dblAddK * dblAddFR
arrPa(i) = dblAddPa * 100
```

```
arrPc(i) = dblAddPc * 100
arrPd(i) = (1 - dblAddPd) * 100
arrFR(i) = dblAddFR * 100

dblParticleDiam = dblParticleDiam * 1.2 'increment particle diam

'==== store recov for grade calc ====
If x = 1 Then
arrGrMineral(i) = arrRecovery(i)
ElseIf x = 2 Then
arrGrMiddling(i) = arrRecovery(i)
Else
arrGrGangue(i) = arrRecovery(i)
End If


Next i

'===== Graphs =====
ChartK.Series.Add(ri)
ChartK.Series(ri).ChartType = SeriesChartType.Line
Chart_Rec2.Series.Add(ri)
Chart_Rec2.Series(ri).ChartType = SeriesChartType.Line
Chart_RecLinear.Series.Add(ri)
Chart_RecLinear.Series(ri).ChartType = SeriesChartType.Line
Chart_Pa.Series.Add(ri)
Chart_Pa.Series(ri).ChartType = SeriesChartType.Line
Chart_Pc.Series.Add(ri)
Chart_Pc.Series(ri).ChartType = SeriesChartType.Line
Chart_Pd.Series.Add(ri)
Chart_Pd.Series(ri).ChartType = SeriesChartType.Line
Chart_FRec.Series.Add(ri)
Chart_FRec.Series(ri).ChartType = SeriesChartType.Line
Chart_Grade.Series.Add(ri)
Chart_Grade.Series(ri).ChartType = SeriesChartType.Line

Dim n As Integer
For n = 0 To 37
ChartK.Series(ri).Points.AddXY(arrPDiam(n), arrRateK(n))
Chart_Rec2.Series(ri).Points.AddXY(arrPDiam(n), arrRecovery(n))
Chart_RecLinear.Series(ri).Points.AddXY(arrPDiam(n), arrRecovery(n))
Chart_Pa.Series(ri).Points.AddXY(arrPDiam(n), arrPa(n))
Chart_Pc.Series(ri).Points.AddXY(arrPDiam(n), arrPc(n))
Chart_Pd.Series(ri).Points.AddXY(arrPDiam(n), arrPd(n))
Chart_FRec.Series(ri).Points.AddXY(arrPDiam(n), arrFR(n))
Next

Call AddToolTip() 'adds info to the graphs

ri = ri + 1 'increment series number and color

Call SizeDistribution()

dblFGRecovery = dblFGRecovery + dblOvrRecovery * arrFeedFractions(x) / 100

If x = 1 Then
dblMinRec = dblOvrRecovery
ElseIf x = 2 Then
dblMidRec = dblOvrRecovery
```

```
Else
dblGangRec = dblOvrRecovery
End If

Next x


''==== plot grade v pdiam ====
Dim m As Integer

For m = 0 To 37
arrSizeGrade(m) = (arrGrMineral(m) * arrFeedFractions(1) + arrGrMiddling
(m) * arrFeedFractions(2) * 0.5) / _
(arrGrMineral(m) * arrFeedFractions(1) + arrGrMiddling
(m) * arrFeedFractions(2) + _
arrGrGangue(m) * arrFeedFractions(3)) * 100
Chart_Grade.Series(ri - 1).Points.AddXY(arrPDiam(m), arrSizeGrade(m))
Next


'===== Temp Outputs for Debugging =====
'MsgBox("Some Outputs Temporary for Debugging")
If dblOvrRecovery > 0 Then
Label_RecoveryOutput.Text = Format(dblFGRecovery, "#.###" & " %")
End If

label_VolCellOutput.Text = Format(dblVolCell, "#.###") & " m³"
label_WaterRecOut.Text = Format(dblR_Water_avg * 100, "#.###") & " %"
Dim dblProductGrade, dblOvrMineralRec As Double
dblProductGrade = (dblMinRec * arrGrades(1) * arrFeedFractions(1) +
dblMidRec * arrGrades(2) * arrFeedFractions(2) + dblGangRec * arrGrades(3) *
arrFeedFractions(3)) _
/ (dblMinRec * arrFeedFractions(1) + dblMidRec *
arrFeedFractions(2) + dblGangRec * arrFeedFractions(3))
dblOvrMineralRec = (dblMinRec * arrGrades(1) * arrFeedFractions(1) +
dblMidRec * arrGrades(2) * arrFeedFractions(2) + dblGangRec * arrGrades(3) *
arrFeedFractions(3)) _
/ (arrGrades(1) * arrFeedFractions(1) + arrGrades(2) *
arrFeedFractions(2) + arrGrades(3) * arrFeedFractions(3))

Label_FeedGrade.Text = FeedGrade.TextBox_OvrFeedGrade.Text & " %"
Label_ProductGrade.Text = Format(dblProductGrade, "#.##") & " %"
Label_ProductRecovery.Text = Format(dblOvrMineralRec, "#.##") & " %"
Label_MineralRec.Text = Format(dblMinRec, "#.##") & " %"
Label_MiddlingsRec.Text = Format(dblMidRec, "#.##") & " %"
Label_GangueRec.Text = Format(dblGangRec, "#.##") & " %"
End Function
Public Function MainCalculation() 'used for single component feed, 100% grade

Dim dblVolBubble, dblVolParticle, dblVolBP As Double
Dim dblMassBubble, dblMassParticle, dblMassBP, dblMassTotal As Double
Dim dblCollisionDiam, dblNumAttached As Double
Dim dblKinVisc As Double 'kinematic viscosity
Dim dblBulkZone, dblImpellerZone, dblDetach_F As Double '2 compartment model
(Lu)
Dim dblVolImpZone As Double = 0.1 'set impeller zone 1
/10
Dim dblEMean, dblEBulk, dblEImpeller As Double 'energy dissipations
Dim dblU1Bulk, dblU2Bulk, dblU1Mean, dblU2Mean As Double
Dim dblBeta, dblNParticle, dblNBubble, dblZBubbParticle As Double
```

```
Dim dblWorkAdhesion, dblKineticEAttach, dblKineticEDetach As Double
Dim gammaMIBC, gammaPPG400, gammaOctanol, gammaPentanol As Double
Dim kMIBC, kPPG400, kOctanol, kPentanol As Double
Dim dblPAtt, dblPDet, dblPCol, dblRe As Double 'probabilities


dblBulkZone = 0.5
dblImpellerZone = 15
dblDetach_F = 1 'adjustable parameter for fitting

'===== Froth Parameters =====
Dim dblR_Entrainment, dblR_Attachment, dblFrothRecoveryFactor As Double 'entrainment and attachment recoveries
Dim dblB As Double = 3.3 'fitting parameter
Dim dblAlpha As Double = 0.01 'fitting parameter
Dim dblCoverage As Double = 0.5 'max particle coverage attached in froth
Dim dblPFTransfer, dblP_i, dblPr As Double
Dim dblEiw, dblEka As Double
Dim dblRmax As Double
'!! see global declarations for more froth parameters !!


'===== Get Textbox Values =====
dblSpPower = Val(TextBox_SpecificPower.Text) * 1000 'x1000 for w/m^3
dblSGasRate = Val(TextBox_SpecificAir.Text) / 100 '/100 for m/s
dblParticleDens = Val(TextBox_ParticleDensity.Text) * 1000 'x1000 for kg/m^3
dblAirFraction = Val(TextBox_AirFraction.Text)
dblSlurryFraction = Val(TextBox_SlurryFraction.Text)
dblImpellerDiam = Val(TextBox_ImpellerDiameter.Text)
dblContactAngle = Val(TextBox_ContactAngle.Text)
dblParticleZ = Val(TextBox_ParticleZ.Text)
dblBubbleZ = Val(TextBox_BubbleZ.Text)
dblCellHeight = Val(TextBox_CellHeight.Text)
dblCellDiam = Val(TextBox_CellDiameter.Text)
dblNumCells = Val(TextBox_NumCells.Text)
dblRetTime = Val(TextBox_RetentionTime.Text)
dblFrothHeight = Val(TextBox_FrothHeight.Text)
dblFrotherConc = Val(TextBox_FrotherConc.Text)

gammaMIBC = 0.000005 'mol/m^2
gammaPPG400 = 0.000001 'mol/m^2
gammaOctanol = 0.000008 'mol/m^2
gammaPentanol = 0.000006 'mol/m^2
kMIBC = 230 'M^-1
kPPG400 = 1700000 'M^-1
kOctanol = 2200 'M^-1
kPentanol = 55 'M^-1

If ComboBox_Frother.Text = "MIBC" Then
dblFrotherConc = dblFrotherConc / 102170 'convert ppm to mol/L
dblSurfaceTension = 0.07243 - 8.314 * (273.15 + 23) * gammaMIBC * Log(kMIBC * dblFrotherConc + 1)
ElseIf ComboBox_Frother.Text = "PPG 400" Then
dblFrotherConc = dblFrotherConc / 134170 'convert ppm to mol/L
dblSurfaceTension = 0.07243 - 8.314 * (273.15 + 23) * gammaPPG400 * Log(kPPG400 * dblFrotherConc + 1)
ElseIf ComboBox_Frother.Text = "Octanol" Then
dblFrotherConc = dblFrotherConc / 130230 'convert ppm to mol/L
dblSurfaceTension = 0.07243 - 8.314 * (273.15 + 23) * gammaOctanol * Log
```

```
(kOctanol * dblFrotherConc + 1)
ElseIf ComboBox_Frother.Text = "Pentanol" Then
dblFrotherConc = dblFrotherConc / 88150 'convert ppm to mol/L
dblSurfaceTension = 0.07243 - 8.314 * (273.15 + 23) * gammaPentanol *
Log(kPentanol * dblFrotherConc + 1)
Else
dblSurfaceTension = 0.07243 'pure water @ 23°C
End If

TextBox_SurfaceTension.Text = Format(dblSurfaceTension * 1000, "##.##")

Call CheckInputs() 'call function check for input values
If blnCheck = False Then
Exit Function 'if inputs missing, exit sub
End If

'===== BEGIN MULTIPARTICLE LOOP =====
dblParticleDiam = 0.000001 '(1 micron)

Dim i As Integer
Dim dblAddRec, dblAddK, dblAddPa, dblAddPc, dblAddPd, dblAddFR As Double

For i = 0 To 37 'particle loop


dblAddRec = 0
dblAddK = 0
dblAddPa = 0
dblAddPc = 0
dblAddPd = 0
dblAddFR = 0
dblR_Water_avg = 0

Call ContactAngleDist()

'===== Energy Dissipation =====

dblTotalDens = dblAirFraction * airDensity + _
(1 - dblAirFraction) * dblSlurryFraction * dblParticleDens _
+ (1 - dblSlurryFraction) * waterDensity
dblEMean = dblSpPower / dblTotalDens
dblEBulk = dblBulkZone * dblEMean
dblEImpeller = dblImpellerZone * dblEMean

If TextBox_BubbleSize.Enabled = False Then
dblBubbleDiam = (2.11 * dblSurfaceTension / (waterDensity *
dblEImpeller ^ 0.66)) ^ 0.6
Else
dblBubbleDiam = Val(TextBox_BubbleSize.Text) / 1000
End If

dblNumAttached = dblCoverage * 4 * (dblBubbleDiam / dblParticleDiam) ^ 2
'num of particles attached to one bubble


'===== Cell Calculations =====
dblCollisionDiam = (dblParticleDiam + dblBubbleDiam) 'avg diam of
collision
dblVolParticle = (4 / 3) * pi * (dblParticleDiam / 2) ^ 3 'vol 1 part.
dblVolBubble = (4 / 3) * pi * (dblBubbleDiam / 2) ^ 3 'vol 1 bubb.
```

```
dblVolBP = dblVolBubble + dblVolParticle 'vol of 1 BP aggregate
dblVolCell = pi * (dblCellDiam / 2) ^ 2 * dblCellHeight
dblKinVisc = waterViscosity / waterDensity
dblMassParticle = dblParticleDens * dblVolParticle 'mass 1 part.
dblMassBubble = airDensity * dblVolBubble 'mass 1 bubb.
dblMassBP = dblMassBubble + dblMassParticle 'mass of 1 BP aggregate
dblMassTotal = dblVolCell * dblTotalDens


'===== Velocities by Dissipation =====
dblU1Bulk = (0.4 * (dblEBulk ^ (4 / 9)) * (dblParticleDiam ^ (7 / 9)) _
* (dblKinVisc ^ (-1 / 3)) * (dblParticleDens / _
waterDensity - 1) ^ (2 / 3)) ^ 2 'for attachment
dblU2Bulk = 2 * (dblEBulk * dblBubbleDiam) ^ (2 / 3)
dblU1Mean = (0.4 * (dblEMean ^ (4 / 9)) * (dblParticleDiam ^ (7 / 9)) _
* (dblKinVisc ^ (-1 / 3)) * (dblParticleDens / _
waterDensity - 1) ^ (2 / 3)) ^ 2
dblU2Mean = 2 * (dblEMean * dblBubbleDiam) ^ (2 / 3)

dblBeta = (2 ^ (3 / 2)) * (pi ^ 0.5) * (dblCollisionDiam ^ 2) * _
Sqrt(dblU1Bulk + dblU2Bulk) 'from Abrahamson model using bulk dissipation

'===== Calc # Density of Bubbles =====
dblNBubble = dblAirFraction / dblVolBubble
dblNParticle = (1 - dblAirFraction) * dblSlurryFraction / dblVolParticle
dblZBubbParticle = dblBeta * dblNBubble * dblNParticle

dblWorkAdhesion = dblSurfaceTension * pi * (dblParticleDiam / 2) ^ 2 _
* (1 - Cos(dblContactAngle * (pi / 180)) ^ 2) 'calc work of adhesion for 1 particle

'===== Energy Barrier =====
Call EnergyBarrier() 'calls function to calc energy barrier

If dblEnergyBarrier <= 0 Then
dblEnergyBarrier = 0
End If

'===== Kinetic Energy of Attachment =====
dblKineticEAttach = 0.5 * dblMassParticle * dblU1Bulk / (dblDragBeta ^ 2)
dblKineticEDetach = 0.5 * dblMassParticle * (dblDetach_F * (dblParticleDiam _
+ dblBubbleDiam) * Sqrt(dblEImpeller / dblKinVisc)) ^ 2

'===== Probabilities =====
dblPAtt = Exp(-dblEnergyBarrier / dblKineticEAttach) 'prob. of attachment
dblPDet = Exp((-dblWorkAdhesion + dblKineticEAttach) _
/ dblKineticEDetach) 'prob. of detachment
dblRe = Sqrt(dblU2Bulk) * dblBubbleDiam / dblKinVisc 'bubble Reynold's number

dblPCol = Tanh(Sqrt(3 / 2 * (1 + (3 / 16 * dblRe) / (1 + 0.249 * dblRe ^ 0.56))) _
```

```
* (dblParticleDiam / dblBubbleDiam)) ^ 2 'prob. collision, modified Luttrell and Yoon

If dblPCol >= 1 Then
dblPCol = 1
End If

dblEiw = gravity / (4 * pi) * (dblKinVisc ^ 3 / dblEBulk) ^ 0.25
dblEka = (dblMassBubble * dblU2Bulk - 2 * (dblBubbleDiam / dblParticleDiam) ^ 2 * dblMassParticle * dblU1Bulk) ^ 2 _
/ (100 * (dblMassBubble + 2 * (dblBubbleDiam / dblParticleDiam) ^ 2 * dblMassParticle))

dblP_i = 13 * Sqrt((9 * waterViscosity ^ 2) / (dblBubbleDiam * dblSurfaceTension * dblTotalDens))
dblPr = Exp(-dblEiw / dblEka)
dblPFTransfer = dblP_i * (1 - dblPr)


'===== Froth Recovery =====
'''===start new froth rec model
Dim dblCoverageFactor = 2
Dim dblAf As Double
Dim dblA0 As Double
Dim dblCoarsenTime As Double
Dim dblFilmThick As Double
Dim dblCoalesceFactor As Double = 2
Dim dblL As Double

dblFilmThick = 3 / 4 * (0.33 / (1 - 0.33)) * dblBubbleDiam
dblCoarsenTime = (4 * waterViscosity * dblFrothHeight) / (waterDensity * gravity * dblFilmThick ^ 2)
dblL = waterViscosity / (airDensity * 0.015) '2 cm/s froth velocity


dblA0 = (dblBubbleDiam) ^ 2

dblAf = (Sqrt(dblSGasRate * waterViscosity / (waterDensity * gravity)) * Tan(Atan(Sqrt(airDensity * gravity * dblA0 / (dblSGasRate * waterViscosity))) _
- dblFrothHeight / 2 * Sqrt(waterDensity * gravity * dblSGasRate) / dblSurfaceTension)) ^ 2

dblRmax = Sqrt(Exp(dblCoalesceFactor * Sqrt(dblAf / dblA0) - dblCoalesceFactor))

dblR_Attachment = dblRmax * Exp(-dblCoverageFactor * (6 * dblFrothHeight / (dblBubbleDiam)) _
* (1 - dblRmax) * (dblParticleDiam / dblBubbleDiam) ^ 2)


dblR_Water_max = (0.33 * dblCoarsenTime * (6 * dblSGasRate) / (dblBubbleDiam / dblRmax)) * Exp(-dblFrothHeight / dblL)

If dblR_Water_max > 1 Then
dblR_Water_max = 1
End If

dblR_Entrainment = dblR_Water_max * Exp(-0.0325 * (dblParticleDens - waterDensity) - 0.063 * dblParticleDiam)

```

```
dblFrothRecoveryFactor = dblR_Entrainment + dblR_Attachment


'===== Rate Constant =====
Dim dblRecovery_ci As Double
Dim dblRecovery_I As Double

dblRateConst = dblBeta * dblNBubble * dblPAtt * dblPCol _
* (1 - dblPDet) * 60 'x60 to make 1/min

dblRecovery_ci = 1 - (1 + dblRateConst * dblRetTime) ^ (-1) 'eq 32 (Do & Yoon)

dblRecovery_I = dblRecovery_ci * dblFrothRecoveryFactor / (dblRecovery_ci * _
dblFrothRecoveryFactor + 1 - dblRecovery_ci) 'eq 6.2 finch & dobby

dblRecovery = 1 - (1 - dblRecovery_I) ^ dblNumCells

'========!!!!!!!!! insert condtional code here for hydrophobic coagulation
'if dblParticleDiam < specified size (~ < 15 micron)
'then multiply by some factors or use equations
' changes recovery for that size fraction

dblAddRec = dblRecovery
dblAddK = dblRateConst
dblAddPa = dblPAtt
dblAddPc = dblPCol
dblAddPd = dblPDet
dblAddFR = dblFrothRecoveryFactor
dblR_Water_avg = dblR_Water_max


'===== Output Results =====

arrRecovery(i) = dblAddRec * 100 '100 for percent
arrPDiam(i) = dblParticleDiam * 1000000 '10^6 for microns
arrRateK(i) = dblAddK * dblAddFR
arrPa(i) = dblAddPa * 100
arrPc(i) = dblAddPc * 100
arrPd(i) = (1 - dblAddPd) * 100
arrFR(i) = dblAddFR * 100

dblParticleDiam = dblParticleDiam * 1.2 'increment particle diam

Next i



'===== Graphs =====
ChartK.Series.Add(ri)
ChartK.Series(ri).ChartType = SeriesChartType.Line
Chart_Rec2.Series.Add(ri)
Chart_Rec2.Series(ri).ChartType = SeriesChartType.Line
Chart_RecLinear.Series.Add(ri)
Chart_RecLinear.Series(ri).ChartType = SeriesChartType.Line
Chart_Pa.Series.Add(ri)
Chart_Pa.Series(ri).ChartType = SeriesChartType.Line
```

```
Chart_Pc.Series.Add(ri)
Chart_Pc.Series(ri).ChartType = SeriesChartType.Line
Chart_Pd.Series.Add(ri)
Chart_Pd.Series(ri).ChartType = SeriesChartType.Line
Chart_FRec.Series.Add(ri)
Chart_FRec.Series(ri).ChartType = SeriesChartType.Line
Chart_Grade.Series.Add(ri)
Chart_Grade.Series(ri).ChartType = SeriesChartType.Line

Dim n As Integer
For n = 0 To 37
ChartK.Series(ri).Points.AddXY(arrPDiam(n), arrRateK(n))
Chart_Rec2.Series(ri).Points.AddXY(arrPDiam(n), arrRecovery(n))
Chart_RecLinear.Series(ri).Points.AddXY(arrPDiam(n), arrRecovery(n))
Chart_Pa.Series(ri).Points.AddXY(arrPDiam(n), arrPa(n))
Chart_Pc.Series(ri).Points.AddXY(arrPDiam(n), arrPc(n))
Chart_Pd.Series(ri).Points.AddXY(arrPDiam(n), arrPd(n))
Chart_FRec.Series(ri).Points.AddXY(arrPDiam(n), arrFR(n))
Next

Call AddToolTip() 'adds info to the graphs

ri = ri + 1 'increment series number and color

Call SizeDistribution()


'===== Temp Outputs for Debugging =====
'MsgBox("Some Outputs Temporary for Debugging")
If dblOvrRecovery > 0 Then
Label_RecoveryOutput.Text = Format(dblOvrRecovery, "#.###")
End If

label_VolCellOutput.Text = Format(dblVolCell, "#.###") & " m³"
label_WaterRecOut.Text = Format(dblR_Water_avg * 100, "#.###") & " %"


End Function
Public Function SizeDistribution()
Dim s15, s425, s250, s180, s150, s106, s75, s45, s38, s25 As Double 'txtbx valu
Dim f15, f425, f250, f180, f150, f106, f75, f45, f38, f25 As Double 'fraction
Dim r15, r425, r250, r180, r150, r106, r75, r45, r38, r25 As Double 'recovery

s425 = Val(SizeDist.TextBox_425.Text)
s250 = Val(SizeDist.TextBox_250.Text)
s180 = Val(SizeDist.TextBox_180.Text)
s150 = Val(SizeDist.TextBox_150.Text)
s106 = Val(SizeDist.TextBox_106.Text)
s75 = Val(SizeDist.TextBox_75.Text)
s45 = Val(SizeDist.TextBox_45.Text)
s38 = Val(SizeDist.TextBox_38.Text)
s25 = Val(SizeDist.TextBox_25.Text)
s15 = Val(SizeDist.TextBox_15.Text)

f425 = s425 - s250
f250 = s250 - s180
f180 = s180 - s150
```

```
f150 = s150 - s106
f106 = s106 - s75
f75 = s75 - s45
f45 = s45 - s38
f38 = s38 - s25
f25 = s25 - s15
f15 = s15

r15 = arrRecovery(15) / 100
r25 = arrRecovery(18) / 100
r38 = arrRecovery(20) / 100
r45 = arrRecovery(21) / 100
r75 = (arrRecovery(23) + arrRecovery(24)) / 2 / 100
r106 = (arrRecovery(25) + arrRecovery(26)) / 2 / 100
r150 = (arrRecovery(27) + arrRecovery(28)) / 2 / 100
r180 = (arrRecovery(28) + arrRecovery(29)) / 2 / 100
r250 = (arrRecovery(30) + arrRecovery(31)) / 2 / 100
r425 = arrRecovery(33) / 100


dblOvrRecovery = f15 * r15 + f25 * r25 + f38 * r38 + f45 * r45 + f75 * r75 + f106 * r106 _
+ f150 * r150 + f180 * r180 + f250 * r250 + f425 * r425

End Function


'save button also writes to file
Private Sub SaveToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles SaveToolStripMenuItem.Click

Dim saveFileDialog1 As New SaveFileDialog()

saveFileDialog1.Filter = "Text File (.txt)|*.txt|Word Document (.doc)|*.doc|Word 2007 Document (.docx)|*.docx" 'these are the file type options shown in the dialog
saveFileDialog1.Title = "Save File"
saveFileDialog1.ShowDialog() ' If the file name is not an empty string open it for saving

If saveFileDialog1.FileName <> "" Then

Dim fs As System.IO.FileStream = CType _
(saveFileDialog1.OpenFile(), System.IO.FileStream) ' Saves file as type selected in the dialog box via FileStream created by OpenFile method.
fs.Close() 'closes the text file

Dim fName As System.IO.StreamWriter
fName = My.Computer.FileSystem.OpenTextFileWriter(fs.Name, True) 'prepares to write to "fs" which was just created by the user

fName.WriteLine("Inputs") 'writes to the file
fName.WriteLine("----------------")
fName.WriteLine(" Specific Power = " & dblSpPower)
fName.WriteLine(" Gas Rate = " & dblSGasRate)
fName.WriteLine(" Particle S.G. = " & dblParticleDens / 1000) 'divide 1000 get get back into SG
fName.WriteLine(" Air Fraction = " & dblAirFraction)
fName.WriteLine(" Slurry Fraction = " & dblSlurryFraction)
```

```
'fName.WriteLine(" Feed Grade = " & dblgrade?)
fName.WriteLine()
fName.WriteLine(" Surface Tension = " & dblSurfaceTension)
fName.WriteLine(" Contact Angle = " & dblContactAngle)
fName.WriteLine(" Dielectric Constant = " & dblDielectric)
fName.WriteLine("Particle Zeta Potential = " & dblParticleZ)
fName.WriteLine(" Bubble Zeta Potential = " & dblBubbleZ)
fName.WriteLine(" Permitivity = " & dblPermitivity)
fName.WriteLine()
fName.WriteLine(" Cell Diameter = " & dblCellDiam)
fName.WriteLine(" Cell Height = " & dblCellHeight)
fName.WriteLine("Impeller Diameter = " & dblImpellerDiam)
fName.WriteLine("Number of Cells = " & dblNumCells)
fName.WriteLine(" Retention Time = " & dblRetTime)
fName.WriteLine(" Froth Height = " & dblFrothHeight)
fName.WriteLine(" Growth Factor = " & dblGrowthFactor)
fName.WriteLine("Max. Water Rec. = " & dblR_Water_max)
fName.WriteLine()
fName.WriteLine()
fName.WriteLine("Outputs")
fName.WriteLine("-----------------")
fName.WriteLine("Cell Volume = " & Format(dblVolCell, "#.####"))
fName.WriteLine("Recovery = " & Format(dblOvrRecovery, "#.####"))
fName.WriteLine()
fName.WriteLine()
fName.WriteLine()
fName.WriteLine("SimuFloat " & Now) 'adds the date and time at the bottom

fName.Close() 'closes the text file

End If

End Sub
Private Sub AboutToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles AboutToolStripMenuItem.Click
AboutBox1.ShowDialog() 'shows the about box
End Sub
Private Sub ExitToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ExitToolStripMenuItem.Click
Me.Close() 'closes the form
End Sub
Private Sub HelpToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles HelpToolStripMenuItem.Click
Help1.ShowDialog() 'shows the help box
End Sub
'Private Sub PrintToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles PrintToolStripMenuItem.Click
' Dim printDialog1 As New PrintDialog()

' printDialog1.ShowDialog()
'End Sub


'enable text input via checkboxes
Private Sub CheckBox_AirFrac_CheckedChanged(ByVal sender As System.Object,
```

```
ByVal
e As System.EventArgs) Handles CheckBox_AirFrac.CheckedChanged

If TextBox_AirFraction.Enabled = False Then
TextBox_AirFraction.Enabled = True
Else
TextBox_AirFraction.Enabled = False
End If

End Sub
Private Sub CheckBox_BubbleSize_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles CheckBox_BubbleSize.CheckedChanged
If TextBox_BubbleSize.Enabled = False Then
TextBox_BubbleSize.Enabled = True
Else
TextBox_BubbleSize.Enabled = False
End If
End Sub
Private Sub CheckBox_BubbleZeta_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles CheckBox_BubbleZeta.CheckedChanged
If TextBox_BubbleZ.Enabled = False Then
TextBox_BubbleZ.Enabled = True
Else
TextBox_BubbleZ.Enabled = False
End If
End Sub
Private Sub CheckBox_ContactDistrib_CheckedChanged(ByVal sender As
System.Object
, ByVal e As System.EventArgs) Handles CheckBox_ContactDistrib.CheckedChanged
CheckBox_FeedGrade.Checked = False

If Button_ContactDistrib.Enabled = False Then
Button_ContactDistrib.Enabled = True
Else
Button_ContactDistrib.Enabled = False
End If
If TextBox_ContactAngle.Enabled = False Then
TextBox_ContactAngle.Enabled = True
Else
TextBox_ContactAngle.Enabled = False
End If
End Sub
Private Sub CheckBox_DielectricConst_CheckedChanged(ByVal sender As System.
Object, ByVal e As System.EventArgs) Handles CheckBox_DielectricConst.
CheckedChanged
If TextBox_DielectricConst.Enabled = False Then
TextBox_DielectricConst.Enabled = True
Else
TextBox_DielectricConst.Enabled = False
End If
End Sub
Private Sub CheckBox_FeedGrade_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles CheckBox_FeedGrade.CheckedChanged
CheckBox_ContactDistrib.Checked = False

If Button_FeedGrade.Enabled = False Then
Button_FeedGrade.Enabled = True
Else
Button_FeedGrade.Enabled = False
End If
```

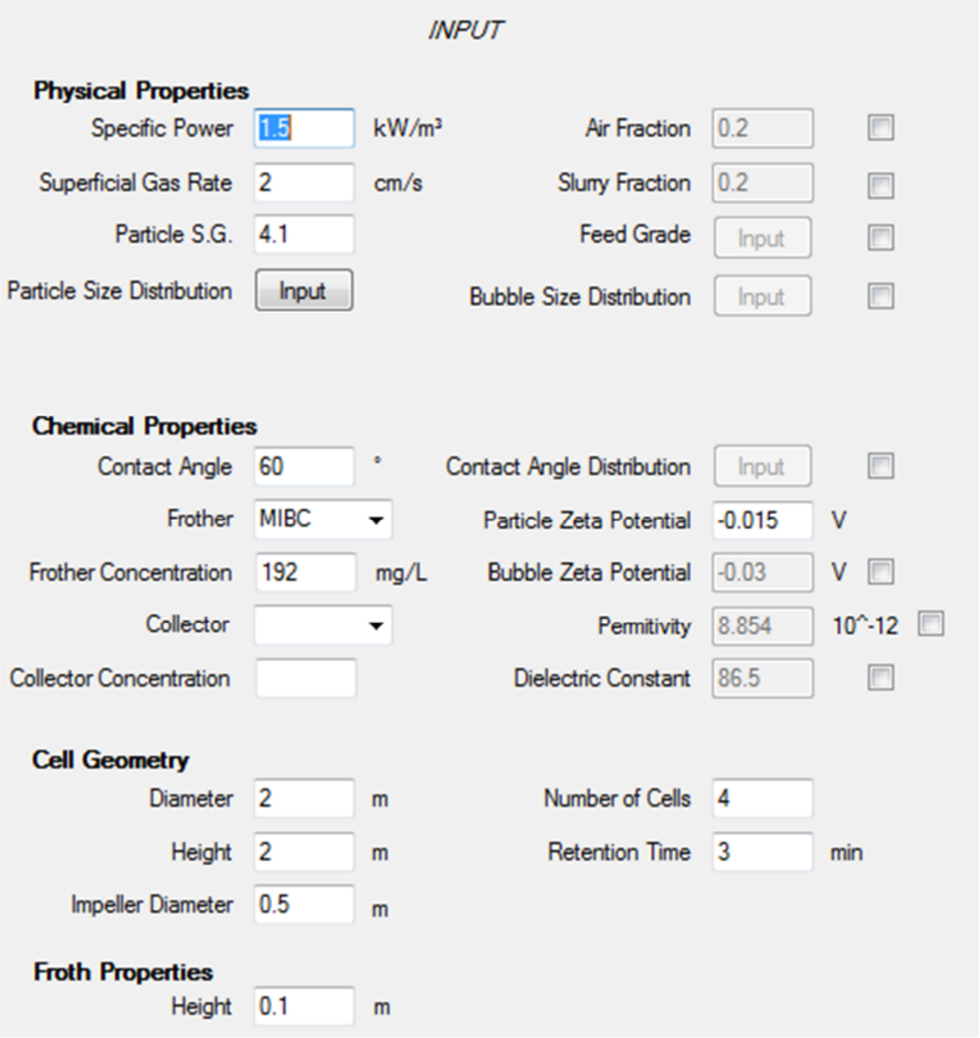
```

If TextBox_ParticleDensity.Enabled = False Then
TextBox_ParticleDensity.Enabled = True
Else
TextBox_ParticleDensity.Enabled = False
End If

If TextBox_ContactAngle.Enabled = False Then
TextBox_ContactAngle.Enabled = True
Else
TextBox_ContactAngle.Enabled = False
End If
End Sub
Private Sub CheckBox_SlurryFrac_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles CheckBox_SlurryFrac.CheckedChanged
If TextBox_SlurryFraction.Enabled = False Then
TextBox_SlurryFraction.Enabled = True
Else
TextBox_SlurryFraction.Enabled = False
End If
End Sub
Private Sub CheckBox_Permitivity_CheckedChanged(ByVal sender As
System.Object,
ByVal e As System.EventArgs) Handles CheckBox_Permitivity.CheckedChanged
If TextBox_Permitivity.Enabled = False Then
TextBox_Permitivity.Enabled = True
Else
TextBox_Permitivity.Enabled = False
End If
End Sub


'Keypress subs for disallowing letters in textboxes
'Allows 0123456789 - . backspace delete
Private Sub TextBox_BubbleZeta_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles TextBox_BubbleZ.KeyPress
Dim allowedChars As String = "0123456789.-" & Chr(8) & Chr(127)

If allowedChars.IndexOf(e.KeyChar) = -1 Then
' Invalid Character
e.Handled = True
End If
End Sub
Private Sub TextBox_ContactAngle_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles TextBox_ContactAngle.KeyPress
Dim allowedChars As String = "0123456789.-" & Chr(8) & Chr(127)

If allowedChars.IndexOf(e.KeyChar) = -1 Then
' Invalid Character
e.Handled = True
End If
End Sub
Private Sub TextBox_NumCell_KeyPress(ByVal sender As Object, ByVal e As
System.
Windows.Forms.KeyPressEventArgs) Handles TextBox_NumCells.KeyPress
Dim allowedChars As String = "0123456789.-" & Chr(8) & Chr(127)

If allowedChars.IndexOf(e.KeyChar) = -1 Then
' Invalid Character
e.Handled = True
```

```
End If
End Sub
Private Sub TextBox_RetTime_KeyPress(ByVal sender As Object, ByVal e As
System.
Windows.Forms.KeyPressEventArgs) Handles TextBox_RetentionTime.KeyPress
Dim allowedChars As String = "0123456789.-" & Chr(8) & Chr(127)

If allowedChars.IndexOf(e.KeyChar) = -1 Then
' Invalid Character
e.Handled = True
End If
End Sub
Private Sub TextBox_SG_KeyPress(ByVal sender As Object, ByVal e As System.
Windows.Forms.KeyPressEventArgs) Handles TextBox_ParticleDensity.KeyPress
Dim allowedChars As String = "0123456789.-" & Chr(8) & Chr(127)

If allowedChars.IndexOf(e.KeyChar) = -1 Then
' Invalid Character
e.Handled = True
End If
End Sub
Private Sub TextBox_SpecificAir_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles TextBox_SpecificAir.KeyPress
Dim allowedChars As String = "0123456789.-" & Chr(8) & Chr(127)

If allowedChars.IndexOf(e.KeyChar) = -1 Then
' Invalid Character
e.Handled = True
End If
End Sub
Private Sub TextBox_SpecificPower_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles TextBox_SpecificPower.KeyPress

Dim allowedChars As String = "0123456789.-" & Chr(8) & Chr(127)

If allowedChars.IndexOf(e.KeyChar) = -1 Then
' Invalid Character
e.Handled = True
End If

End Sub
Private Sub TextBox_SurfaceTension_KeyPress(ByVal sender As Object, ByVal e
As
System.Windows.Forms.KeyPressEventArgs) Handles TextBox_SurfaceTension.KeyPress
Dim allowedChars As String = "0123456789.-" & Chr(8) & Chr(127)

If allowedChars.IndexOf(e.KeyChar) = -1 Then
' Invalid Character
e.Handled = True
End If
End Sub

'subs for adding tooltips to labels
Private Sub Label_FrotherConc_MouseHover(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Label_FrotherConc.MouseHover
Dim ttfc As New ToolTip()

ttfc.AutoPopDelay = 5000
ttfc.InitialDelay = 500
ttfc.ReshowDelay = 500
```

```
ttfc.ShowAlways = True
ttfc.SetToolTip(Me.Label_FrotherConc, "Milligrams of frother added per liter of slurry")

End Sub
Private Sub Label_CellNum_MouseHover(ByVal sender As Object, ByVal e As System.EventArgs) Handles Label_CellNum.MouseHover
Dim ttfc As New ToolTip()

ttfc.AutoPopDelay = 5000
ttfc.InitialDelay = 500
ttfc.ReshowDelay = 500
ttfc.ShowAlways = True
ttfc.SetToolTip(Me.Label_CellNum, "Number of identical cells in the flotation bank")
End Sub
Private Sub Label_Bub_MouseHover(ByVal sender As Object, ByVal e As System.EventArgs) Handles Label_Bub.MouseHover
Dim ttfc As New ToolTip()

ttfc.AutoPopDelay = 5000
ttfc.InitialDelay = 500
ttfc.ReshowDelay = 500
ttfc.ShowAlways = True
ttfc.SetToolTip(Me.Label_Bub, "Enter bubble size if known, SimuFloat will calculate otherwise")
End Sub
Private Sub Label_ContactDist_MouseHover(ByVal sender As Object, ByVal e As System.EventArgs) Handles Label_ConDist.MouseHover
Dim ttfc As New ToolTip()

ttfc.AutoPopDelay = 5000
ttfc.InitialDelay = 500
ttfc.ReshowDelay = 500
ttfc.ShowAlways = True
ttfc.SetToolTip(Me.Label_ConDist, "Enable this function to use a distribution of contact angles for a single component feed")
End Sub
Private Sub Label_FGrade_MouseHover(ByVal sender As Object, ByVal e As System.EventArgs) Handles Label_FGrade.MouseHover
Dim ttfc As New ToolTip()

ttfc.AutoPopDelay = 5000
ttfc.InitialDelay = 500
ttfc.ReshowDelay = 500
ttfc.ShowAlways = True
ttfc.SetToolTip(Me.Label_FGrade, "Enable this function to input parameters for a multi-component feed")
End Sub

End Class

```

Appendix B

Standard Conditions Chalcopyrite



INPUT

Physical Properties

Specific Power 1.5 $\mathrm{kW/m^3}$

Superficial Gas Rate 2 cm/s

Particle S.G. 4.1

Particle Size Distribution Input

Air Fraction 0.2 ☐

Slurry Fraction 0.2 ☐

Feed Grade Input ☐

Bubble Size Distribution Input ☐

Chemical Properties

Contact Angle 60 °

Frother MIBC

Frother Concentration 192 mg/L

Collector

Collector Concentration

Contact Angle Distribution Input ☐

Particle Zeta Potential -0.015 V

Bubble Zeta Potential -0.03 V ☐

Permitivity 8.854 10^{-12} ☐

Dielectric Constant 86.5 ☐

Cell Geometry

Diameter 2 m

Height 2 m

Impeller Diameter 0.5 m

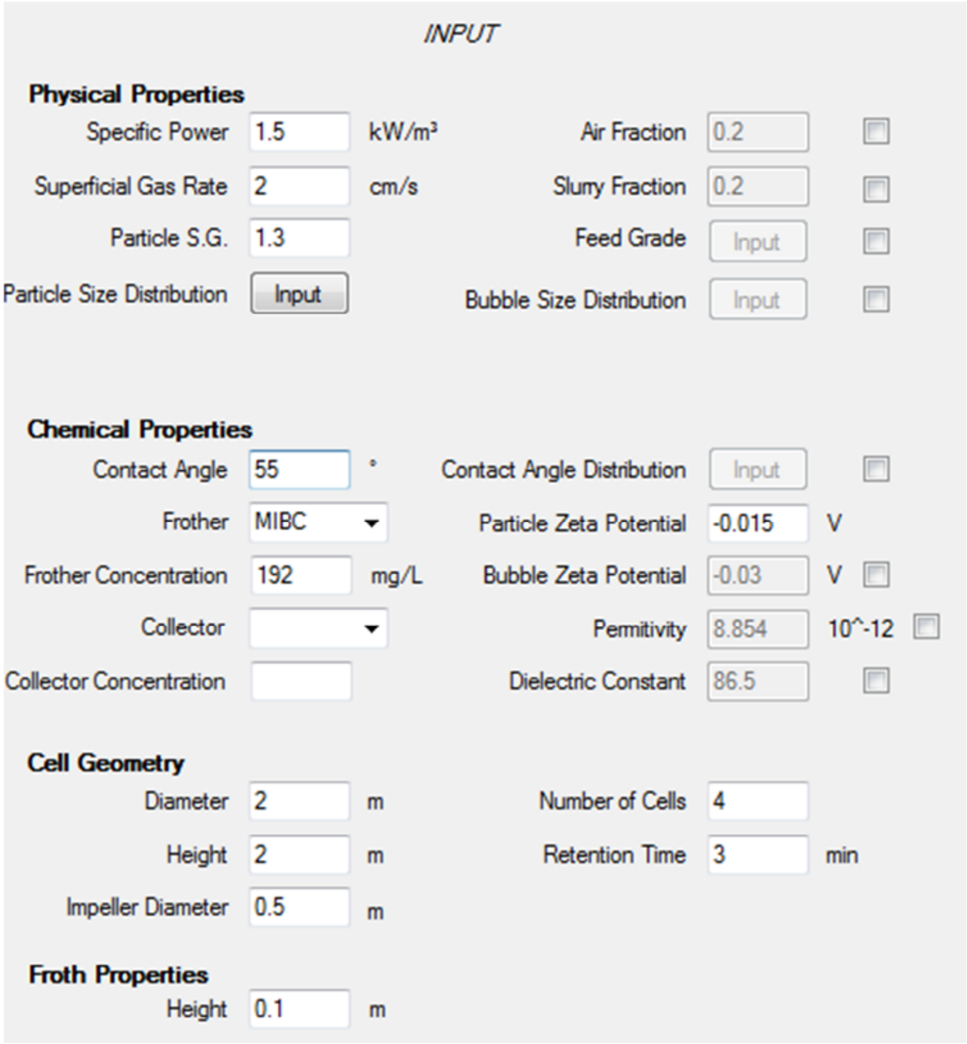
Number of Cells 4

Retention Time 3 min

Froth Properties

Height 0.1 m

Standard Conditions Coal



INPUT

Physical Properties

Specific Power 1.5 kW/m^3

Superficial Gas Rate 2 cm/s

Particle S.G. 1.3

Particle Size Distribution Input

Air Fraction 0.2 ☐

Slurry Fraction 0.2 ☐

Feed Grade Input ☐

Bubble Size Distribution Input ☐

Chemical Properties

Contact Angle 55 °

Frother MIBC

Frother Concentration 192 mg/L

Collector

Collector Concentration

Contact Angle Distribution Input ☐

Particle Zeta Potential -0.015 V

Bubble Zeta Potential -0.03 V ☐

Permitivity 8.854 10^-12 ☐

Dielectric Constant 86.5 ☐

Cell Geometry

Diameter 2 m

Height 2 m

Impeller Diameter 0.5 m

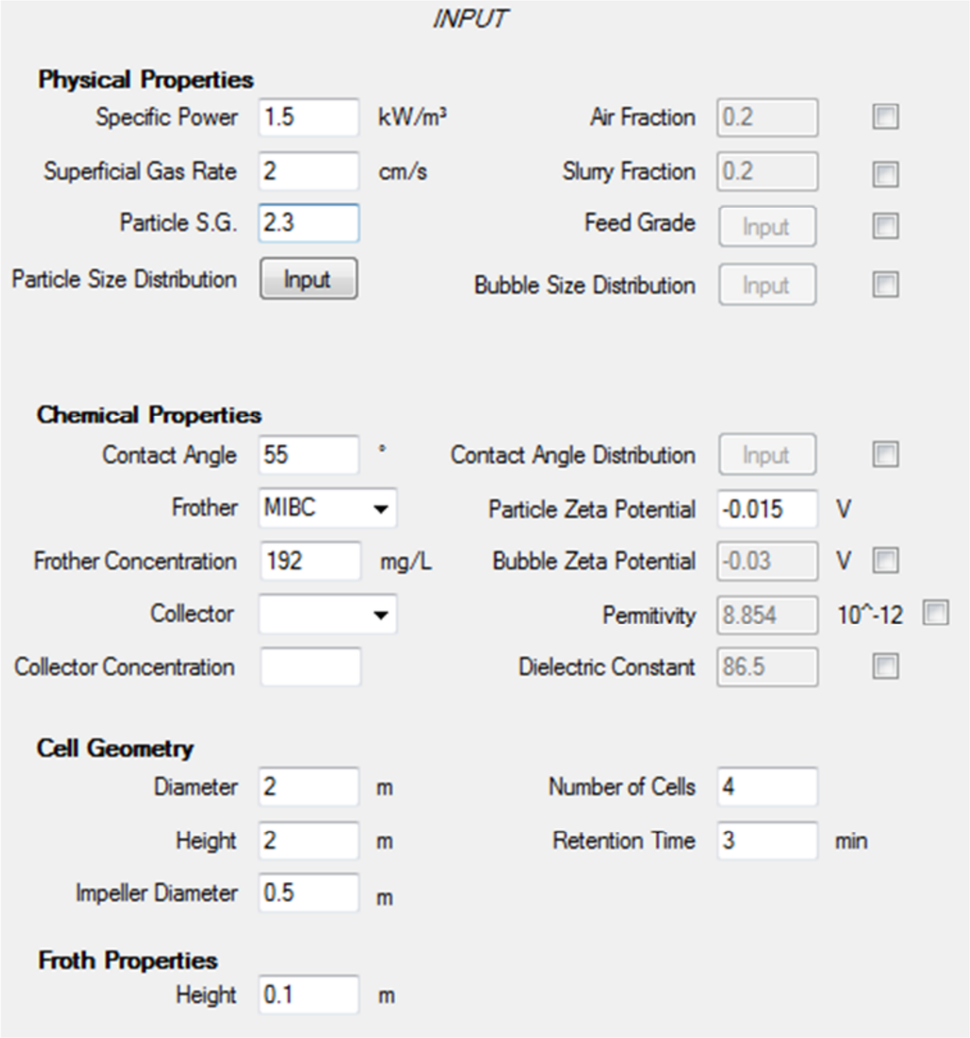
Number of Cells 4

Retention Time 3 min

Froth Properties

Height 0.1 m

Standard Conditions Phosphate


INPUT

Physical Properties

Specific Power 1.5 kW/m³
Air Fraction 0.2 ☐
Superficial Gas Rate 2 cm/s
Slurry Fraction 0.2 ☐
Particle S.G. 2.3
Feed Grade Input ☐
Particle Size Distribution Input
Bubble Size Distribution Input ☐

Chemical Properties

Contact Angle 55 °
Contact Angle Distribution Input ☐
Frother MIBC
Particle Zeta Potential -0.015 V
Frother Concentration 192 mg/L
Bubble Zeta Potential -0.03 V ☐
Collector
Permitivity 8.854 10^-12 ☐
Collector Concentration
Dielectric Constant 86.5 ☐

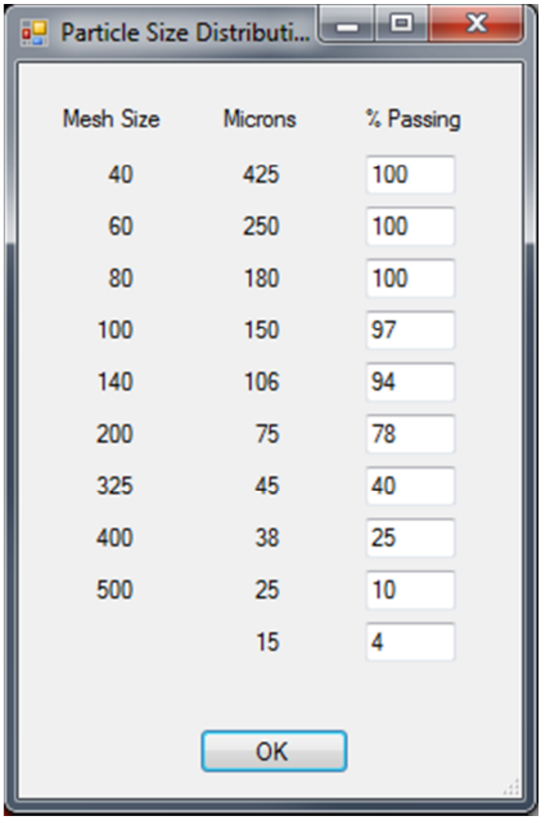
Cell Geometry

Diameter 2 m
Number of Cells 4
Height 2 m
Retention Time 3 min
Impeller Diameter 0.5 m

Froth Properties

Height 0.1 m

Particle Size Distribution Used in Simulations



Mesh Size	Microns	% Passing
40	425	100
60	250	100
80	180	100
100	150	97
140	106	94
200	75	78
325	45	40
400	38	25
500	25	10
	15	4

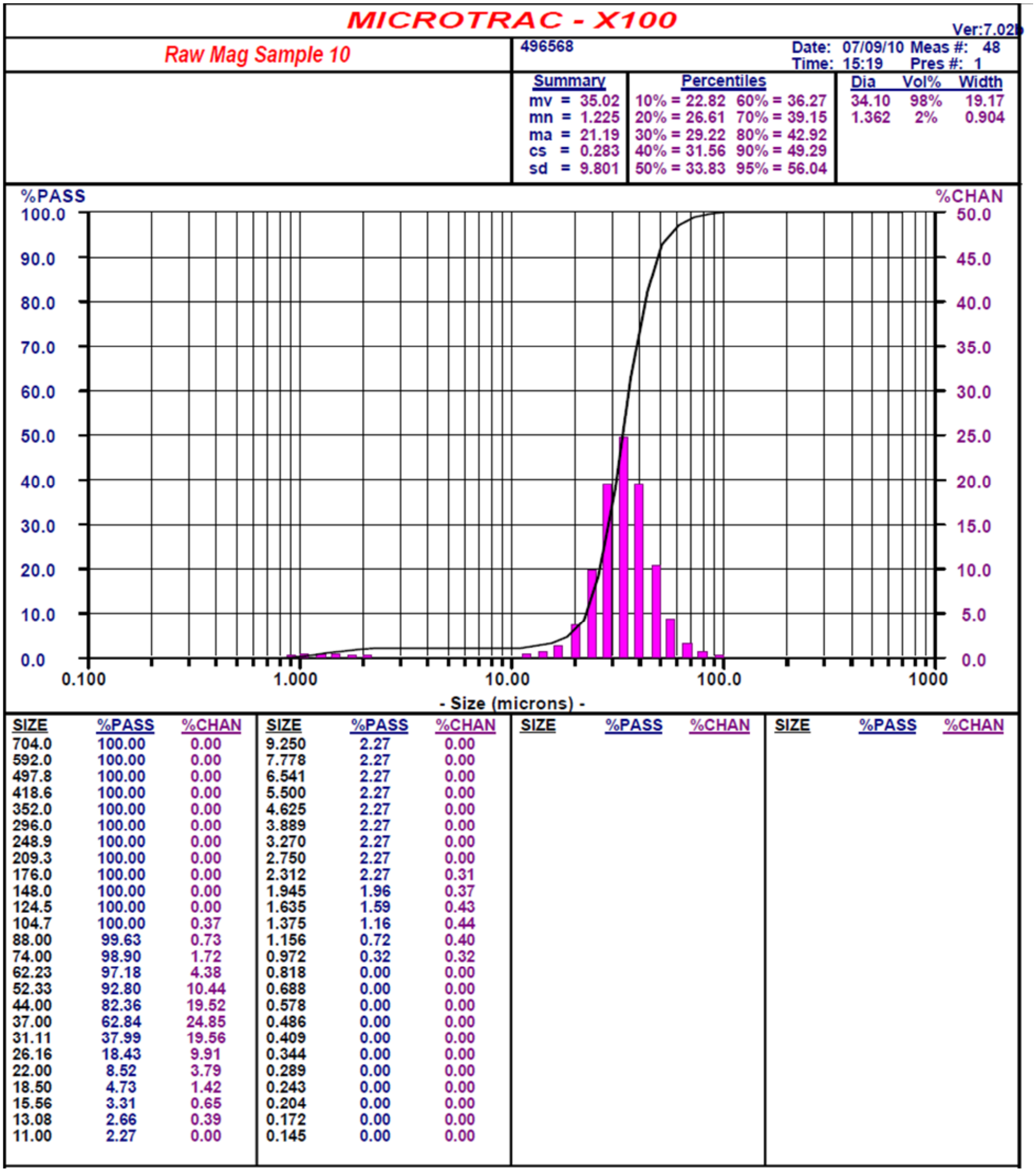
Figures 16 and 17 Input Parameters

Liberation Class (% chalcopyrite)	Contact Angle
0-30	25
30-60	33
60-90	40
90-100	60

Figure 28 Input Parameters

Test	% of Total Feed		
	Gangue	Mids	Mineral
No Middlings, 40°	96	0	4
No Middlings, 60°	96	0	4
Low Middlings	93	4	3
Intermediate Middlings	91	7	2
High Middlings	87	12	1

Appendix C



MICROTRAC - X100

Ver:7.02b

Raw Mag Sample 10

496568

Date: 07/09/10 Meas #: 48
Time: 15:19 Pres #: 1

Summary	Percentiles		Dia	Vol%	Width
mv = 35.02	10% = 22.82	60% = 36.27	34.10	98%	19.17
mn = 1.225	20% = 26.61	70% = 39.15	1.362	2%	0.904
ma = 21.19	30% = 29.22	80% = 42.92			
cs = 0.283	40% = 31.56	90% = 49.29			
sd = 9.801	50% = 33.83	95% = 56.04			

SIZE	%PASS	%CHAN	SIZE	%PASS	%CHAN	SIZE	%PASS	%CHAN	SIZE	%PASS	%CHAN
704.0	100.00	0.00	9.250	2.27	0.00						
592.0	100.00	0.00	7.778	2.27	0.00						
497.8	100.00	0.00	6.541	2.27	0.00						
418.6	100.00	0.00	5.500	2.27	0.00						
352.0	100.00	0.00	4.625	2.27	0.00						
296.0	100.00	0.00	3.889	2.27	0.00						
248.9	100.00	0.00	3.270	2.27	0.00						
209.3	100.00	0.00	2.750	2.27	0.00						
176.0	100.00	0.00	2.312	2.27	0.31						
148.0	100.00	0.00	1.945	1.96	0.37						
124.5	100.00	0.00	1.635	1.59	0.43						
104.7	100.00	0.37	1.375	1.16	0.44						
88.00	99.63	0.73	1.156	0.72	0.40						
74.00	98.90	1.72	0.972	0.32	0.32						
62.23	97.18	4.38	0.818	0.00	0.00						
52.33	92.80	10.44	0.688	0.00	0.00						
44.00	82.36	19.52	0.578	0.00	0.00						
37.00	62.84	24.85	0.486	0.00	0.00						
31.11	37.99	19.56	0.409	0.00	0.00						
26.16	18.43	9.91	0.344	0.00	0.00						
22.00	8.52	3.79	0.289	0.00	0.00						
18.50	4.73	1.42	0.243	0.00	0.00						
15.56	3.31	0.65	0.204	0.00	0.00						
13.08	2.66	0.39	0.172	0.00	0.00						
11.00	2.27	0.00	0.145	0.00	0.00						

Vita

Kyle R. Kelley was born to Scott and Sandy Kelley in September 1987 in Fairfax, Virginia. After Graduating from Bishop O'Connell High School in 2005, he began his undergraduate studies at Virginia Tech. In May 2009 Kyle received his Bachelor's degree in Mining and Minerals Engineering. He expects to complete the requirements for a Master's degree in Mining and Minerals Engineering in January 2011.