An Assessment of Software Packages as Applied to Particles in a Rotating Mill

Leigh M. Sonn

This is in partial fulfilment for the degree of
Master of Science (Applied Science)

Department of Mechanical Engineering
University of Cape Town

September, 2001

Declaration

I, Leigh M. Sonn, submit this thesis in part fulfillment of the requirements for the degree of Masters of Science in Applied Science. I claim that this is my original work and has not been submitted in this or any other form for a degree at any university.

Acknowledgements

Thanks are due to AMIRA and the National Research Foundation (NRF) for financial assistance.

Thanks are also due to the De Beers Research Laboratory (DRL) especially Mr. P. Sargeant, for the financial assistance with regard to the ELFEN licence for the duration of the project.

I would like to thank the staff at the Department of Mechanical Engineering and CERECAM for their assistance, especially Mr. V. Balden. Thanks are also due to my supervisor Prof. G.N. Nurick for his continued support and assistance. To Mr. D. Bailey and Mr. X. N. Middleton for their input regarding the final version of this thesis. Also to my close friends and family for their encouragement.

Abstract

This thesis evaluates three different software packages to test their potential use in comminution. The packages are ELFEN, ABAQUS and Particle Flow Code (PFC^{2D}).

The numerical codes differ in the method of inputting data as well as the theoretical models used. ELFEN and ABAQUS are finite element (FEM) packages whereas PFC^{2D} uses the discrete element method (DEM). The packages also differ in the integration and contact schemes, as well as the method used to calculate the critical time step used in the analysis.

Simulations are conducted for a single rectangular particle in mills with different liner configurations. The liners include a smooth, a corrugated and the Skorupa liner. The experimental angles of departure are compared to the numerical results for a range of mill speeds.

The software packages are in close agreement with one another for the smooth liner, but differ from the experimental results for speeds higher than 105% critical. Damping had to be applied in PFC^{2D}, but not for the other packages.

ELFEN and PFC^{2D} simulate the angles of departure for the corrugated liner and Skorupa liner are satisfactorily close to the experimental trends. Damping was applied in both packages for this liner configuration. PFC^{2D} was found to be sensitive to changes in damping implying a dependence on mill speed. The trend of the damping values used in the corrugated and Skorupa liners differed in PFC^{2D}. This implies that the damping value is related to the liner configuration of the mill.

A range of damping values was used for the corrugated and Skorupa liners for ABAQUS. The numerical results did not correlate well to the experimental values. The motion of the particle in ABAQUS was erratic and this made it difficult to determine the angle of departure.

PFC^{2D} was the most time efficient package. A simulation that took 7 to 8 hours in ELFEN and approximately 30 minutes in ABAQUS, took approximately 2 minutes in PFC^{2D}.

Contents

List of Figures

Nomenclature

R = Normal reaction in liner (N)
T = Tangential frictional force (N)
θ = Angular position of the particle on the liner (degrees)
$\dot{\theta}$ = Angular velocity of the particle about the Mill Centre (rad/s)
$\ddot{\theta}$ = Angular acceleration of the particle about the Mill Centre (rad/s^2)
m = mass (kg)
ρ = Mill radius (m)
Ω_m = Mill Rotational speed (rad/s)
t = Time (s)
μ = Coefficient of friction
β = Relative angular position of the particle (degrees)
c_1 = Amplitude of corrugations for corrugated liner (m)
ζ_1 = Frequency of corrugations for corrugated liner
g = Gravitational acceleration (9.8 m/s^2)
a = Ball radius (m)
γ = Angular position of lifter tangent at contact point from the horizontal (rad)
S = Sliding motion
x_i = Position of disc i
c_i = damping constant of disc i
K_i = Material stiffness of disc i
F_i = Force acting in direction i
I = Moment of inertia
s_i = The perpendicular line from the line of action of the force to the centroid of the disc

M	= Moment about the centroid of the disc
f	= Factor of critical step in ELFEN
u	= Displacement
$\dot{u}$	= Velocity
$\ddot{u}$	= Acceleration
Δt	= Time increment in ABAQUS
$\omega_{\max}$	= highest eigenvalue in the system in ABAQUS
ξ	= Fraction of critical damping in the highest mode in ABAQUS
α	= Mass proportional damping factor in ABAQUS
δ	= Stiffness proportional damping in ABAQUS
t_{crit}	= Critical timestep in PFC^{2D}
k^{tran}	= Translational stiffness in PFC^{2D}
k^{rot}	= Rotational stiffness in PFC^{2D}

Glossary

Angle of Departure	: Angle where the particle comes away from the liner
Cascading	: Description of a particle that is rolling down the face of the 'en Masse' region
Cataracting	: Description of a particle that is in flight above the body of the charge
Centrifuging	: Charge does not leave the liner while mill rotates
Critical Speed	: The mill speed where the particle becomes centrifuged to the liner
Shoulder of Charge	: The uppermost part of the charge where particles start to come away from the liner
Toe of Charge	: The lowest part of the charge, onto which cataracting particles impact
DEM	: Discrete Element Method
FEM	: Finite Element Method

Chapter 1

Introduction

Mining in South Africa contributes about 8% to the country's Gross Domestic Product [7]. It is one of the world's largest producers of platinum group elements (PGE's), as well as gold [8]. PGE's include platinum, palladium, iridium, osmium, rhodium and ruthenium[9]. These minerals are used mainly by the automotive industry, where they are used in catalytic converters to control exhaust emissions [10]. South Africa holds approximately 55% of the world's platinum and palladium reserves [9] and 40% of the gold reserves.

PGE's are found in mafic and ultramafic rocktypes [10]. Gold is found primarily in hard, abrasive quartzic ore. Mills are used to grind this rock down to a required particle size before the mineral is extracted from the ore. The size of the mills used varies from 2m to 12m in diameter[11]. The ratio of the diameter to the length of a tube mill ranges from 3:1 to 1:5. Steel balls with diameters ranging from 50 to 20mm in diameter are added to the ore to assist in grinding the ore to a powder. The mills consume large amounts of power which is costly to the mine. It is therefore essential to run the mills using optimal operating conditions. To achieve this, an understanding of the motion inside the tube mill has to be reached.

There have been many studies conducted in the past century to better understand the dynamics inside a mill. This work has been made difficult by the harsh environment that exists inside a mill. There has been a move towards using numerical methods to simulate the motion inside a mill and hence achieve a better understanding of the parameters that affect the performance of a tube mill. The discrete element method, originally used in rock mechanics [12], was first applied to

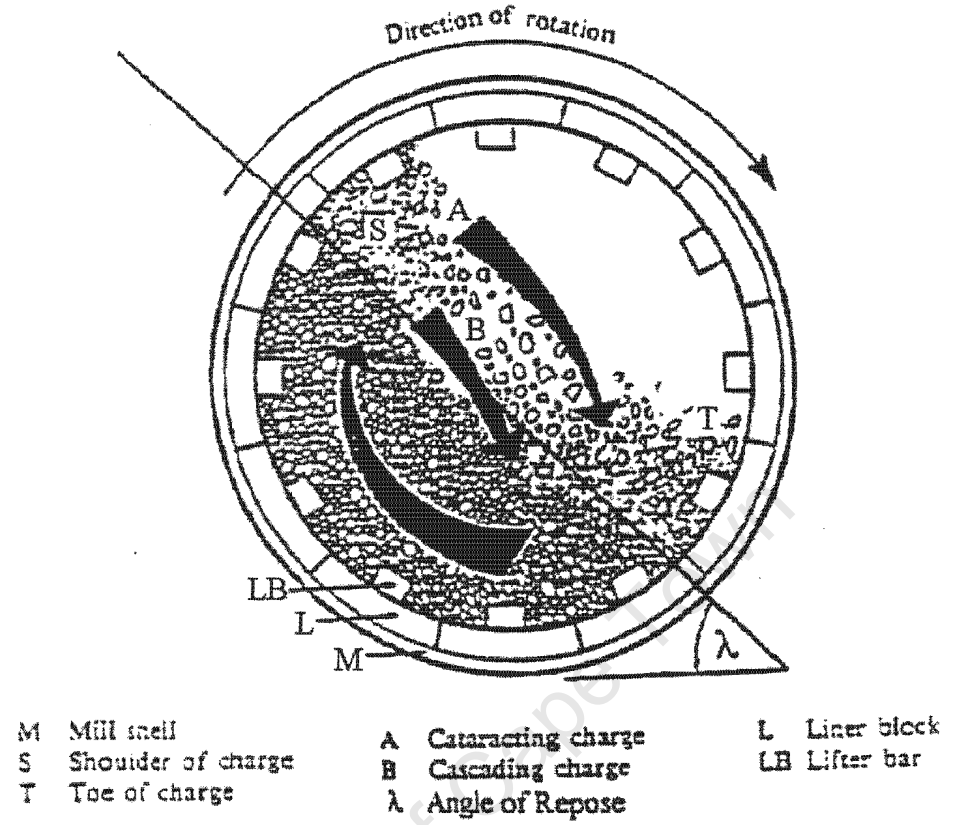


Figure 1.1: Cross section of a rotating mill [3]

the comminution process by Mishra and Rajamani [5]. There have since been many studies conducted to validate this numerical method for use in comminution such as those by Cleary[13], Cleary and Hoyer[14] and Van Nierop, Glover, Hinde and Moys [15].

During grinding, steel balls are added to the ore so that when the mill rotates the rotary action causes the balls to grind the ore to the required particle size. This mixture of balls and ore is known as the charge inside the mill. The amount of charge inside a mill is known as the percentage fill. A cross section of a mill is shown in figure 1.1. There are two types of motion that take place inside a rotating mill. These are cataracting motion (A in figure 1.1) and cascading motion (B in figure 1.1). Cascading motion occurs when the balls roll over one another and do not leave the bulk of the charge or 'en Masse' region (area below B in figure1.1). Cataracting motion applies to those particles that have left the 'en Masse' region.

The angle at which the charge leaves the liner is known as the angle of departure. This angle is dependent on the speed of the mill. The faster the mill rotates, the higher the angle of departure. The mill speed can be increased until the charge no longer leaves the liner but sticks to it. This is known as centrifuging. The speed at which this occurs is known as the critical speed of the mill.

Nates [1] investigated the motion of a single steel rectangular particle in a mill with a smooth liner and developed a theoretical model to this end. The aim of this work was to contribute to the understanding of which parameters affect the motion of the charge inside a mill. The experimental work conducted by Nates [1] concurred with the results obtained from the numerical model developed. Von Bentheim [2] continued by developing a numerical method that modelled the motion of a single particle in a mill with a corrugated liner. Milner [3] proceeded, formulating a model for the single particle in a mill with arbitrary liners. The model allowed for the wear of the liner to be taken into account. Experimental work was conducted by Milner [3] with various percentages fill.

This thesis aims to evaluate various modelling software packages by simulating the works of Nates [1] and Von Bentheim [2]. The aim is to investigate the suitability of these codes for the use in comminution. The numerical results obtained from these software packages will be compared to the experimental results. The software packages that will be used are ELFEN, ABAQUS and Particle Flow Code (PFC^{2D}). ELFEN has relatively long runtimes. For example, a single particle simulation can take 7 to 8 hours to complete. Therefore experimental work by Milner[3] was not simulated.

Chapter 2 presents a background of the work conducted about ball paths and trajectories. The works of Nates [1], Von Bentheim [2] and Milner [3] in particular are reviewed. The numerical method, the discrete element method together with its application to comminution and specifically ball mills is then discussed.

Chapter 3 gives a description of the software packages used.

Chapter 4 presents the results obtained form the numerical packages. These results are compared to the experimental work of Nates [1] and Von Bentheim [2] for each of the liners used.

Chapter 5 will discuss the observations made regarding the numerical results.

Chapter 6 will present the conclusions made regarding each software package

and recommendations for future work.

In summary: This thesis aims to assess the potential use of ELFEN, ABAQUS and PFC^{2D} in comminution. The experimental work that is simulated was conducted by Nates [1] and Von Bentheim [2]. The simulations gave an opportunity to investigate the parameters that influence the motion of a single particle.

Chapter 2

Literature Review

A background of the studies conducted to better understand charge motion is presented in this literature survey. The dissertations of Nates [1], Von Bentheim [2] and Milner [3] are also reviewed.

The numerical method used when modelling the comminution process is primarily the discrete element method (DEM). The theory regarding this method will be presented. Experimental work conducted with the aim of verifying DEM is reviewed.

2.1 Ball Paths and trajectories

In 1905 White [16] proposed the first mathematical model to predict the path of balls in a mill. Experimental work was conducted to validate the model. Davis [17] extended White's [16] work in 1919 and derived expressions for the motion of a single ball. Davis [17] and White [16] both assumed that the particle moved without slippage. Davis [17] also presented an expression for the critical velocity of a mill. It was observed from experimental work conducted by Davis [17] that the balls inside a mill packed in distinct layers. This assumption was incorporated into the theoretical model.

In 1922 Haultain and Dyer [18] tried to reproduce the results Davis [17] achieved theoretically by conducting experimental investigations. The behaviour predicted by the Davis model could not be reproduced experimentally. Haultain and Dyer [18] found that the charge in the mill did not centrifuge at the velocity predicted by the Davis model. This discrepancy was attributed to slippage that occurs between

the charge and the liner. This highlighted a shortcoming in Davis' model. Haultain and Dyer [18] also noted from experimental work that a certain amount of radial segregation occurred within the charge.

Gow, Campbell and Coghill [19] proposed a different model to that of Davis in 1929. The proposed model suggested that a ball does not follow a distinct path within the charge because of bunching and pushing from other balls in the charge. The model proposed that balls will move further than predicted by the Davis model. Anomalies arose in experimental work conducted which was attributed to the interaction of the balls with the mesh screen at the ends of their test mill. This work was reviewed by Fahrenwald and Lee [20] in 1931. Fahrenwald and Lee [20] suggested that the results obtained were an exaggeration and that this was due to the jamming of the charge inside the test mill. Fahrenwald and Lee [20] also suggested that parabolic trajectories were a closer description of what occurs inside an operating mill. An expression for the angle of departure which included the effect of friction on the motion of the ball was derived.

In 1958 Rose and Sullivan [21] addressed the assumptions used in the Davis model. An alternative approach which results in a similar model was proposed. The resulting model incorporated friction and was extended to predict the shape of the charge. The charge itself was assumed to have no slip.

A theoretical model for the motion of a single ball was derived by McIvor [22] in 1983. This model included the effect of friction and the angle of the lifter bar. A conclusion was made that the trajectories of the balls in the mill are independent of the diameter of the mill. This conclusion implied that the optimal grinding conditions in a mill are only influenced by the mill operating speed. It was also concluded that the paths of the balls were sensitive to the lifter face-angle. This implied that the trajectories of the ball charge would change as the liner and the lifters began to wear.

Vermeulen, Olsen De Fine and Schakowski [23] conducted experimental work in 1984 using piezoelectric sensors. These sensors were placed in the liner of an operating mill. In this way it was possible to measure the angle where the charge left the liner as well as where it impacted at the toe of the charge. The experimental results obtained by Vermeulen *et al* [23] did not correlate to the values predicted by the Davis model. Vermeulen *et al* [23] proceeded to develop a theoretical model

to predict the ball paths using the same assumptions as Davis [17]. The difference with this model was that it included the effect of the adhesion forces within the charge. The predicted values obtained from this model showed a good correlation to the experimental work. Vermeulen [24] continued by including the effect of lifter bars in the model. Experimental work was conducted using lifter bars that had a 90°angle with the mill surface. These results were in close agreement with the predicted values.

In 1988 Powell [25] investigated the effect different lifter bars have on the motion of the charge in a rotary mill. Experimental work was performed to compare to theoretical values. Powell [25] discovered that the angle of departure of the charge inside a mill increases as the height of the lifter bar is increased. The angle of departure stops increasing when the height of the lifter bar is about the same as the radius of a grinding-element. By increasing the lifter-bar face-angle, there is an increase in the angle of departure and a decrease in slip. It was also found that a linear relationship exists between the mill speed and the angle of impact.

Morrell [26] conducted experimental work with a laboratory mill to determine the effect changes in speed and percentage fill had on the motion of the charge in 1992. It was found that with an increase of percentage fill, the toe of the charge also increased. This angle was not affected by the speed of the mill. The shoulder of the charge was found to increase as the percentage fill and the speed of the mill increased. The power model developed by Morrell [26] incorporates the motion of the charge.

In 1996 Powell and Nurick ([27],[28],[29]) investigated the motion of particles within the charge. Experimental work was conducted using an innovative tracking technique. This technique involved the use of a bi-planar angioscope, usually reserved for the treatment of heart patients. The angioscope was used to record the motion of charge inside a perspex mill. The aim of the experiments was not only to study the motion of the charge, but also to determine the effect of the liner configuration and mill speed on the charge. Powell and Nurick ([28],[29]) found among other things, that the profile of the charge changes as the speed of the mill varies. The energy consumption of the mill can be significantly reduced if the correct liner is used.

The experimental technique used by Powell and Nurick [28] is currently being

used by Govender, Powell and Nurick [30] to gather data that can be used to verify the discrete element method. Much of the experimental investigations into the motion of charge and the parameters that affect it, have been done with the aim of verifying DEM. This will be discussed in section 2.5.

Much work conducted with regards to the motion of charge in a mill since 1992 was done in conjunction with the simulations of those experiments using DEM. These works will be presented in the section that reviews the DEM method and its use in comminution.

2.2 The behaviour of a single particle on the inside of a rotating cylinder: Theoretical and Experimental work (Nates [1])

Nates [1] embarked on an investigation into the parameters that affect the motion of single particle inside a mill with a smooth liner. A theoretical model was formulated. A numerical code was developed to implement the theoretical model. Nates [1] conducted experimental work to compare to the numerical results obtained.

A single particle has three types of motion. When a particle is restricted to two dimensions there is only one degree of freedom if no rolling is assumed. This translates into the motion of a rectangular block. Nates [1] decided that this was suitable as in reality balls that are removed from operating mills are found to be cubic in shape because of the grinding and wear experienced inside the mill.

The forces acting on a rectangular particle are shown in figure 2.1. A frictional force T is assumed to exist between the block and the liner in this model. It is taken as being tangential to the mill. The friction is assumed to move in the direction of the mill initially until it is exceeded and sliding occurs. It then moves in the opposite direction of the mill. The equations for the model are derived by summing all the radial and tangential forces that act on the block.

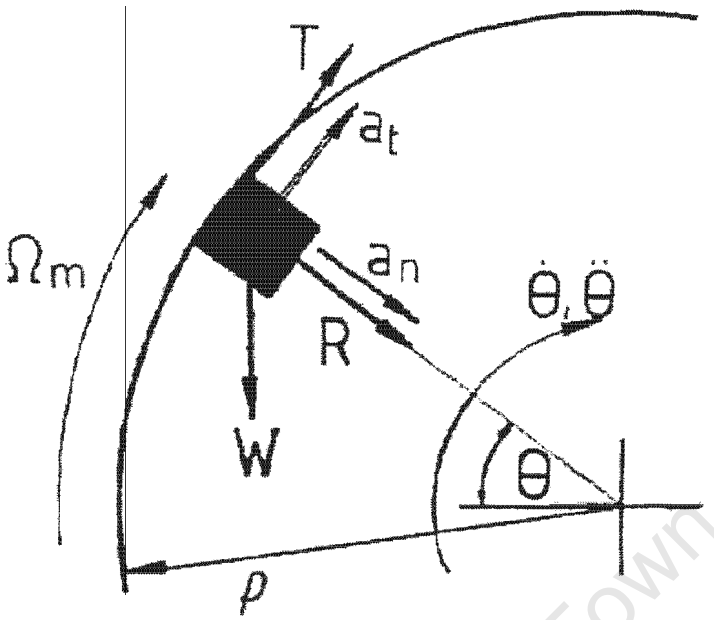


Figure 2.1: Orientation of the forces acting on a block [1]

This results in the following equations:

$$R = m\,\dot{\theta}^2\,\rho - mg\sin\theta \tag{2.1}$$

$$T = m\,\ddot{\theta}\,\rho + mg\cos\theta \tag{2.2}$$

where R = Normal Reaction on Liner (N)
T = Tangential Frictional Force (N)
θ = Angular Position of a Particle on the liner (degrees)
m = Mass (Kg)
ρ = Mill Radius (m)

There are three unknowns in equations 2.1 and 2.2. These are (θ, R, T). Another equation is necessary to solve this system. To achieve this an initial condition must be set. Nates [1] stated that before the particle starts to slip, it sticks to the mill

liner. Thus there is no acceleration acting on the particle. This led to the assumption that the velocity of the particle is equal to the velocity of the mill .

Thus for no sliding:

$$\dot{\theta} = \Omega_m \tag{2.3}$$

where $\dot{\theta}$= the angular velocity of the particle (rad/s)
Ω_m = mill velocity (rad/s)

Integrating equation 2.3 results in:

$$\theta = \Omega_m t + \beta \tag{2.4}$$

where t = time (seconds)
β = constant

Substituting equations 2.3 and 2.4 into equations 2.1 and 2.2, results in

$$R = m\Omega_m^2 \rho - mg \sin(\Omega_m t + \beta) \tag{2.5}$$

$$T = m\Omega_m \rho + mg \cos(\Omega_m t + \beta) \tag{2.6}$$

Once the particle starts sliding, the Tangential Friction Force (T) is proportional to the Reaction Force (R). That is $T = \mu R$. This results is a second order, non-linear, homogenous, ordinary differential equation.

$$\ddot{\theta}\,\rho - \mu_k \dot{\theta}^2 \rho + g(\mu_k \sin\theta + \cos\theta) = 0 \tag{2.7}$$

The high level of non-linearity of the equation necessitates the use of a numerical solution. The Euler Forward Step Approximation method was found by Nates [1] to be the best choice.

Nates [1] implemented the theoretical model using a Fortran code. As the problem was not a self terminating one, end conditions need to be set. These were found to be as follows:

1. The particle does not gain enough energy to stick to the liner and thus slides back down.
2. The particle leaves the liner in a parabolic trajectory.
3. The particle centrifuges.
4. The particle tumbles down the liner.

Relating to the third end condition, Nates[1] found that there are two types of centrifuging. The first is slipping centrifuging where the particle's angular velocity is less than the mill's velocity. The second being total centrifuging where the particle's angular velocity is equal to the mill's velocity.

From the numerical results it was found that the Davis model [17] was inadequate. The speed for centrifuging that was predicted by Davis[17] was too low. It was found that the block's motion was affected by changes in friction and mill velocity. When the friction increased the particle moved further up the mill. If the particle moves above the horizontal it produced greater trajectories across the mill when friction was increased. The same behaviour was noted when the mill speed was changed.

Nates[1] conducted experimental work to compare to the numerical results. The work concentrated on the sliding model. An axially mounted drum was used with a diameter of 404mm and a depth of 100mm. The bottom of the drum was labelled -90°. This is illustrated in figure 2.2. The equation used to determine the critical speed was $N_c = \frac{29.9}{\sqrt{\rho}} rpm$ where ρ is the diameter, derived by Davis [17]. This gave a critical speed of 66.7rpm used by Nates[1]. Speeds will be referred to as percentages of this critical speed. Experiments were conducted for speeds ranging from 30% to 165% of the critical speed.

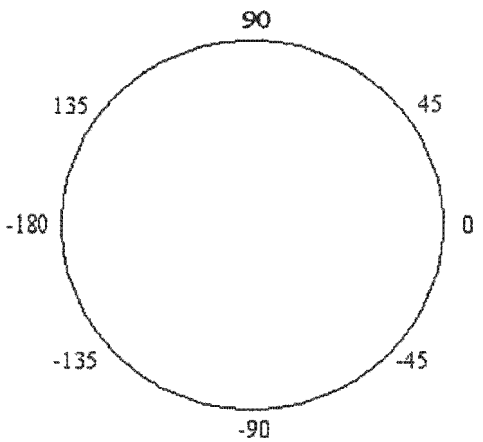


Figure 2.2: Labelling of the drum by Nates [1]

Rectangular blocks of various masses were used to determine whether results were independent of the mass of the particle. This was confirmed. The remaining experiments to find angles of departure were conducted with a block of dimensions 15mm by 15mm. In order to vary the friction, two different surfaces were tested, a steel drum and a cloth covered drum. A video camera and recorder were used to record the motion of the particle. The resulting frames were used to obtain results.

The initial conditions assumed were that the block started at -90°and that the initial velocity of the particle was equal to that of the mill. That is $\theta(0) = -90\,^{\circ}$ and $\dot{\theta}\,(0) = \Omega_m$. It was later found by Nates [1] that for the particle to have a velocity equal to the mill at -90°, the particle had to be released at approximately -135°. At higher speeds it also had to be released further from the drum's surface.

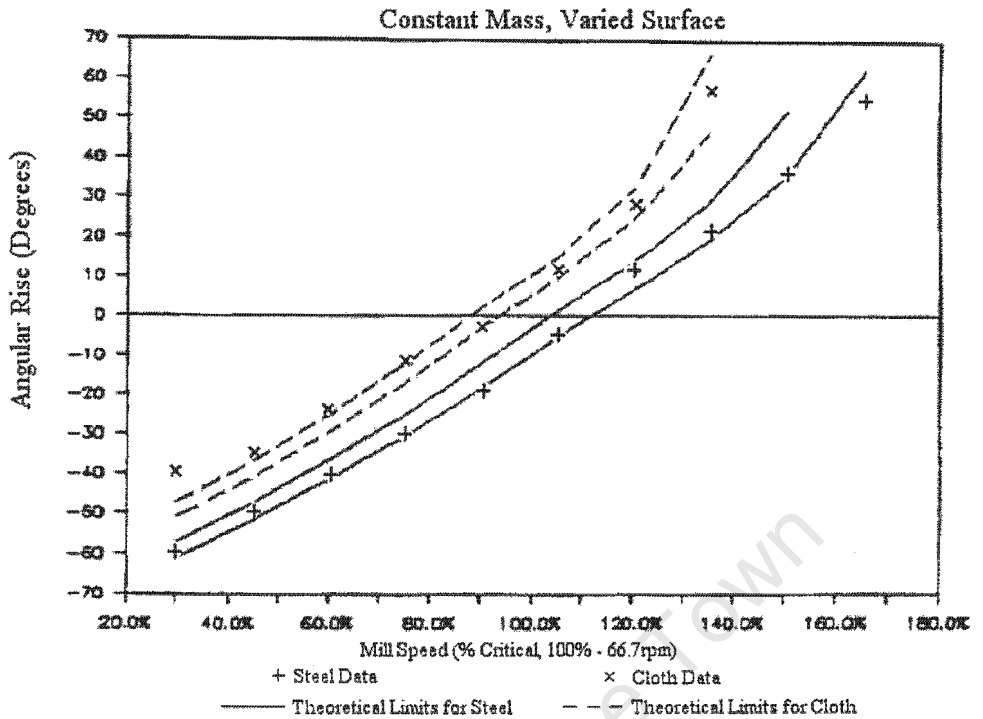


Figure 2.3: Comaprison of theoretical and experimental results by Nates [1]

The angle of departure was taken to be the highest point reached by the particle if it did not move higher than the horizontal (i.e. 0°). If the particle did move beyond the horizontal, the angle of departure was taken to be where it leaves the liner. From the experimental results, Nates [1] obtained an upper and lower bound angle of departure for each speed. A mean value was also calculated for each speed. A comparison between the theoretical predictions and the experimental results is shown in figure 2.3. The experimental values illustrated are the mean values obtained. The theoretical formulation used an upper bound and lower bound for the coefficient of friction which was found experimentally. In the figure, the top curves correspond to the upper bound of the coefficient of friction and the bottom curve to the lower bound. The results obtained experimentally correlated well with the numerical values obtained using the model developed by Nates [1].

2.3 The behaviour of a single particle on a corrugated liner inside a rotating cylinder : Theoretical and Experimental work (von Bentheim [2])

Von Bentheim [2] extended Nates' [1] work to include different liner configurations. The first was a corrugated liner and the second a Skorupa liner. The equation of the corrugated liner is as follows

$$f(\beta) = \rho + c_1 \cos(\zeta_1 \beta) \tag{2.8}$$

where $\beta =$ Relative Angular Position of Block
$\rho =$ mean radius of liner
$c_1 =$ Amplitude of Corrugations
$\zeta_1 =$ Frequency of Corrugations

A corrugated liner with an amplitude of 3mm and 16 corrugations is shown in figure 2.4. A single corrugation is also illustrated.

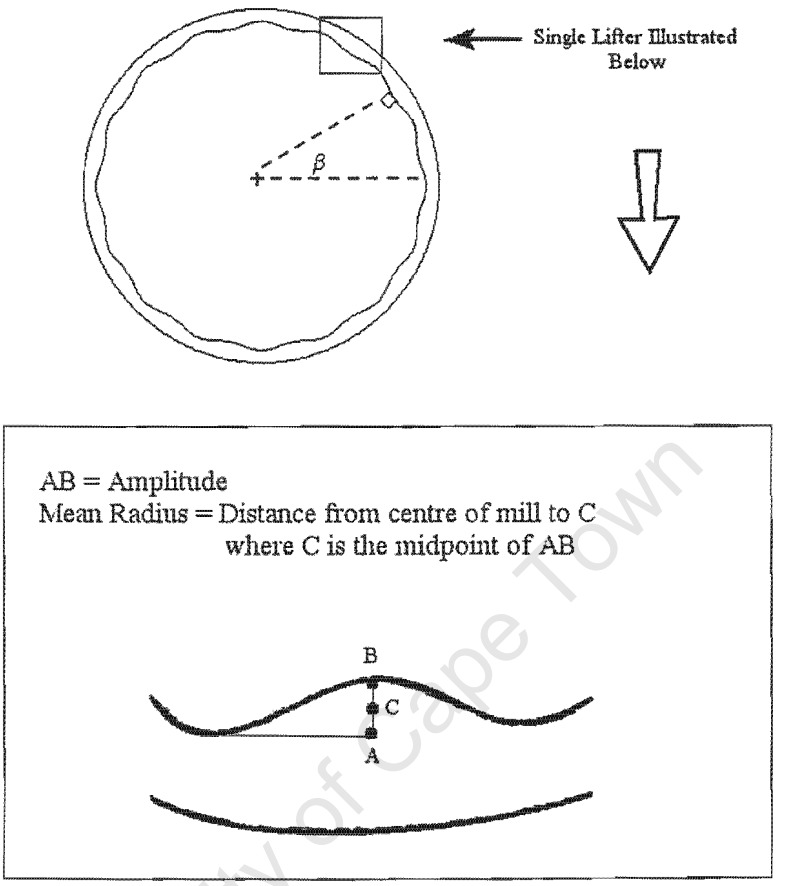


Figure 2.4: Corrugated liner with a liner amplitude of 3mm

The Skorupa liner is a liner configuration that was found to be the most wear resistant by Skorupa [31]. The liner was initially modelled using a Fourier Series Approximation. This was found to be inadequate and a series of straight lines were used to define the liner shape. Von Bentheim [2] only modelled one segment of the Skorupa liner in the theoretical model.

The set of lines to make up one segment or lifter is as follows:

$$
\begin{aligned}
&\text{Section 1:} && 0^\circ \leq \beta \leq 5.5^\circ \\
& && f(\beta) = 0.1925 \\
&\text{Section 2:} && 5.5^\circ \leq \beta \leq 9^\circ \\
& && f(\beta) = -0.21773\beta - 0.2141 \\
&\text{Section 3:} && 9^\circ \leq \beta \leq 13.5^\circ \\
& && f(\beta) = 0.1792 \\
&\text{Section 4:} && 13.5^\circ \leq \beta \leq 21.5^\circ \\
& && f(\beta) = 0.09524\beta + 0.34379 \\
&\text{Section 5:} && 21.5^\circ \leq \beta \leq 22.5^\circ \\
& && f(\beta) = 0.1925
\end{aligned}
$$

An illustration of one lifter is shown in figure 2.5. An indication of where the segment fits in the whole Skorupa liner is also illustrated.

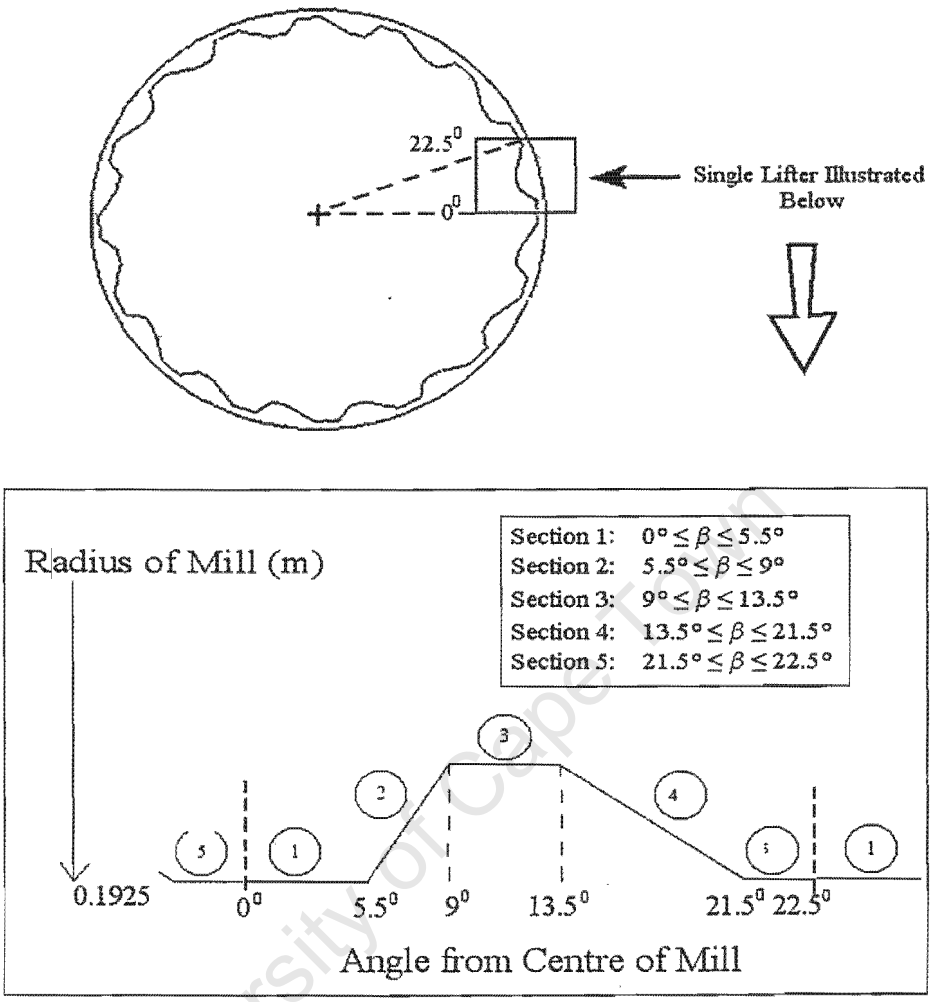


Figure 2.5: The complete Skorupa Liner with an illustration of a Single lifter

Von Bentheim[2] derived the following equations that describe the motion of a particle on a corrugated liner.

$$B\ddot{\theta} + C\dot{\theta} + D\dot{\theta}^2 + E = 0 \tag{2.9}$$

where $B = \frac{1}{\alpha^2}$

$C = -2\Omega_m \{f''(\beta)[f'(\beta) + \mu f(\beta)] - f'(\beta)[f(\beta) - \mu f'(\beta)]\}$

$D = 2f'(\beta)\{f(\beta) - \mu f'(\beta)\} + \{f'(\beta) + \mu f(\beta)\}\{f''(\beta) - f(\beta)\}$

$E = g\{[f(\beta) - \mu f'(\beta)]\cos\theta + [f'(\beta) + \mu f(\beta)]\sin\theta\} + \Omega_m^2 f''(\beta)[f'(\beta) + \mu f(\beta)]$

θ = Angular position of particle to liner (rad)

β = Angular position of particle relative to its starting position

$\alpha = \frac{1}{\sqrt{f(\beta)^2 + f'(\beta)^2}}$

Equation 2.9 had to be solved numerically because of its non-linear nature. Von Bentheim[2] found that the fourth order Runge-Kutta method to be the most efficient computationally. The initial conditions used were the block's initial position and its initial angular velocity, i.e. $\theta = -90°$ and $\dot{\theta} = \Omega_m$.

The numerical results showed that the motion of the particle was affected by the coefficient of friction, the amplitude of the liner and the speed of the mill. It was found that as the friction and speed were increased, the particle's departure angle increased. An increase in liner amplitude also results in an increase in the angle of departure.

Von Bentheim[2] conducted experimental work to compare to the theory developed. A mill with a diameter of 405mm and a depth of 100m was used. The critical speed of the mill was 70rpm. The liners were made of Polyvinyl Chloride (PVC) plastic. Two corrugated liners were used. These liners had amplitudes of 3mm and 8mm. In order to investigate the changes in friction, a bare PVC liner and a cloth covered one were used. A rectangular steel particle with dimensions 16mm by 16mm was used. The initial conditions assumed for the block were that it started at the bottom of the mill and its initial velocity was equal to the mill's velocity.

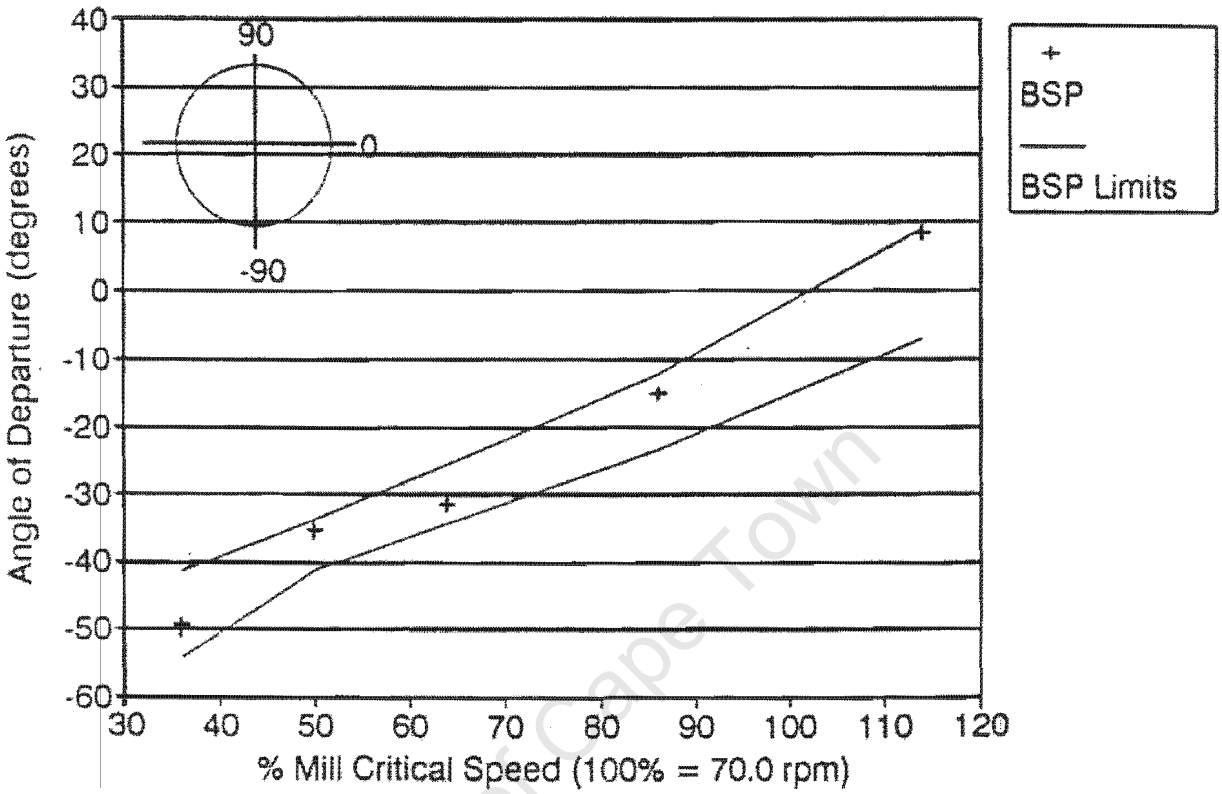


Figure 2.6: The angle of departure vs the mill speed for a steel particle on the corrugated liner

Figure 2.6 shows the angles of departure versus the mill speed for the corrugated liner. The steel block is referred to as BSP in the diagram. The upper and lower bounds found using the theoretical model developed by Von Bentheim [2] are referred to as BSP limits. The angles of departure for the steel block on the Skorupa is shown in figure 2.7. The experimental results correlated satisfactorily with the numerical results obtained using the theoretical model of Von Bentheim[2]

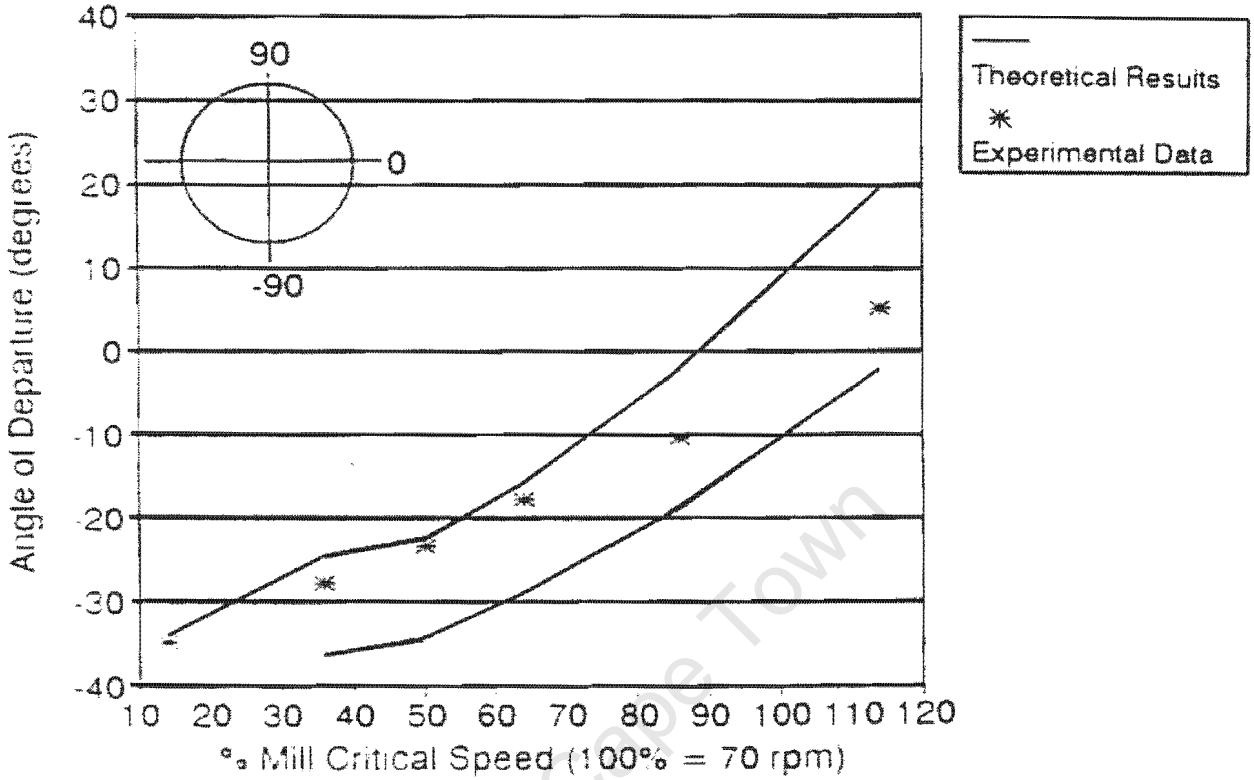


Figure 2.7: Tha angle of departure vs the mill speed for the steel particle on the Skorupa liner [2]

2.4 The prediction of outermost trajectory of media in a grinding mill for the lifter bars with rounded or worn profiles (Milner[3])

Milner[3] developed a theoretical model for the motion of a particle on a liner with arbitrary lifter bars. The model included the effect of mill speed and friction on the particle's outer trajectory. Experiments were conducted to compare to the numerical results.

The model developed predicted the outer trajectories of a particle in lifter bar profiles that had a parabolic shape. The shape of the lifter bar followed the equation $y = Ax^2 + C$ where A and C are constants that can be changed to obtained the desired lifter bar. An example of this is shown in figure 2.8 with A=-70 and C=0.01.

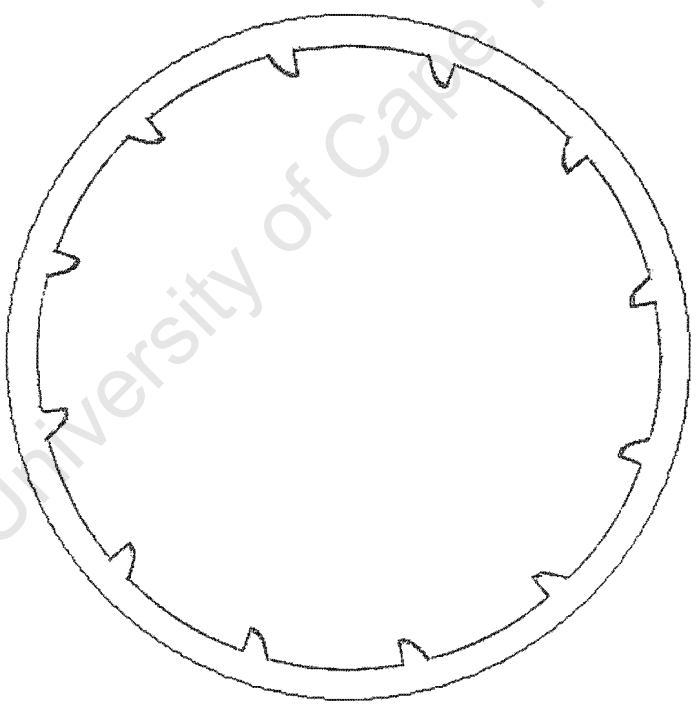


Figure 2.8: Liner with lifter bars have the form $y = Ax^2 + C$, where A=-70 and C=0.01

The motion of the single ball is divided into various stages. As the mill rotates, the particle moves with the liner until an equilibrium point is reached where the tangential forces sum to zero. This state of equilibrium of the particle is described by equation 2.10.

$$g\sin\gamma - \mu(g\cos\gamma + \Omega^2(R-a)\sin\alpha_0) - \Omega^2(R-a)\cos\alpha_0 = 0 \tag{2.10}$$

g = gravitational acceleration ($9.8m/s^2$)
γ = angular position of lifter tangent at contact point from the horizontal (rad)
μ = friction coefficient
Ω = mill rotational speed (rad/s)
R = mill radius (m)
a = ball radius (m)
β = angular position of ball fron the horizontal (rad)
α_0 = $\gamma_0 - \beta_0$ (the lowerscript relates to where the rotation angle of the mill is zero)

Once the particle has reached equilibrium, it will start to slide down the liner of the mill. Milner [3] found the equation governing this sliding motion to be :

$$\ddot{S} = g\sin\gamma_t - \Omega^2 r_t\cos\alpha_t - \mu_k(g\cos\gamma_t + \Omega^2 r_t\sin\alpha_t + \frac{\dot{S}}{\rho} \tag{2.11}$$

$$\dot{S} = \frac{\dot{x}}{\cos(\gamma_t - \Omega t)} \tag{2.12}$$

$S, \dot{S}, \ddot{S}$ = displacement, velocity and acceleration due to the sliding motion
t = time dependent
r = distance from the mill centre to the ball's centre (m)
ρ = instantaneous radius of rotation of the ball (m)

At some point after sliding, the particle will leave the mill's liner. This will happen if the particle reaches the tip of the parabolic lifter and falls off or if the

normal force acting on the ball is zero. The parabolic trajectory that results is given by equations 2.13 and 2.14.

$$X = X_{td} + V_{xd}t_p \tag{2.13}$$

$$Y = Y_{td} + V_{yd}t_p - \frac{1}{2}gt^2 \tag{2.14}$$

(X, Y)	= global coordinates
V	= velocity
t_p	= time form departure
d	= relates to the contact point between ball and lifter

Milner[3] proceeded to conduct experimental work using a mill with a diameter of 250mm and a width of 110mm. Six sets of twelve lifter bars with the form $y = Ax^2 + C$ were used. Metal balls with diameters of 5mm, 5.5mm and 6mm were used. The mill was filled with approximately 1%, 10% and 40% of the balls.

The experimental work showed a satisfactory correlation to the predictions made by the model developed for the angle of departure and the trajectories of the outer layer of balls inside the mill. The following conclusions were made with respect to the angles of departure of the charge:

- The angles of departure were found to decrease the flatter the lifter became, i.e. as the amount of wear of the lifter bar increases. The worn lifter bars were shown to be more efficient than new ones under certain operating conditions.
- The percentage fill also affected the angle of departure. An increased percentage fill gave rise to an increased pressure in the charge. This led to an increase in the angle of departure.
- The lifter height had an effect on the trajectory of the charge. This correlated to findings by Powell[25] and Vermeulen et al [23]. Figure 2.9 shows that for various lifter heights, the departure angle reaches a maximum value and then starts to decrease. Note that the lower the value of $|A|$ is, the higher the lifter.

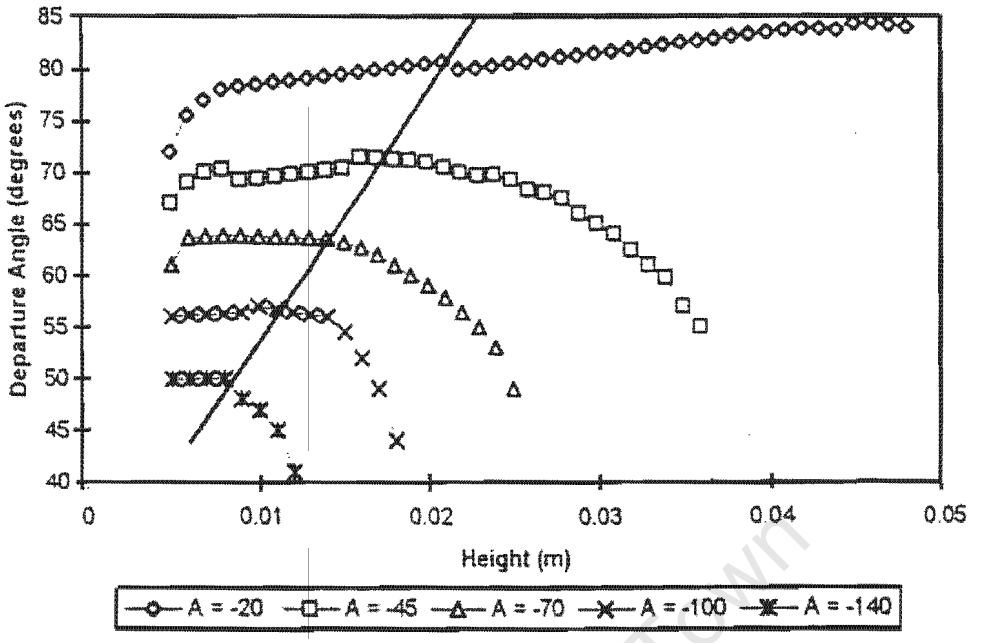


Figure 2.9: The angle of departure vs the lifter height [3]

- The angles of departure were affected by the static coefficient of friction. This relationship is illustrated in figure 2.10. As the friction increased, so did the angle of departure.
- It was found that as the size of the balls increase, the angles of departure decreased.

Milner[3] proposed that lifter bars that had the form of functions higher than the second order be incorporated into the theoretical model. This, together with the effect of charge pressure was thought would present more realistic trajectories. It was concluded that the biggest problem would be to determine a charge shape that would relate to different operating conditions.

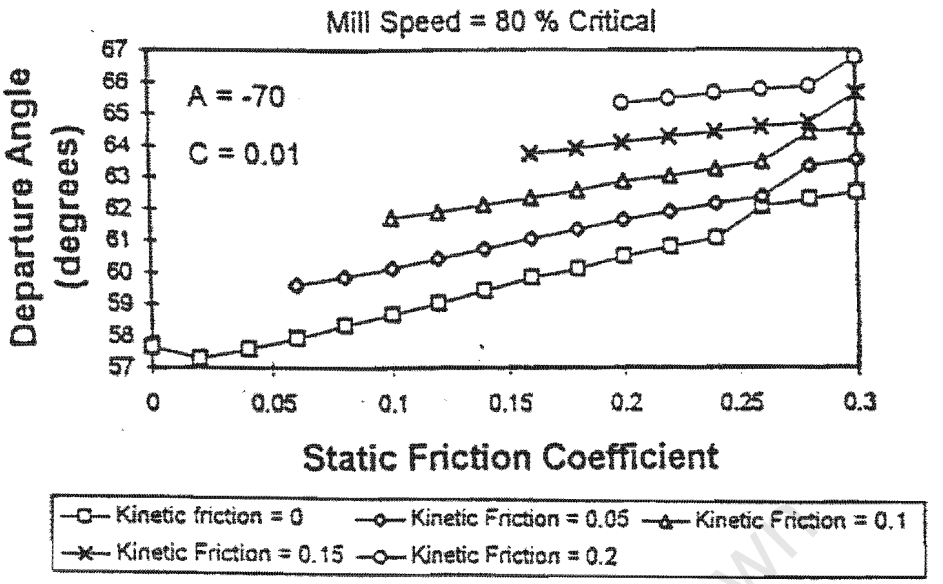


Figure 2.10: The coefficient of friction vs the angle of departure [3]

2.5 Numerical modelling applied to Comminution

The discrete element method (DEM) was first presented by Cundall and Strack [12] in 1979 as a means of modelling problems relating to granular mediums. Mishra and Rajamani[5] applied this method to comminution and specifically ball mills. One of the motivations for using this method was to be able to predict the power draw of the ball mill, the distribution of the collision forces inside the mill and finally the wear rate of the liner.

DEM simulates the behaviour of distinct, interacting bodies. In the case of ball mills, the balls are treated as discs. DEM is based on the notion that a time step can be chosen so that the forces that occur between two discs upon contact, do not propagate beyond the neighbours of those discs[12]. The time step chosen by the algorithm is therefore very small. As a result, a problem can have lengthy run times.

In the DEM algorithm, there are two applications performed during each time step. The first is the force displacement law used to determine the contact forces between discs that are in collision. The second application is Newton's second law to give a description of the motion of the disc.

The equations to describe the motion of a disc are as follows:

$$m\ddot{x}_i + C_i\dot{x} + K_i x_i = F_i \qquad i = 1, 2 \tag{2.15}$$

$$I\ddot{\theta} + \sum_{i=1}^{2}(K_i x_i + C_i \dot{x}_i)s_i = M \tag{2.16}$$

where x_i = position of disc i
m = mass of disc i
C_i = damping constant of disc i
K_i = material stiffness of disc i
F_i = the force acting in direction i
I = the moment of inertia
s_i = the perpendicular line from the line of action of the force to the centroid of the disc
M =the moment about the centroid of the disc

The contact between discs is illustrated in figure 2.11. The disc i is shown to be in contact with the other discs in the assembly. Contact between particles is modelled as a pair of normal and tangential spring dashpots. The spring and dashpot parameters are obtained from the material properties that are determined by the user. The spring component is obtained from the stiffness of the material and the dashpot from the coefficient of restitution. At each step of the analysis, the equations are integrated using a finite-difference approximation method.

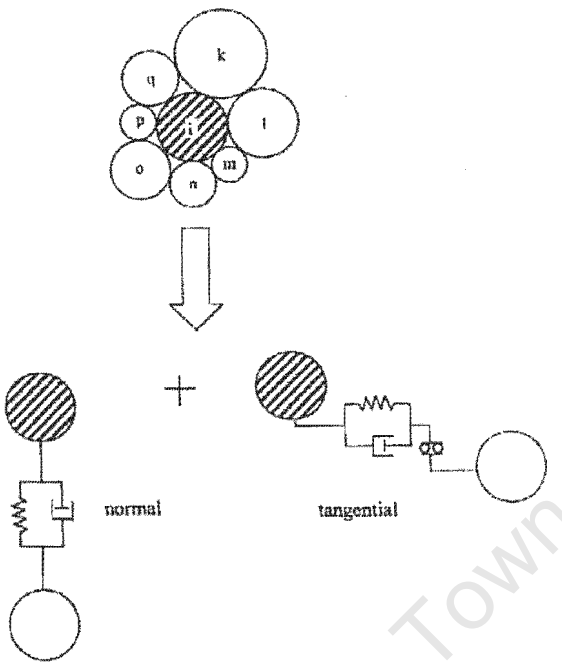


Figure 2.11: Representation of contact between discs [4]

The derivatives are approximated to be:

$$\ddot{x} = \frac{\dot{x}\,(t+\delta t) - \dot{x}\,(t)}{\delta t} \tag{2.17}$$

$$\dot{x} = \frac{\dot{x}\,(t+\delta t) + \dot{x}\,(t)}{2} \tag{2.18}$$

Datta, Mishra and Rajamani[32] outlined the following as the major computational tasks of the DEM algorithm in each timestep.

a) Each disc is stored having a radius, mass, moment of inertia, and collision properties. The contact between a disc and its neighbour is then established. This takes up the bulk of computational time.

b) The contact forces between the elements are calculated using the contact deformation equation together with the contact properties of the elements.

c) The contact forces are summed to determine the total force acting on the disc.

d) The acceleration is calculated from the force. This is followed by calculating the velocity and displacement by integrating the acceleration.

e) Once all the above is completed the position of the ball and wall elements are updated.

Mishra and Rajamani[5] were the first to simulate the motion of charge inside a ball mill using DEM. The input parameters of the model used were the material properties of the bodies involved in the motion. The results obtained from the numerical simulation were verified from experimental work conducted by Mishra and Rajamani[5]. The experiments conducted used a 550mm diameter mill with square and rectangular liner configurations. Eighteen lifters were used in each case. The charge was comprised of 12mm, 18mm and 25mm radii balls. The mill was run at a critical speed of 57.8rpm with a ball filling of 38%. The results compared were the torque and power draw of the mill. Mishra and Rajamani[5] found an overall agreement between the experimental and simulated values as can be seen in figure 2.12. The data illustrated are for two different liner configurations. It was found that when the correct material properties were inputted into the numerical model, the values for the power draw compared with the experimental values. This highlighted the importance of accurate input parameters. These parameters relate to the material stiffness, damping and friction values.

In later work, Mishra and Rajamani [33] tracked two particles within the ball charge of a 300mm diameter mill. The motion of the particles was simulated using a DEM based numerical code. The experimental and numerical trajectories correlated well. Further experiments were conducted using a 250mm diameter mill with 25.4mm diameter discs. The mill was divided into sections with circular plates. This configuration allowed Mishra and Rajamani [33] to create a two dimensional situation. The power draft was accurately predicted by the DEM code. The code was used to find the energy distribution between collisions. The authors proposed that this, together with the breakage properties of particles, a distribution of the size of the particles at the end of milling can be predicted.

The DEM code developed by Mishra and Rajamani [33] was used to simulate the behaviour inside of industrial size mills [6]. The aim was to illustrate that DEM

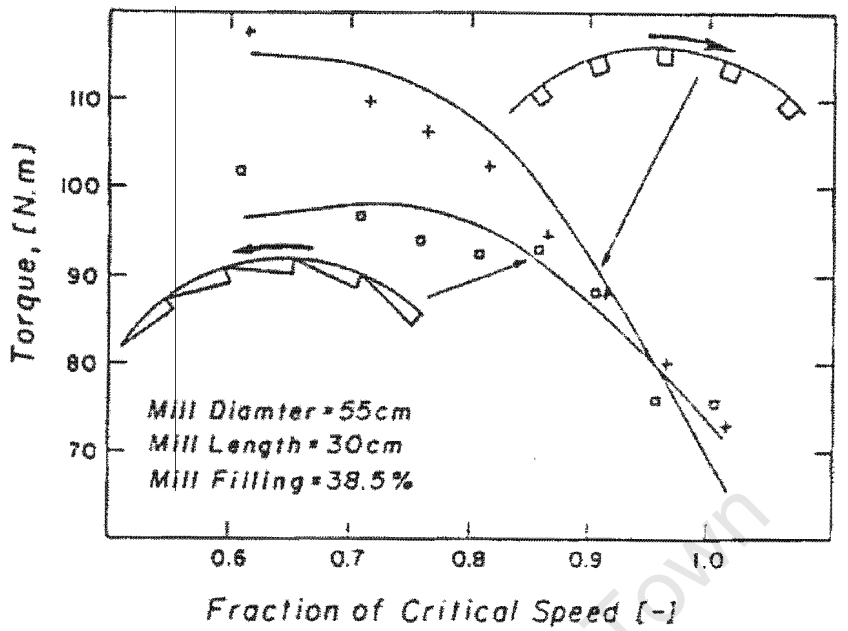


Figure 2.12: Comparison of experimentally and numerically determined torque [5]

is capable of simulating a multi-body system like the motion of charge inside a ball mill accurately. It was found that at high speeds the larger particles within the ball charge moved to the centre. At lower speeds these particles moved to the shell of the mill. The friction value used affected the motion of the charge and hence the power draft of the mill. For a mill with no lifter bars, the motion of the charge was highly dependent on the friction between the balls and the shell of the mill. For a mill with lifter bars, a high friction value resulted in more particles cataracting [6]. This is illustrated in figure 2.13. It shows two mills rotating at 70% of critical speed with the same number of particles. The mill on the left has a coefficient of friction of 0.7 and the one on the right has a value of 0.2.

Cleary[13] simulated the motion of charge inside a 5m ball mill. The mill, with 23 lifter bars, was rotated at various percentages of a critical speed of 19.5 rpm. The sensitivity of the power draw of the mill to changes in the material properties of the particles inside was investigated. A charge of only steel balls was first investigated and then a mixture of rocks and steel balls was considered. The balls ranged from 50mm to 20mm in diameter. The rock ranged from 5mm to 50mm in diameter. In the mixed charge there was 10 times more rock than balls. The power draw

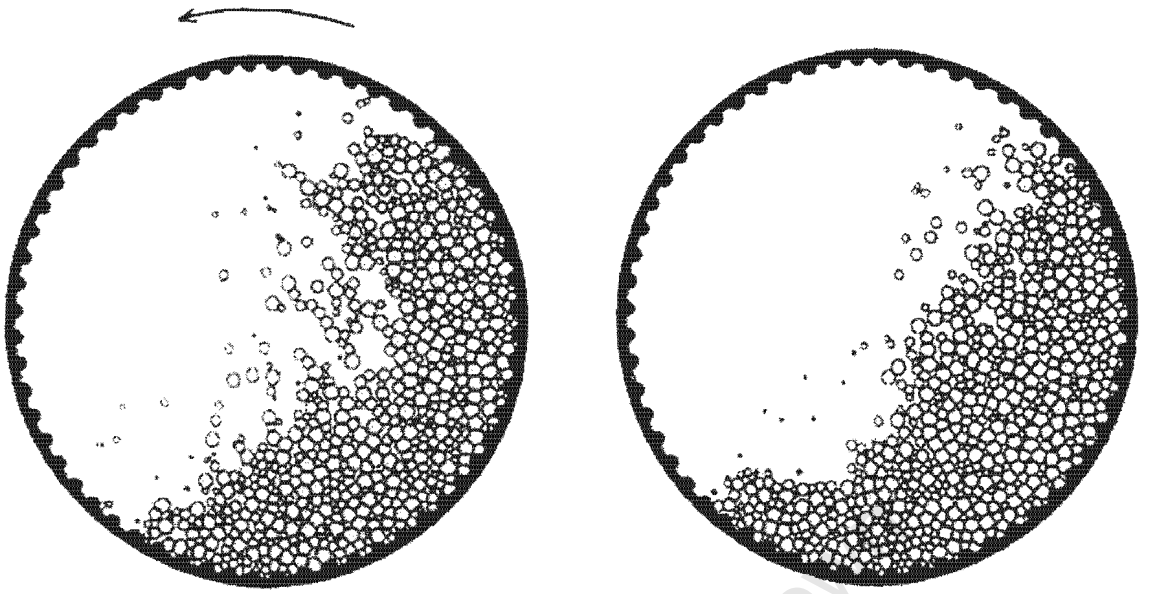


Figure 2.13: Comparison of a variation in the coefficient of friction at 70% critical speed. The mill on the left has a coefficient friction of 0.7 and on the right a value of 0.2 [6]

was found to be insensitive to changes in material properties for speeds less than 100% of critical speed. For speeds greater than 100% of critical, the power draw increased when friction was decreased. Cleary[13] found that including non-circular particles did not change the power draw dramatically. Cleary[13] concluded that further validation of DEM relating to ball mills was necessary.

Cleary and Hoyer [14] compared the of motion of charge using DEM to experimental work. High speed photographs were captured of a 150mm diameter mill with a depth of 150 mm. The mill charge comprised of particle with mean size of 3mm. The speed of the mill was 695rpm. The experimental charge profiles closely matched the numerically simulated ones.

Radziszewski [34] compared three different approaches to simulating the motion of charge. The first is to track a single particle within the charge. The second is as Mishra and Rajamani [33] did and that is to track each particle within the ball charge. The final is to group the charge together using a discretisation scheme. Each method is used to simulate the motion of charge in a 12m diameter mill. The power draw is predicted using each method. There are advantages and disadvantages to

using each of these methods. Radziszewski [34] found that each of the methods results in similar charge motions. The single particle method was found to predict the outer trajectory of the charge accurately. The latter two methods predict power fairly accurately.

Rajamani, Mishra, Venugopal and Datta [4] also formulated 2D and 3D DEM models. Simulation results were compared to experimental work conducted. In the model a linear spring, dashpot contact model was used. For the 2D experiments a 254mm diameter by 292mm long mill. The mill was partitioned and filled with 25.4mm diameter discs. This is to simulate the two dimensional situation. It was found that the 2D DEM model satisfactorily predicted the motion of the charge as well as the power draw of the mill. Further experiments were conducted to determine the effect a change of mill dimensions would have on the predictor power of the DEM model. The mills varied from 240mm to 4.11m in diameter. There was still a good correlation between experimental and numerically calculated power. This led to the conclusion that this DEM model for 2D was an adequate predictor of power regardless of the mill size. The 3D model proved to be even more accurate. The numerical predictions matched the experimental values more closely than the 2D results.

Van Nierop, Glover, Hinde and Moys [15] stated that it was difficult to verify DEM as the numerical method needs some degree of accuracy with regards to the input values used. This is a challenge as the conditions in a mill changes over time. Van Nierop et al conducted experiments with a laboratory mill to measure power and observe the charge motion. The mill was constructed to simulate the two dimension. The dimensions were a diameter of 550mm and length of 23mm. The ends of the mill were covered with glass plates. Ball bearings with diameter 22.24 mm were used as charge. The percentage fill and the mill speed were varied.

Two dimensional and three dimensional DEM codes were used to simulate the experimental work. The 3D code more accurately predicted the charge motion as was also found by Rajamani *et al* [4]. Unlike the 2D code, the 3D code was able to account for the length of the mill as well as the interaction between the glass plates and the mill charge. The numerical code was able to accurately predict the power of the mill as well as the centrifuging speed. Van Nierop *et al* [15] cautioned that the material properties used needed to be accurate.

Zhang and Whiten[35] addressed the contact model used in DEM. A linear spring and dashpot is commonly used to determine the contact forces between discs. It was found by Zhang and Whiten[35] that when damping is non-zero, the initial force of the particle is large. This was found to be contrary to experimental work previously conducted. An alternative to the linear spring and dashpot was suggested. Zhang and Whiten [35] concluded by reiterating the importance of knowing the nature of the forces between the surfaces involved. These forces are complex and the simple assumptions that are commonly used are not sufficient to accurately model the ball mill problem.

In recent years there has been much work done to verify the discrete element method. This numerical method has the potential to give greater insight into the motion of charge inside a ball mill. In order for DEM to be used in industry with confidence, it is necessary to validate the method and codes based on this method thoroughly. The ultimate goal would be that a user could run simulations using the numerical code without having in depth knowledge regarding the theory of the method.

Currently there is other work being conducted which simulates mills with numerous particles [36]. Govender, Balden, Powell and Nurick [36] have conducted experimental and numerical work using a perspex mill with a diameter of 142mm. The particles used were plastic balls that were rotated for more than 250 circulations. However, in order to fully understand the variables that affect the motion of a particle when using the different software packages, it is necessary to understand the motion of a single particle. It was with this goal in mind that this research was initiated.

Chapter 3

Numerical Analysis

The aim of this project was to test various numerical codes to determine whether they would be suitable for use in comminution. The results from these simulations were compared to experimental work done with a single particle on various liners. The software packages investigated are ELFEN, ABAQUS and Particle Flow Code (PFC^{2D}).

ELFEN became available with the aid of industry funding. ABAQUS is currently being used for many other research work at the University of Cape Town. PFC^{2D} became available in December 2000.

3.1 Description of the Model

The experiments simulated were those of Nates [1] and Von Bentheim [2]. The mills had different liner configurations and were rotated with a single rectangular particle inside. The following were the dimensions and material properties used.

The smooth steel liner with a steel rectangular particle as used by Nates [1]:

Block dimensions :	$15mm$ by $15mm$
Diameter of drum :	$404mm$
Material properties of steel :	
Young's Modulus :	$200e9N/m^2$
Poisson's Ratio :	0.3
Density :	$7800kg/m^3$

The corrugated and Skorupa PVC liner with a steel rectangular block as used by Von Bentheim [2]:

Block dimensions :	$16mm$ by $16mm$
Diameter of drum :	$405mm$
Material properties of PVC :	
Young's Modulus :	$3.5e9N/m^2$
Poisson's Ratio :	0.38
Density :	$1400kg/m^3$

Each drum was assumed to be rigid.

3.2 General Introductory Comments

Each software package has essentially three parts. The first part requires the user to input the data regarding the problem that is to be simulated. The second part analyses the problem using a numerical method. The third involves extracting the data required by the user from the completed simulation.

The input section is different for each of the packages being assessed. ELFEN has a user interface where the user simply inputs loading conditions, material properties, etc. For ABAQUS the user is required to write an input deck (refer to Appendix B). PFC^{2D} also requires the user to write an input deck or what is called a data file (refer to Appendix B).

ELFEN was developed over a long period of time as new modelling problems arose. While there is a help manual to assist with the use of the package, there is no information available regarding the theory used.

3.3 ELFEN

ELFEN version 2.8.3b is a finite element numerical modelling package that was developed by Rockfield Software Limited in Swansea, United Kingdom. The package includes the option of simulating discrete problems. These problems are solved using a central difference integration scheme.

Any analysis problem is essentially divided into four steps[37]. The user executes the first which is to examine and define the problem. The second is to input the geometry and process information into a model. This step is associated with the Modelling section in ELFEN. The third step is to perform an analysis of the problem. The fourth and final step is to analyse the results of the analysis. The last two steps are captured as the Analysis and Results sections. The window showing these sections on the computer is shown in figure 3.1.

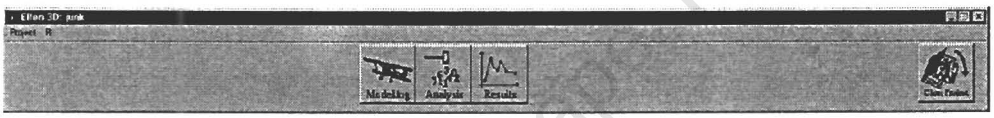


Figure 3.1: ELFEN display showing different sections

3.3.1 Modelling

This section is user interactive where the user enters the information needed for the analysis of a problem. It is divided into geometry, loading conditions, constraints and boundary conditions, material properties, meshing and process control. The window showing the different sections to Modelling is shown in figure 3.2. The loading conditions, constraint and boundary conditions are crucial in determining the geometry definition of the structure. These, together with mesh generation, will be discussed in greater detail in sections 3.3.1a, 3.3.1b and 3.3.1c.

Any problem to be analysed has to be divided into various stages of motion. The ball mill problem was modelled as a single stage problem. The mill is rotated while the particle is dropped into it. The rationale behind this is that as the particle touches the liner there is velocity acting on it. This is to fulfil the assumption made by Nates [1] and Von Bentheim [2] that the velocity of the particle is equal to the

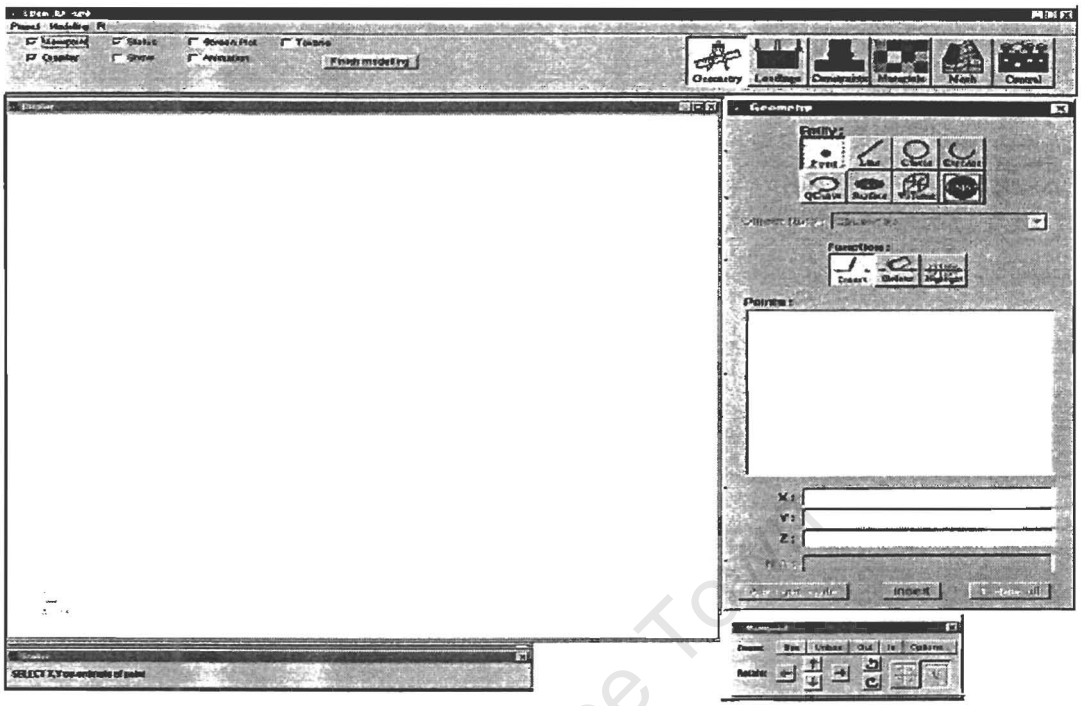


Figure 3.2: The Modelling section of ELFEN

velocity of the drum when the two come into contact for the first time.

a) Loading Conditions

ELFEN allows for mechanical, thermal and fluid loads. The package allows for a problem to be defined under several load cases with different loading functions applied to each case. A loading function is a curve that defines how a load will act on an object over a certain period of time.

In the ball mill problem, mechanical loads are used. Two load cases were identified. These include a load case to apply gravity to the particle and a load to allow the mill to rotate. Under the 'gravity load' case a drop load is implemented. This is constant loading function and is applied to the rectangular particle as a body force. The graph for the drop load is shown in figure 3.3.

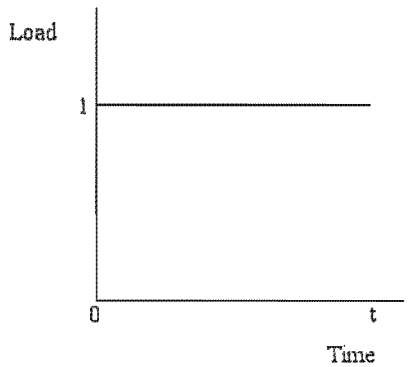


Figure 3.3: The load function used to apply gravity

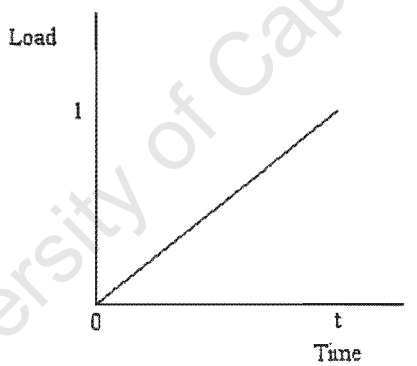


Figure 3.4: The load function used to rotate the mill

The 'rotate load' case is associated to a ramp load. This is a linear loading function that is applied to the drum to simulate a rotational load. The gradient of this linear function is equal to the speed of the drum. The graph for this function is seen in figure 3.4. The load of 1 in both diagrams is a value that equals a 360° revolution. A value of 2 would equal 720°.

b) Constraints and Boundary conditions

ELFEN has the option of computing the behaviour of surfaces in continuum contact or discrete contact. The nature of the ball mill dictates that discrete contact be used. The global properties for the problem have to be defined. These include inputting the normal and tangential penalties as well as the contact damping. Surface properties can be applied to different parts of the model. In the ball mill problem the same friction is applied to the block as well as the drum. The values used are those determined by Nates[1] and Von Bentheim[2] experimentally. The drum was fixed in the x and y direction to allow for rotational movement.

c) Mesh Definition

The mesh used in a problem is critical in the accuracy of the finite element model. A higher mesh density is required in regions where sharp changes are expected. There are two mesh generation types. These are the structured and unstructured types. To use the structured type, the model's domain must be divided into four sided surfaces for two-dimensional mesh generation. This type was used for the smooth liner (fig 3.5). The bulge indicated in the figure was included to keep track of the rotation of the mill when viewing an animation sequence. The unstructured type was applied to the corrugated (fig. 3.6) and the Skorupa liners (fig. 3.7). This was chosen, as it is suited to geometry with polygonal planar surfaces.

It should be noted that different mesh generation types cannot be mixed in a single problem. Linear elements were chosen in both cases. The unstructured mesh generation type uses a mesh generation algorithm.

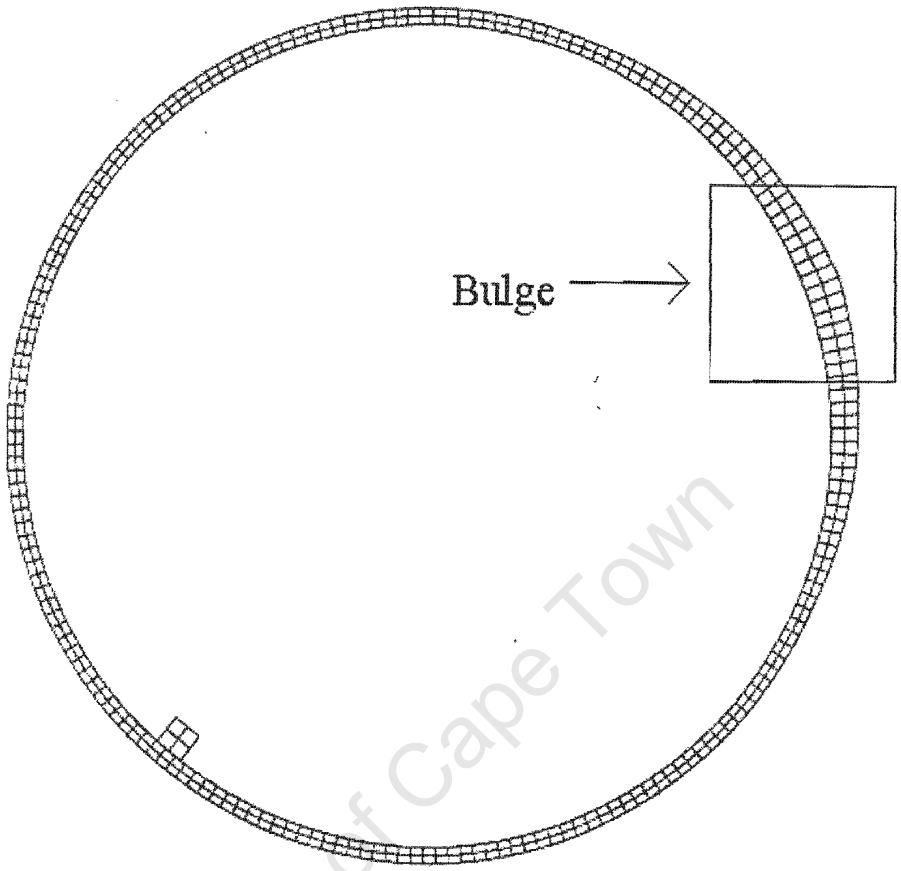


Figure 3.5: Mill with smooth liner with a structured mesh in ELFEN. The structure mesh divides the model's domain into four sided surfaces.

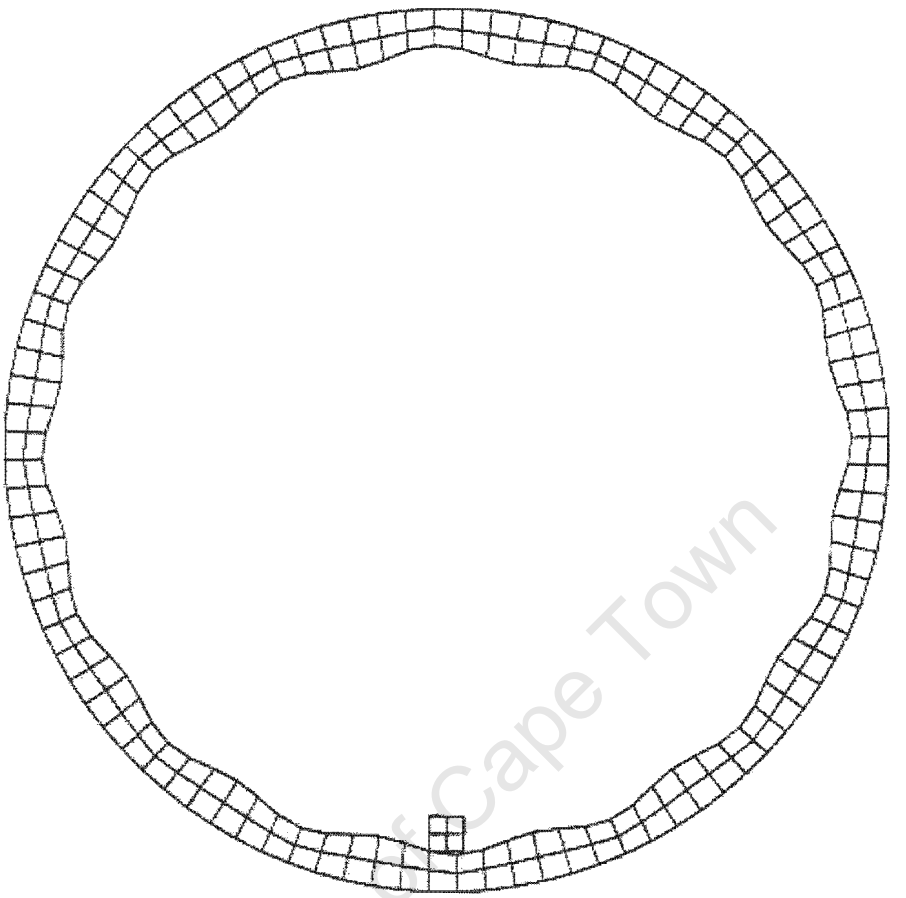


Figure 3.6: Mill with corrugated liner with an unstructured mesh in ELFEN. The unstructured mesh is suitable for polygonal planar surfaces.

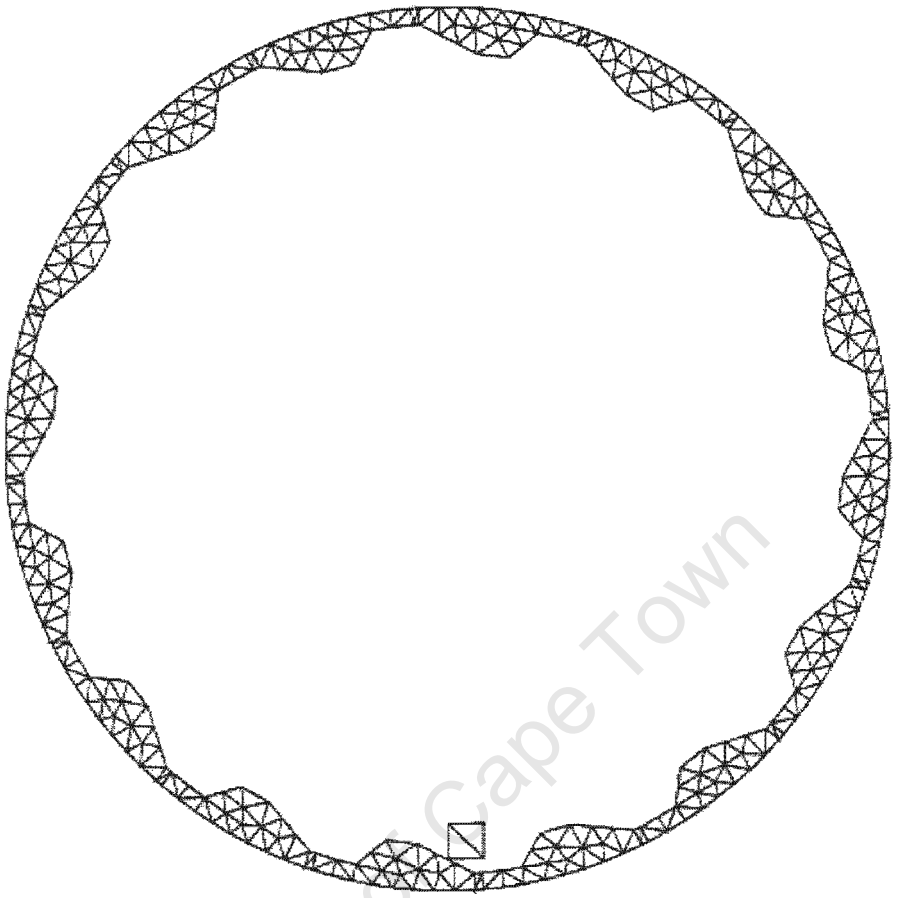


Figure 3.7: Mill with Skorupa liner with an unstructured mesh in ELFEN. The unstructured mesh is suitable for polygonal planar surfaces.

3.3.2 Control

In this section the various load cases and constraints are assigned to predetermined stages of the problem. One is also able to indirectly affect the timestep of the analysis. This is done by changing the factor of the critical step. This value is defined as a factor (f) of the critical timestep Δt_{cr} which is computed automatically by ELFEN. Then Δt_n for timestep n is defined as $\Delta t_n = f * \Delta t_{cr}$.

The type of output required by the user is inputted in this section. For the single particle problem with various liners, the displacement values for the four corner nodes of the particle are saved. These output values are determined at a time interval defined by the user.

3.3.3 Analysis

When the modelling section is complete, all the data is summarised into a single file called the neutral file which is used during the analysis stage. As ELFEN was not initially designed for discrete problems, the neutral file had to be edited before analysis could take place. Rockfield Software Limited provided the required modifications.

A discrete problem can be computationally expensive. At the start of this research, a 233 MHz processor was used to run this problem. The smooth liner problem ran for approximately seven days, completing one revolution. The hardware was then upgraded to a 733 MHz processor and the runtime for the same problem was reduced to eight hours. The introduction of more particles increases the runtime exponentially. A simulation with 36 particles takes a week to complete one rotation at a speed of 60rpm. Therefore because of the long run times, Milner's [3] experimental work was not simulated.

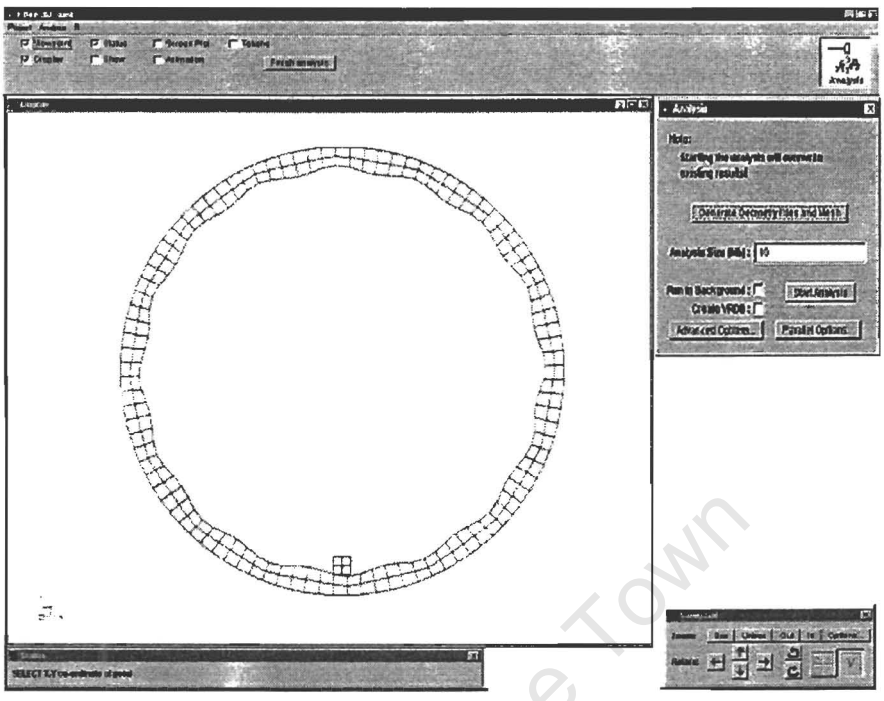


Figure 3.8: The Analysis section in ELFEN

3.4 ABAQUS

ABAQUS version 5.8 is a finite element package developed by Hibbitt, Karlsson and Sorenson, Inc in Rhode Island, USA. The software package requires the user to write an input deck detailing the geometry, material properties, constraints and boundary conditions of the problem (See Appendix B). This input deck is akin to the neutral file produced by ELFEN.

For the problem of the single particle in a ball mill, ABAQUS/Explicit was used. This analysis uses an explicit integration rule together with diagonal element mass matrices [38]. The equations of motion used in the analysis are integrated using an explicit central difference integration rule:

$$\dot{u}^{(i+\frac{1}{2})} = \dot{u}^{(i-\frac{1}{2})} + \frac{\Delta t^{(i+1)} + \Delta t^{(i)}}{2} \ddot{u}^{(i)} \tag{3.1}$$

$$u^{(i+1)} = u^{(i)} + \Delta t^{(i+1)} \dot{u}^{(i+\frac{1}{2})} \tag{3.2}$$

where $\dot{u}$= velocity
$\ddot{u}$= acceleration
i = increment number

The computational efficiency of the explicit procedure is due to the use of diagonal element mass matrices. The accelerations are computed at the beginning of each increment using the mass matrix, the applied load vector and the internal force vector. This procedure also requires no iterations as well as no tangent stiffness matrix.

The explicit procedure, like the discrete element method, uses many small time increments. The following condition must be satisfied for the central difference scheme used to remain conditionally stable :

$$\Delta t \leq \frac{2}{\omega_{\max}} \tag{3.3}$$

where $\omega_{\max}$ = the highest eigenvalue in the system

A small amount of damping is introduced to control high frequency oscillations. The time increment is given as

$$\Delta t \leq \frac{2}{\omega_{\max}}(\sqrt{1+\xi^2} - \xi) \tag{3.4}$$

where ξ = the fraction of critical damping in the highest mode

The determination of the time step used in the analysis is fully automated and does not require user intervention.

There are two contact models that are available in ABAQUS/Explicit. These are the kinematic contact scheme and the penalty contact scheme. The kinematic contact scheme being the default model. This scheme is a predictor/corrector algorithm and does not affect the stable time increment. During each time step of the analysis ABAQUS/Explicit first predicts the behaviour of the model without considering the contact between entities. The simulation proceeds to determine which surfaces are penetrating one another. The depth of penetration, the mass of the body associated with it and the time increment are used to calculate the force required to prevent penetration. This is the force that would have been used in the time step if contact had been taken into account.

The penalty contact algorithm is a weaker algorithm than the kinematic contact scheme in that it is less strict in the enforcement of the contact constraints. It does however allow for the treatment of more general types of contact. This scheme introduces more stiffness into the model. As a result, the stable time increment is affected.

Material damping can also be introduced to the system. This can be in the form of a mass proportional damping value or a stiffness proportional damping value. ABAQUS/Explicit provides an option to use Rayleigh damping. This is defined as a linear combination of the mass and the stiffness matrices (equation 3.5).

$$C^{MN} = \alpha M^{MN} + \delta K^{MN} \tag{3.5}$$

α	= mass propotional damping factor
δ	= stiffness propotional damping factor
K^{MN}	= mass matrix
M^{MN}	= stiffness matrix
C^{MN}	= liner combination

The simulations conducted in this research uses the mass proportional damping value. This value ranges from 0 to 1 where 0 is equivalent to no damping.

3.5 Particle Flow Code (PFC^{2D})

PFC^{2D} was developed by the Itasca Consulting Group, Inc in Minnesota, USA to model the movement and interaction of circular particles in two dimensions. A data file is written by the user as is done with ABAQUS (See Appendix C). This file is then used by the package to run an analysis of the problem. The numerical scheme that is used is the discrete element method. The circular particle can be bonded to form arbitrary shapes. The developers of PFC^{2D} propose that the advantages of this package are three fold. The first is that the detection of contact between the circular particles is easier than that of angular particles making it more time efficient. The second is that there is no limit to the magnitude of displacements that can be modelled. Finally as the shapes used in the problem are comprised of circular particles, breakage is possible [39].

The rectangular block used in the single particle simulations, was composed of an array of discs as shown in figure 3.9. The rectangular particle was assumed to be rigid and was therefore made into a clump. PFC^{2D} defines a clump as a group of discs that are bound together in such a manner that the distance between the discs do not change. During analysis the contact forces between the discs are ignored thus allowing the particle to be rigid [39].

An explicit timestepping scheme is used in PFC^{2D}. At each timestep Newton's 2nd law is integrated to obtain the velocities and displacements of the particle.

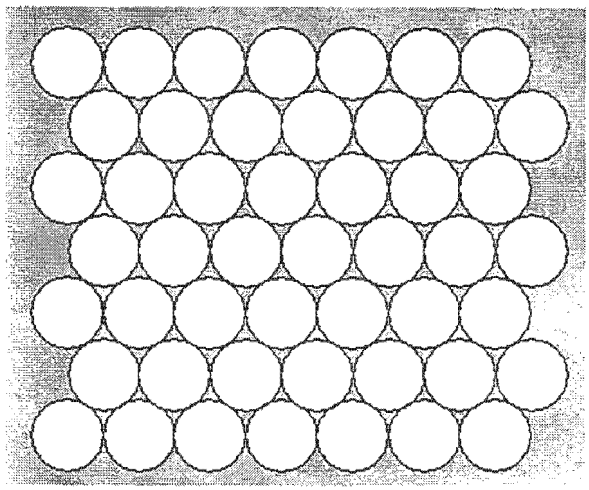


Figure 3.9: Rectangular particle generated by PFC^{2D}

These are calculated using the contact forces present. The new contact forces acting on the particle are calculated from its new position and displacements relative to the other particles present. The actual timestep used in the calculation is taken as a factor of a calculated critical value. This critical timestep is defined as

$$t_{crit} = \left\{ \begin{array}{ll} \sqrt{m/k^{tran}}, & \text{(translational motion)} \\ \sqrt{I/k^{rot}}, & \text{(rotational motion)} \end{array} \right\} \tag{3.6}$$

where m = mass
k^{tran} = translational stiffness
k^{rot} = rotational stiffness
I = the moment of inertia of the paticle

There are two contact models in PFC^{2D}. These are the Hertz-Midlin and linear contact models. The Hertz-Midlin model is only applicable to spheres in contact. Thus the linear contact model was employed for the problem in this thesis. This model is defined by the normal and shear stiffnesses of the two bodies in contact. These values are specified by the user.

Hysteretic damping can be introduced to the model by assigning a damping value to the desired object. In the case of the single particle simulations, that object is the cluster that forms the rectangular particle. When using this type of damping,

the normal stiffness of the particle before contact is greater than the normal stiffness after contact. The hysteretic damping value used must lie in the interval 0.04 and 1. The value of 1 equates to no damping.

3.6 Comparison of the different Software Packages

i) The biggest noticeable difference by the user is the method of inputting the data of the problem to be simulated. Where ELFEN has a user interface, ABAQUS and PFC require the user to write an input or data file.

ii) The type of damping used in each package also differs. ELFEN has a point damping scheme that is applied to surfaces of objects. ABAQUS has a material damping that is used in the Rayleigh damping equation. PFC^{2D} has hysteretic damping.

iii) ELFEN and ABAQUS use an explicit central difference scheme, whereas PFC^{2D} uses the finite-difference approximation method shown in Chapter 2.5.

iv) The contact schemes used in ABAQUS and PFC^{2D} differ. ABAQUS offers a choice between the kinematic contact scheme and the penalty method contact scheme. For PFC^{2D}, a linear contact model was used.

v) The time step used for each package is different. The critical timestep used by ELFEN can be influenced directly by the user by changing f. In ABAQUS, as with PFC^{2D}, the timestep cannot be directly affected by the user.

Chapter 4

Results

This section presents the results obtained from the software packages for each liner configuration. The method of processing the results is explained. The modifications necessary in each package to obtain the results are also discussed. The results are compared to the experimental values obtained by Nates [1] and Von Bentheim [2].

4.1 Method of processing

In the software packages ELFEN and ABAQUS, nodal displacement values can be retrieved from the analysis. The nodal displacements were taken at each corner of the block and the average of the four nodes was taken to be the angle of departure. In PFC^{2D}, displacement values were taken for the clump generated, hence only one value was used and no average was necessary.

The displacements were generated into a file for each software package. The displacement for each node has a x and y coordinate. An example is as follows. Figure 4.1 illustrates positions A, B and C of a single block at different time intervals. Let time at A $= t_A$, B $= t_B$ and C $= t_C$.

The files containing the displacement values for the separate software packages have different formats. This affects the manner in which the files are loaded into the MATLAB code used to process the data. For ELFEN there are two files generated, one for the x coordinates and one for the y coordinates.

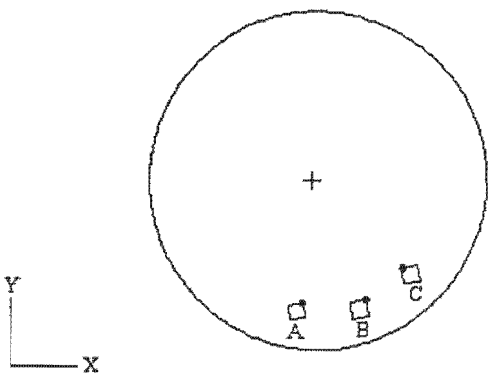


Figure 4.1: A single block at different time intervals

This file is edited to get the information in the following form:

Time	X-coord
t_A	x_A
t_B	x_B
t_C	x_C

for the x values and

Time	Y-coord
t_A	y_A
t_B	y_B
t_C	y_C

for the y values.

The values recorded in the file for ABAQUS are as follows:

Time	X-coord	Y-coord
t_A	x_A	y_A
t_B	x_B	y_B
t_C	x_C	y_C

For PFC^{2D} a single file is generated for the clump. This file has the same format as that generated for ABAQUS:

Time	X-coord	Y-coord
t_A	x_A	y_A
t_B	x_B	y_B
t_C	x_C	y_C

A MATLAB program was coded to compute the angle of departure. The basic steps in the program are shown in figure 4.2.

Step 1 : The file, or files depending on the software package, containing the displacement values is loaded.

Step 2 : The values for time, the x-coordinates and the y-coordinates are inserted into an array.

Step 3 : The values are converted into polar coordinates (r, θ) and inserted into a new array.

Step 4 : The new array is searched for the highest angle, θ_h.

Step 5 : The radius r_h corresponding to θ_h is found. A test is conducted to check whether the particle at angle θ_h is in flight or whether it is still on the liner, i.e. if $r_h \leq$ (mill's radius-longest length of the particle), the particle is in flight.

Step 6a : If the test is confirmed, the particle has already left the liner. Step 4 is repeated.

Step 6b : If the test fails the highest point by the particle before it leaves the liner would have been reached. Save the angle, θ_h. This is the angle of departure for the node.

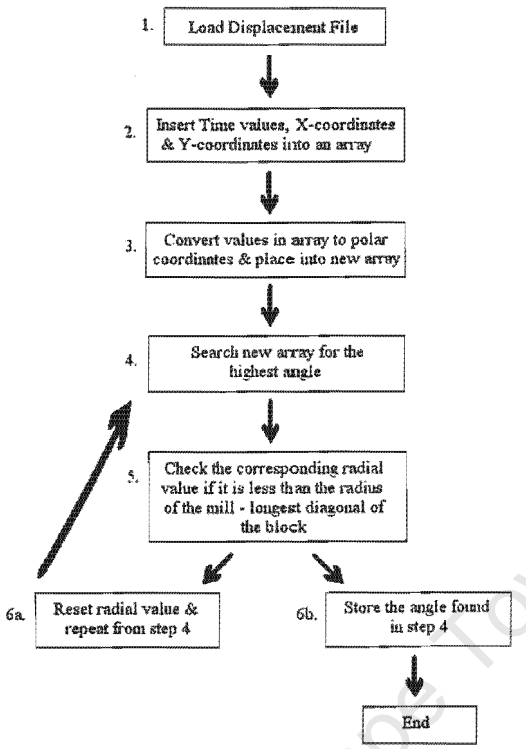


Figure 4.2: The basic steps in the MATLAB code used to process data

4.2 Smooth Liner

Nates [1] conducted experiments with a single steel rectangular block in a mill with a smooth steel liner. The angle of departures were determined from video footage taken during the experiments. When the block moved away from the liner a shadow was cast. The shadow resulted from the lighting conditions during the experiments. The angle was taken when this shadow occurred. In the experimental work the particle was released at -135° (refer to figure 2.2. in Chapter 2). This is so that the particle's velocity is equal to the drum's velocity at -90°. This follows the initial condition assumed in Nates' [1] theoretical work. Nates [1] presented an upper bound and a lower bound for the particle's experimental angle of departure at each speed. The speeds ranged from 30% to 165% of the critical speed.

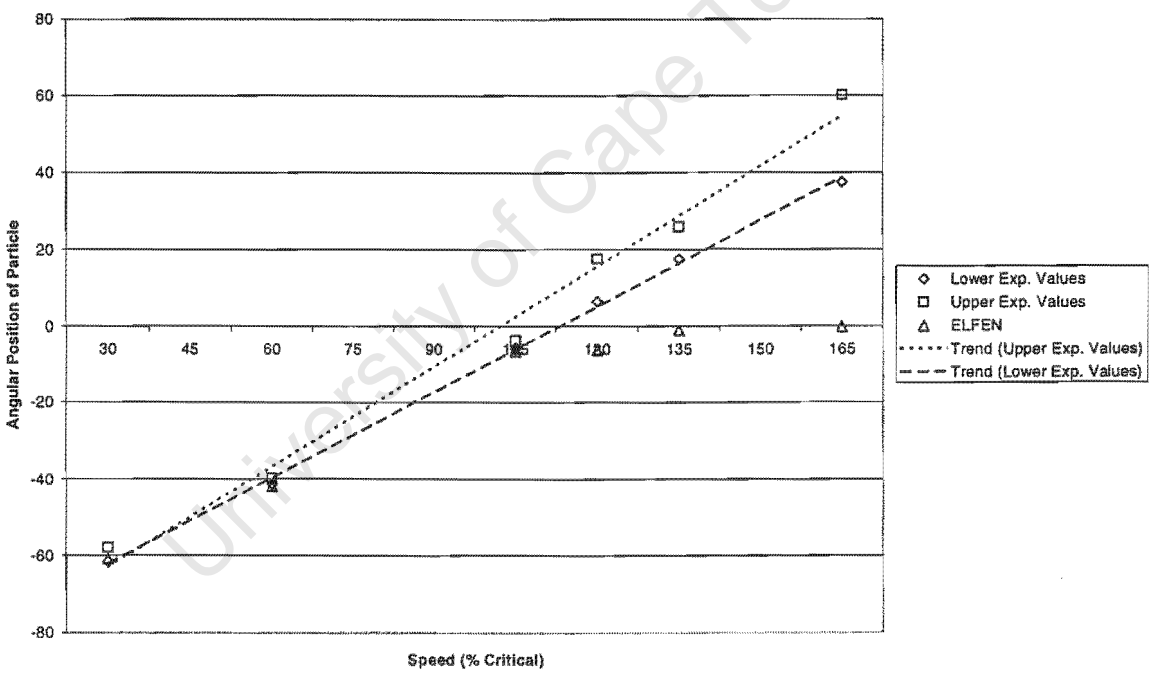


Figure 4.3: Angles of departure in ELFEN for the smooth liner

The results obtained from ELFEN correlate well to the trend of the lower experimental values until 105% critical speed, as can be seen in figure 4.3. For the 120% to 165% critical speed the numerical angles of departure were significantly lower than the experimental values. The values for these speeds do not move higher than 0°.

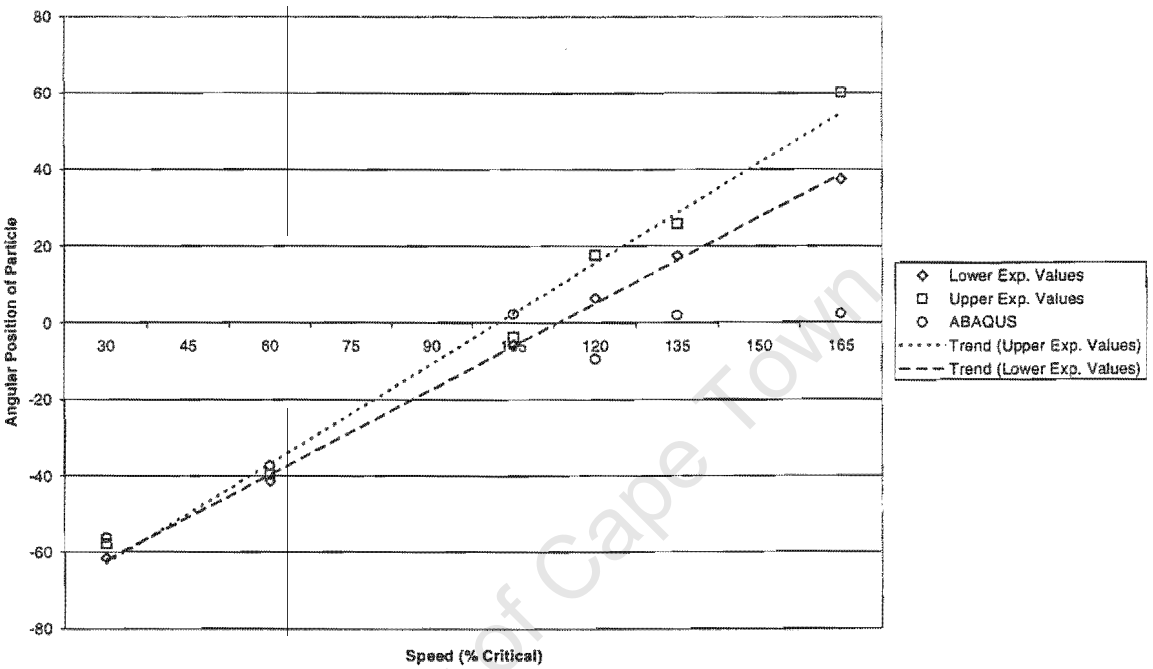


Figure 4.4: Angles of departure in ABAQUS for the smooth liner

The contact algorithm used in ABAQUS was the kinematic contact scheme. The angles of departure obtained follow the trend of the upper experimental values until 105 % critical speed (figure 4.4). The numerical results for speeds higher than 105% diverge from the experimental values. Unlike ELFEN these angles of departure move slightly higher than 0°.

The numerical results for PFC^{2D} correlate to the trend of the lower experimental values until 60% critical speed (figure 4.5). The angles of departure are lower than the experimental values for the speeds 105% to 165% critical speed. In PFC^{2D} it was found that a hysteretic damping value of 0.7 for each speed was necessary to stabilise the movement of the particle. It was seen that when there was no damping applied, the particle rocked in the moving mill.

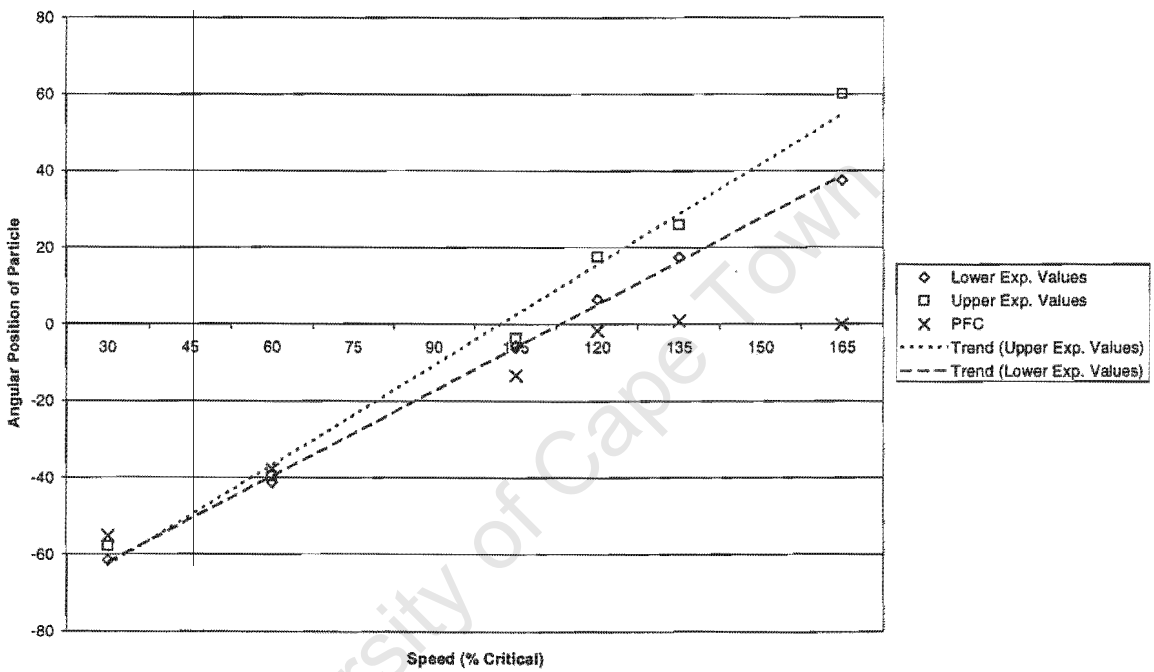


Figure 4.5: Angles of departure in PFC^{2D} for the smooth liner

None of the packages were able to simulate the results for speeds higher than 120% critical speed (figure 4.6). The particle does not seem to have enough energy to move significantly beyond the horizontal or 0°. Instead, the particle rolls back down the liner.

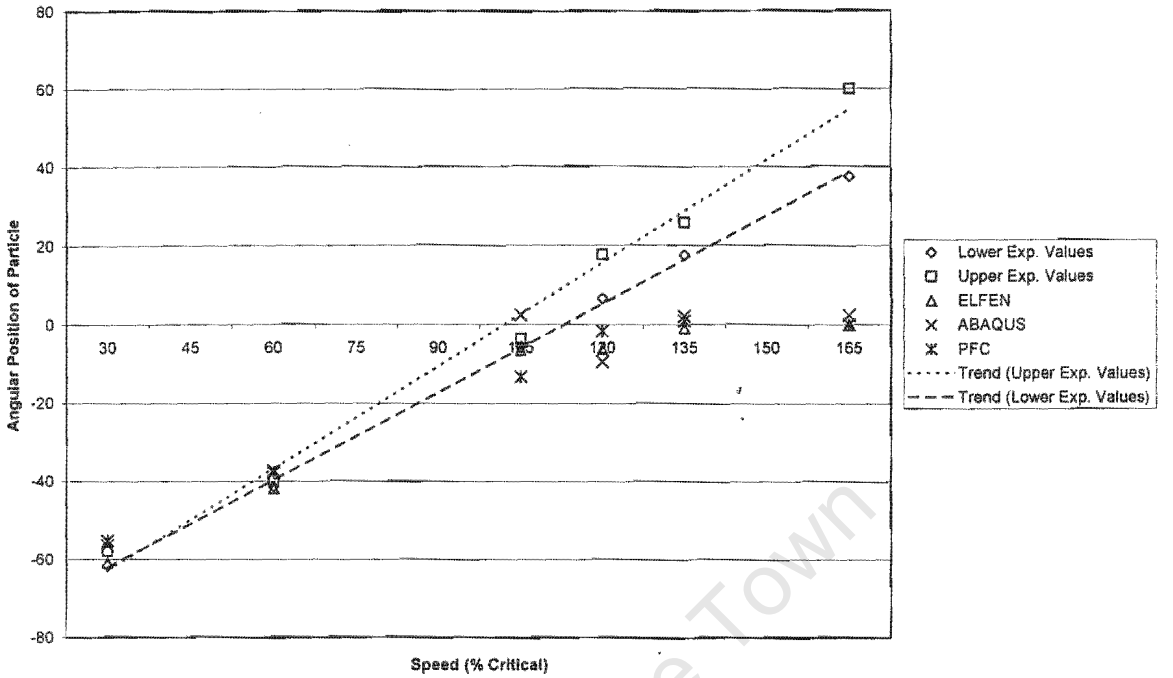


Figure 4.6: Angles of departure for all the software packages for the smooth liner

4.3 Corrugated Liner

Von Bentheim[2] conducted experiments using a steel particle on a corrugated PVC liner with an amplitude of 3mm. The angles of departure were determined in the same fashion as for the smooth liner. The friction value used in all the simulations is 0.337. This value falls in the range that was determined experimentally by Von Bentheim[2].

In ELFEN the normal and tangential penalties were changed from the values used in the smooth liner. This was necessary as the smooth liner is steel and the corrugated liner PVC. The penalties were decreased from $2*10^9 N/m^2$ to $1*10^7 N/m^2$. These changes softened the contact between the particle and the liner. Together with these changes, it was found that the factor of critical step (f) influences the motion of the particle. The effect of a change in f on the angles of departure can be seen in figure 4.7. The simulations for $f = 0.9$ are compared to those for $f = 0.2$. The results for $f = 0.2$ compare better to the experimental trends than do the results for $f = 0.9$. There was no damping applied in these simulations.

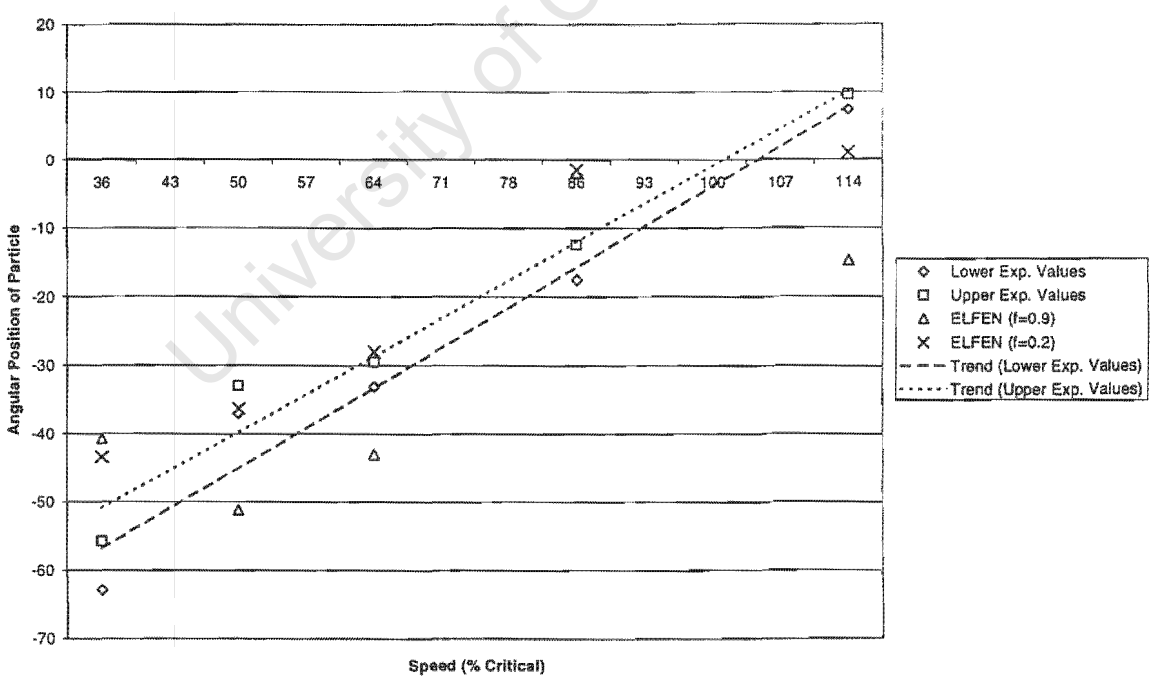


Figure 4.7: Angles of departure in ELFEN with factor of critical steps 0.2 and 0.9 with no damping for the corrugated liner

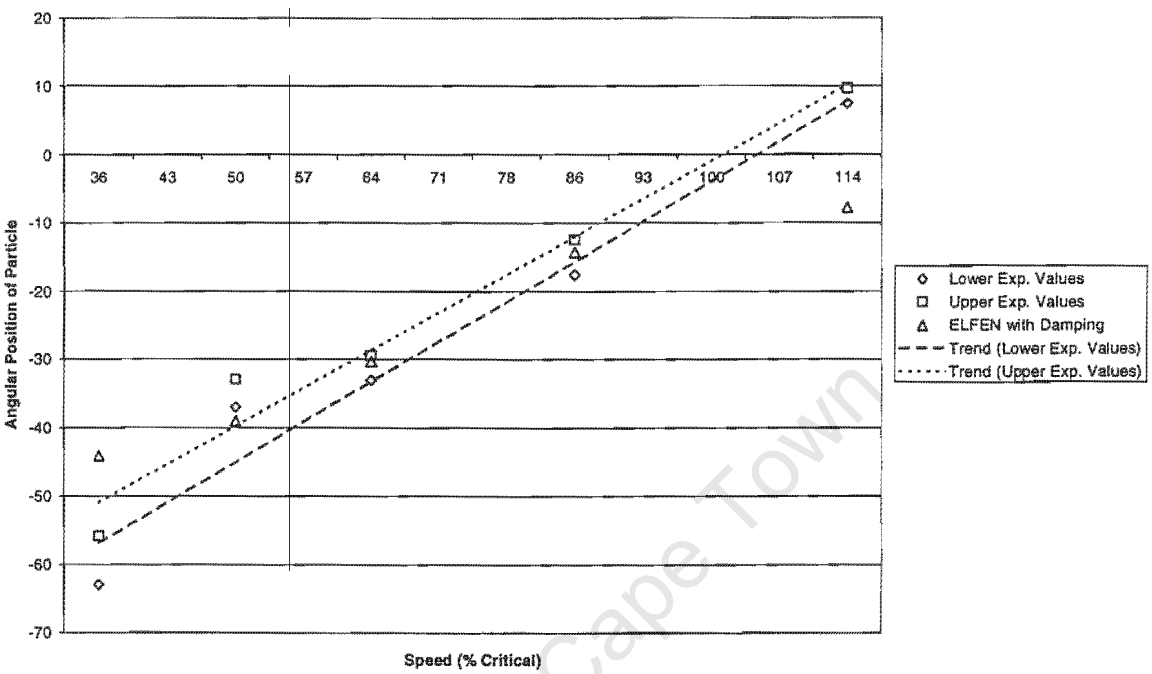


Figure 4.8: Angles of departure in ELFEN with damping and $f = 0.2$ for the corrugated liner

The simulations were continued using $f = 0.2$. As with the simulations for the smooth liner using PFC^{2D}, damping was applied to the particle to stabilise its motion. The angles of departure obtained from the simulations in ELFEN with damping are shown in figure 4.8. From the graph it can be seen that the angle of departure obtained for 36% critical speed is higher than the experimental values. The angles of departure for speeds 50%, 64% and 86% of critical speed are satisfactorily close to the experimental trends. At 114% of critical speed the angle of departure obtained is less than the experimental values. This could also be as a result of the initial condition assumed not being met. That is that the velocity of the particle does not have the same velocity of the drum at $-90\,^{\circ}$ and thus is not able to move beyond $0\,^{\circ}$.

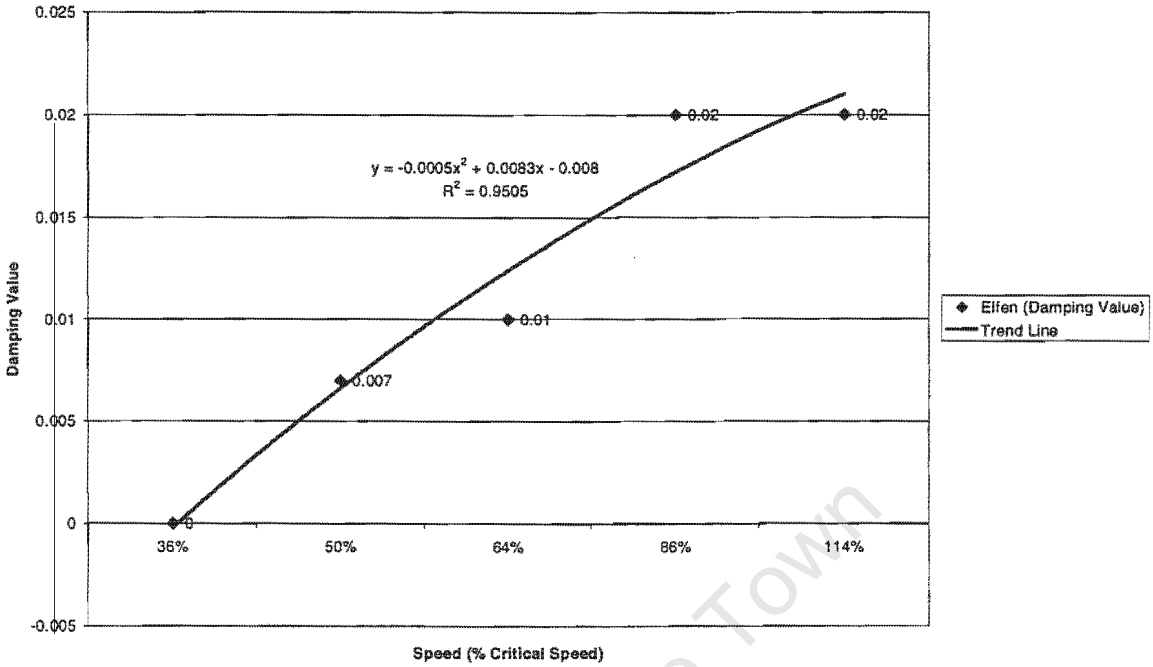


Figure 4.9: The damping values used in ELFEN with a parabolic function fitted to the damping values for the Corrugated Liner

The damping values used are illustrated in figure 4.9. It was found that the values used fit a parabolic function. This function is shown in the figure 4.9 as the trend line. The equation of the parabolic function is also shown. The damping values fit the parabolic function with a 95% correlation.

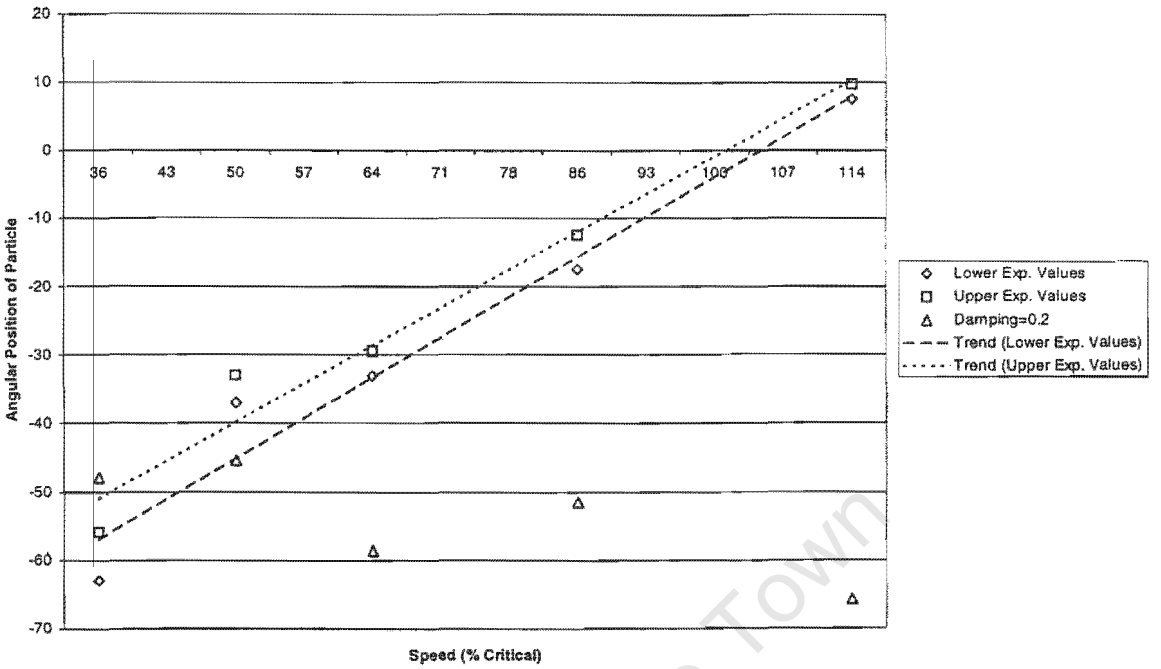


Figure 4.10: Angles of departure in ABAQUS with kinematic contact and damping=0.2 for the corrugated liner

The corrugated liner was simulated in ABAQUS using both the kinematic contact scheme as well as the penalty method. Material damping was applied to the particle at each speed for both contact schemes. A range of damping values were simulated for each type of contact. Figure 4.10 shows the angles of departure obtained for the simulations with kinematic contact and a damping of 0.2. This graph shows that the values for 36% and 50% critical speed follow the trends of the experimental values. The numerical results for 64% to 114% critical speed do not correlate at all to the experimental values. This was found to be true for the other values of damping simulated (refer to Appendix E for figures).

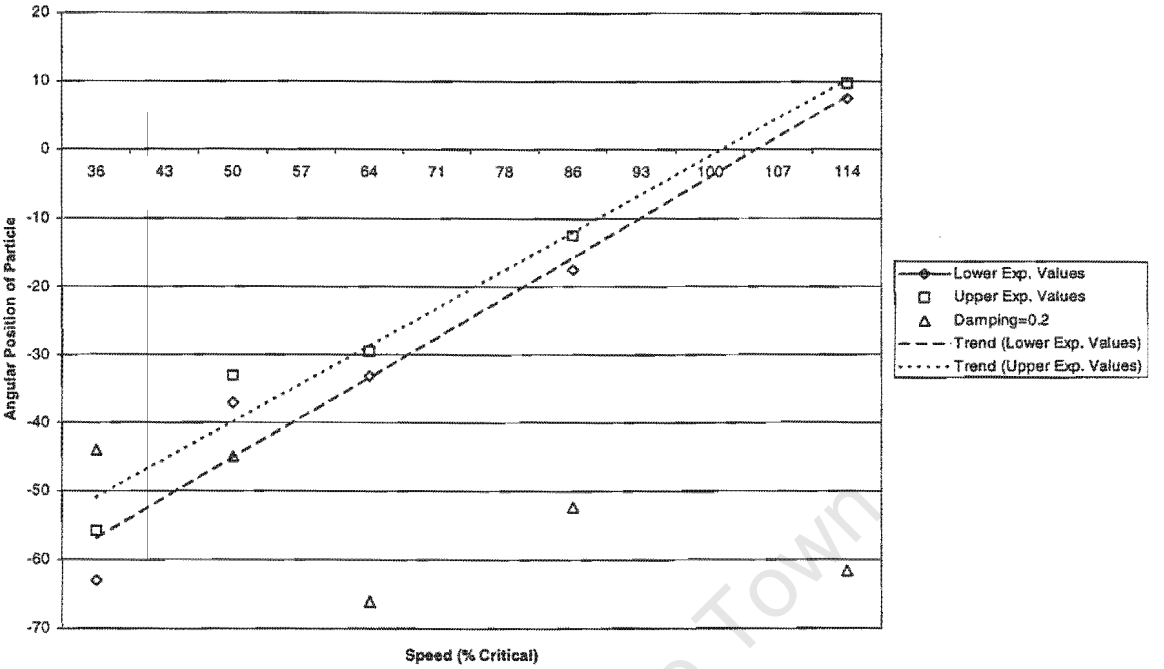


Figure 4.11: Angles of departure in ABAQUS with penatly contact and damping=0.2 for the corrugated liner

The penalty contact scheme is shown in figure 4.11. A damping of 0.2 was used in this simulation. As with the kinematic contact scheme, it was found that the values for 36% and 50% critical correlate well to the experimental trends. The values for speeds greater than 50% are significantly lower than the experimental values.

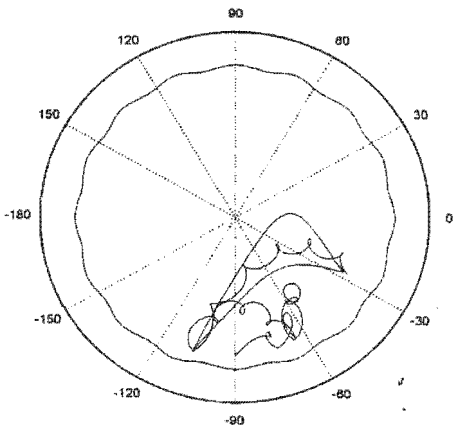


Figure 4.12: Kinematic contact with damping of 0.2

For the speeds 50% to 114%, the motion of the particle was found to be erratic for both contact schemes. This made it difficult to determine the angle of departure. An example of this is shown in figure 4.12. This diagram illustrates the motion of the particle at 64% critical speed using kinematic contact with a damping value of 0.2. Figure 4.13 shows the same speed with penalty contact scheme with damping of 0.2.

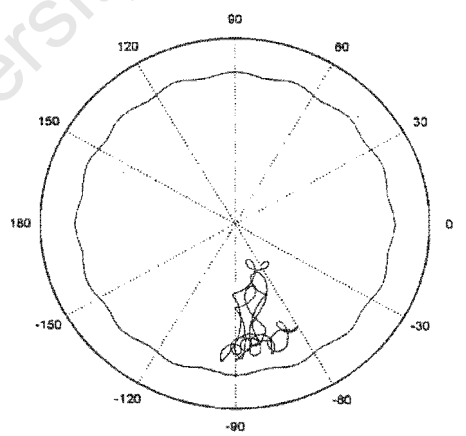


Figure 4.13: Penalty contact with damping of 0.2

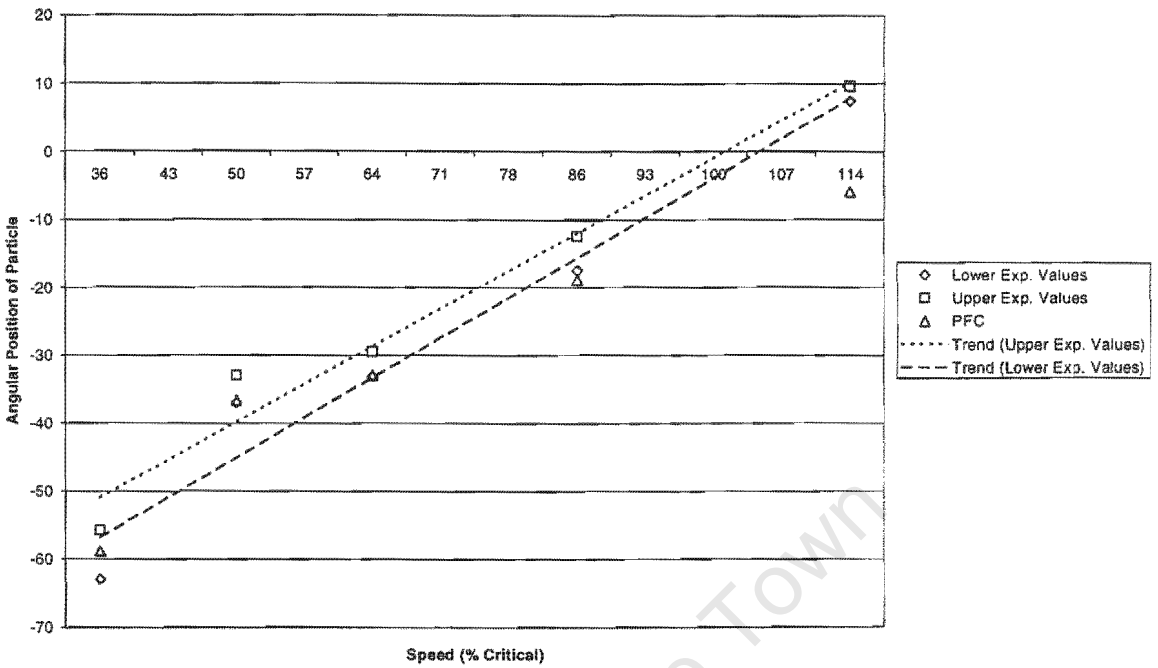


Figure 4.14: Angles of departure in PFC^{2D} with damping for the corrugated liner

The angles of departure obtained from PFC^{2D} correlate well with the experimental work until the 114% critical speed is reached. The angles of departure found using this package are compared to the experimental values in figure 4.14. For 114% of critical speed the results is lower than that of the experimental work.

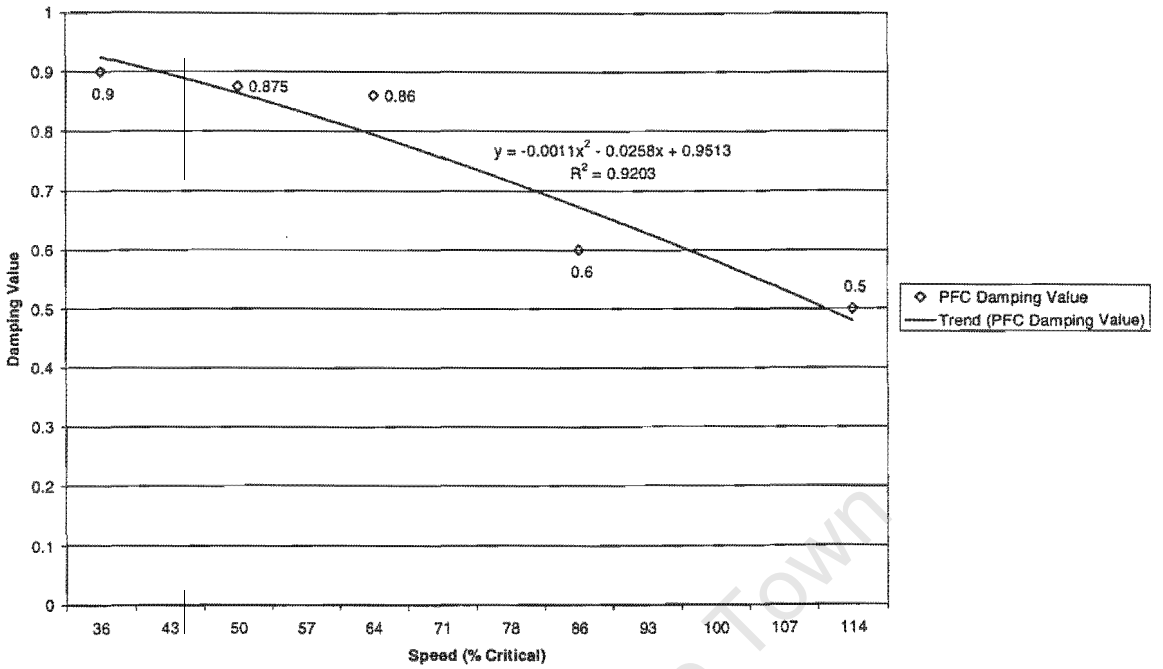


Figure 4.15: Damping values used in PFC^{2D} with the parabolic fit for the corrugated liner

A different damping value was applied to the particle for each speed. The values used were found to follow a parabolic function as seen in figure 4.15. The parabolic function has a 93% fit to the damping values used. The equation of the function is illustrated in figure 4.15 as the trend line.

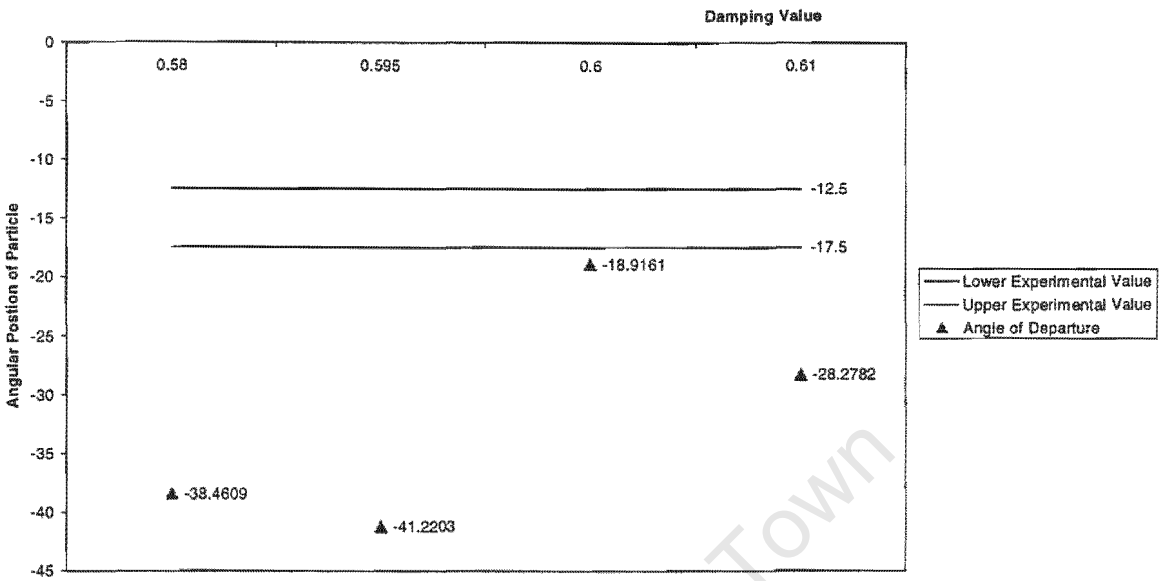


Figure 4.16: Angles of departure at 86% critical speed with various damping values in PFC^{2D} for the corrugated liner

The angle of departures were found to be very sensitive to the damping value used. Figure 4.16 show the angles of departure for 86% critical with different damping values. The closest angle to the experimental work is -18.9161 with a damping value of 0.6. When the damping value is decreased by 0.005, the angle drops to -41.2203. When the damping values is increased from 0.6 to 0.61, the angle of departure is -28.2782.

4.4 Skorupa Liner

The experimental work conducted with the Skorupa liner used a steel particle on a PVC liner. The speeds modelled in the software packages are the same as those used for the corrugated liner. The mills with the corrugated liner and the skorupa liner have the same diameters.

In ELFEN the normal and tangential penalties used were the same as for the corrugated liner . The factor of critical step (f) was again found to affect the numerical results. As can be seen in figure 4.17 the angle increased for 64% of critical when $f = 0.2$ was used. The angle of departure dropped significantly for 114% of critical speed for $f = 0.2$.

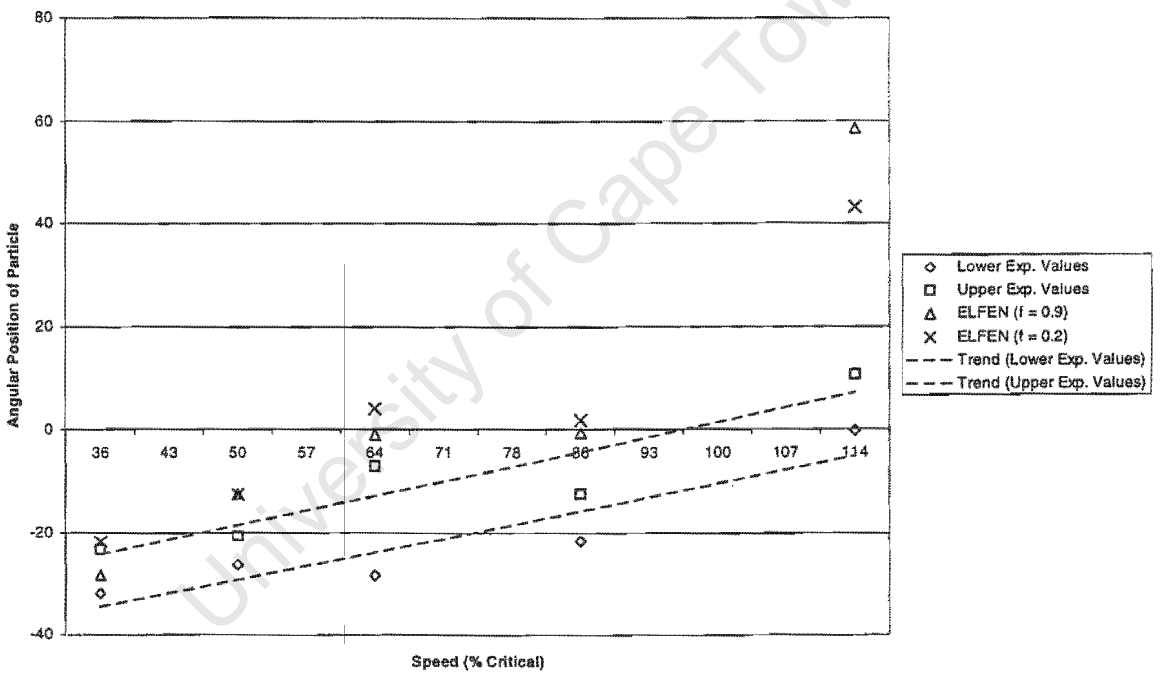


Figure 4.17: Angles of departure in ELFEN with $f = 0.2$ and $f = 0.9$ for the corrugated liner

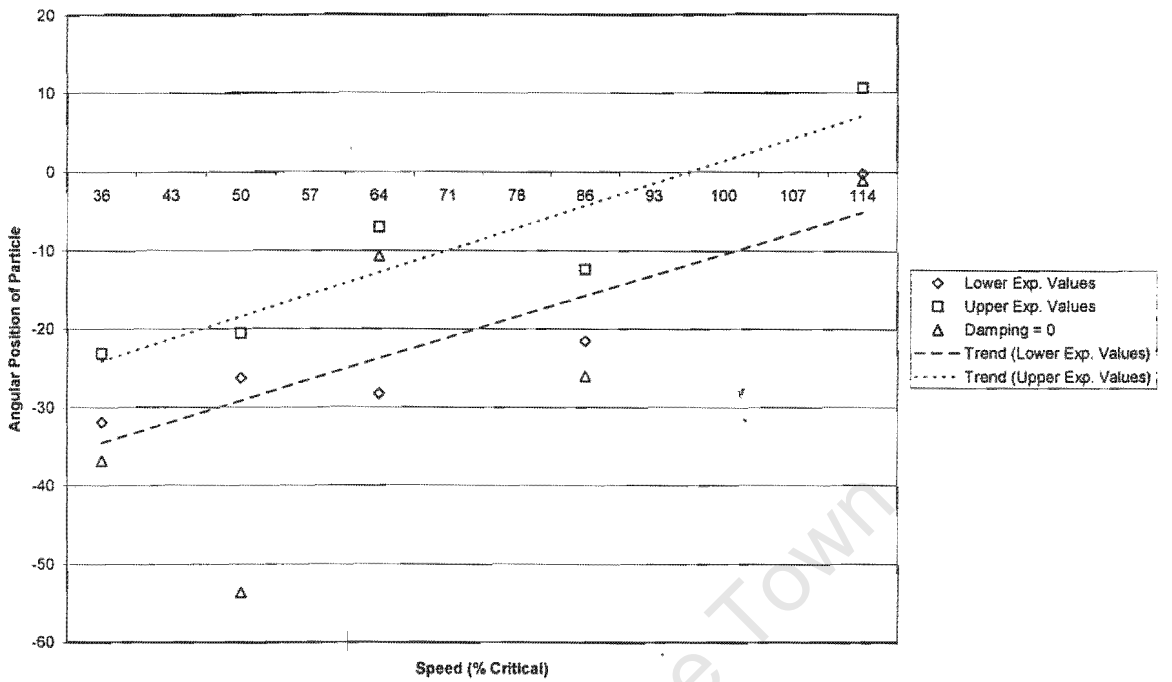


Figure 4.18: Angles of departure in ABAQUS with kinematic contact and damping=0 for the Skorupa liner

The simulations in ABAQUS were run with the two different contact schemes for various damping values. The kinematic contact algorithm was run with damping values ranging from 0 to 0.4. The resulting angles of departure for the kinematic contact scheme with damping equal to 0 are shown in figure 4.18. The numerical angle of departure for 36% critical speed correlates the closest to the experimental trends for zero damping. All the other damping values used results in a result that is much lower (refer to Appendix F for figures). It was found that the results for 64% critical speed correlated well for all damping values used. For 86% and 114% critical speed the results for zero damping follow the experimental trends the best.

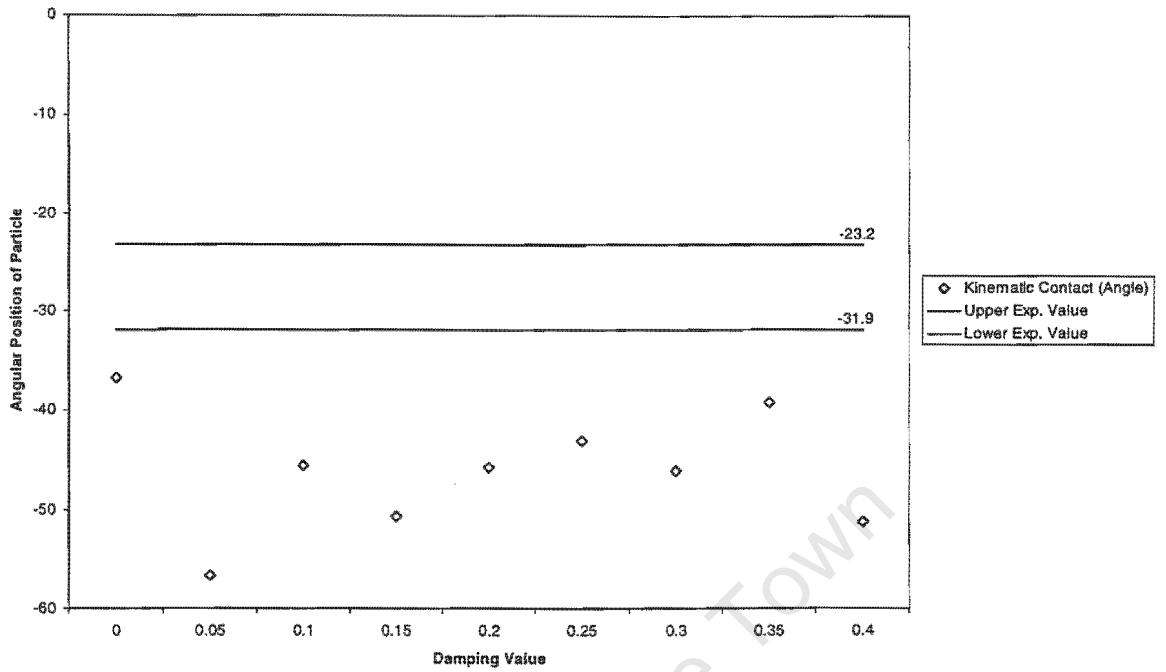


Figure 4.19: Kinematic contact with various damping values for 36% critical speed for the Skorupa liner

Figure 4.19 shows the angles of departure at various damping values using the kinematic contact scheme for 36% critical speed. The damping value appears to be sensitive to changes. Thus it is possible that the experimental value can be reached at a very specific damping value.

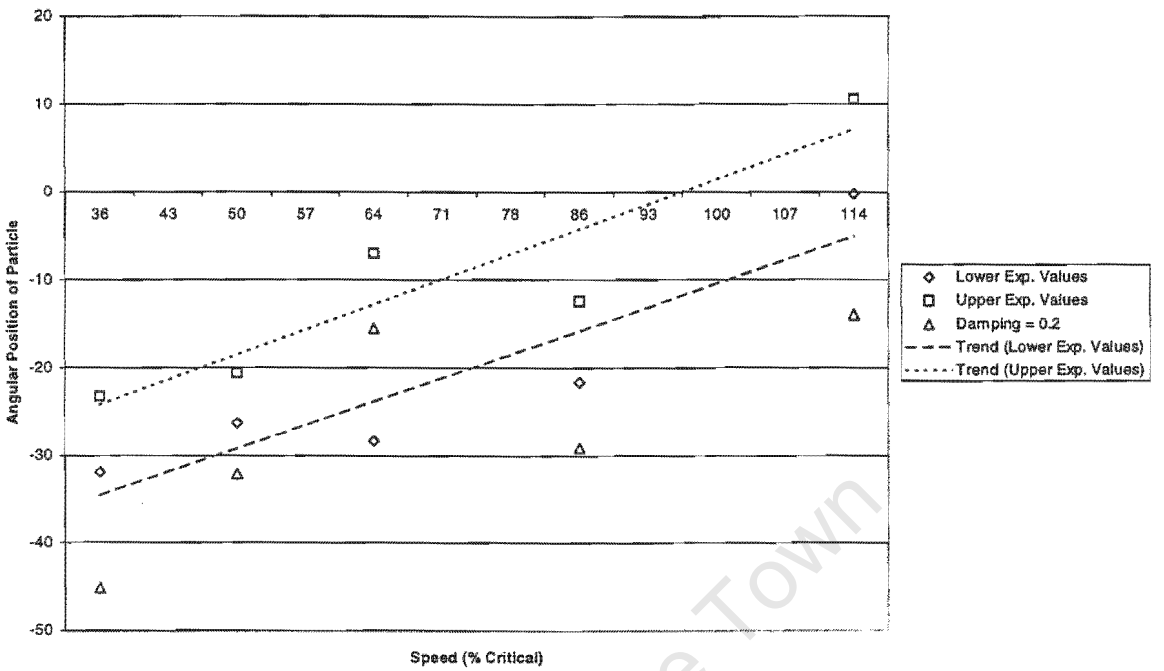


Figure 4.20: Angles of departure in ABAQUS with penalty contact and damping=0.2 for the Skorupa liner

The penalty contact method was used with a scale penalty of 0.5. The angles of departure calculated for a damping of 0.2 are illustrated in figure 4.20. The numerical results for 50% of critical speed is satisfactorily close to the trend for the lower experimental values. This is not the case for the other damping values used (refer to Appendix F). The angles of departure for the 64% of critical speed lie within the experimental values for all the damping values used except zero damping (refer to Appendix F). The results for the other speeds do not correlate well to the experimental values for any of the damping values.

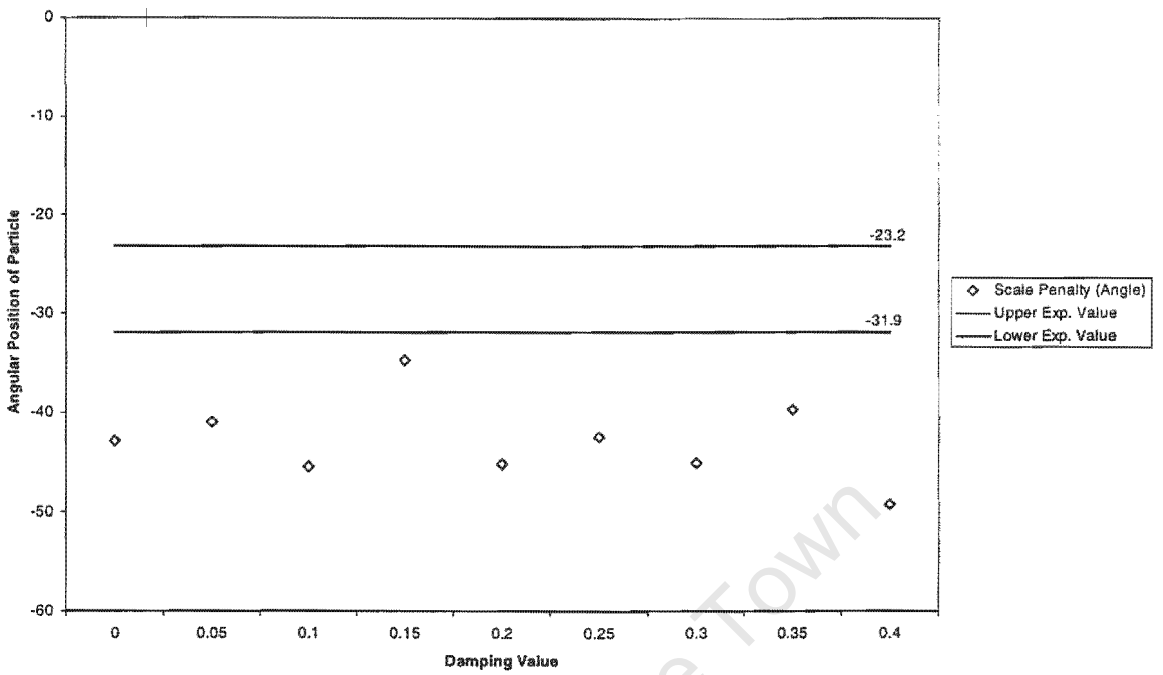


Figure 4.21: Penalty contact with various damping values for 36% critical speed for the Skorupa liner

As with the kinematic contact scheme, the damping value appears to be sensitive to changes when using the penalty method contact scheme. Figure 4.21 shows the results using the penalty method with various damping at the speed of 36% critical speed. The damping value of 0.15 gives a result closest to the experimental values.

The values for the angles of departure obtained from PFC^{2D} correlated well with the trends of the experimental work (figure 4.22). It was found that the damping applied to this model behaved in a similar fashion to the damping values for the corrugated model. The damping values also followed a parabolic function shown in figure 4.23. The resulting function is flatter than that of the corrugated liner. As found previously, the angles of departure were sensitive to changes in the damping value.

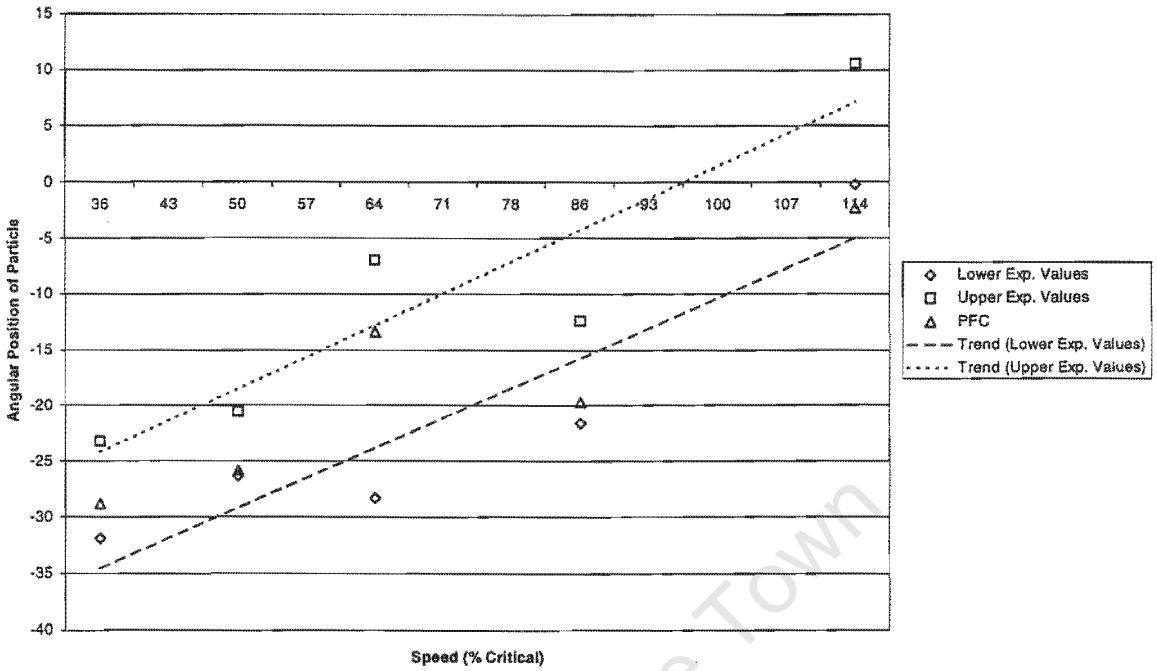


Figure 4.22: Angles of departure in PFC^{2D} for the Skorupa liner

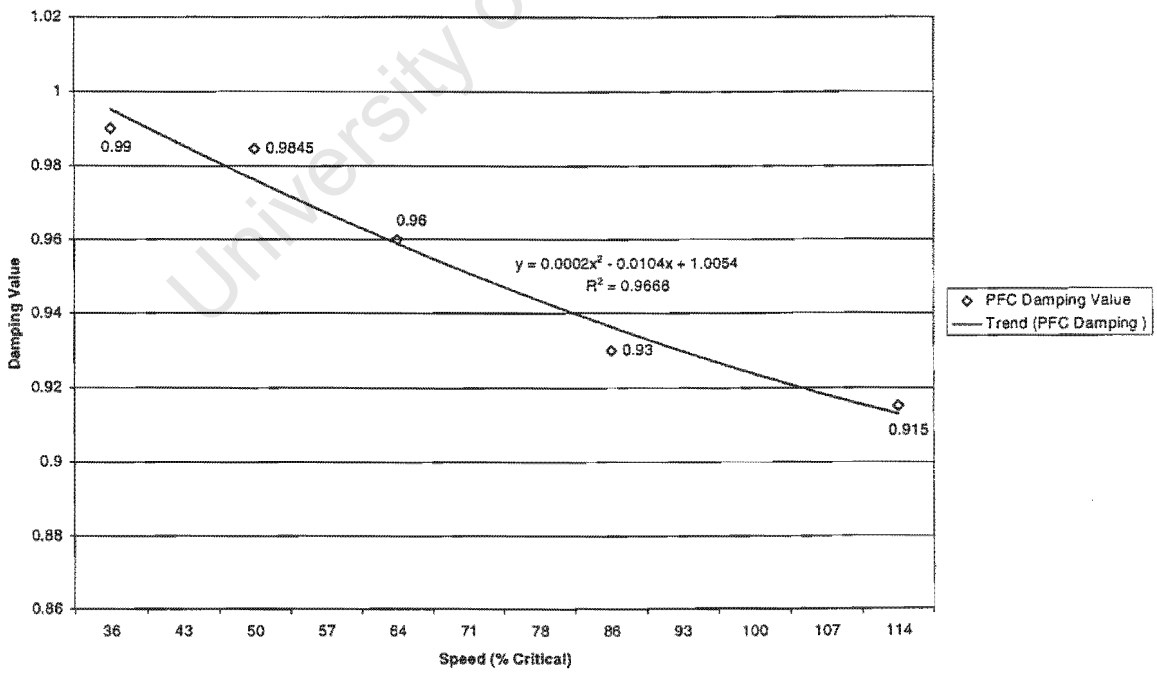


Figure 4.23: Damping values in PFC^{2D} with parabolic fit for the Skorupa liner

Chapter 5

Discussion

This research has allowed an opportunity not only to assess the software packages, but also to investigate the parameters which affect the motion of a single particle.

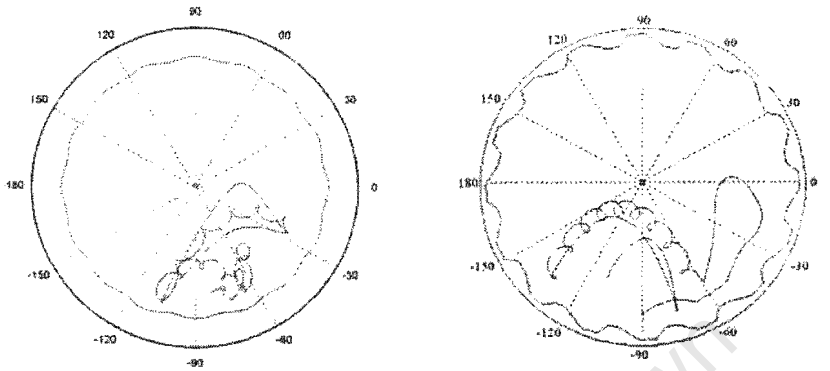
For the smooth liner all the codes reproduced the experimental results until 105% critical speed. For speeds higher than this, the particle did not move significantly higher than the horizontal (i.e. 0°). The same occurrence can be seen in the corrugated liner for ELFEN and PFC^{2D}, where the particle does not move beyond 0° for 114% of critical speed. It seems as though the particle does not gain enough energy to move beyond 0° . Instead it slides back down the liner.

The results for ABAQUS correlate well with the other software packages for the smooth liner. The same is not true for the corrugated and Skorupa liners. The motion of the particle on the corrugated liner can be seen in figures 5.1a and 5.2a. It seems that once the particle is in motion and leaves the liner, it is unable to dissipate energy. The particle does not settle on the liner again, but bounces around. This behaviour is not as pronounced in the skorupa liner as can be seen from the particle's motion in figures 5.1b and 5.2b.

This reason for the difference in particle motion could be due to the liner configuration. For the skorupa liner there is a section that is the similar to the smooth liner indicated in figure 5.3. It is possible that this section allows the particle to dissipate energy thus allowing the particle's motion to be more stable. The corrugated liner does not have any flat section to allow energy dissipation.

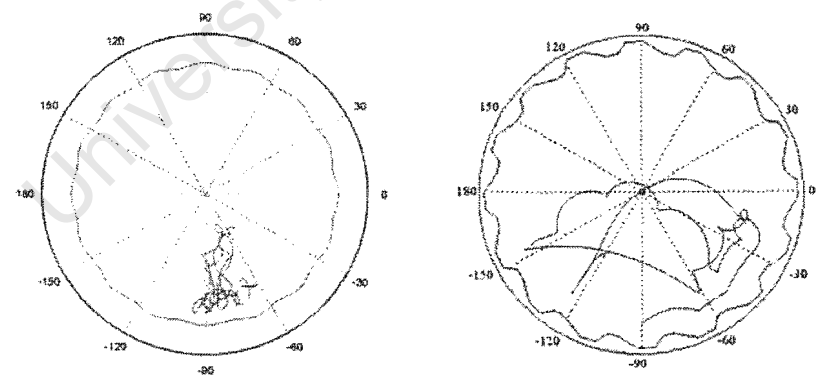
However, even though the particle's motion is more stable for the skorupa liner, the numerical results did not correlate with the experimental work for all the speeds.

Damping was introduced to the simulations. This did not however improve the results.



a.) Corrugated Liner b.) Skorupa Liner

Figure 5.1: Motion of particle with two differents liners in ABAQUS at speed 64% critical with kinematic contact and 0.2 damping



a.) Corrugated Liner b.) Skorupa Liner

Figure 5.2: Motion of particle with two differents liners in ABAQUS at speed 64% critical with penalty contact and 0.2 damping

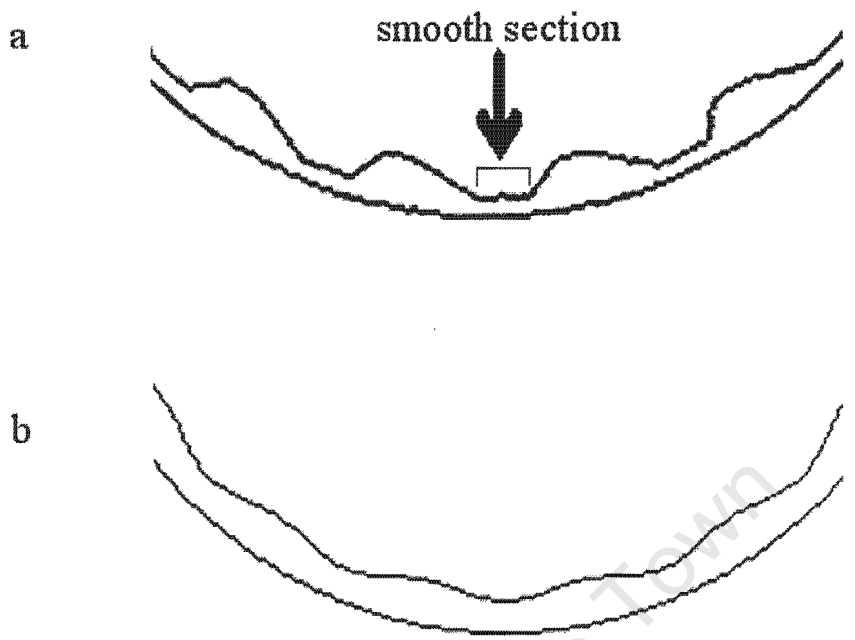


Figure 5.3: Comparison between the corrugated liner and the Skorupa liner

Damping was found to have a great influence in the simulations using PFC^{2D}. The trend of damping is different for each liner. The smooth liner used a constant damping value. The corrugated and skorupa liner both have damping values that follow a parabolic function. The function for the corrugated liner is steeper than for the skorupa liner. A comparison of this trend is shown in figure 5.4. This implies a relationship between the damping coefficient and the liner configuration.

The angles of departure obtained from PFC^{2D} exhibited a sensitivity to changes in the damping value for the corrugated liner. This would imply a dependency on the speed of the mill. ELFEN however did not show the same sensitivity. When comparing the trend of the damping values for ELFEN and PFC^{2D}, the resulting parabolic function for ELFEN is found to be flatter than for PFC^{2D} (figure 5.5).

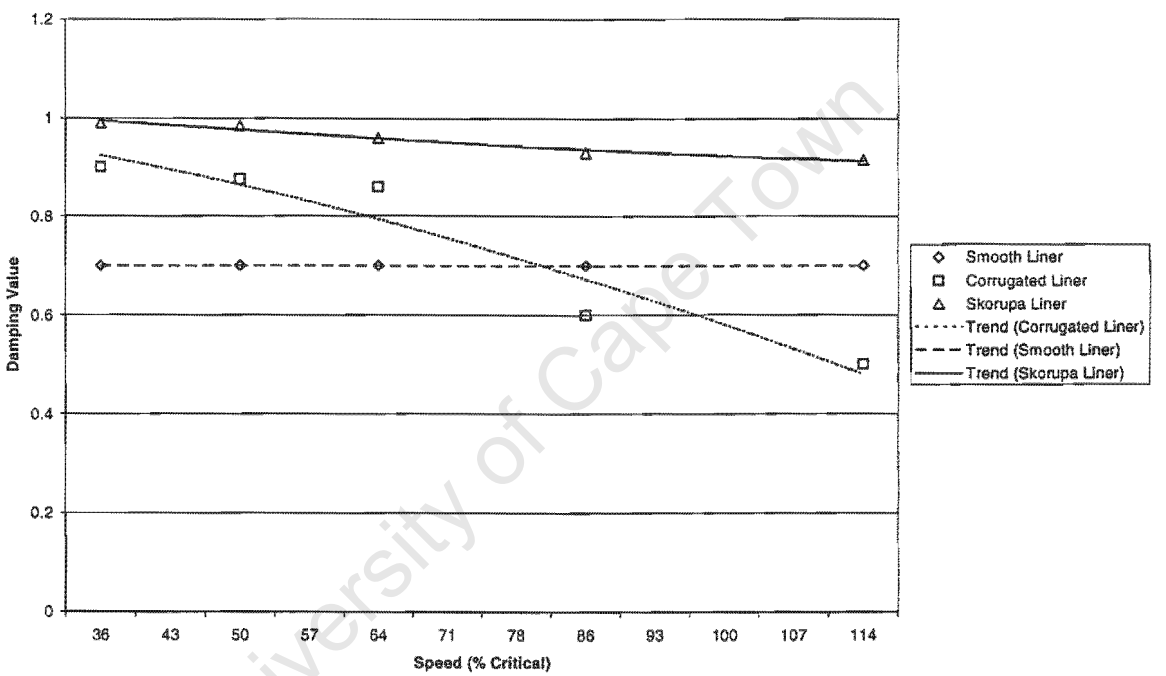


Figure 5.4: A comparison of the trends of the damping values used for the different liners in PFC^{2D}

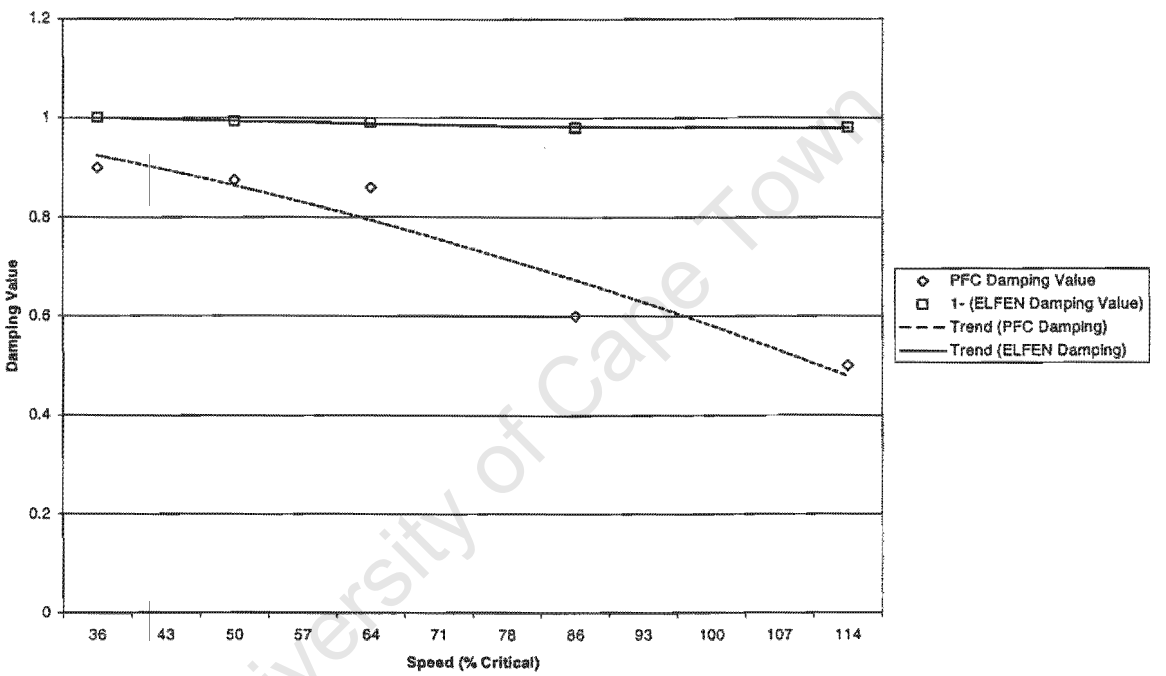


Figure 5.5: Comparison of the damping values for ELFEN and PFC^{2D} for the corrugated liner

Chapter 6

Conclusions

As a user, there are considerations besides the numerical results that are important. Issues such as user friendliness, the speed of the package and the level of understanding of the theory used by the package also need to be considered.

ELFEN has an appealing user interface. The various parts to the package are complex as there are different options available. A thorough help manual is therefore essential. In the version used for this research, modifications had to be made by the user to the data file that is compiled before an analysis is run. The modifications were prescribed by the developers. The necessary changes had not yet been implemented into the package itself to allow for a problem of a discrete nature. However once these modifications have been made, the package would be more user friendly. At a glance it would seem that one does not need to know a great deal of theoretical background but this is not the case. The package needs to be very robust for a user with little or no knowledge relating to the problem to use it.

ABAQUS, as with PFC^{2D}, allows the user a greater deal of flexibility with regards to the inputting of the data. The input or data file used by these two packages has to be written in a particular format. The packages also require the user to be more knowledgable about the numerical method used in the analysis of the problem.

In comparison to the other packages, ELFEN had a long runtime. For the mill with a smooth liner, ELFEN completed the simulation in approximately 7 to 8 hours. ABAQUS completed the same problem in approximately 30 minutes and PFC^{2D} in about 2 minutes. Therefore in terms of time efficiency , PFC^{2D} is superior. All

packages allow the user to run more than one simulation at a time.

All the packages allow easy extraction of the data required by the user. It is necessary however to know beforehand exactly what data is required. The package has to be told before the simulation which information to store. If the user wishes to analyse a different variable and it was not included in the simulations, the problem has to be rerun to extract that data. This can be problematic if the runtimes are long, such as with ELFEN.

The assessment of the software packages afforded an opportunity to examine the parameters that affect the motion of a single particle in a mill. It was found that damping is a contributing factor. The value used for damping in PFC^{2D} was influenced by the liner configuration and the speed of the mill. The extent of the effect of damping was different for each package. ELFEN was found not to be as sensitive to changes in damping as PFC^{2D} was.

While the numerical results for ELFEN were satisfactory, the time taken to complete the simulations was not. For ABAQUS the time efficiency was superior to ELFEN, but the overall results did not correlate well with the experimental work. Although the damping values in PFC^{2D} were found to be sensitive, the package was superior to the ELFEN and ABAQUS with regards to time efficiency and the correlation of the results to the experimental work.

It is recommended that further work be conducted with PFC^{2D}. The factors influencing the interaction between particles can be investigated by conducting simulations with various liner configurations and only two particles. In order to understand the effect of damping on the motion of the two particles, other input values such as the coefficient of friction can be kept constant.

Simulations using various percentages fill with a different liner configuration could be compared to the experimental work conducted by Milner [3]. Work that is currently being done using PFC^{2D} for mills with numerous particles has shown that the runtime of the simulations will not be as long as was for ELFEN [36]. Thus PFC^{2D} is more viable.

References

[1] M.B. Nates. An investigation into the parameters affecting the performance of tube mills: The behaviour of a single particle on the inside of a rotating cylinder. Masters Dissertation, University of Cape Town, September 1989.

[2] K. Von Bentheim. An investigation into the parameters affecting the performance of tube mills: The behaviour of a single particle on a corrugated liner inside a rotating cylinder. Masters Dissertation, University of Cape Town, April 1991.

[3] A.L. Milner. The prediction of outermost trajectory of media in a grinding mill for the lifter bars with rounded or worn profiles. Masters Dissertation, University of Cape Town, November 1996.

[4] R.K. Rajamani, B.K. Mishra, R. Venugopul, and A. Datta. Discrete element analysis of tumbling mills. *Powder Technology*, 109:105–112, 2000.

[5] B.K. Mishra and R.K. Rajamani. The discrete element method for the simulation of ball mills. *Applied Mathematical Modelling*, 16:598–604, 1992.

[6] B.K. Mishra and R.K. Rajamani. Simulation of charge in ball mills. part 2: Numerical simulations. *International Journal of Mineral Processing*, 40:187–197, 1994.

[7] Http://www.mbendi.co.za/indy/ming/af/sa.

[8] Http://www.mbendi.co.za/indy/ming/gold/af/sa.

[9] Http://www.mbendi.co.za/indy/ming/plat/af/sa.

[10] Http://www.mbendi.co.za/indy/ming/plat.

[11] M.S. Powell. A study of charge motion in rotary mills, with particular reference to the grinding action. Doctoral Dissertation, University of Cape Town, March 1993.

[12] P.A. Cundall and O.D.L. Strack. A discrete numerical model for granular assemblies. *Geotechnique*, 29(1):47–65, 1979.

[13] P.W. Cleary. Predicting charge motion, power draw, segregation and wear in ball mills using discrete element methods. *Minerals Engineering*, 11(11):1061–1080, 1998.

[14] P.W. Cleary and D. Hoyer. Centrifugal mill charge motion and power draw: Comparison of dem predictions with experiment. *Int. J. Min. Process.*, 59:131–148, 2000.

[15] A.L. Hinde M.A. van Nierop, G. Glover and M.H. Moys. A discrete element investigation of the charge motion and power draw of an experimental two-dimensional mill. *Int. J. Min. Process.*, 61:77–92, 2001.

[16] H.A. White. The theory of the tube mill. *The Journal of the Chemical Metallurgical and Mining Society of South Africa*, pages 290–305, May 1905.

[17] E. W. Davis. Fine crushing in ball mills. *Trans of the American Inst. of Mining and Metallurgical Engineers*, 61:250–296, 1919.

[18] H. E. Haultain and F. C. Dyer. Ball paths in tube mills. *Canadian Institute of Mining and Metallurgy Engineers*, 25:276–291, 1922.

[19] A. M. Gow, A. B. Campbell, and W. H. Coghill. A laboratory investigation of ball milling. *AIME Trans*, pages 51–81, October 1929.

[20] A. W. Fahrenwald and H. E. Lee. Ball mill studies. *AIME Technical Publication No*, 375, 1931.

[21] H.E. Rose and R.M. Sullivan. *Ball, Tube and Rod Mills.* Constable, London, 1958.

[22] R. E.McIvor. Effects of speed and liner configuration on ball mill performance. *Mining Engineering*, pages 617–622, June 1983.

[23] L.A. Vermeulen, M.J. Ohlsen De Fine, and F. Schakowski. Physical information from the inside of a rotary mill. *J. S. Afr. Inst. Min. Metall*, 84(8):247–253, August 1984.

[24] L.A. Vermeulen. The lifting action of lifter bars in rotary mills. *J. S. Afr. Inst. Min. Metall*, pages 51–63, January 1985.

[25] M.S. Powell. The effect of liner design upon charge motion in a rotary mill. Masters Dissertation, University of Cape Town, September 1988.

[26] S. Morrell. Prediction og grinding-mill power. *Trans. Inst. Min. Metall.*, 101:C25–C32, 1992.

[27] M.S. Powell and G.N. Nurick. A study of charge motion in rotary mills: part 1 - extension of the theory. *Minerals Engineering*, 9(2):259–268, 1996.

[28] M.S. Powell and G.N. Nurick. A study of charge motion in rotary mills: part 2 - experimental work. *Minerals Engineering*, 9(3):343–350, 1996.

[29] M.S. Powell and G.N. Nurick. A study of charge motion in rotary mills: part 3 - analysis of results. *Minerals Engineering*, 9(4):399–418, 1996.

[30] I. Govender, M.S. Powell, and G.N. Nurick. 3d particle tracking in a mill: A rigorous technique for verifying dem predictions. *Minerals Engineering,to appear.*

[31] J.J. Skorupa. Wear of tube mill liners for south african power industry. Doctoral Dissertation, University of Cape Town, March 1989.

[32] A. Datta, B.K. Mishra, and R.K. Rajamani. Analysis of power draw in ball mills by the discrete element method. *Canadian Metallurgical Quarterly*, 38(2):133–140, 1999.

[33] B.K. Mishra and R.K. Rajamani. Simulation of charge in ball mills. part 1: Experimental verifications. *International Journal of Mineral Processing*, 40:171–186, 1994.

[34] P. Radziszewski. Comparing three dem charge motion models. *Minerals Engineering*, 12(12):1501–1520, 2000.

[35] D. Zhang and W.J. Whiten. Contact modelling for discrete element modelling of ball mills. *Minerals Engineering*, 11(8):689–698, 1998.

[36] I. Govender, V. Balden, M.S. Powell, and G.N. Nurick. Validated DEM - potential major improvements to sag mill modelling. In *SAG Conference*, 2001.

[37] Rockfield Software Limited. *Elfen User Manual Version 2.8*, 1999.

[38] Hibbitt, Karlsson and Sorenson, Inc., USA. *ABAQUS Theory Manual*, 1997.

[39] Itasca Consulting Group, Inc., USA. *PFC Theory and Background*, 1999.

Appendices

Appendix A

Code used for ELFEN

A.1 MATLAB code used to process results

```
clear all
    clc

    load x_values_filename.grf;
    dummy(:,1:2)= filename;

    load y_values_filename.grf;
    dummy(:,3)= filename (:,2);
    index=size(dummy,1)/4
    time=dummy(1:index,1);
    dummy_time=input('enter time of one revolution ');
    theta_shell=time.*(2*pi)/dummy_time;

    displ_node1=dummy(1:index,2:3);
    displ_ node1 (:,1)=displ_ node1 (:,1);
    displ_ node1 (:,2)=displ_ node1 (:,2)+(-0.186);

    displ_ node2=dummy(index+1:index*2,2:3);
    displ_ node2 (:,1)=displ_ node2 (:,1);
    displ_ node2 (:,2)=displ_ node2 (:,2)+(-0.186);

    displ_ node3=dummy(2*index+1:index*3,2:3);
    displ_ node3 (:,1)=displ_ node3 (:,1);
    displ_ node3 (:,2)=displ_ node3 (:,2)+(-0.201);

    displ_ node4=dummy(3*index+1:index*4,2:3);
    displ_ node4 (:,1)=displ_ node4 (:,1);
    displ_ node4 (:,2)=displ_ node4 (:,2)+(-0.201);

    [theta_node1,R_ node1] = cart2pol(displ_ node1 (:,1),displ_ node1 (:,2));
    [theta_ node2,R_ node2] = cart2pol(displ_ node2 (:,1),displ_ node2 (:,2));
    [theta_ node3,R_ node3] = cart2pol(displ_ node3 (:,1),displ_ node3 (:,2));
```

```
[theta_ node4,R_ node4] = cart2pol(displ_ node4 (:,1),displ_ node4 (:,2));

interval_flag=0;
while interval_flag==0
    lower_bound=input('Enter the lower boundary for the time interval to be
plotted ');
    upper_bound=input('Enter the upper boundary for the time interval to be
plotted ');

    %Plot displacements for the given time intervals for each node
    figure(1)
    subplot(2,2,1)
  polar(theta_node1(lower_bound:upper_bound),R_ node1 (lower_bound:upper_bound))
    title(['displacement for node 1'])
    subplot(2,2,2)
  polar(theta_ node2(lower_bound:upper_bound),R_ node2 (lower_bound:upper_bound))
    title(['displacement for node 2'])

    subplot(2,2,3)
  polar(theta_ node3 (lower_bound:upper_bound),R_ node3 (lower_bound:upper_bound))
    title(['displacement for node 3'])

    subplot(2,2,4)
  polar(theta_ node4(lower_bound:upper_bound),R_ node24(lower_bound:upper_bound))
    title(['displacement for node 4'])

    %Enter whether you are satisfied with the chosen time interval
    verdict=input('Satisfied with time intervals plotted? (yes=1 no=0)');
    %If yes then jump out of loop
    if verdict==1
        interval_flag=1
    end
end
```

```
temp_node1(:,1)=theta_ node1 (lower_bound:upper_bound);
flag_ node1 =0;
while flag_ node1 ==0
    [theta_depart_ node1,i]=max(temp_ node1);
    real_i=i+lower_bound;
    radius=R_ node1 (real_i);
    if radius<0.165
        temp_ node1 (i)=-20;
    else
        flag_ node1 =1;
    end
end

temp_ node2 (:,1)=theta_ node2 (lower_bound:upper_bound);
flag_ node2 =0;
while flag_ node2 ==0
    [theta_depart_ node2,i]=max(temp_ node2);
    real_i=i+lower_bound;
    radius=R_ node2 (real_i);
    if radius<0.165
        temp_ node2 (i)=-20;
    else
        flag_ node2 =1;
    end
end

temp_ node3 (:,1)=theta_ node3 (lower_bound:upper_bound);
flag_ node3 =0;
while flag_ node3 ==0
    [theta_depart_ node3,i]=max(temp_ node3);
    real_i=i+lower_bound;
    radius=R_ node 3(real_i);
```

```
        if radius<0.165
            temp_ node3 (i)=-5;
        else
            flag_ node3 =1
        end
    end

    temp_ node4 (:,1)=theta_ node4 (lower_bound:upper_bound);
    flag_ node4 =0;
    while flag_ node4 ==0
        [theta_depart_ node4,i]=max(temp_ node4);
        real_i=i+lower_bound;
        radius=R_ node4 (real_i);
        if radius<0.165
            temp_ node4 (i)=-20;
        else
            flag_ node4 =1;
        end
    end

    AOD_ node1 =(theta_depart_ node1).*(180/pi)
    AOD_ node2=(theta_depart_ node2).*(180/pi)
    AOD_ node3 =(theta_depart_ node3).*(180/pi)
    AOD_ node4 =(theta_depart_ node4).*(180/pi)

    AOD=(AOD_ node1+AOD_ node2+AOD_ node3+AOD_ node4)/4
```

Appendix B

Code used for ABAQUS/Explicit

B.1 Input deck for the Smooth Liner

```
*heading
  Mill with smooth liner at 20rpm
  **
  *preprint, model=yes, history=yes, echo=no, contact=yes
  **
  **_____
  ** Define rigid shell, radius = 0.202m
  **_____
  *node, nset=ns_ref
  1000, 0.0, 0.0, 0.0
  *rigid body, ref node=1000
  **_____
  ** Define block with dimensions 0.015 by 0.015m
  **_____
  *node, nset=bot
  1, 0.0, 0.0, 0.0
  5, 0.015, 0.0, 0.0
  *ngen, nset=bot
  1,5,1
  *ncopy, old set=bot, new set=top, multiple=1, shift, change number=40
  0.0, 0.015, 0.0
  0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0
  *nfill, nset=ns_temp
  bot, top, 4, 10
  **
  *ncopy, old set=ns_temp, new set=ns_block, multiple=1, change number=100,
shift
  0.0, -0.201, 0.0
  0.0, 0.0, 0.0, 0.0, 0.0, 1.0, -45.
  **
  **_____
  ** Define elements on block
```

```
**_____________________________________________
*element, type=cpe4r, elset=el_block
1, 101,102,112,111
*elgen, elset=el_block
1, 4,1,1, 4,10,10
**
**_____________________________________________
** Define material properties of block
**_____________________________________________
*solid section, elset=el_block, material=steel
0.015
*material, name=steel
*elastic
200.e9, 0.3
*density
7800.
**_____________________________________________
** Define step: drop block under gravity
**_____________________________________________
*step, nlgeom=yes
drop block under gravity
*dynamic, explicit
,0.05
*rigid surface, type=segments, name=shell, ref node=1000
start, 0.202, 0.0
circl, 0.0 , 0.202, 0.0,0.0
circl, -0.202 ,0.0 , 0.0,0.0
circl, 0.0 ,-0.202, 0.0,0.0
circl, 0.202, 0.0 , 0.0,0.0
**
**_____________________________________________
** Define contact surfaces & pairs
**_____________________________________________
```

```
*surface definition, name=block
el_block,
*contact pair, interaction=block2shell
block, shell
*surface interaction, name=block2shell
*friction
0.194
**
*boundary, type=displacement
1000, 1,2, 0.0
**
*dload, op=new
el_block, grav, 9.81, 0.0, -1.0, 0.0
**
*restart, write, number intervals=50
**
*end step
**_____
** define step: rotate shell
**_____
*step
rotate shell 20rpm
*dynamic, explicit
,3.0
*restart, write, number intervals=500
*file output, number interval=10000
*node file, nset=ns_block
U,V,RF
*node file, nset=ns_ref
U,V,RF
**
*boundary, type=velocity
1000, 6,6, 2.0944
```

```
*end step
**
```

B.2 Input deck for the Corrugated Liner

```
*heading
    Mill with a Corrugated liner at 25.2rpm
    **
    *preprint, model=yes, history=yes, echo=no, contact=yes
    **
    **_____
    ** define rigid shell, radius = 0.202m
    **_____
    *node, nset=ns_ref
    1000, 0.0, 0.0, 0.0
    *rigid body, ref node=1000
    **_____
    ** define square block with dimensions 0.016 by 0.016m
    **_____
    *node, nset=bot
    1, 0.0, 0.0, 0.0
    5, 0.016, 0.0, 0.0
    *ngen, nset=bot
    1,5,1
    *ncopy, old set=bot, new set=top, multiple=1, shift, change number=40
    0.0, 0.016, 0.0
    0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0
    *nfill, nset=ns_temp
    bot, top, 4, 10
    *ncopy, old set=ns_temp, new set=ns_block, multiple=1, change number=100,
shift
    0.0, -0.182, 0.0
    0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0
    **
    **_____
    ** Define elements on block
    **_____
```

```
*element, type=cpe4r, elset=el_block
1, 101,102,112,111
*elgen, elset=el_block
1, 4,1,1, 4,10,10
**
**_____
** Define material properties of block
**_____
*solid section, elset=el_block, material=steel
0.016
*material, name=steel
*elastic
200.e9, 0.3
*density
7800.
*damping, alpha=0.1
**_____
** define step: drop block under gravity & rotate shell
**_____
*step, nlgeom=yes
drop block under gravity & rotate shell
*dynamic, explicit
,2.49
**
*rigid surface, type=segments, name=shell, ref node=1000
*include,input=abq_skorupa_liner.txt
**
**_____
** Define contact surfaces & pairs
**_____
*surface definition, name=block
el_block,
*contact pair, penalty, interaction=block2shell, cpset=shell2block
```

```
block, shell
*contact controls, cpset=shell2block, scale penalty=0.5
*surface interaction, name=block2shell
*friction
0.337
**
*boundary, type=displacement
1000, 1,2, 0.0
**
*dload, op=new
el_block, grav, 9.81, 0.0, -1.0, 0.0
*restart, write, number intervals=500
*file output, number interval=10000
*node file, nset=ns_block
U,V,RF
*node file, nset=ns_ref
U,V,RF
**
*boundary, type=velocity
1000, 6,6, 2.64
*end step
```

B.3 Input deck for the Skorupa Liner

```
*heading
    mill for von Bentheim with Corrugated liner at 25.2rpm
    **
    *preprint, model=yes, history=yes, echo=no, contact=yes
    **
    **_____
    ** define rigid shell, radius = 0.202m
    **_____
    *node, nset=ns_ref
    1000, 0.0, 0.0, 0.0
    *rigid body, ref node=1000
    **
    **_____
    ** define square block with dimensions 0.016 by 0.016m
    **_____
    *node, nset=bot
    1, 0.0, 0.0, 0.0
    5, 0.016, 0.0, 0.0
    *ngen, nset=bot
    1,5,1
    *ncopy, old set=bot, new set=top, multiple=1, shift, change number=40
    0.0, 0.016, 0.0
    0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0
    *nfill, nset=ns_temp
    bot, top, 4, 10
    *ncopy, old set=ns_temp, new set=ns_block, multiple=1, change number=100,
shift
    0.0, -0.182, 0.0
    0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0
    **
    **_____
    ** Define elements on block
```

```
**_____
*element, type=cpe4r, elset=el_block
1, 101,102,112,111
*elgen, elset=el_block
1, 4,1,1, 4,10,10
**
**_____
** Define material properties of block
**_____
*solid section, elset=el_block, material=steel
0.016
*material, name=steel
*elastic
200.e9, 0.3
*density
7800.
**_____
** define step: drop block under gravity & rotate shell
**_____
*step, nlgeom=yes
drop block under gravity & rotate shell
*dynamic, explicit
,2.49
*rigid surface, type=segments, name=shell, ref node=1000
*include,input=abq_skorupa_liner.txt
**
**_____
** Define contact surfaces & pairs
**_____
*surface definition, name=block
el_block,
*contact pair, penalty, interaction=block2shell, cpset=shell2block
block, shell
```

```
*contact controls, cpset=shell2block, scale penalty=0.5
*surface interaction, name=block2shell
*friction
0.337
**
*boundary, type=displacement
1000, 1,2, 0.0
**
*dload, op=new
el_block, grav, 9.81, 0.0, -1.0, 0.0
*restart, write, number intervals=500
*file output, number interval=10000
*node file, nset=ns_block
U,V,RF
*node file, nset=ns_ref
U,V,RF
**
*boundary, type=velocity
1000, 6,6, 2.64
*end step
**
```

B.4 Matlab code used to process results

```
clear all
    clc
    load filename.txt;
    %Load in time values
    dummy(:,1)= filename (:,1);
    %Load in x values
    dummy(:,2)= filename (:,2);
    %Load in y values
    dummy(:,3)= filename (:,3);

    index=size(dummy,1)
    time=dummy(1:index,1);
    dummy_time=input('enter time of one revolution in rad/s ');
    theta_shell=time.*dummy_time;
    displacement_data=dummy(1:index,2:3);
    displacement_data(:,1)=displacement_data(:,1);
    displacement_ data (:,2)=displacement_data(:,2)+(-0.182);
    [theta_ data, R_ data] = cart2pol(displacement_data(:,1), displacement_data(:,2));

    interval_flag=0;
    while interval_flag==0
        lower_bound=input('Enter the lower boundary for the time interval to be
plotted ');
        upper_bound=input('Enter the upper boundary for the time interval to be
plotted ');
        %Plot displacements for the given time intervals for the node
        figure(1)
      polar(theta_ data (lower_bound:upper_bound),R_ data (lower_bound:upper_bound))
        title(['Motion of particle'])
        hold off
        %Enter whether you are satisfied with the chosen time interval
        verdict=input('Satisfied with time intervals plotted? (yes=1 no=0)');
```

```
        if verdict==1
            interval_flag=1
        end
    end

    temp_data(:,1)=theta_data(lower_bound:upper_bound);
    flag=0;
    while flag==0
        [theta_depart,i]=max(temp_data);
        real_i=i+lower_bound;
        radius=R_data(real_i);
        if radius<0.165
            temp_data(i)=-20;
        else
            flag=1
        end
    end

    AOD=(theta_depart).*(180/pi)
```

Appendix C

Code used for PFC^{2D}

C.1 Data file used for the Smooth liner

```
new
    set max_balls 1000
    set gen_error on
    set random
    set disk on
    set pinterval=100
    trace energy on
    set logfile temp.log
    set log on overwrite
    define MillPower
        while_stepping
          Real_Time = time
          Power = e_bound / time
    end
    ;-----------------------------------------------------------------
    ; DEFINE MILL
    ;-----------------------------------------------------------------
    define Shell_Smooth
    ;
    xc = Shell_Xcenter ; x - center of circle
    yc = Shell_Ycenter ; y - center of circle
    ;
    omega = Shell_RotationSpeed
    n_stiff = Shell_Nstiff
    s_stiff = Shell_Sstiff
    friction= Shell_Friction
    ;
    dia = Shell_InnerDiameter
    num = Shell_NumSegments
    ;
    theta = (2*pi) / (num)
    ;
```

```
loop i (1,num)
   x1 = (dia * 0.5) * cos((i-1)*theta)
   y1 = (dia * 0.5) * sin((i-1)*theta)
   x2 = (dia * 0.5) * cos((i)*theta)
   y2 = (dia * 0.5) * sin((i)*theta)
;
   command
      wall id i kn n_stiff ks n_stiff fric friction &
;        spin omega x xc xv 0.0 y yc yv 0.0 &
      nodes x1,y1 x2,y2
   endCommand
;
endLoop
end
;
define Shell_Rotate
xc = Shell_Xcenter ; x - center of circle
yc = Shell_Ycenter ; y - center of circle
;
num = Shell_NumSegments
;
omega = Shell_RotationSpeed
loop i (1,num)
   command
      wall id=i spin omega x xc xv 0.0 y yc yv 0.0
   endCommand
endLoop
end
;
;-------------------------------------------------------------------------
; GENERATE BALLS
;-------------------------------------------------------------------------
define Balls_Packing
```

```
xc = x0
yc = y0
rc = radius
idc = id_start
r2 = 2.0 * radius
yinc = radius * sqrt(3.0)
loop row (1,n_row)
   loop col (1,n_col)
      command
         ball id=idc x=xc y=yc rad=rc
      end_command
      idc = idc + 1
      xc = xc + r2
   end_loop
yc = yc + yinc
xc = x0 + radius * (row - (row/2) * 2)
end_loop
end
;
```

```
;-------------------------------------------------------------------------------
; DEFINE MILL PARAMETERS
;-------------------------------------------------------------------------------
set Shell_InnerDiameter = .404
set Shell_NumSegments = 180
set Shell_Xcenter = 0.0
set Shell_Ycenter = 0.0
set Shell_Nstiff = 1e6
set Shell_Sstiff = 1e6
set Shell_RotationSpeed = 2.09
set Shell_Friction = 0.194
;
;-------------------------------------------------------------------------------
; DEFINE HEX ARRAY OF BALLS PARAMETERS
;-------------------------------------------------------------------------------
set x0 = -0.141
set y0 = -0.121
set id_start=201
set radius=0.001
set n_col=7
set n_row=7
;
;-------------------------------------------------------------------------------
; BUILD MODEL
;-------------------------------------------------------------------------------
set echo off
Shell_Smooth
Balls_Packing
set echo on
;
;-------------------------------------------------------------------------------
; SET GLOBAL MATERIAL PROPERTIES FOR ALL BALLS
;-------------------------------------------------------------------------------
```

```
property density 7800
property kn = 1e6
property ks = 1e6
property fric = 0.194
;
;-----------------------------------------------------------------
; MAKE CLUMP OF ALL BALLS
;-----------------------------------------------------------------
property n_bond=1e6
property s_bond=1e6
clump id=999, full on, perm, range id=201,1000
;
;-----------------------------------------------------------------
; DEFINE DAMPING
;-----------------------------------------------------------------
define zero_damp
    bp = ball_head
    loop while bp # null
      b_damp(bp) = 0.0
      bp = b_next(bp)
    end_loop
end
zero_damp
;
define catch_contact_hys
cp = fc_arg(0);
c_model(cp) = 'hysdamp'
c_prop(cp,'damp_n')= setv
c_prop(cp,'notension')=1;
end
set setv=0.7
model hysdamp
set fishcall 6 catch_contact_hys
```

```
;
;-------------------------------------------------------------
; SET GLOBAL GRAVITY
;-------------------------------------------------------------
set grav 0 -9.81
;
;-------------------------------------------------------------
; PLOTTING ROUTINES
;-------------------------------------------------------------
plot create pic
plot set title text 'clump geometry 7x7@rad 1.0mm'
plot add ball lblue
plot add clump
plot add wall black
plot add axes brown
plot show
;
;-------------------------------------------------------------
; ADDS SHELL MARKER
;-------------------------------------------------------------
wall id=999 nodes 0.2,0 .21,0
;
;-------------------------------------------------------------
; SET GLOBAL MATERIAL PROPERTIES FOR ALL BALLS
;-------------------------------------------------------------
history id=1 nstep=10 ball xvel id=201
history id=2 nstep=10 ball yvel id=201
history id=3 nstep=50 ball xposition id=201
history id=4 nstep=50 ball yposition id=201
history id=9 nstep=50 Real_Time
history id=10 nstep=10 Power
;
;-------------------------------------------------------------
```

```
; RUN FOR 5000 ITERATIONS - NO ROTATION
;-------------------------------------------------------------------
solve average=1e-20 maximum=1e-20 cycle=900000000 clock=100000000 time
0.1
save temp.sav
;
;-------------------------------------------------------------------
; RUN WITH ROTATION
;-------------------------------------------------------------------
set echo off
Shell_Rotate
wall id=999 spin Shell_RotationSpeed x 0 xv 0 y 0 yv 0
set echo on
;
solve average=0.001 maximum=0.001 cycle=900000000 clock=100000000 time
3.1
;
```

```
;-------------------------------------------------------------
; OUTPUT
;-------------------------------------------------------------
plot create history1
plot add history 3 4 vs 9
plot show
plot create history2
plot add history 10 vs 9
plot show
history write 3 4 vs 9 filename.his
save filename.sav
```

C.2 Data file used for the Corrugated liner

```
new
    set max_balls 1000
    set gen_error on
    set random
    set disk on
    set pinterval=100
    trace energy on
    set logfile temp.log
    set log on overwrite
    define MillPower
    while_stepping
    Real_Time = time
    Power = e_bound / time
    end
    ;-----------------------------------------------------------------
    ; DEFINE MILL WALL
    ;-----------------------------------------------------------------
    define Shell_Corrugated
    ;
    xc = Shell_Xcenter ; x - center of circle
    yc = Shell_Ycenter ; y - center of circle
    ;
    omega = Shell_RotationSpeed
    n_stiff = Shell_Nstiff
    s_stiff = Shell_Sstiff
    friction= Shell_Friction
    ;
    dia = Shell_InnerDiameter
    num = Shell_NumSegments
```

```
;
theta = (2*pi) / (num)
;
loop i (1,num)
   radius1=(dia * 0.5) + (0.003 * (cos (16*(i-1)*theta)))
   radius2=(dia * 0.5) + (0.003 * (cos (16*(i)*theta)))
   x1 = radius1 * cos ((i-1)*theta)
   y1 = radius1 * sin ((i-1)*theta)
   x2 = radius2 * cos ((i)*theta)
   y2 = radius2 * sin ((i)*theta)
;
   command
      wall id i kn n_stiff ks n_stiff fric friction &
            ; spin omega x xc xv 0.0 y yc yv 0.0 &
      nodes x1,y1 x2,y2
   endCommand
endLoop
end
;
define Shell_Rotate
xc = Shell_Xcenter ; x - center of circle
yc = Shell_Ycenter ; y - center of circle
;
num = Shell_NumSegments
;
omega = Shell_RotationSpeed
loop i (1,num)
   command
      wall id=i spin omega x xc xv 0.0 y yc yv 0.0
   endCommand
endLoop
end
;
```

```
;-----------------------------------------------------------------
; GENERATE BALLS
;-----------------------------------------------------------------
define Balls_Packing
xc = x0
yc = y0
rc = radius
idc = id_start
r2 = 2.0 * radius
yinc = radius * sqrt(3.0)
   loop row (1,n_row)
      loop col (1,n_col)
         command
            ball id=idc x=xc y=yc rad=rc
         end_command
      idc = idc + 1
      xc = xc + r2
      end_loop
   yc = yc + yinc
   xc = x0 + radius * (row - (row/2) * 2)
   end_loop
end
;
```

```
;-------------------------------------------------------------------------
; DEFINE MILL PARAMETERS
;-------------------------------------------------------------------------
set Shell_InnerDiameter = .404
set Shell_NumSegments = 360
set Shell_Xcenter = 0.0
set Shell_Ycenter = 0.0
set Shell_Nstiff = 1e7
set Shell_Sstiff = 1e7
set Shell_RotationSpeed = 2.64
set Shell_Friction = 0.337
;
;
;-------------------------------------------------------------------------
; DEFINE HEX ARRAY OF BALLS PARAMETERS
;-------------------------------------------------------------------------
set x0 = 0
set y0 = -0.182
set id_start=201
set radius=0.001
set n_col=7
set n_row=7
;
;-------------------------------------------------------------------------
; BUILD MODEL
;-------------------------------------------------------------------------
set echo off
Shell_Corrugated
Balls_Packing
set echo on
;
;-------------------------------------------------------------------------
; SET GLOBAL MATERIAL PROPERTIES FOR ALL BALLS
```

```
;-----------------------------------------------------------------
property density 7800
property kn = 1e6
property ks = 1e6
property fric = 0.194
;
;-----------------------------------------------------------------
; MAKE CLUMP OF ALL BALLS
;-----------------------------------------------------------------
property n_bond=1e10
property s_bond=1e10
clump id=999, full on, perm, range id=201,1000
;
;-----------------------------------------------------------------
; DEFINE DAMPING
;-----------------------------------------------------------------
define zero_damp
bp = ball_head
loop while bp # null
b_damp(bp) = 0.0
bp = b_next(bp)
end_loop
end
zero_damp
;
define catch_contact_hys
cp = fc_arg(0);
c_model(cp) = 'hysdamp'
c_prop(cp,'damp_n')= setv
c_prop(cp,'notension')=1;
end
;
set setv=0.55
```

```
model hysdamp
set fishcall 6 catch_contact_hys
;
;-------------------------------------------------------------------------
; SET GLOBAL GRAVITY
;-------------------------------------------------------------------------
set grav 0 -9.81
;
;-------------------------------------------------------------------------
; PLOTTING ROUTINES
;-------------------------------------------------------------------------
plot create pic
plot set title text 'clump geometry 7x7@rad 1.0mm'
plot add ball lblue
plot add clump
plot add wall black
plot add axes brown
plot show
;
```

```
;-------------------------------------------------------------
; ADDS SHELL MARKER
;-------------------------------------------------------------
wall id=999 nodes 0.2,0 .21,0
;
;-------------------------------------------------------------
; SET GLOBAL MATERIAL PROPERTIES FOR ALL BALLS
;-------------------------------------------------------------
history id=1 nstep=10 ball xvel id=201
history id=2 nstep=10 ball yvel id=201
history id=3 nstep=50 ball xposition id=201
history id=4 nstep=50 ball yposition id=201
history id=9 nstep=50 Real_Time
history id=10 nstep=10 Power
;
;-------------------------------------------------------------
; RUN FOR 5000 ITERATIONS - NO ROTATION
;-------------------------------------------------------------
solve average=1e-20 maximum=1e-20 cycle=900000000 clock=100000000 time
0.1
save temp.sav
;
;-------------------------------------------------------------
; RUN WITH ROTATION
;-------------------------------------------------------------
set echo off
Shell_Rotate
wall id=999 spin Shell_RotationSpeed x 0 xv 0 y 0 yv 0
set echo on
solve average=0.001 maximum=0.001 cycle=900000000 clock=100000000 time
1.1
;
;-------------------------------------------------------------
```

```
; OUTPUT
;-------------------------------------------------------------------------------
plot create history1
plot add history 3 4 vs 9
plot show
;
plot create history2
plot add history 10 vs 9
plot show
;
history write 3 4 vs 9 filename.his
save filename.sav
```

C.3 Data file used for the Skorupa liner

```
new
    set max_balls 1000
    set gen_error on
    set random
    set disk on
    set pinterval=100
    trace energy on
    set logfile temp.log
    set log on overwrite
    define MillPower
    while_stepping
    Real_Time = time
    Power = e_bound / time
    end
    ;
    ;-------------------------------------------------------------------
    ; DEFINE MILL
    ;-------------------------------------------------------------------
    define Shell_Skorupa
    xc = Shell_Xcenter ; x - center of circle
    yc = Shell_Ycenter ; y - center of circle
    ;
    omega = Shell_RotationSpeed
    n_stiff = Shell_Nstiff
    s_stiff = Shell_Sstiff
    friction= Shell_Friction
    ;
    dia = Shell_InnerDiameter
    ;
    array abc (8)
    array inner_radius (8)
    array theta_inner (8)
```

```
abc(1)=0
abc(2)=5.5*pi/180
abc(3)=9*pi/180
abc(4)=13.5*pi/180
abc(5)= 21.5*pi/180
inner_radius(1)=0.1925
inner_radius(2)=0.1925
inner_radius(3)=0.1792
inner_radius(4)=0.1792
inner_radius(5)=0.1925
;
count=0
;
loop i (1,16)
   loop j (1,5)
     if j=5 then
        theta1=abc(5)+(22.5*pi/180*(i-1))
        theta2=abc(1)+(22.5*pi/180*(i))
        x1 = inner_radius(5) * cos (theta1)
        y1 = inner_radius(5) * sin (theta1)
        x2 = inner_radius(1) * cos (theta2)
        y2 = inner_radius(1) * sin (theta2)
        count = count + 1
     else
        theta1=abc(j)+(22.5*pi/180*(i-1))
        theta2=abc(j+1)+(22.5*pi/180*(i-1))
        x1 = inner_radius(j) * cos (theta1)
        y1 = inner_radius(j) * sin (theta1)
        x2 = inner_radius(j+1) * cos (theta2)
        y2 = inner_radius(j+1) * sin (theta2)
        count = count + 1
     endif
     command
```

```
            wall id count kn n_stiff ks n_stiff fric friction &
              ; spin omega x xc xv 0.0 y yc yv 0.0 &
            nodes x1,y1 x2,y2
        endCommand
    endLoop
endLoop
end
;
define Shell_Rotate
xc = Shell_Xcenter ; x - center of circle
yc = Shell_Ycenter ; y - center of circle
;
omega = Shell_RotationSpeed
loop i (1,count)
    command
        wall id=i spin omega x xc xv 0.0 y yc yv 0.0
    endCommand
endLoop
end
;
```

```
;-------------------------------------------------------------------------
; GENERATES BALLS
;-------------------------------------------------------------------------
define Balls_Packing
xc = x0
yc = y0
rc = radius
idc = id_start
r2 = 2.0 * radius
yinc = radius * sqrt(3.0)
loop row (1,n_row)
   loop col (1,n_col)
      command
         ball id=idc x=xc y=yc rad=rc
      end_command
      idc = idc + 1
      xc = xc + r2
   end_loop
yc = yc + yinc
xc = x0 + radius * (row - (row/2) * 2)
end_loop
end
;
;-------------------------------------------------------------------------
; DEFINE MILL PARAMETERS
;-------------------------------------------------------------------------
set Shell_InnerDiameter = .404
set Shell_Xcenter = 0.0
set Shell_Ycenter = 0.0
set Shell_Nstiff = 1e8
set Shell_Sstiff = 1e8
set Shell_RotationSpeed = 2.64
set Shell_Friction = 0.337
```

```
;
;-------------------------------------------------------------------
; DEFINE HEX ARRAY OF BALLS PARAMETERS
;-------------------------------------------------------------------
set x0 = 0
set y0 = -0.182
set id_start=201
set radius=0.001
set n_col=7
set n_row=7
;
;-------------------------------------------------------------------
; BUILDS MODEL
;-------------------------------------------------------------------
set echo off
Shell_Skorupa
Balls_Packing
set echo on
;
;-------------------------------------------------------------------
; SET GLOBAL MATERIAL PROPERTIES FOR ALL BALLS
;-------------------------------------------------------------------
property density 7800
property kn = 1e7
property ks = 1e7
property fric = 0.194
;
;-------------------------------------------------------------------
; MAKE CLUMP OF ALL BALLS
;-------------------------------------------------------------------
property n_bond=1e10
property s_bond=1e10
clump id=999, full on, perm, range id=201,1000
```

```
;
;-----------------------------------------------------------------
; DEFINE DAMPING
;-----------------------------------------------------------------
define zero_damp
bp = ball_head
loop while bp # null
b_damp(bp) = 0.0
bp = b_next(bp)
end_loop
end
zero_damp
;
define catch_contact_hys
cp = fc_arg(0);
c_model(cp) = 'hysdamp'
c_prop(cp,'damp_n')= setv
c_prop(cp,'notension')=1;
end
;
set setv=0.995
model hysdamp
set fishcall 6 catch_contact_hys
;
;-----------------------------------------------------------------
; SET GLOBAL GRAVITY
;-----------------------------------------------------------------
set grav 0 -9.81
;
;-----------------------------------------------------------------
; PLOTTING ROUTINES
;-----------------------------------------------------------------
plot create pic
```

```
plot set title text 'clump geometry 7x7@rad 1.0mm'
plot add ball lblue
plot add clump
plot add wall black
plot add axes brown
plot show
;
;-----------------------------------------------------------------
; ADDS SHELL MARKER
;-----------------------------------------------------------------
wall id=999 nodes 0.2,0 .21,0
;
;-----------------------------------------------------------------
; SET GLOBAL MATERIAL PROPERTIES FOR ALL BALLS
;-----------------------------------------------------------------
history id=1 nstep=10 ball xvel id=201
history id=2 nstep=10 ball yvel id=201
history id=3 nstep=50 ball xposition id=201
history id=4 nstep=50 ball yposition id=201
history id=9 nstep=50 Real_Time
history id=10 nstep=10 Power
;
;-----------------------------------------------------------------
; RUN FOR 5000 ITERATIONS - NO ROTATION
;-----------------------------------------------------------------
solve average=1e-20 maximum=1e-20 cycle=900000000 clock=100000000 time
0.1
save temp.sav
;
;-----------------------------------------------------------------
; RUN WITH ROTATION
;-----------------------------------------------------------------
set echo off
```

```
Shell_Rotate
wall id=999 spin Shell_RotationSpeed x 0 xv 0 y 0 yv 0
set echo on
solve average=0.001 maximum=0.001 cycle=900000000 clock=100000000 time
2.59
;
;-------------------------------------------------------------------
; OUTPUT
;-------------------------------------------------------------------
plot create history1
plot add history 3 4 vs 9
plot show
plot create history2
plot add history 10 vs 9
plot show
;
history write 3 4 vs 9 filename.his
save filename.sav
```

C.4 Matlab code sued to process results

```
clear all
    clc
    load filename.his;
    %Load in time values
    dummy(:,1)= filename (:,1);
    %Load in x values
    dummy(:,2)= filename (:,2);
    %Load in y values
    dummy(:,3)= filename (:,3);

    index=size(dummy,1)
    time=dummy(1:index,1);
    dummy_time=input('enter time of one revolution in rad/s ');
    theta_shell=time.*dummy_time;
    data=dummy(1:index,2:3);
    displacement_data(:,1)=displacement_data(:,1);
    displacement_data(:,2)=displacement_data(:,2);
    [theta_ data,R_ data] = cart2pol(displacement_data(:,1),displacement_data(:,2));

    interval_flag=0;
    while interval_flag==0
        lower_bound=input('Enter the lower boundary for the time interval to be
plotted ');
        upper_bound=input('Enter the upper boundary for the time interval to be
plotted ');
        %Plot displacements for the given time intervals for each node
        figure(1)
      polar(theta_ data (lower_bound:upper_bound),R_ data (lower_bound:upper_bound))
        title(['Motion of particle'])
        %Enter whether you are satisfied with the chosen time interval
        verdict=input('Satisfied with time intervals plotted? (yes=1 no=0)');
        %If yes then jump out of loop
```

```
        if verdict==1
            interval_flag=1
        end
    end

    temp_ data (:,1)=theta_ data (lower_bound:upper_bound);
    flag=0;
    while flag ==0
        [theta_depart,i]=max(temp_ data);
        real_i=i+lower_bound;
        radius=R_ data (real_i);
        if radius<0.165
            temp_ data (i)=-20;
        else
            flag=1
        end
    end

    AOD_=(theta_depart).*(180/pi)
```

Appendix D

Numerical and Experimental Results

Table 1: Numerical Values for the Smooth Liner

% Crit Speed	ELFEN	ABAQUS	PFC^{2D}
30%	-60.8685	-56.3251	-55.2298
60%	-41.8764	-37.3341	-37.8448
105%	-6.5488	2.2788	-13.3272
120%	-6.2963	-9.4272	-1.688
135%	-1.0558	2.0565	0.8395
165%	-0.1621	2.3214	-0.0092

Table 2 : Experimental Values for the Smooth Liner [1]

% Crit Speed	Lower Exp Values	Upper Exp Values
30%	-61.6	-57.8
60%	-41.3	-39.7
105%	-5.6	-3.7
120%	6.4	17.6
135%	17.4	25.8
165%	37.4	60

Table 3 :Experimental Values for the Corrugated Liner [2]

% Crit Speed	Lower Exp Values	Upper Exp Values
36%	-63	-55.8
50%	-37	-33
64%	-33.1	-29.5
86%	-17.5	-12.5
114%	7.5	9.7

Table 4 :Numerical Values for ELFEN for the Corrugated Liner

% Crit Speed	ELFEN (f = 0.9)	ELFEN (f = 0.2)
36%	-40.68	-43.4849
50%	-51.1844	-36.2931
64%	-43.0307	-27.9819
86%	-1.936	-1.445
114%	-14.7879	1.0553

% Crit Speed	ELFEN (with Damping)	Damping Value
36%	-44.1346	0
50%	-38.9916	0.007
64%	-30.3359	0.01
86%	-14.27	0.02
114%	-7.8569	0.02

Table 5 : Numerical Values for ABAQUS with Kinematic Contact for the Corrugated Liner

% Crit Speed	Damping = 0	Damping = 0.1	Damping = 0.2
36%	-44.6	-44.992	-47.8897
50%	-48.91	-45.5955	-45.3573
64%	-42.4337	-59.0848	-58.6217
86%	-54.4906	-39.1301	-51.5243
114%	-63.76	-33.8362	-65.834

% Crit Speed	Damping = 0.3	Damping = 0.4
36%	-50.956	-44.754
50%	-49.4721	-43.8474
64%	-60.4627	-60.1767
86%	-25.3247	-32.7724
114%	-31.5834	-67.9059

Table 6 : Numerical Values for ABAQUS with Penalty Contact for the Corrugated Liner

% Crit Speed	Damping = 0	Damping = 0.1	Damping = 0.2
36%	-44.1328	-48.2279	-43.9937
50%	-60.7179	-53.6883	-45.0285
64%	-66.0689	-54.4801	-66.1353
86%	-53.4204	-44.3987	-52.3621
114%	-62.498	-69.1546	-61.6686

% Crit Speed	Damping = 0.3	Damping = 0.4
36%	-48.9121	-47.7608
50%	-44.3776	-50.4491
64%	-66.5277	-64.6194
86%	-50.1255	-49.7299
114%	-58.822	-68.4814

Table 7 : Numerical Values for PFC^{2D} with Damping for the Corrugated Liner

% Crit Speed	PFC^{2D}	Damping Value Used
36%	-58.8409	0.9
50%	-36.6848	0.875
64%	-32.9599	0.86
86%	-18.9161	0.6
114%	-5.9177	0.5

Table 8 : Experimental Values for the Skorupa Liner [2]

% Crit Speed	Lower Exp Values	Upper Exp Values
36%	-31.9	-23.2
50%	-26.3	-20.6
64%	-28.3	-7
86%	-21.6	-12.4
114%	-0.2	10.6

Table 9 : Numerical Values for ELFEN for the Skorupa Liner

% Crit Speed	ELFEN (f = 0.9)	ELFEN (f = 0.2)
36%	-28.2282	-21.7323
50%	-12.3683	-12.5616
64%	-0.9201	4.113
86%	-0.5876	1.8285
114%	58.556	43.1388

Table 10 : Numerical Values for ABAQUS with Kinematic Contact for the Skorupa Liner

% Crit Speed	Damping = 0	Damping = 0.1	Damping = 0.2
36%	-36.8029	-45.5659	-45.6901
50%	-53.6456	-54.5401	-60.3624
64%	-10.659	-11.2127	-14.78
86%	-26.0535	-27.4866	-29.0546
114%	-1.0198	-17.8799	-19.8233

% Crit Speed	Damping = 0.3	Damping = 0.4
36%	-46.0878	-51.24
50%	-60.3265	-24.4045
64%	-19.3839	-17.5169
86%	-29.0739	-29.6361
114%	-21.0923	-22.4819

Table 11 : Numerical Values for ABAQUS with Kinematic Contact for 36% critical Speed for the Skorupa Liner

Damping Value	Angle of Departure
0	-36.8029
0.05	-56.6581
0.1	-45.5659
0.15	-50.6485
0.2	-45.6901
0.25	-43.0909
0.3	-46.0878
0.35	-39.2243
0.4	-51.24

Table 12 : Numerical Values for ABAQUS withPenalty Contact for the Skorupa Liner

% Crit Speed	Damping = 0	Damping = 0.1	Damping = 0.2
36%	-42.8924	-45.4643	-45.1692
50%	-47.2870	-48.5086	-32.1227
64%	-6.083	-14.8016	-15.4424
86%	-44.1125	-44.8818	-29.0546
114%	-7.2339	-11.2875	-13.9529

% Crit Speed	Damping = 0.3	Damping = 0.4
36%	-45.1022	-49.269
50%	-54.2646	-55.625
64%	-16.6529	-14.3057
86%	-45.6175	-42.0298
114%	-16.0344	-41.9937

Table 13 : Numerical Values for ABAQUS with Penalty Contact for 36% critical Speed for the Skorupa Liner

Damping Value	Angle of Departure
0	-42.8924
0.05	-40.9868
0.1	-45.4643
0.15	-34.727
0.2	-45.1692
0.25	-42.4725
0.3	-45.1022
0.35	-39.7287
0.4	-49.269

Table 14 : Numerical Values for PFC^{2D} for the Skorupa Liner

% Crit Speed	PFC^{2D}	Damping Value Used
36%	-28.7854	0.99
50%	-25.8094	0.9845
64%	-13.3601	0.96
86%	-19.7132	0.93
114%	-2.3357	0.915

Appendix E

Figures for ABAQUS (Corrugated Liner)

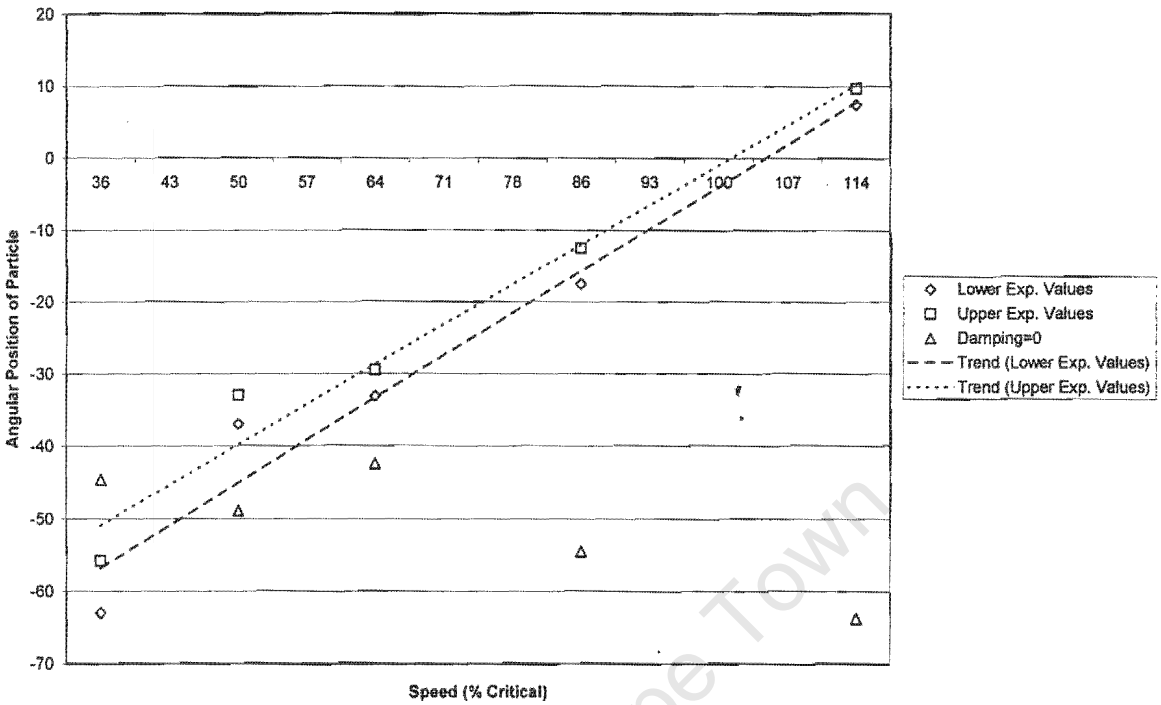


Figure E.1: Kinematic contact with damping = 0 for the Corrugated Liner

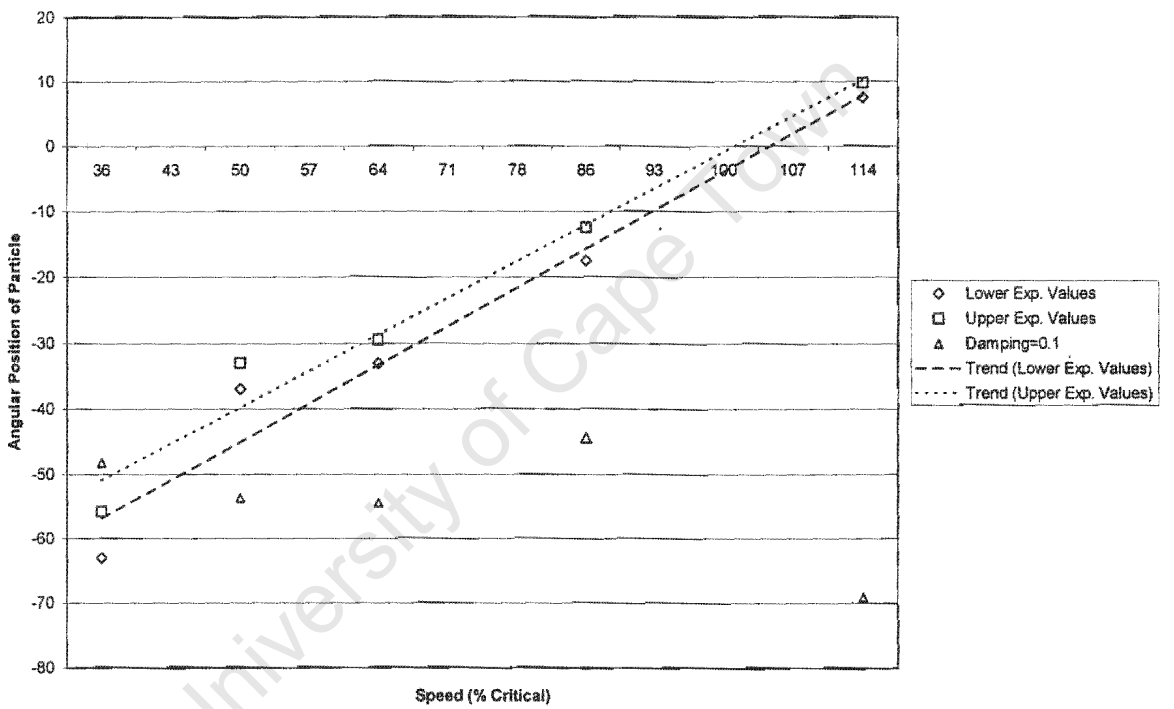


Figure E.2: Kinematic contact with damping = 0.1 for the Corrugated Liner

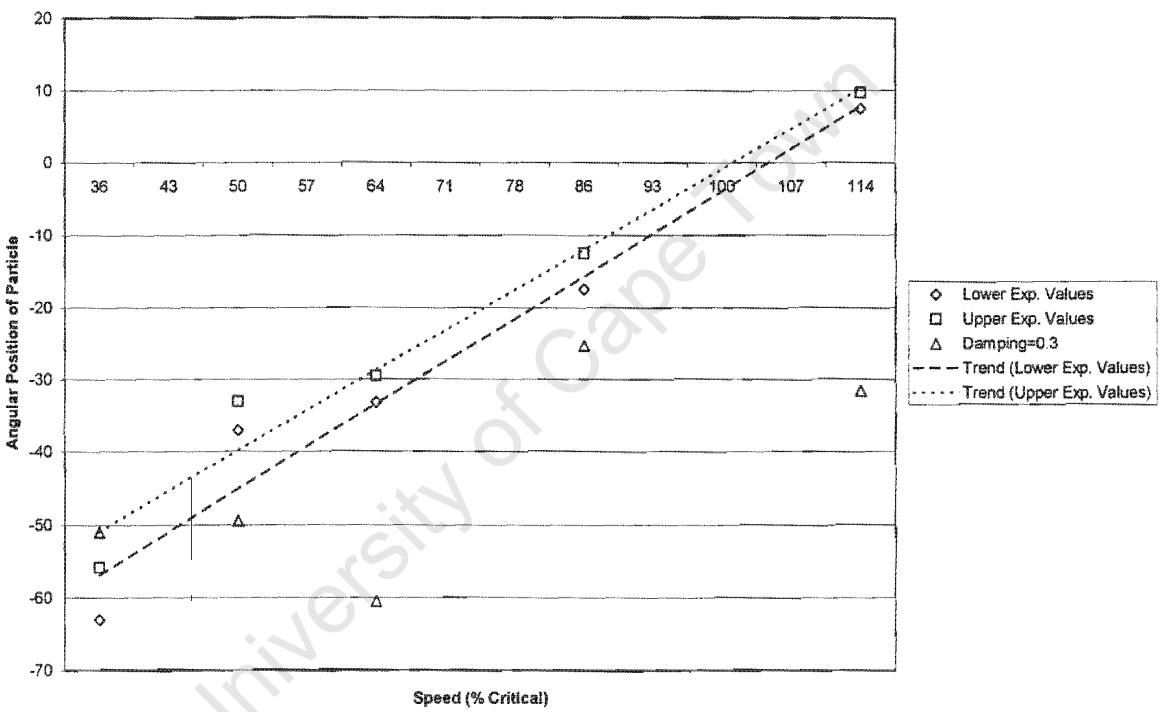


Figure E.3: Kinematic contact with damping = 0.3 for the Corrugated Liner

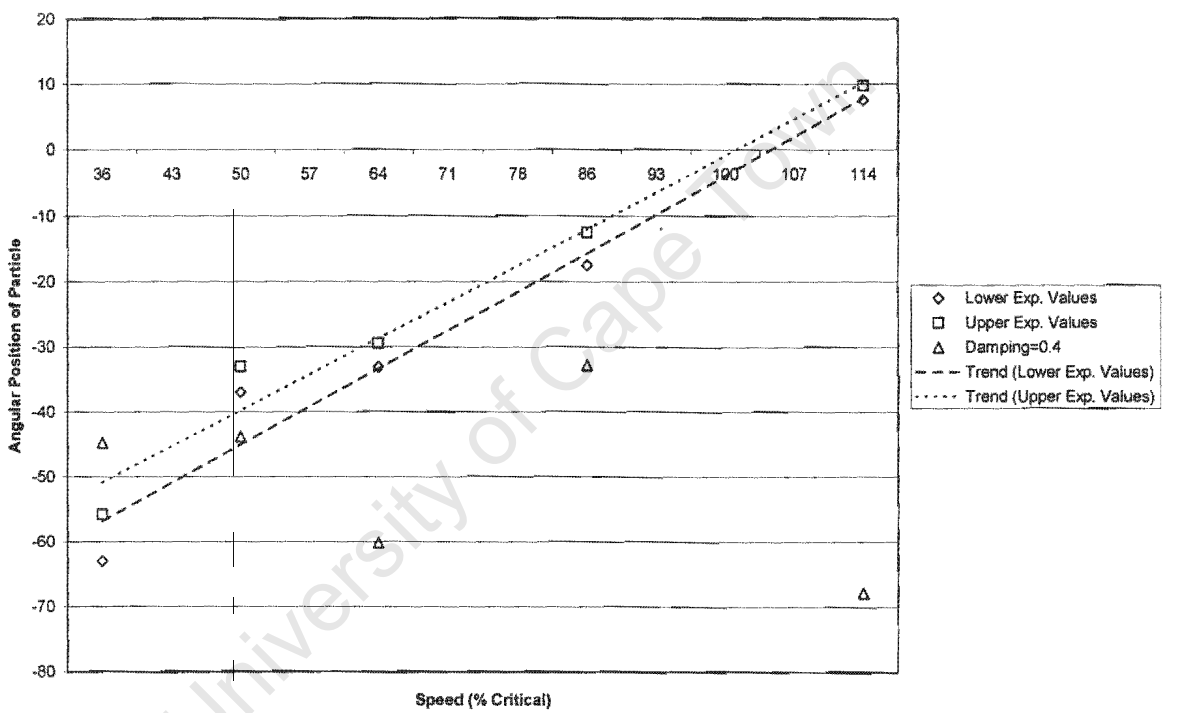


Figure E.4: Kinematic contact with damping = 0.4 for the Corrugated Liner

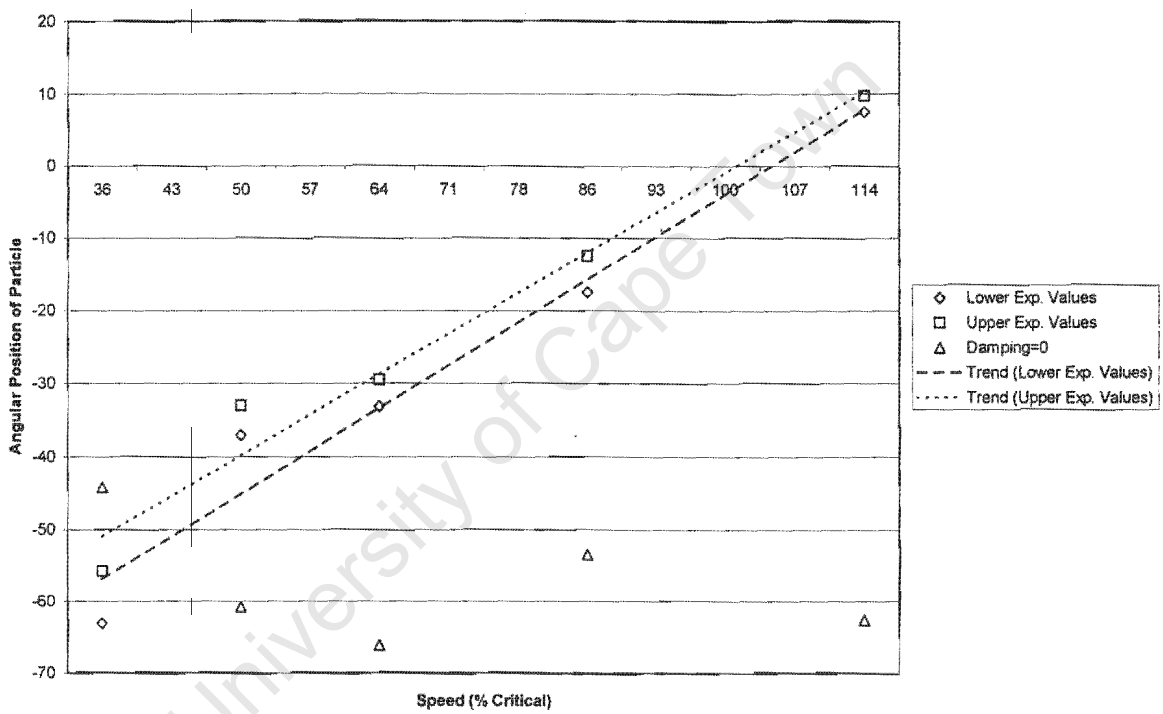


Figure E.5: Penalty contact damping = 0 for the Corrugated Liner

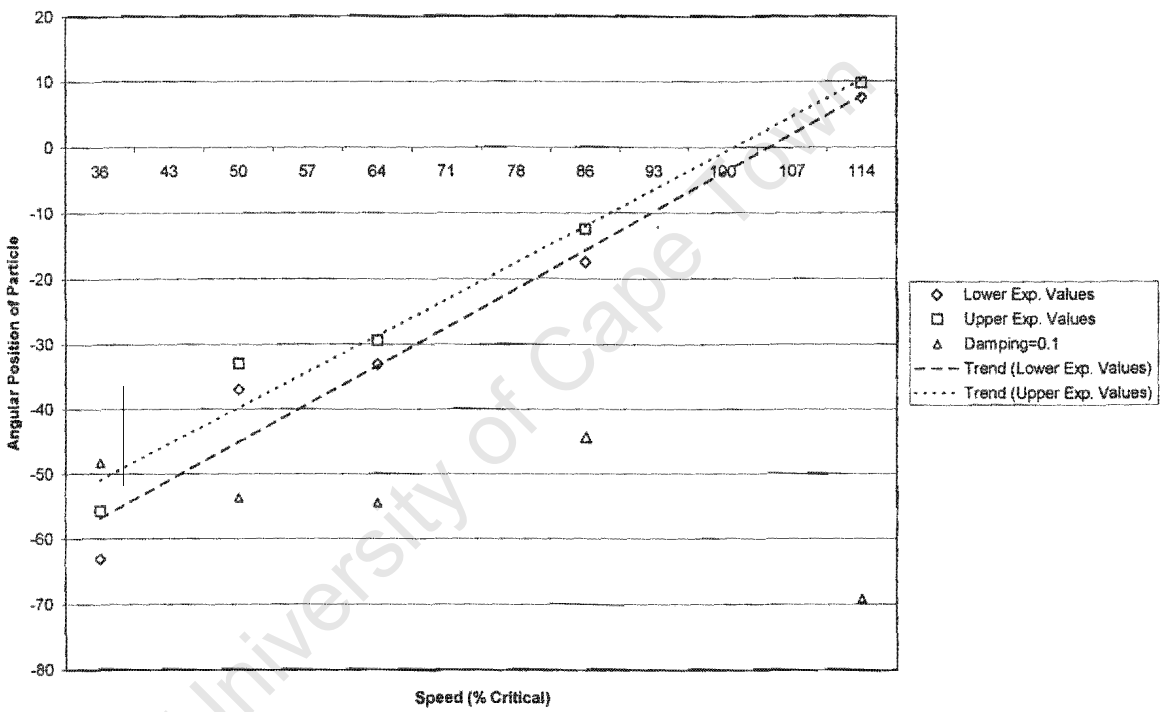


Figure E.6: Penalty contact damping = 0.1 for the Corrugated Liner

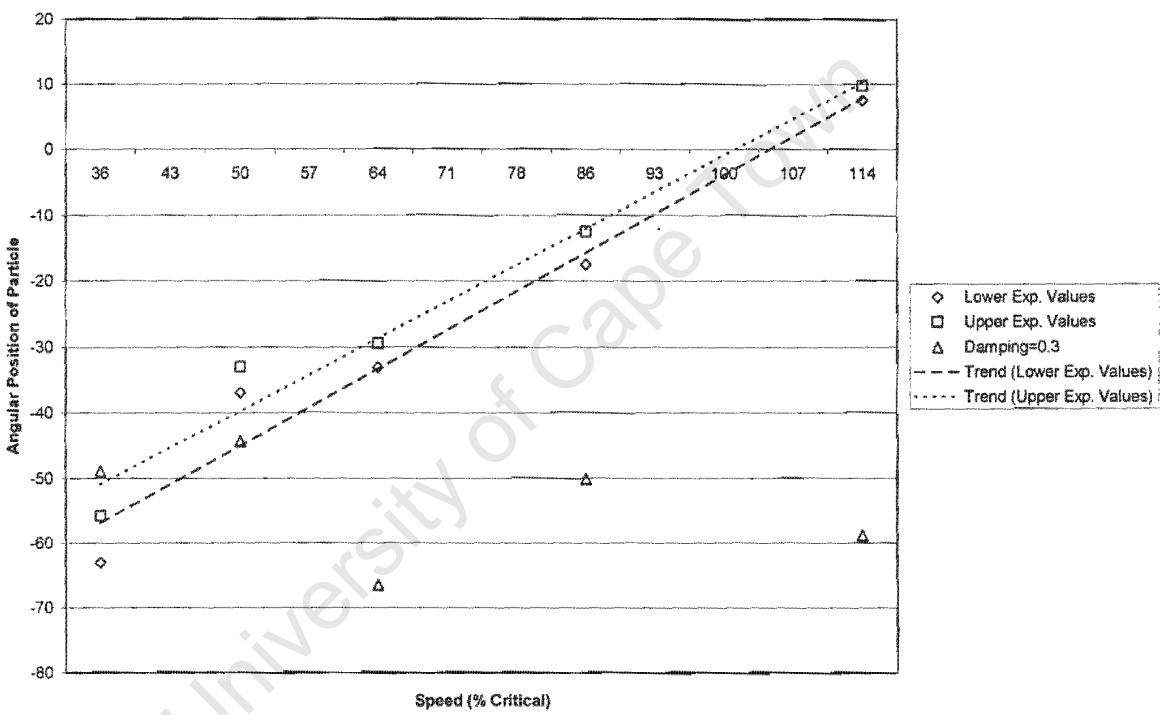


Figure E.7: Penalty contact damping = 0.3 for the Corrugated Liner

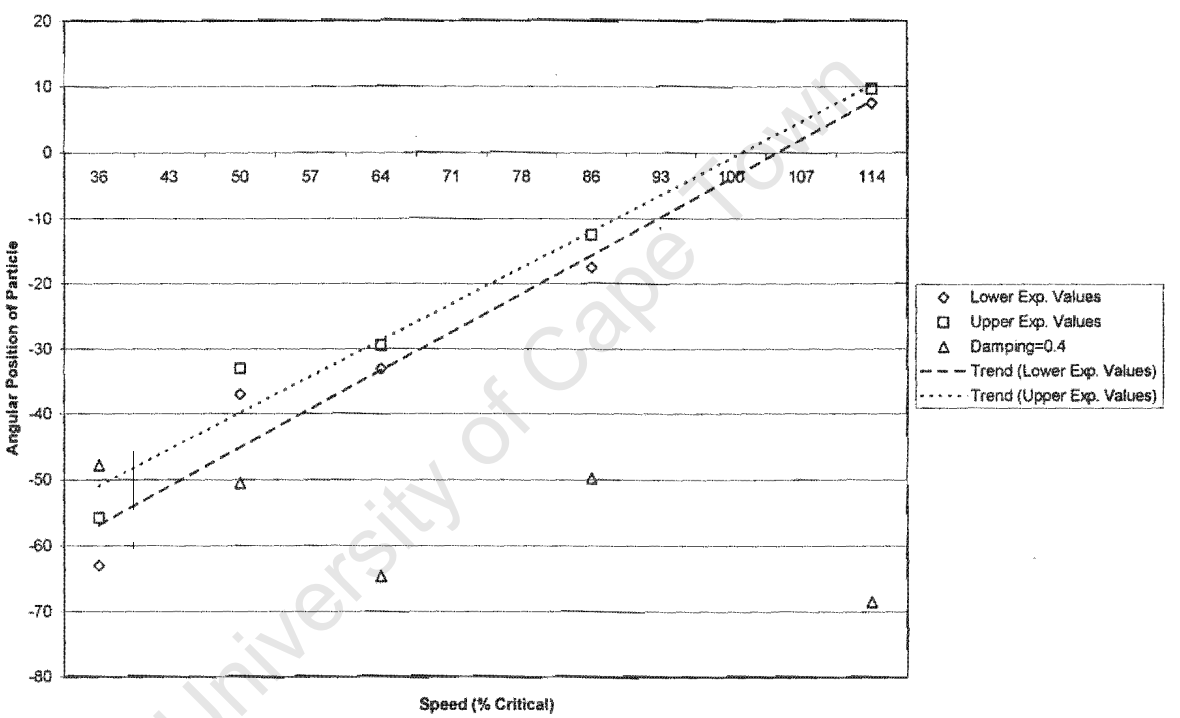


Figure E.8: Penalty contact damping = 0.4 for the Corrugated Liner

Appendix F

Figures for ABAQUS (Skorupa Liner)

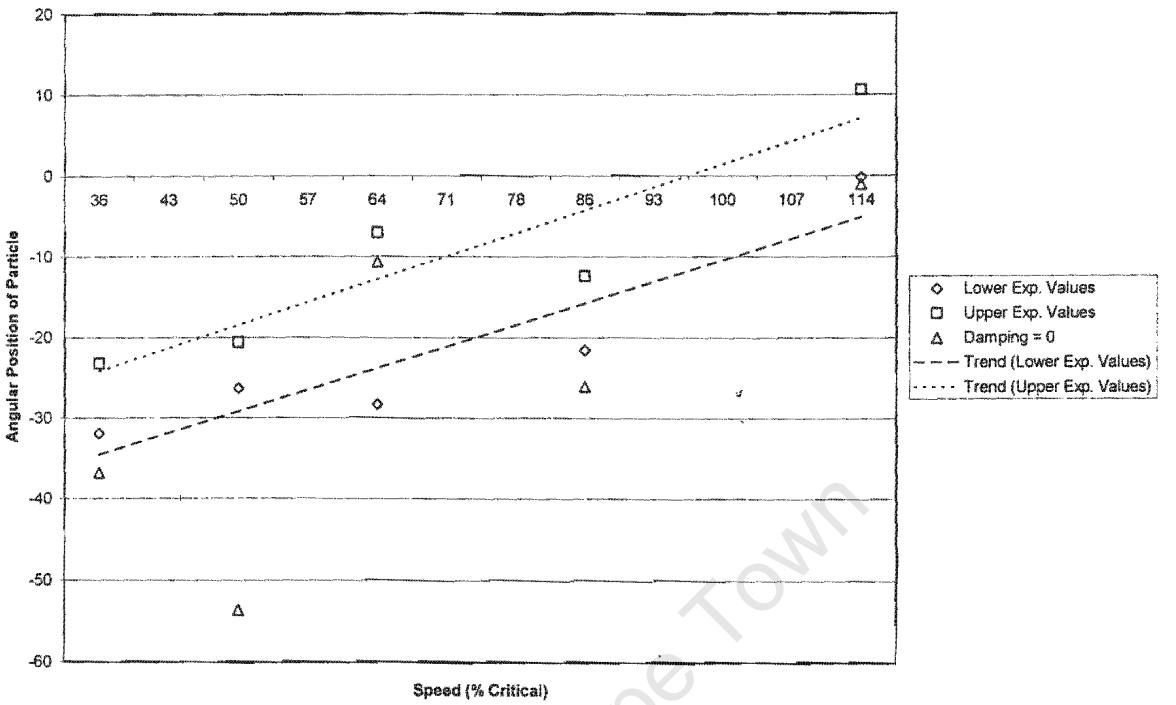


Figure F.1: Kinematic contact with damping = 0 for the Skorupa Liner

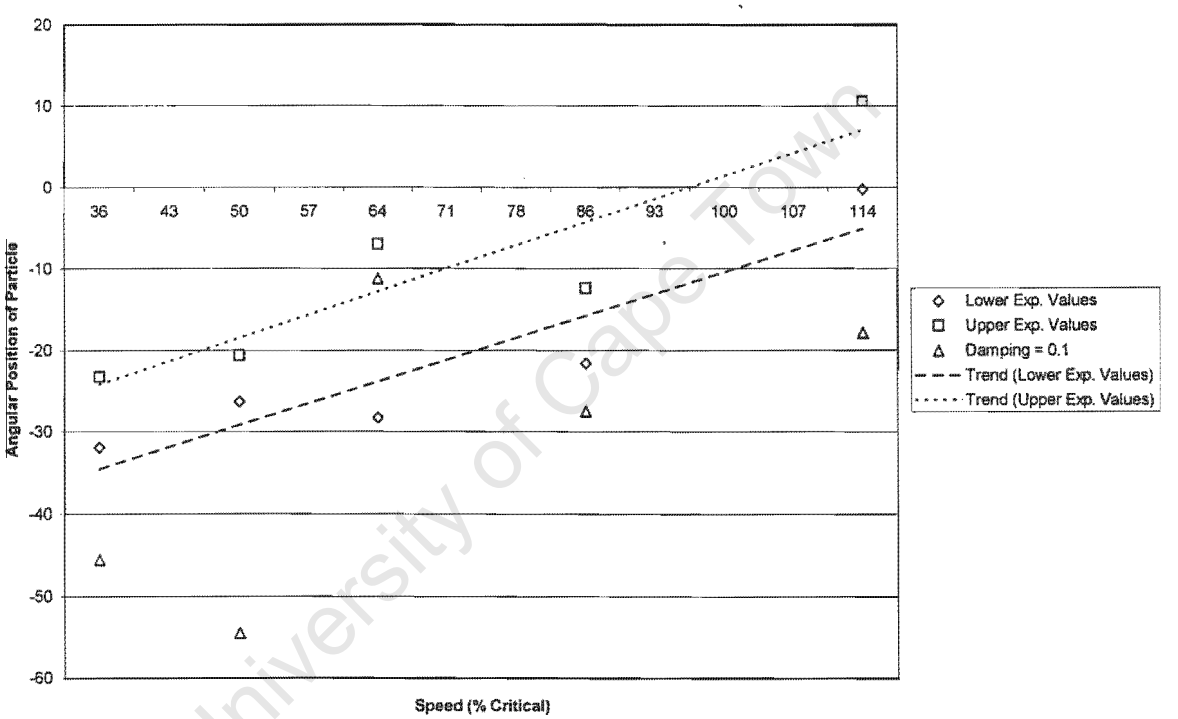


Figure F.2: Kinematic contact with damping = 0.1 for the Skorupa Liner

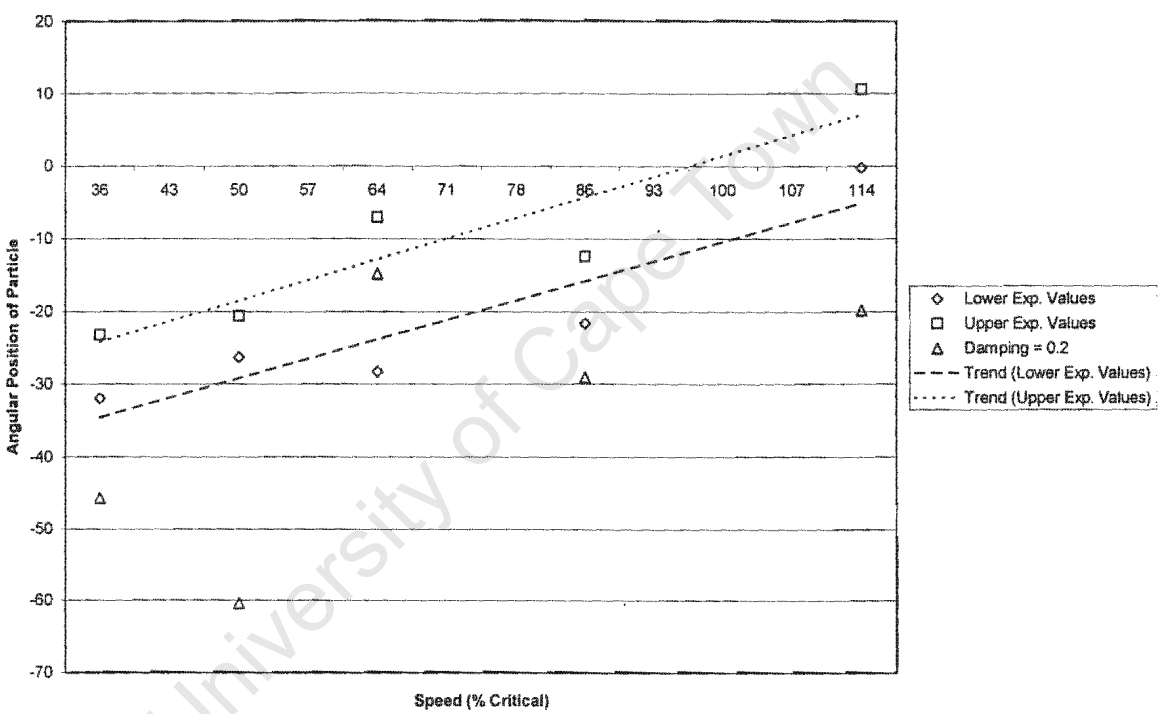


Figure F.3: Kinematic contact with damping = 0.2 for the Skorupa Liner

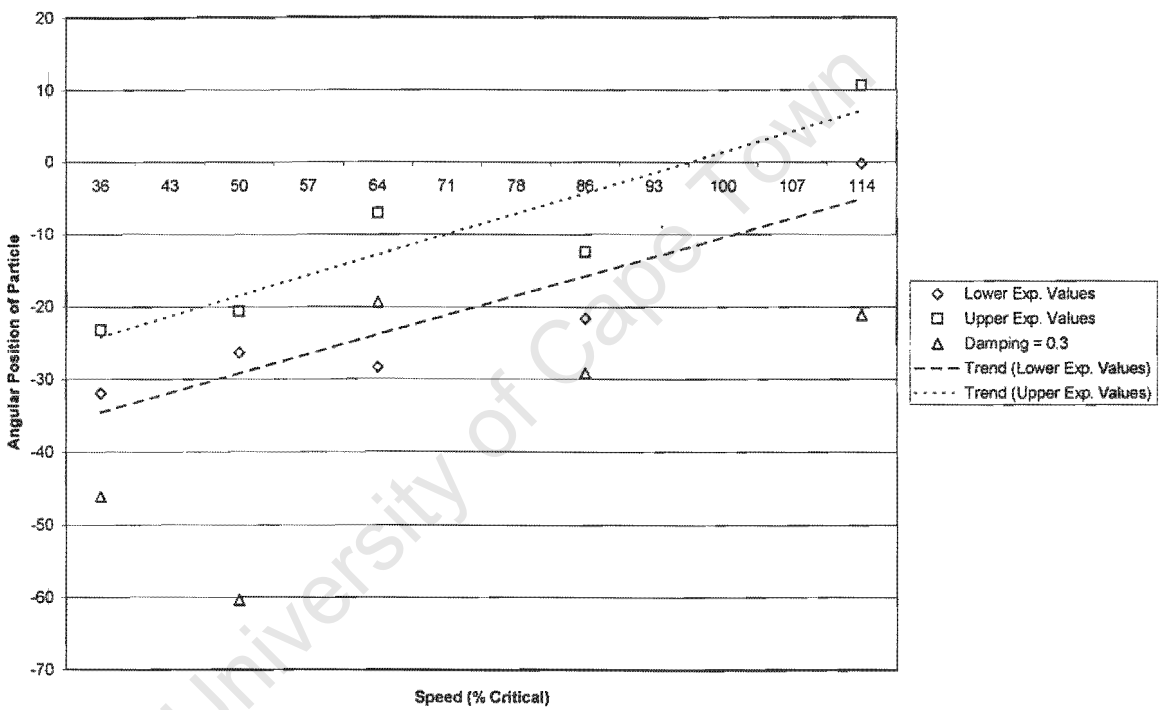


Figure F.4: Kinematic contact with damping = 0.3 for the Skorupa Liner

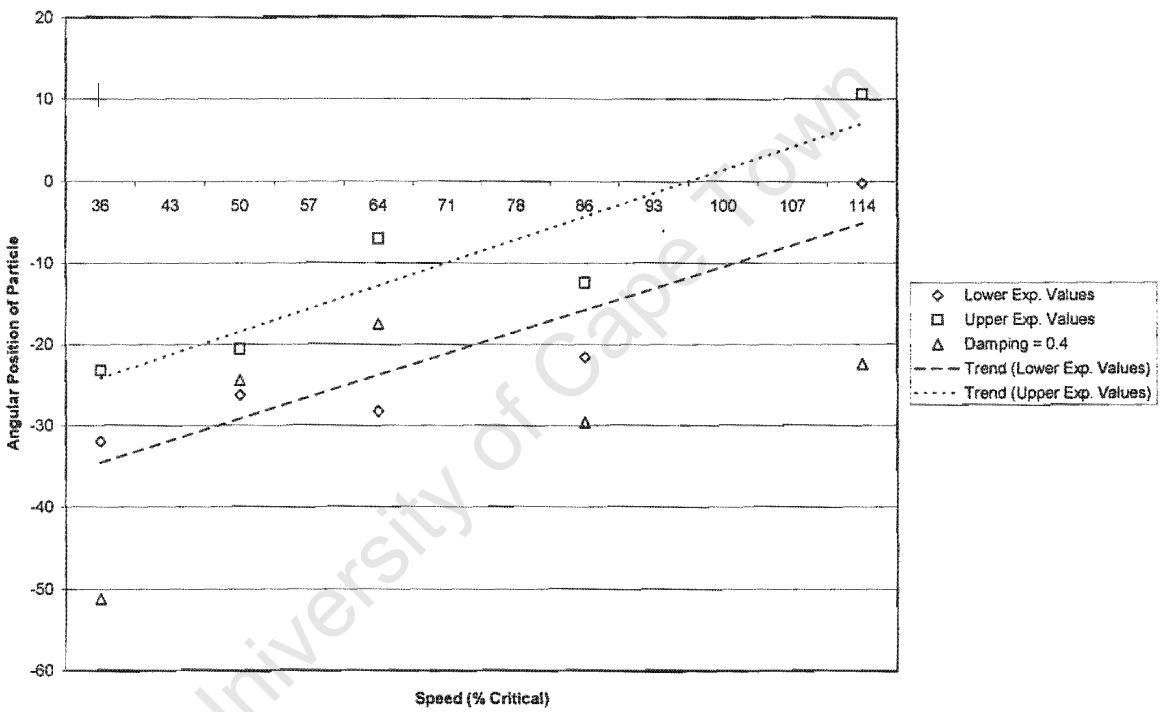


Figure F.5: Kinematic contact with damping = 0.4 for the Skorupa Liner

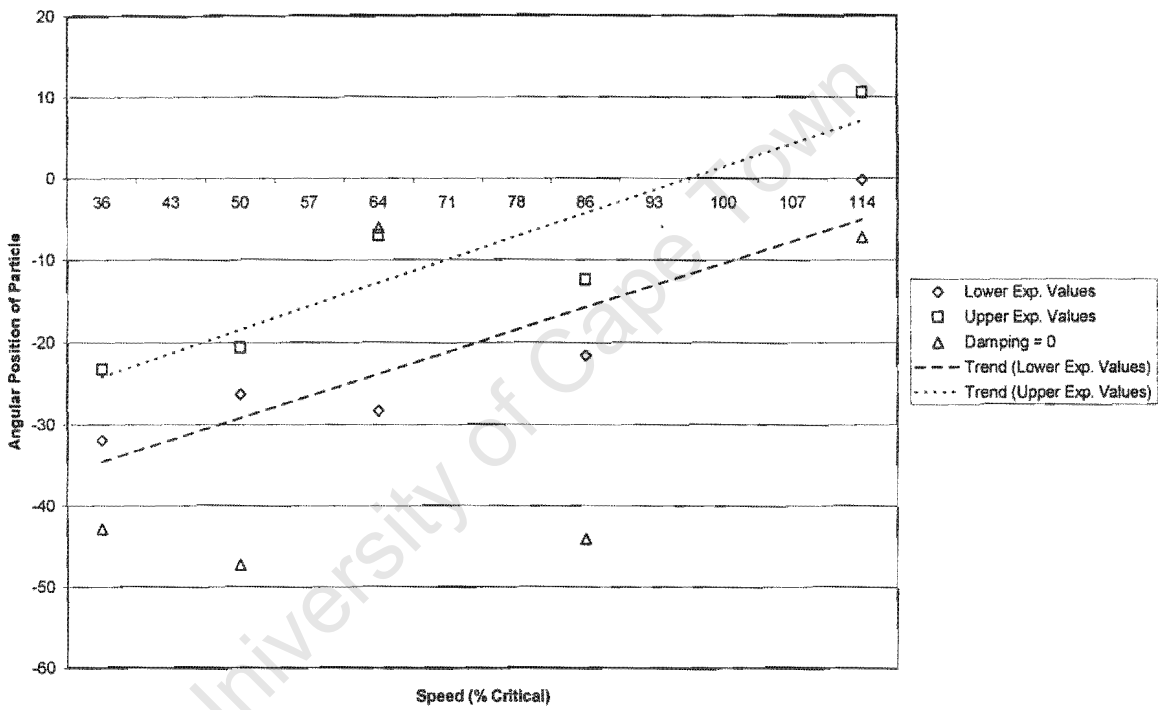


Figure F.6: Penalty contact with damping = 0 for the Skorupa Liner

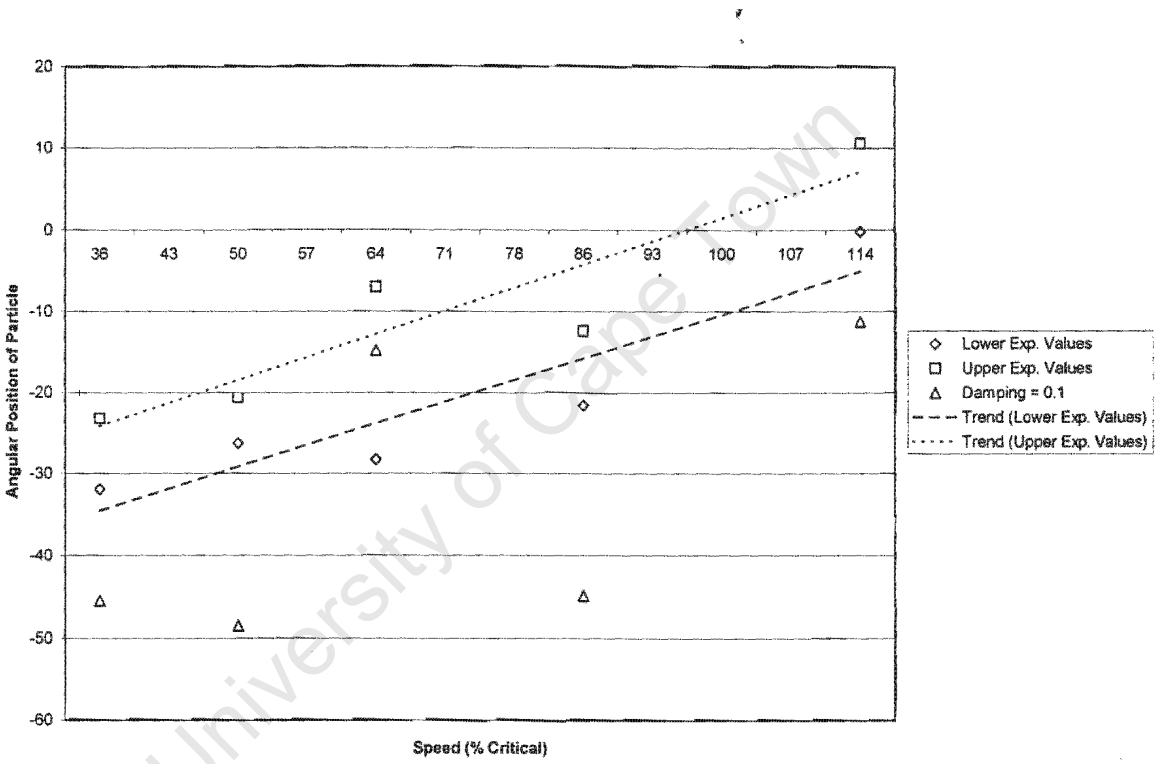


Figure F.7: Penalty contact with damping = 0.1 for the Skorupa Liner

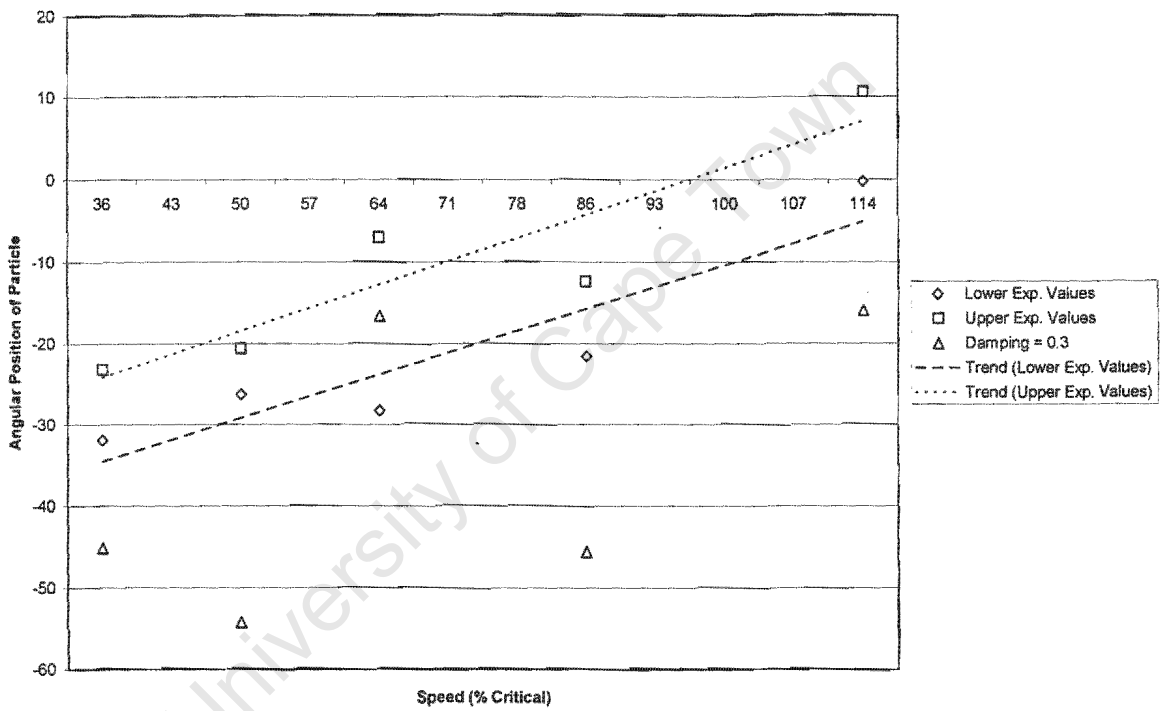


Figure F.8: Penalty contact with damping = 0.3 for the Skorupa Liner

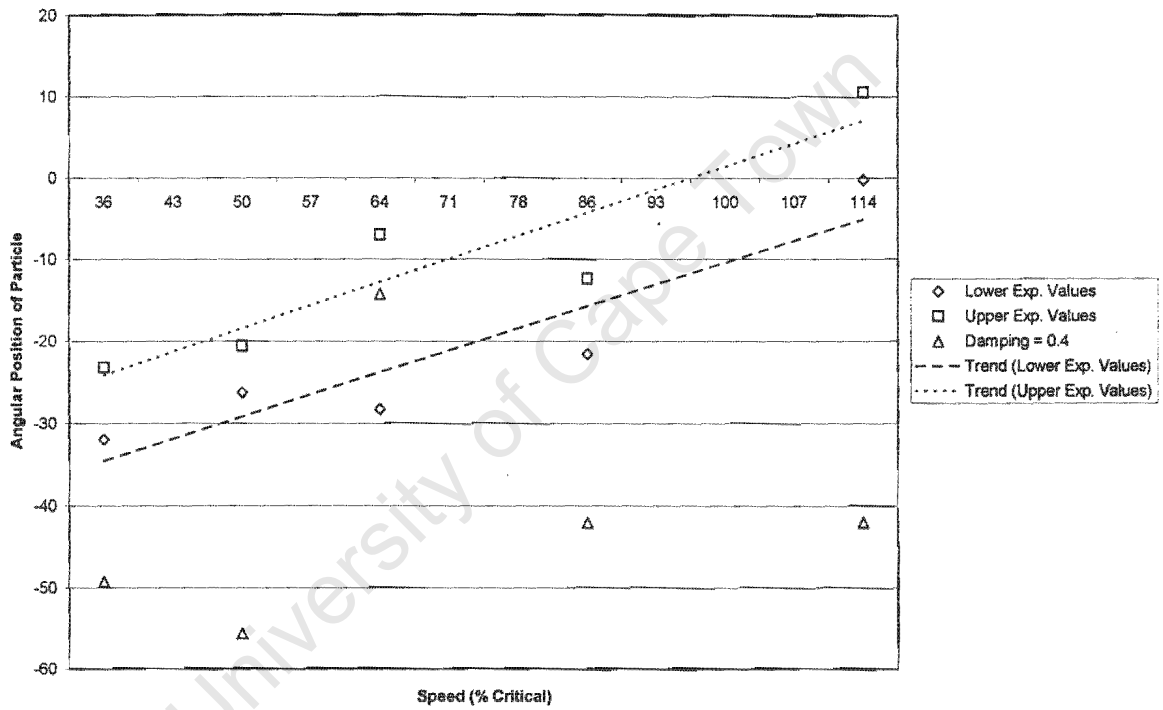


Figure F.9: Penalty contact with damping = 0.4 for the Skorupa Liner