Spring 1-1-2012

The Influence of Representative Volume Element Size, Soil Fabric, and Interparticle Elasto-Plasticity in Three-Dimensional Ellipsoidal Discrete Element Modeling of Granular Assemblies

Yevgeniy Kaufman
University of Colorado at Boulder, yevgeniy.kaufman@colorado.edu

Recommended Citation

Kaufman, Yevgeniy, "The Influence of Representative Volume Element Size, Soil Fabric, and Interparticle Elasto-Plasticity in Three-Dimensional Ellipsoidal Discrete Element Modeling of Granular Assemblies" (2012). *Civil Engineering Graduate Theses & Dissertations*. 283.
https://scholar.colorado.edu/cven_gradetds/283

The Influence of Representative Volume Element Size, Soil Fabric, and Interparticle Elasto-Plasticity in Three-dimensional Ellipsoidal Discrete Element Modeling of Granular Assemblies

by

Yevgeniy Kaufman

A thesis submitted to the
Faculty of the Graduate School of the
University of Colorado in partial fulfillment
of the requirements for the degree of
Concurrent Bachelor's/Master's Degree
Department of Civil, Environmental, and Architectural Engineering
2012

This thesis entitled:

The Influence of Representative Volume Element Size, Soil Fabric, and Interparticle Elasto-Plasticity in Three-dimensional Ellipsoidal Discrete Element Modeling of Granular Assemblies

written by Yevgeniy Kaufman

has been approved for the

Department of Civil, Environmental, and Architectural Engineering

____

Dr. Richard Regueiro

____

Dr. Franck Vernerey

____

Dr. Ronald Pak

Date ________________

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above-mentioned discipline

ABSTRACT

Yevgeniy Kaufman

The Influence of Different Size Representative Volume Elements (RVEs), and Isotropic Compression of Soil Fabric vs. Gravity Deposition Before Triaxial
Compression in Three-dimensional Ellipsoidal Discrete Element Modeling (DEM) of Granular Assemblies in Addition to DEM Interparticle Elasto-Plasticity of Metallic Powder Simulations vs. Abaqus Results.

Directed by Prof. Richard Regueiro, University of Colorado, Boulder

In this comparative study of RVEs, a DEM code ELLIP3D is utilized to simulate quartz sand in triaxial compression tests with particle assemblies attained from an *in situ* fabric generated by synchrotron X-ray microcomputed tomography. In order to attain meaningful results for engineered systems at the macroscopic scale using DEM, RVE volume sizes play a significant role in the accuracy of simulating physical experiments. Simulation time or the number of particles limits the capacity of DEM to simulate large-scale granular systems, which is why it is crucial to determine an efficient quantity and quality of RVEs.

ELLIP3D simulations have been carried out to evaluate a minimum threshold for RVE size for which a comparison of results from various sized boxed particle assemblies provides valuable information on the quantitative behavior of RVEs. The mid-progress and final results of the ELLIP3D simulations are compared to

experimental data from synchrotron micro-computed tomography (SMT) scans. These data containing the initial numerical packing from the scans provide us with the particles' radii, positions, and orientations which are approximated as ellipsoids, which is a current limitation of ELLIP3D, but extension to poly-ellipsoids is being planned. Additionally, a comparison between simulations of triaxial compression on *in situ* fabric versus a slightly displaced fabric due to gravity deposition on an assembly of ellipsoidal particles is explored.

Furthermore, a bilinear elasto-plasticity constitutive model is implemented into ELLIP3D to explore a more realistic phenomenon within the interparticle interaction for materials that exhibit such behavior. Elasto-plastic deformation is appropriate in studies involving metallic powders. The results of a two-particle compaction simulation using DE modeling are compared to finite element (FE) simulations.

ACKNOWLEDGEMENTS

Foremost, I wish to express my utmost appreciation to my advisor, Professor Richard Regueiro for his continuous support, patience, motivation, immense knowledge, and guidance throughout my undergraduate and graduate studies. Additionally, I would not have been able to complete my master's research without the support of everyone who has helped me along the way. Particularly: research associate Dr. Beichuan Yan for the countless hours he spent writing and refining ELLIP3D, BS/MS student Austin Nossokoff for providing me with meaningful DE simulation results and a pleasant collaborative research experience, and Dr. Volkan Isbuga for his mentoring. Dr. Franck Vernerey for serving on my thesis committee, introducing me to research methods through UROP, guiding me during the Earn-Learn Apprenticeship, and an invaluable classroom experience. Dr. Ronald Pak for serving on my thesis committee and for challenging me to stretch my mind and realize that complex systems of equations are intrinsically simple. My soccer buddies and former and current roommates at CU for the distractions and memorable experiences. Lastly, my father and brother for their encouragement and love.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

This section summarizes the symbols, notations, and units (when applicable) throughout this thesis, in the order that they are referred to.

m_i = mass of particle i (kg)
$\mathbf{u}_i$ = particle centroid displacement vector (m)
$\mathbf{F}_i$ = resultant force vector at centroid of particle i (N)
I_i = rotary inertia of particle i (kg·m^2)
$\boldsymbol{\theta}_i$ = rotation vector of particle i (radians)
$\mathbf{M}_i$ = resultant moment vector at centroid about the principal axes of inertial frame of particle i (N-m)
n_c = number of contacts for the ith particle
$F_x^{j,i}$ = scalar component of the contact force exerted on the ith particle by the jth particle in the x direction (N)
$M_x^{j,i}$ = scalar component of the contact moment exerted on the ith particle by the jth particle in the x direction (N-m)
$\mathbf{M}_i$ = mass matrix (kg)
$\mathbf{a}_i$ = acceleration vector (m/s^2)
$\mathbf{v}_i$ = velocity vector (m/s)
$\mathbf{K}_i$ = stiffness matrix (N/m)
$\mathbf{C}_i$ = damping matrix (N·s/m)
$\mathbf{P}_i$ = contact loads (N)
α_1 = coefficient of mass proportional damping
α_2 = coefficient of stiffness proportional damping
c_r = normal damping coefficient (kg/s)
$\mathbf{v}_r$ = normal relative velocity at the center of two particles i and j in contact (m/s)
$\mathbf{F}_d$ = interparticle damping force (N)
$\mathbf{n}$= normal vector at contact
R_1 and R_2 = radius of particles 1 and 2 (m)
E_1 and E_2 = Young's modulus of particles 1 and 2 (N/m^2)
ν_1 and ν_2 = Poisson's ratio of particles 1 and 2
P = normal contact force (N)
ρ = normal displacement (m)
ν = Poisson's ratio
f = inter-particle constant static coefficient of friction
T = tangential force (N)
G = shear modulus (N/m^2)
δ_t = tangential displacement (m)
k_t = tangential stiffness (N/m)
a = radius of contact area (m^2)

Δt = time increment (s)
ω_{max} = angular velocity of the particles (rad/s)
δ_N and δ_T = normal and tangential displacements (m)
ε = point of contact between two particles
$\boldsymbol{\delta}$ = vector of contact displacement (m)
e = elastic component
p = plastic component
F = contact force vector (N)
k = stiffness vector (N/m)
k_N = normal stiffness (N/m)
k_T = tangential stiffness (N/m)
$\dot{\mathbf{F}}^{\varepsilon}$ = rate of change of contact force vector (N/s)
$\dot{F}_N^{\ \varepsilon}$ = rate of change of normal contact force (N/s)
$\dot{F}_T^{\ \varepsilon}$ = rate of change of tangential contact force (N/s)
$\dot{\boldsymbol{\delta}}$ = rate of change of displacement vector (m/s)
f = yield function (N)
κ = single force-like internal state variable (ISV) (N)
$\hat{\boldsymbol{n}}^{\varepsilon}$ = direction of the contact force vector
H = hardening parameter (N/m)
$\dot{\gamma}$ = plastic multiplier (m/s)
$(\bullet)_{n+1}$ = variable at timestep n+1
$(\bullet)_{n}$ = variable at timestep n

Chapter 1: Introduction to DEM

1.1 Motivation

The purpose of this research is to gain a better understanding of dense, dry granular material shear behavior through representative volume element (RVE) modeling using the discrete element method (DEM) as applied to RVE assemblies obtained directly from sub-volume synchrotron X-ray computed tomography (CT) images, as well as investigating elasto-plastic interparticle constitutive response of metallic particles using DEM.

Predicting the behavior of engineered systems is significantly improved with numerical models and higher resolution experimental methods. Numerical models broaden the abilities of engineers by providing accurate and efficient calculations of physical phenomena. In general, physical experiments can be challenging and expensive to perform, which creates the demand for computational models that can accurately depict physical behavior. For one, granular materials may exhibit solid-like behavior and in an instant, transition to flowing like a fluid. By treating individual particles as discrete bodies, we can capture this overall system behavior of particle motion. One application to granular flow can be observed in modeling hopper grain flow as shown in Figure 1. In addition to particle flow, DEM can also model the shear resistance of granular materials, which is closely related to the subject of this research. For instance, Martian soil to tire interaction as shown in Figure 2, can be represented by a DEM model of the soil interacting with a FE tire model (Horner et. al., 2001). Three-dimensional discrete element modeling has the potential to convey accurate multiscale mechanical behavior of granular materials by considering the microscopic properties at the particle scale. Scaling up, or relating the particle scale behavior to the continuum or macroscale is an ongoing topic of research, and is beyond the scope of this thesis.

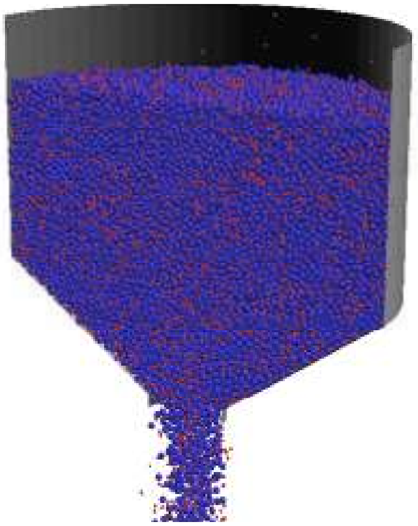


Figure 1: DEM model of hopper flow (Wassgren and Sarker, 2008)

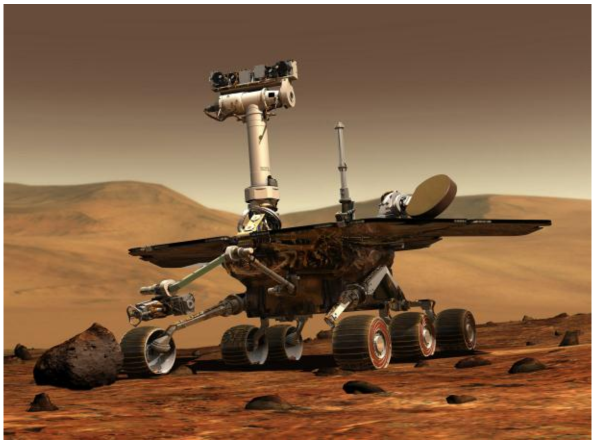


Figure 2: Martian soil to tire interaction (Photo source: National Aeronautics and Space Administration)

In designing geotechnical structures such as building foundations, earth embankment dams, or levees, engineers utilize continuum methods including the finite element method (FEM) for computational efficiency. However, for granular media and powder mechanics, continuum methods dismiss the essential mechanical behavior which is driven by the particle-to-particle interaction and potential for large motion. With the rapid advancements in computational power, DE methods have the potential to accurately model systems of granular materials and metallic powders. This paper considers two methods for simulating particle mechanics: Hertz-Mindlin contact DEM model for small elastic strains at contact in granular materials and an elasto-plastic DEM contact model, reasonable for simulations such as metallic powder compaction. Needless to say, the accuracy of DE and FE methods is limited by the accuracy of the constitutive models, and also the computing power available. For instance, a cylindrical specimen 20.2 mm in height and 9.46 mm in diameter contains approximately 4200 ASTM 20/30 quartz Ottawa sand particles. Depending on the number of contacts and particle motion, a triaxial compression simulation may take roughly 24 hours on a single Intel Xeon 3.07GHz processor. A cubic meter of the same sand may have $1\sim2 \times 10^9$ particles and would require ELLIP3D more than 10^6 hours (≈114 years) to run on a similar single processor. Although ELLIP3D has parallel processing capabilities in OpenMP, there is still a large interdependence of particle interactions which increases the complexity of parallel computing, and it would still require immense computing power to simulate large samples, and places more emphasis on the relevance and accuracy of RVEs.

1.2 Literature Review

DEM was first introduced by Cundall (1971) while being applied to rock mechanics to assess the failure of jointed, blocky rock systems and was later applied to granular materials (Cundall and Strack, 1979). Its main limitation is the number of particles in an assembly or the simulation time, which are related to each other. With advancing computing power and parallel processing capabilities, these restraints are greatly reduced. Yet, it is currently too computationally costly to consider particle-to-particle interaction within the soil in commercial applications, such as tool or bucket digging through gravel/sand, tire-soil interaction (Horner et. al., 2001; Mak et. al., 2011), cone penetrometer testing (Yan, 2008; Breul et. al., 2009), estimation of shear strength (Yan and Ji, 2010), etc. Laboratory tests such as triaxial compression tests give insight to the physical behavior of granular materials, which makes it a suitable measure for DEM simulations as performed by various authors (Ng, 2009; Lu, 2010; Plassiard et. al., 2009; Yan et. al., 2010; Yimsiri and Soga, 2010; Yimsiri and Soga, 2011).

The use of DEM models places great emphasis on the fabric used in the simulations. The term “fabric” is coined as a qualitative and quantitative property which describes the geometric configuration in terms of (1) position and density, (2) shape and dimension, and (3) orientation of discrete particles (Oda, 1982). It describes the geometrical properties of discontinuities in geomaterials which are crucial in determining the macroscopic mechanical behavior of the material. In terms of its quantitative measure, a second-order fabric tensor was proposed by Oda and Nakayama (1989). Several authors including Yimsiri and Soga (2010), Yimsiri and Soga

(2011), and Gao and Zhao (2012) incorporated the concept of the fabric tensor proposed by Oda and Nakayama (1989) into DEM simulations of sands in order to understand the effect of soil fabric on mechanical behavior.

The Hertz-Mindlin contact law formulates the interparticle mechanical behavior for simulations of granular material. Hertz (1882) developed the first contact law between elastic bodies in which he considered normal contact between isotropic, linearly elastic spherical bodies. Mindlin (1949) extended the classical Hertzian contact theory to various loading situations between two spheres in contact and tangential force-displacement relationships. The Hertz-Mindlin contact theory was extended by Misra (1995) in order to capture the mechanical behavior of asperity contact which characterizes the interface geometry of the particles. Misra's formulation of the Hertz-Mindlin theory accounts for normal elastic deformation and frictional sliding for rough surfaces.

The Hertz-Mindlin contact theory is appropriate for modeling elastic behavior of inherently stiff materials such as quartz sand; however, the intrinsically irreversible deformative properties of metallic powders exhibit an elasto-plastic constitutive relationship for which the Hertz-Mindlin contact laws are not feasible during particle compaction. Micromechanical models of the powder compaction process have been analyzed using elastic, perfectly plastic, and elasto-plastic models.

In the industry of powder metallurgy (PM), powder compaction is studied because it is an essential step in forming metals to a desired shape, prior to sintering.

The utilization of powder delivers a cost-efficient method for the production of low stressed components in the automotive and white goods industries (e.g., household appliances, dish washers, microwave ovens, etc.). Higher loads on metallic parts require compaction of powders to higher densities in order to achieve defect-free compacts (Khoei and Iranfar, 2003). Accurate modeling of the powder under an applied stress will ultimately enable manufacturers to build higher quality fabrications with proper design. Particle tracking by Computed Tomography (CT) techniques, combined with compaction-hardening continuum plasticity models (Fleck, 1995; Gu et. al., 2001; Zeuch et. al., 2001), and DEM modeling (Wu et. al., 2003; Wu and Cocks, 2006) are various methods that collectively aim to accurately capture the behavior of metallic powders.

Basic deformation behavior of metallic powder was described early on through empirical means, in which simple uniaxial strain in compression tests were performed on iron powder by Kuhn and Downey (1971). In the industry of powder metallurgy, there are two conventional processes to isostatically compress a powder: cold compaction and hot compaction. Cold compaction is characterized by rate-independent plasticity while hot compaction is constitutively defined by power law and creep and/or diffusional flow (Storåkers, Fleck, and McMeeking, 1999). Wilkinson and Ashby (1975) attempted to model the constitutive relation for power-law creep to solve time dependent flow during the sintering process of metallic powders. Further work in modeling deformation of metallic powders by power-law creep during hot isostatic pressing (HIP) was extended by Kuhn and McMeeking (1992). Fischmeister and others attempted to evaluate plastic flow of spherical particle powders by isostatic and uniaxial compression (Fischmeister et. al., 1978; Fischmeister et. al., 1983). Many extensions have been made to rate-independent plasticity models (Fleck et. al., 1992; Fleck, 1995). Storåkers et. al. (1999) consider only plasticity and no elasticity in which they formulate yield surfaces and creep dissipation parameters to analyze strain hardening plasticity. They performed isostatic compaction on copper and tin powders for which a homogeneous strain field assumption is imposed. Essentially, the homogenous strain field assumption disallows local particle rearrangement or rotation during compaction (Martin, Bouvard, and Shima, 2003). Intricate constitutive models for powder mechanics have been developed to model compaction; however, this research grasps the fundamentals of elasto-plasticity through a bilinear force-displacement constitutive relationship for powder compaction.

1.3 Background Overview and Governing Equations:

1.3.1 Governing and Constitutive Equations for Elastic Frictional Granular Materials

In this paper, a 3D DEM code ELLIP3D (Yan, 2008) is utilized to simulate over several thousand ellipsoidal particle assemblies in conventional triaxial compression tests (CTCs). The code uses contact detection algorithms which allow particles to freely come into contact and break contacts with neighboring particles. ELLIP3D abides by Hertz-Mindlin contact laws for centric/eccentric impacting, rolling/sliding, and depositing of granular particles. It is capable of incorporating Mindlin or Coulomb friction models. Additionally, velocity proportional damping and mass/stiffness proportional damping at interparticle contacts are utilized in the code. ELLIP3D is coded in C++ and uses an explicit time integration algorithm (central difference method), with mass scaling as well as “global” (background) mass-proportional damping for Dynamic Relaxation (DR) (Underwood, 1983) to simulate quasi-static loading of granular assemblies. DR determines a static solution by using a dynamic transient analysis and ultimately aims to resolve simulations with stable solutions in order to prevent erratic dynamic transitions. For dynamic loading, DEM is naturally well-suited.

The numerical stability of the simulations is restricted by the size of the time step which is controlled by the smallest particle mass and stiffest elastic contact (Yan, 2008). Since a considerably small time step may result in extremely long-computation time (i.e., many time steps) for larger assemblies, a fictitious background damping can be applied to maintain particle stability in order to accommodate a larger time step for quasi-static simulations using DR. Specific values for parameters used will be discussed in the proceeding sections.

DEM is a method that simulates the interaction between distinct bodies by integrating the equations of motion of linear and angular momentum. A contact detection algorithm is utilized along with the fundamental equations of motion to compute particle movement by calculating particles’ displacements over a determined time step. The overall system behavior is determined by the governing equations: Equation (1.1) and Equation (1.2) for which these equations evaluate motion in the local coordinate system for each particle to be assembled into the global coordinate system.

$$m_i\ddot{\mathbf{u}}_\mathrm{i} = \mathbf{F}_\mathrm{i} \qquad (1.1)$$

$$I_\mathrm{i}\ddot{\boldsymbol{\theta}}_\mathrm{i} = \mathbf{M}_\mathrm{i} \qquad (1.2)$$

$\mathrm{m_i}$ = mass of particle i

$\mathbf{u}_\mathrm{i}$ = particle centroid displacement vector in internal coordinate frame

$\mathbf{F}_\mathrm{i}$ = resultant force vector at centroid of particle i

$\mathrm{I_i}$ = rotary inertia of particle i

$\boldsymbol{\theta}_\mathrm{i}$ = rotation vector of particle i

$\mathbf{M}_\mathrm{i}$ = resultant moment vector at centroid about the principal axes of inertial frame of particle i

The contact loads acting on each particle are determined by the magnitude of the resultant forces and resultant moments acting in the directions defined by the global coordinate system which is described by Equation (1.3).

$$\boldsymbol{P}_\mathrm{i}^\mathrm{T} = \left[\sum_{\boldsymbol{j}=1}^{\boldsymbol{n_c}} \mathrm{F_x}^{\boldsymbol{j,i}}, \sum_{\boldsymbol{j}=1}^{\boldsymbol{n_c}} \mathrm{F_y}^{\boldsymbol{j,i}}, \sum_{\boldsymbol{j}=1}^{\boldsymbol{n_c}} \mathrm{F_z}^{\boldsymbol{j,i}}, \sum_{\boldsymbol{j}=1}^{\boldsymbol{n_c}} \mathrm{M_x}^{\boldsymbol{j,i}}, \sum_{\boldsymbol{j}=1}^{\boldsymbol{n_c}} \mathrm{M_y}^{\boldsymbol{j,i}}, \sum_{\boldsymbol{j}=1}^{\boldsymbol{n_c}} \mathrm{M_z}^{\boldsymbol{j,i}}\right] \qquad (1.3)$$

$\mathrm{n_c}$ = number of contacts for the ith particle

$\mathrm{F_x}^{\mathrm{j,i}}$ = scalar component of the contact force exerted on the ith particle by the jth particle in the x direction

$\mathrm{M_x}^{\mathrm{j,i}}$ = scalar component of the contact moment exerted on the ith particle by the jth particle in the x direction

Ultimately, the resultant forces acting in the global system on particle i are formulated by Equation (1.4). It consists of three terms, which are described by Equations (1.3), (1.5), and (1.6). The terms associated with mass-proportional damping and velocity proportional damping are fictitious quantities assigned in order to obtain a static solution using a minimum number of time steps (Yan, 2008). The $\mathbf{P}_\mathbf{i}$ and $\mathbf{F}_\mathbf{i}$ for particle i represent physical quantities in quasi-static DR simulations.

$$\boldsymbol{M}_{\boldsymbol{i}}\boldsymbol{a}_{\boldsymbol{i}} + \boldsymbol{C}_{\boldsymbol{i}}\boldsymbol{v}_{\boldsymbol{i}} + \boldsymbol{P}_{\boldsymbol{i}} = \boldsymbol{F}_\mathrm{i} \qquad (1.4)$$

From the given values of mass densities and volumes of the particles, the mass matrix for each particle can be assembled from the particle masses and moment of rotary inertia as demonstrated in Equation (1.5).

The general viscous damping matrix consists of mass and stiffness proportional damping. It is formulated by the mass matrix and stiffness matrix as shown in Equation (1.6). The equations of motion for the rigid body particles are applied to a system with viscous background damping, i.e. DR for quasi-static simulations. The viscous background damping applies a force acting in the opposite direction of the velocity vector. It is proportional to the velocity of the particles as shown in Equation (1.7), where the relative velocity of two particles in contact can be determined by the displacement history in time as in Equation (1.8). Equation (1.7) is essentially a force that contributes to the right-hand side of Equation (1.4).

$$\boldsymbol{M}_{\mathrm{i}} = \begin{bmatrix} \mathrm{m_i} & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathrm{m_i} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathrm{m_i} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathrm{I_i} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathrm{I_i} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathrm{I_i} \end{bmatrix} \qquad (1.5)$$

$$\boldsymbol{C_i} = \alpha_1 \boldsymbol{M_i} + \alpha_2 \boldsymbol{K_i} \qquad (1.6)$$

$$\boldsymbol{F}_{\mathrm{d}} = \mathrm{c_r} \boldsymbol{v}_{\mathrm{r}} \qquad (1.7)$$

$$\mathbf{v_r} = [(\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_i) \cdot \mathbf{n}]\mathbf{n} \qquad (1.8)$$

$\mathbf{M}_i$ = mass matrix

$\mathbf{a}_i$ = acceleration vector

$\mathbf{v}_i$ = velocity vector

$\mathbf{K}_i$ = stiffness matrix

$\mathbf{C}_i$ = damping matrix

$\mathbf{P}_i$ = contact loads

α_1 = coefficient of mass proportional damping

α_2 = coefficient of stiffness proportional damping

c_r = normal damping coefficient

$\mathbf{v}_r$ = normal relative velocity at the center of two particles i and j in contact

$\mathbf{F}_d$ = interparticle damping force

$\mathbf{n}$= normal vector at contact

The combined balance of linear and angular momentum equation in matrix form in Equation (1.8) are assembled for all particles i=1,.., $n_{particles}$, to obtain the global matrix balance equation to be interpreted in time using central difference method as in (Yan, 2008).

The governing equations are based on Newton's laws of motion, while the constitutive force-displacement relationships for granular materials are established from the Hertz-Mindlin contact theory. The values of relative radius and relative Young's Modulus of two particles in contact is defined by Equations (1.9) and (1.10). A conceptual schematic of the geometric parameters of two particles in contact is shown in Figure 3.

$$R_0 = \left(\frac{1}{R_1} + \frac{1}{R_2} \right)^{-1} \quad (1.9)$$

$$E_0 = \left(\frac{1-\nu_1^2}{E_1} + \frac{1-\nu_2^2}{E_2} \right)^{-1} \quad (1.10)$$

R_1 and R_2 = radius of particles 1 and 2

E_1 and E_2 = Young's modulus of particles 1 and 2

ν_1 and ν_2 = Poisson's ratio of particles 1 and 2

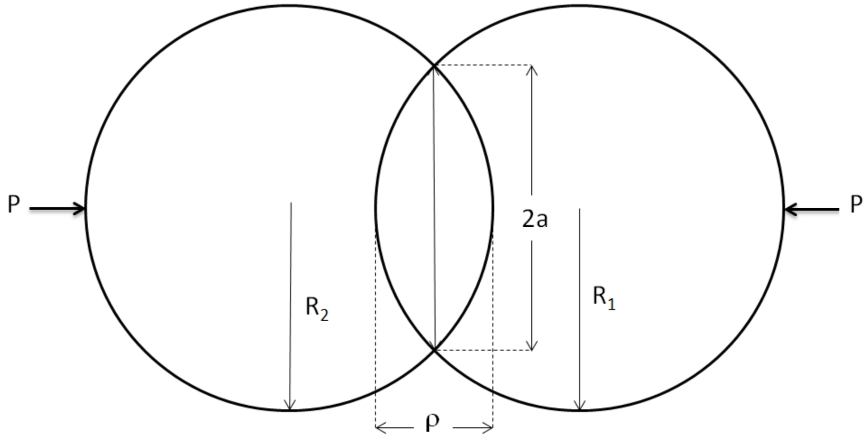


Figure 3: Schematic of Hertz contact between two spheres

The normal contact force P at particle contact is defined by Equation (1.11), following Hertz's contact theory. This equation can be rearranged to formulate displacement as a function of normal force: Equation (1.12). The derivative of normal force in terms of normal displacement formulates the normal stiffness: Equation (1.13).

$$\mathrm{P} = \frac{4}{3} * \mathrm{E}_0 \cdot \mathrm{R}_0^{1/2} \rho^{3/2} \qquad (1.11)$$

P = normal contact force

ρ = normal displacement

$$\rho = \mathrm{R}_0^{-1/3} \left(\frac{3\mathrm{P}}{4\mathrm{E}_0} \right)^{2/3} \qquad (1.12)$$

$$k_n = \frac{dP}{d\rho} = (6P\mathrm{R}_0 E_0^{\,2})^{1/3} \qquad (1.13)$$

In ELLIP3D, tangential contact forces can be described by one of Mindlin's three tangential-force/displacement relationships: no slip conditions, partial slip, or a decreasing tangential force. For general DEM applications to granular materials, it is physically feasible to use a partial slip condition since some slippage is likely to occur, where the tangential displacement is a function of normal and tangential forces acting at the point of contact. For the partial slip condition, the relationship between tangential force and tangential displacement is described by Equation (1.14). Similarly to the normal displacement, the tangential displacement and tangential stiffness can be derived as expressed in Equation (1.15) and (1.16) respectively.

$$T = \mathrm{f} \cdot \mathrm{P}\left\{1 - \left[1 - \left(1 - \frac{8\mathrm{Ga}\delta_{\mathrm{t}}}{3(2-\nu)\mathrm{f} \cdot \mathrm{P}}\right)\right]^{3/2}\right\} \qquad (1.14)$$

$$\delta_{\mathrm{t}} = \frac{3(2-\nu)\mathrm{f} \cdot \mathrm{P}}{8\mathrm{Ga}}\left[1 - \left(1 - \frac{\mathrm{T}}{\mathrm{f} \cdot \mathrm{P}}\right)\right]^{2/3} \qquad (1.15)$$

$$k_{\mathrm{t}} = \frac{\mathrm{dT}}{\mathrm{d}\delta_{\mathrm{t}}} = \frac{4Ga}{2-\nu}\left[1 - \frac{8\mathrm{Ga}\delta_{\mathrm{t}}}{3(2-\nu)\mathrm{f} \cdot \mathrm{P}}\right]^{1/2} \qquad (1.16)$$

ν = Poisson's ratio
f = inter-particle constant static friction
T = tangential force
G = shear modulus
a = radius of contact area

Equation (1.17) gives the Regularized Coulomb Friction model with sliding, which is the linear approximation to the partial slip condition. The equation is a function of the radius of the contact area a, which changes with normal displacement. Figure 4 illustrates this constitutive relationship where sliding occurs when $|F_t| \geq \mu F_n$.

$$k_{\mathrm{t}} = \frac{\mathrm{dT}}{\mathrm{d}\delta_{\mathrm{t}}} = \frac{4Ga}{2-\nu} \qquad (1.17)$$

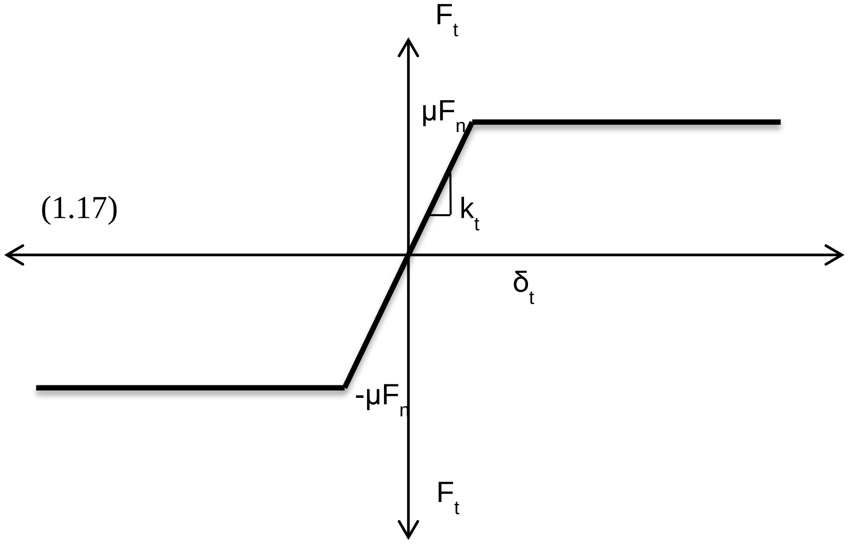


Figure 4: Regularized Coulomb Friction Model with Sliding

Additionally, mass scaling can be used when there is a requirement for a very small time step when simulating small particles with stiff contacts (Yan, 2008). Mass and moment of inertia are inversely proportional to the acceleration and angular acceleration of the particles by the definition of Newton's second law. Thus, applying mass and moment of inertia scaling will reduce the linear and angular acceleration of the particles. Equation (1.18) conveys the theoretical critical time step obtained from linear stability analysis for which the solution will grow unboundedly if the critical time step is exceeded (Yan, 2008).

$$\Delta t < \frac{2}{\omega_{max}} = 2\sqrt{m/k} \qquad (1.18)$$

Although mass scaling permits a larger time step by slowing down the particles, it also causes instability due to large contact forces. For instance, in a well-graded granular material, it is possible that a very small particle comes in contact with another particle that is an order of magnitude larger. Normal force is a function of particle mass and the increase in relative mass due to mass-scaling would cause the smaller particle to repel at a high velocity, producing unrealistic results. Thus, an efficient value of mass-scaling should be weighted based on stability and computation time. Specific parameters used in ELLIP3D simulations will be discussed in the following chapter.

1.3.2 Constitutive Theory for Elastoplastic Metallic Powders

Elasto-plastic DE modeling is relevant for modeling particulate materials with elasto-plastic particles such as metallic powders (Storåkers et. al., 1999). This elasto-plastic constitutive model assumes that particles do not crush or fracture. For simplicity for now, the vector form of this model described below, incorporates a constant elastic stiffness k and constant hardening parameter H to define the normal contact force F and a single force-like internal state variable (ISV) κ. Major limitations of the model are the following: (1) particle shapes remain spherical (or ellipsoidal) upon deformation at contact; (2) the displacement vector at contact δ is additively decomposed into elastic and plastic parts which is typically appropriate only for small deformations. A comparison of full finite element (FE) simulations and DE simulations are discussed in Section 2.2.1. The formulas in the preceding section depict the general scheme of ELLIP3D for elastic, frictional contact. The simple 1-D elastoplasticity constitutive model for powder compaction holds the following general formulation to be true:

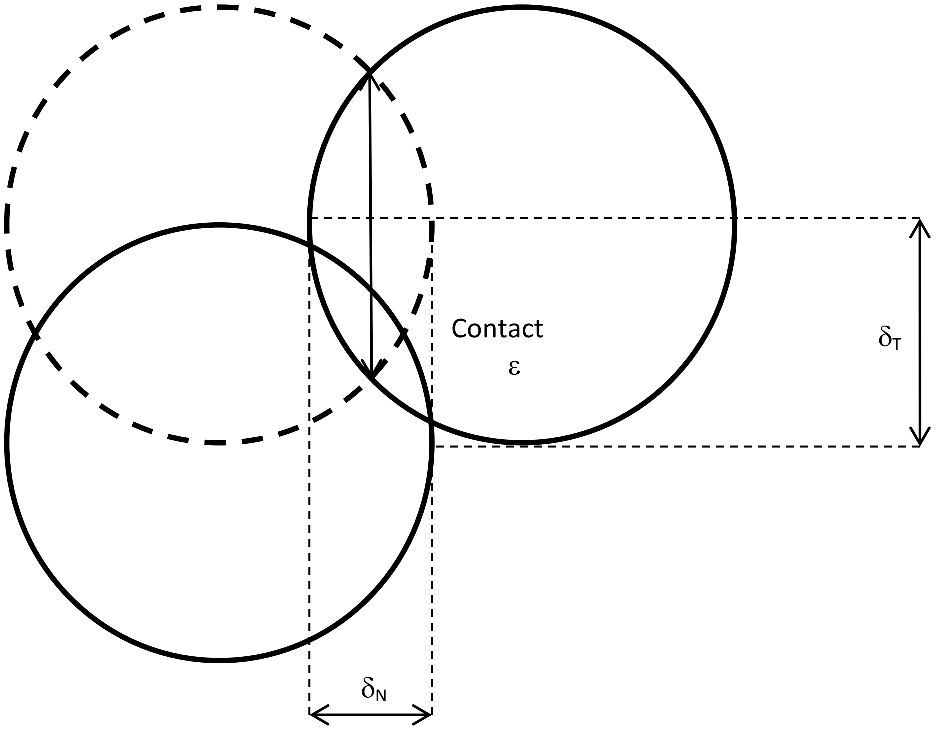


Figure 5: Two-Particle Contact Schematic with normal overlap displacement δ_N and tangential displacement δ_T

<u>Vector Form</u>

Vector of contact displacement is given as:

$$\boldsymbol{\delta} = \begin{bmatrix} \delta_N \\ \delta_T \end{bmatrix} \qquad (1.19)$$

for which the normal (labeled with subscript N) and tangential (labeled with subscript T) displacements are δ_N and δ_T, respectively. Additive decomposition into elastic and plastic parts is denoted as

$$\boldsymbol{\delta} \ = \boldsymbol{\delta}^e + \boldsymbol{\delta}^p \qquad (1.20)$$

Where the superscript e is the elastic component and p represents the plastic component. The contact force vector **F** at contact ε is expressed as

$$\mathbf{F}^\varepsilon = \mathbf{k} \ \cdot \boldsymbol{\delta}^e \qquad (1.22)$$

where **k** is the stiffness matrix, and will be assumed isotropic for now where $k_N = k_T$.

$$\mathbf{k} = \begin{bmatrix} k_N & 0 \\ 0 & k_T \end{bmatrix} = k \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad (1.23)$$

In rate form,

$$\dot{\mathbf{F}}^\varepsilon = \mathbf{k} \cdot \dot{\boldsymbol{\delta}}^e = \mathbf{k} \cdot (\dot{\boldsymbol{\delta}} - \dot{\boldsymbol{\delta}}^p), \quad \dot{\boldsymbol{\delta}}^e = \begin{bmatrix} \dot{\delta}_N{}^e \\ \dot{\delta}_T{}^e \end{bmatrix} \qquad (1.24)$$

$$\dot{\mathbf{F}}^\varepsilon = \begin{bmatrix} \dot{F}_N{}^\varepsilon \\ \dot{F}_T{}^\varepsilon \end{bmatrix} = k \begin{bmatrix} \dot{\delta}_N{}^e \\ \dot{\delta}_T{}^e \end{bmatrix} = k \, \dot{\boldsymbol{\delta}}^e \qquad (1.25)$$

Yielding is determined by the yield function *f*, which is a function of the contact force vector $\mathbf{F}^{\varepsilon}$ and the single force-like internal state variable (ISV) κ.

$$f(\mathbf{F}^{\varepsilon}, \kappa) = \left\| \mathbf{F}^{\varepsilon} \right\| - \kappa \leq 0 \qquad (1.26)$$

$$f < 0 \qquad \text{elastic} \qquad (1.27)$$

$$f = 0 \qquad \text{plastic} \qquad (1.28)$$

$$\left\| \mathbf{F}^{\varepsilon} \right\| = \sqrt{(\mathrm{F_N})^2 + (\mathrm{F_T})^2} \qquad (1.29)$$

With the associative plasticity, the plastic potential can be observed as $g = f$; the flow rule can be stated as

$$\dot{\boldsymbol{\delta}}^{p} = \dot{\gamma} \frac{\partial g}{\partial \mathbf{F}^{\varepsilon}} = \dot{\gamma} \frac{\partial f}{\partial \mathbf{F}^{\varepsilon}} \qquad (1.30)$$

where γ is the plastic multiplier. The direction of the contact force vector $\hat{\boldsymbol{n}}^{\varepsilon}$, which is the direction of plastic displacement at contact ε is described by

$$\frac{\partial f}{\partial \mathbf{F}^{\varepsilon}} = \frac{\mathbf{F}^{\varepsilon}}{\left\| \mathbf{F}^{\varepsilon} \right\|} = \hat{\mathbf{n}}^{\varepsilon} \qquad (1.31)$$

Through the definitions of the ISV evolution

$$\dot{\kappa} = H \dot{\gamma} \qquad (1.32)$$

consistency condition

$$\dot{\gamma}\ \dot{f} = 0 \qquad (1.33)$$

and Kuhn-Tucker conditions

$$\dot{\gamma} \geq 0,\quad f \leq 0,\quad \dot{\gamma} f = 0 \qquad (1.34)$$

With three equations (1.35) – (1.37) and three unknowns ($\mathbf{F}^e$, κ, γ), the ODEs of the system are integrated using Backward Euler in the following fashion:

$$\dot{\mathbf{F}}^{\varepsilon} = \mathbf{k} \cdot \dot{\boldsymbol{\delta}}^{e} = \mathbf{k} \cdot (\dot{\boldsymbol{\delta}} - \dot{\boldsymbol{\delta}}^{p}) \qquad (1.35)$$

$$\kappa = H\dot{\gamma} \qquad (1.36)$$

$$\dot{\gamma}\dot{f} = 0 \qquad (1.37)$$

$$(\mathbf{F}^{\varepsilon})_{n+1} = (\mathbf{F}^{\varepsilon})_{n} + \mathbf{k} \cdot (\Delta\boldsymbol{\delta} - \Delta\gamma\hat{\mathbf{n}}^{\varepsilon,tr}) = \mathbf{F}^{\varepsilon,tr} - \Delta\gamma\mathbf{k} \cdot \hat{\mathbf{n}}^{\varepsilon,tr} \qquad (1.38)$$

$$\hat{\mathbf{n}}^{\varepsilon,tr} = \frac{\mathbf{F}^{\varepsilon,tr}}{\left\|\mathbf{F}^{\varepsilon,tr}\right\|} \qquad (1.39)$$

$$\kappa_{n+1} = \kappa_{n} + H\Delta\gamma \quad \text{(N/m)} \qquad (1.40)$$

$$f_{n+1} = 0 \qquad (1.41)$$

To solve for Δγ,

$$(f)_{n+1} = \left\|\mathbf{F}^{\varepsilon}_{n+1}\right\| - \kappa_{n+1} = 0 \qquad (1.42)$$

$$\mathbf{F}^{\varepsilon}_{n+1}\cdot\mathbf{F}^{\varepsilon}_{n+1} = (\mathbf{F}^{\varepsilon,tr} - \Delta\gamma\mathbf{k}\cdot\hat{\mathbf{n}}^{\varepsilon,tr})\cdot(\mathbf{F}^{\varepsilon,tr} - \Delta\gamma\mathbf{k}\cdot\hat{\mathbf{n}}^{\varepsilon,tr}) \qquad (1.43)$$

let $\mathbf{k} = k\mathbf{1} = k\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$ (1.44)

so that $\mathbf{k}\cdot\hat{\mathbf{n}}^{\varepsilon,tr} = k\hat{\mathbf{n}}^{\varepsilon,tr}$ (1.45)

$$\mathbf{F}^{\varepsilon}_{n+1} = \left(1 - \frac{\Delta\gamma k}{\left\|\mathbf{F}^{\varepsilon,tr}\right\|}\right)\mathbf{F}^{\varepsilon,tr} \qquad (1.46)$$

$$\mathbf{F}^{\varepsilon}_{n+1}\cdot\mathbf{F}^{\varepsilon}_{n+1} = \left(1 - \frac{\Delta\gamma k}{\left\|\mathbf{F}^{\varepsilon,tr}\right\|}\right)^{2}\mathbf{F}^{\varepsilon,tr}\cdot\mathbf{F}^{\varepsilon,tr} \qquad (1.47)$$

$$\left\|\mathbf{F}^{\varepsilon}_{n+1}\right\| = \left|1 - \frac{\Delta\gamma k}{\left\|\mathbf{F}^{\varepsilon,tr}\right\|}\right|\sqrt{\mathbf{F}^{\varepsilon,tr}\cdot\mathbf{F}^{\varepsilon,tr}} = \left|1 - \frac{\Delta\gamma k}{\left\|\mathbf{F}^{\varepsilon,tr}\right\|}\right|\left\|\mathbf{F}^{\varepsilon,tr}\right\| = \left\|\mathbf{F}^{\varepsilon,tr}\right\| - \Delta\gamma k \qquad (1.48)$$

Then we can solve for the plastic multiplier increment $\Delta\gamma$ when plastic loading occurs by determining the yield function *f* at the current (n+1) timestep

$$f_{n+1} = \left\|\mathbf{F}^{\varepsilon,tr}\right\| - \Delta\gamma k - \kappa_n - H\Delta\gamma = 0 \qquad (1.49)$$

$$= \left\|\mathbf{F}^{\varepsilon,tr}\right\| - \kappa_n - \Delta\gamma(k + H) = 0 \qquad (1.50)$$

$$= f^{tr} - \Delta\gamma(k + H) = 0 \qquad (1.51)$$

$$\Rightarrow \Delta\gamma = \frac{f^{tr}}{k+H} \quad (1.52)$$

With the return mapping algorithm:

Given $\Delta\boldsymbol{\delta} = \boldsymbol{\delta}_{n+1} - \boldsymbol{\delta}_n,\ \kappa_n,\ \gamma_n$

Parameters *k* (N/m), *H* (N/m), κ_0 (N)

1. Check for yielding:

$$f^{tr} = \left\|\mathbf{F}^{\varepsilon,tr}\right\| - \kappa_n,\ \mathbf{F}^{\varepsilon,tr} = \mathbf{F}_n^{\varepsilon} + k\Delta\boldsymbol{\delta} \qquad (1.53)$$

If $f^{tr} > 0$, plastic go to step 2.

else if $f^{tr} < 0$, elastic:

$$\mathbf{F}_{n+1}^{\varepsilon} = \mathbf{F}^{\varepsilon,tr} \quad (1.54)$$

$$\kappa_{n+1} = \kappa_n \quad (1.55)$$

$$\gamma_{n+1} = \gamma_n \quad (1.56)$$

2. Plastic loading:

$$\Delta\gamma = \frac{f^{tr}}{k+H} \qquad (1.57)$$

Update:

$$\mathbf{F}_{n+1}^{\varepsilon} = \mathbf{F}^{\varepsilon,tr} - k\Delta\gamma\hat{\mathbf{n}}^{\varepsilon,tr},\ \hat{\mathbf{n}}^{\varepsilon,tr} = \frac{\mathbf{F}^{\varepsilon,tr}}{\left\|\mathbf{F}^{\varepsilon,tr}\right\|} \qquad (1.58)$$

$$\kappa_{n+1} = \kappa_n + H\Delta\gamma \qquad (1.59)$$

Notes: 1) There is no consistent tangent needed because ELLIP3D is explicit. 2) The preceding notation is in vector form with a single ISV κ0, stiffness k, and hardening parameter H; the component form with different values of κ0, k, and H for normal and tangential directions is implemented as follows in order attain results in Figures 27/28 :

<u>Decoupled Component Form:</u>

$$\boldsymbol{\kappa} = \begin{bmatrix} \kappa_N & 0 \\ 0 & \kappa_T \end{bmatrix} \qquad (1.60)$$

$$\mathbf{k} = \begin{bmatrix} k_N & 0 \\ 0 & k_T \end{bmatrix} \qquad (1.61)$$

$$\mathbf{H} = \begin{bmatrix} H_N & 0 \\ 0 & H_T \end{bmatrix} \qquad (1.62)$$

$$\Delta\delta_N = (\delta_N)_{n+1} - (\delta_N)_n \qquad (1.63)$$

$$\Delta\delta_T = (\delta_T)_{n+1} - (\delta_T)_n \qquad (1.64)$$

$$\Delta\boldsymbol{\delta} = \begin{bmatrix} \Delta\delta_N \\ \Delta\delta_T \end{bmatrix} \qquad (1.65)$$

$$F_N^{\varepsilon,tr} = (F_N^{\varepsilon})_n + k_N \Delta\delta_N \qquad (1.66)$$

$$F_T^{\varepsilon,tr} = (F_T^{\varepsilon})_n + k_T \Delta\delta_T \qquad (1.67)$$

$$\mathbf{F}^{\varepsilon,tr} = \begin{bmatrix} F_N^{\varepsilon,tr} \\ F_T^{\varepsilon,tr} \end{bmatrix} = (\mathbf{F}^{\varepsilon})_n + \mathbf{k}\Delta\boldsymbol{\delta} \qquad (1.68)$$

$$\left\|\mathbf{F}^{\varepsilon,tr}\right\| = \sqrt{(\mathbf{F}_N^{\varepsilon,tr})^2 + (\mathbf{F}_T^{\varepsilon,tr})^2} \qquad (1.69)$$

$$\hat{n}_N^{\varepsilon,tr} = \frac{F_N^{\varepsilon,tr}}{\left\|\mathbf{F}^{\varepsilon,tr}\right\|},\ \hat{n}_T^{\varepsilon,tr} = \frac{F_T^{\varepsilon,tr}}{\left\|\mathbf{F}^{\varepsilon,tr}\right\|} \qquad (1.70)$$

$$\hat{\mathbf{n}}^{\varepsilon,tr} = \begin{bmatrix} \hat{n}_N^{\varepsilon,tr} \\ \hat{n}_T^{\varepsilon,tr} \end{bmatrix} = \frac{\mathbf{F}^{\varepsilon,tr}}{\left\|\mathbf{F}^{\varepsilon,tr}\right\|} \qquad (1.71)$$

$$f_N^{\ tr} = \left\|\mathbf{F}_N^{\varepsilon,tr}\right\| - \kappa_{N,n} \qquad (1.72)$$

$$f_T^{\ tr} = \left\|\mathbf{F}_T^{\varepsilon,tr}\right\| - \kappa_{T,n} \qquad (1.73)$$

$$\Delta\gamma_N = \frac{f_N^{\ tr}}{k_N + H_N} \qquad (1.74)$$

$$\Delta\gamma_T = \frac{f_T^{\ tr}}{k_T + H_T} \qquad (1.75)$$

$$(F_N^{\varepsilon})_{n+1} = F_N^{\varepsilon,tr} - k_N\,\Delta\gamma_N\,\hat{n}_N^{\varepsilon,tr} \qquad (1.76)$$

$$(F_T^{\varepsilon})_{n+1} = F_T^{\varepsilon,tr} - k_T\,\Delta\gamma_T\,\hat{n}_T^{\varepsilon,tr} \qquad (1.77)$$

$$\kappa_{N,n+1} = \kappa_{N,n} + H_N\,\Delta\gamma_N \qquad (1.78) \qquad\qquad \kappa_{T,n+1} = \kappa_{T,n} + H_T\,\Delta\gamma_T \qquad (1.79)$$

Chapter 2: Method

2.1 Triaxial Compression

2.1.1 Three-dimensional Particle Arrangement (Fabric)

Various methods are used to generate initial particle assemblies. Most commonly, these methods lead to synthetic fabrics of randomly generated particles produced to achieve a specified size gradation and sample porosity (Wellmann, 2008; Pizette, 2010). The term fabric refers to the 3D particle arrangement within a RVE of granular particles. This research aims to simulate conventional triaxial compression tests with particle assemblies in their *in situ* state by obtaining the specimen fabric through X-ray synchrotron micro-computed tomography (SMT) imaging. The represented volume elements of the particle assemblies are generated from *in situ* SMT imaging provided by Professor Khalid Alshibli and co-workers at the University of Tennessee, Knoxville. These data of initial numerical packing are approximated as two-axis ellipsoids based on the numerical algorithm used to interpret particle shape from X-ray CT data (Thompson et. al., 2006). The SMT imaging provides the sand grain particles' radii, positions, and orientation. Triaxial compression tests were performed on F-75 Ottawa sand and ASTM 20/30 Ottawa sand with SMT scans occurring throughout the test in order to provide initial, intermediate, and final images. This 3D imaging was achieved using the Beamline at the Advanced Photon Source (APS) of Argonne National Laboratory by having an X-ray source and imaging unit take cross-sectional images of a cylindrical soil specimen that rests on a rotating platform, for which cross-sectional images were taken every ¼ of a degree. The SMT imaging enables the validation of numerical and visual results of ELLIP3D simulations in terms of particle displacements for the various assemblies as the specimen is compressed. In the ASTM 20/30 sample, there are apparently granular particulates which appear as very small particles in the DEM simulation. They too are approximated as ellipsoids and are not removed from the simulation in order to maintain the fabric. Instructions on converting the experimental data to ELLIP3D readable files can be found in Section 2.1 in Appendix C. An image of a CT scan from a full cylindrical specimen 20.2 mm height and 9.46 mm in diameter of ASTM 20/30 sand is shown in Figure 6 below. At this time, ELLIP3D does not have capabilities to perform simulations with cylindrical boundaries so a boxed RVE is trimmed from the cylindrical specimen as shown in Figure 6.

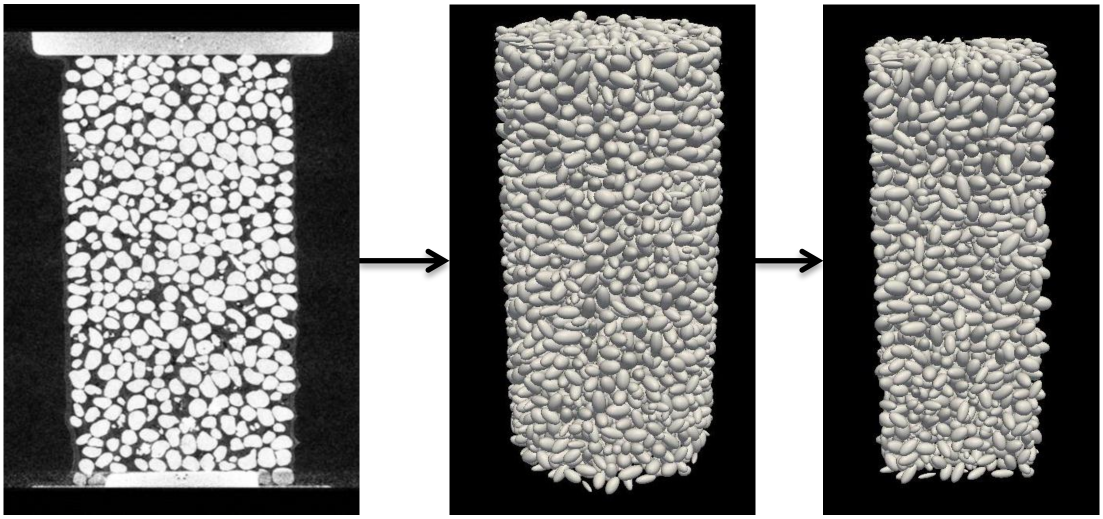


Figure 6: 2-D CT Cross-Section, Computed Cylindrical Assembly, and Boxed RVE Trimmed from Cylindrical Assembly

2.1.2 Fabric Modifications for Computational Stability

The initial particle assembly is formulated by an algorithm that interprets the raw X-ray SMT volume data (Thompson et. al., 2006). It approximates the sand grain geometries as two-axis ellipsoids which result in small gaps and overlaps. In order to maintain the fabric (three dimensional geometric arrangement), overlapping particles are separated to the point where their surfaces are just in contact as shown in the initial and final configurations of the particles in Figure 7. This is done by applying a parameter that controls viscous background damping and a relatively small time step while allowing the body forces to push the particles apart.

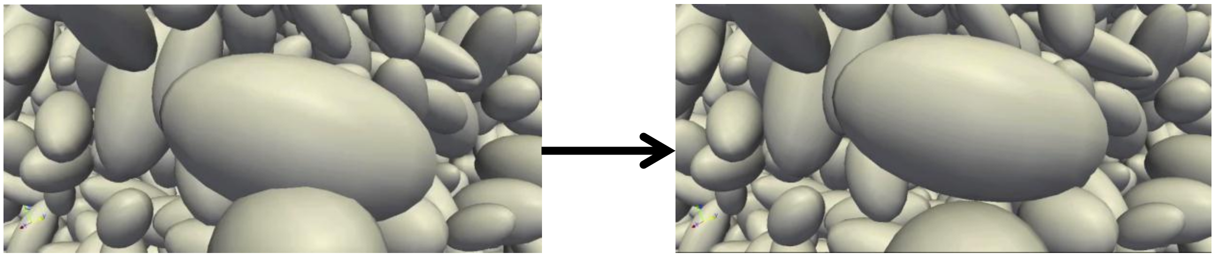


Figure 7: Initial and Final States of Particle Separation

Next, the small gaps are closed by isotropically compressing the particles to achieve the confining pressures used in the experiment (see Table 1 for confining pressures and other experimental parameters). Once the confining pressure is achieved by isotropic compression, triaxial compression simulations are performed. This is the procedure for triaxial compression tests on *in situ* fabric particle assemblies; these results are compared to triaxial compression tests where the fabric is lost due to deposition by gravity. Detailed instructions and the location of a repository directory containing example simulations with the various stages involved can be found in Appendix C.

2.1.3 Experiment and ELLIP3D Parameters

The two granular samples considered in this research have D50s, median grain sizes, of 0.160 mm (F-75 quartz Ottawa Sand) and 0.725 mm (ASTM 20/30 quartz

Ottawa Sand). See Figures 34 and 35 in Appendix A for gradation curves of the granular assemblies. Professor Alshibli and his team performed triaxial compression tests on the F-75 and ASTM 20/30 samples with vacuum confining pressures of 58 kPa and 70 kPa, respectively. The samples were compressed with the same displacement rates and were equal in height and diameter, see Table 1. A cross-sectional image of the X-ray SMT scan for the F-75 specimen is displayed in Figure 8.

	F-75	**ASTM 20/30**	
Confining Pressure	58	70	kPa
Specimen Height	20.2	20.2	mm
Specimen Diameter	9.46	9.46	mm
Displacement Rate	0.2	0.2	mm/min
Initial Void Ratio	0. 518	0. 523	

Table 1: F-75 and ASTM 20/30 Ottawa Sands Triaxial Compression Experimental Data

Young's Modulus, E	2.9×10^{10} Pa
Poisson's ratio ν	0.25
Specific Gravity G_s	2.65

Table 2: Particle Parameters for Numerical Computation

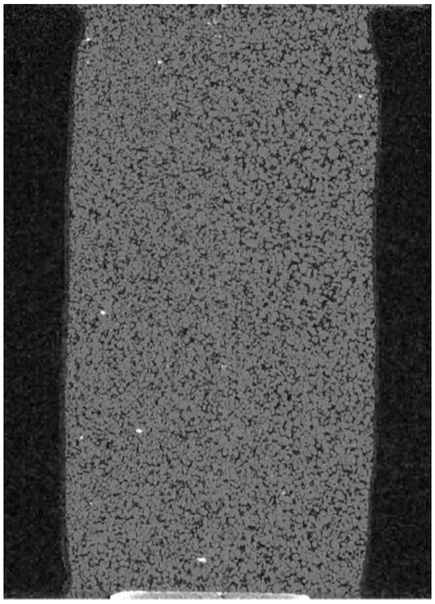


Figure 8: X-ray SMT Cross-section of the F-75 Quartz Ottawa Sand, 20.2 mm in Height

2.1.4 Deposition by Gravity

The constitutive results of triaxial compression simulations with particle assemblies attained from SMT imaging are compared to those where the fabric of the particle assemblies is lost by gravity deposition. In the previous simulations, the parameter that controls the gravity force was set to zero, but in order to alter the fabric for the initial assembly prior to triaxial compression, the particles drop in such a way that the large particle motion displaces and alters the fabric. These simulations were run dynamically with zero background damping, i.e. no dynamic relaxation. The resulting assemblies will be discussed in Section 3.1.

2.2 Elasto-Plastic Modeling of Powder Compaction

Compaction simulations of metallic powder were performed on two spherical particles (courtesy of Y. Hammi) using the Finite Element Analysis (FEA) program Abaqus, and comparing the results to that of the DEM code ELLIP3D. The goal of the FE modeling is to understand the phenomenon of deformation, interaction, and flow behavior of metallic particles through the use of continuum plasticity models and contact laws (Hammi, 2011). In the Abaqus model, the nonlinear geometric effect function (NLGEOM) is used in order to account for finite deformations. Although FE methods achieve results with greater efficiency than DE methods for large scale geo-structures, the ratio of finite elements to discrete bodies for simulations of powder compaction is in favor of the DEs in terms of computational cost at the particle scale. For instance, each particle in the Abaqus simulation is meshed with 2560 brick elements, while the ELLIP3D simulations will solely have as many “elements” as particles simulated. A two-particle compaction simulation run implicitly in Abaqus requires approximately 16 hours to run; however, the two-particle compaction in ELLIP3D runs for about 30 seconds to reach 1e5 time steps. These two-particle simulations in Abaqus and ELLIP3D are compressed to 50% strain and appear in Figures 9 and 10 below, respectively. In these figures, it can be noted that the FE particles can become deformed; however, the DE particles undergo overlap rather than deformation. In order to obtain a proper calibration of material properties, the two-particle compaction in Abaqus and ELLIP3D are simulated simply to serve as a precursor to the 1000 plus particle assemblies to be modeled in future research. Dr. Youssef Hammi of Mississippi State University set up the Abaqus models to assess the normal and shear behavior of spherical nickel particles. In this study, the normal and tangential force-displacement elasto-plasticity behavior is considered. To compare the results, several measures were involved.

First and foremost, the elasto-plasticity constitutive model discussed in Section 1.3.2 required implementation into ELLIP3D, since the code had only Hertz-Mindlin contact theory for rigid, elastic bodies, combined with Coulomb friction. The ELLIP3D code needed to be calibrated with the Abaqus results in order to obtain the yield force κ_0^n and κ_0^t, stiffness k_n and k_t, and hardening parameters H_n and H_t, as denoted in the component form of Section 1.3.2. The sum of the forces acting on the top platen as it displaces results in the force-displacement curve illustrated in Figure 11. For simplicity, a bilinear elasto-plastic curve is determined.

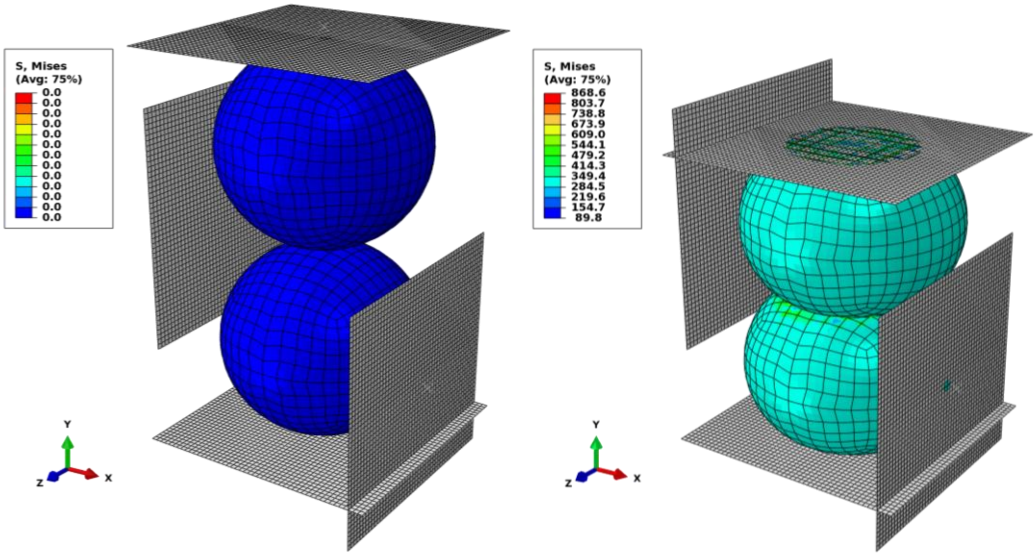


Figure 9: Two-Particle Compaction in Abaqus (Hammi, 2011)

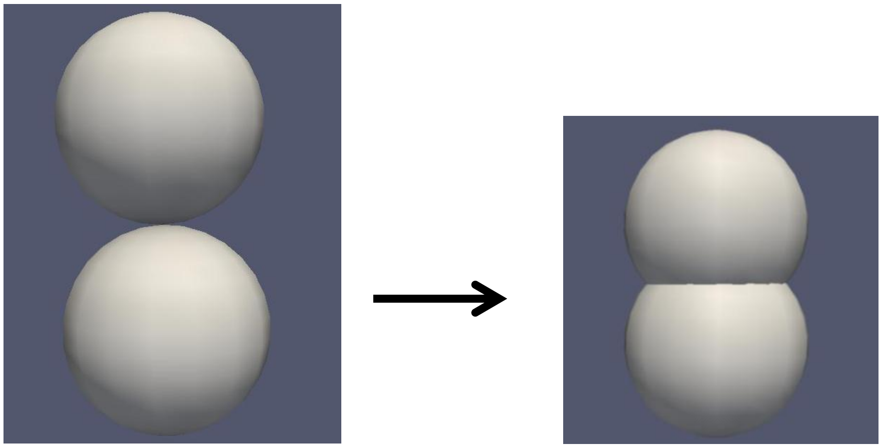


Figure 10: Two-Particle Compaction in ELLIP3D

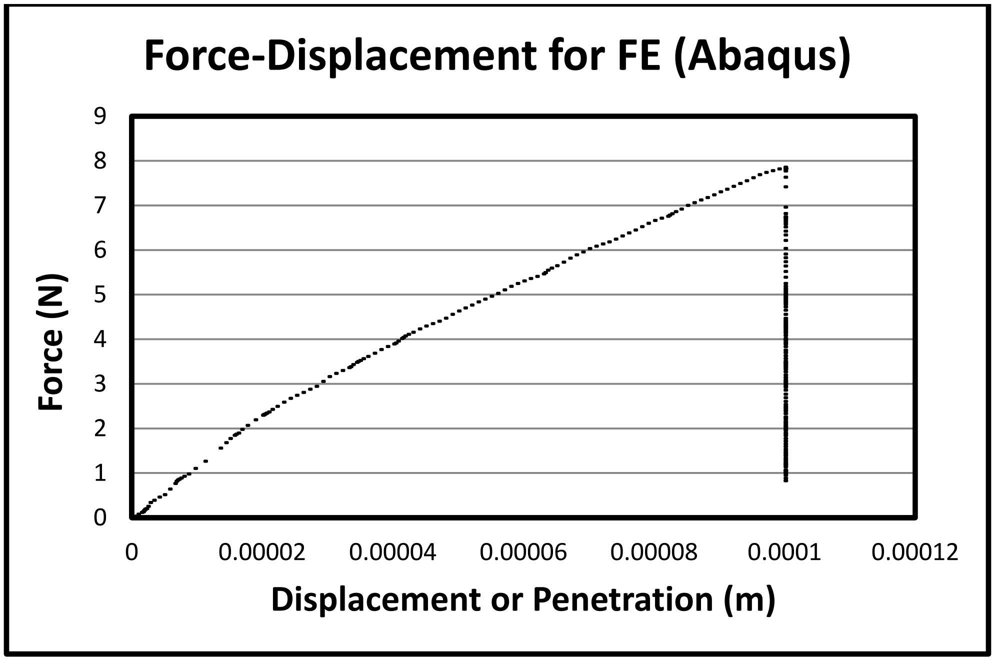


Figure 11: Force Displacement Curve from Abaqus simulations (Hammi, 2011)

Since the bilinearity of the curve is difficult to assess on the force-displacement curve, the stress-strain curve is plotted in order to determine the normal yield force, $\kappa_0{}^n$. Note that the stress is the sum of the contact forces divided by the contact area, where the contact area of two spheres in contact at small strain is $a = (R_0\rho)^{1/2}$. The bilinear approximation to the stress-strain curve is shown in Figure 12, from which the yield stress is approximated to be $\sigma_y = 75{,}000$ Pa. The FE model consists of particles that have a diameter of $D = 2 \times 10^{-4}$ m, thus the formulation $\kappa_0{}^n = \sigma_y{}^n \cdot a = \sigma_y{}^n \cdot (R_0\varepsilon_y D)^{1/2}$ where the yield strain is $\varepsilon_y = 0.05$. This results in

$\kappa_0^n = \left(75000\frac{N}{m^2}\right)\sqrt{\left(\frac{1}{1x10^{-4}m}+\frac{1}{1x10^{-4}m}\right)^{-1}(0.05)(2.0x10^{-4}m)} = 1.6771$ N so the yield force is $\kappa_0{}^n = 1.68$ N. The elastic and elasto-plastic slopes result in a normal elastic stiffness of k_n = 1.68e5 N/m and a normal hardening parameter of H_n = 1.18e5 N/m. The FE particles were prescribed with the provided material properties for nickel powder: E = 69,000 MPa for the Young's modulus and ν = 0.3 for the Poisson's ratio. The plasticity model used in Abaqus can be found in Appendix A, Figure 36. The plasticity model used in Abaqus is a Von Mises tabular isotropic hardening with linear isotropic elasticity. The mechanical constraint formulation, i.e., friction model used is the kinematic contact method with finite sliding, incorporating a coefficient of friction of 0.1 for both, particle-to-particle and particle-to-wall interaction. From the tangential force-displacement results in Abaqus, the ELLIP3D elasto-plasticity parameters could be calibrated with a tangential stiffness of k_t = 3.65e5 N/m and tangential hardening parameter H_t = -1.08e4 N/m which can be referred to in Figure 28. The ELLIP3D and Abaqus simulations are shown in Figure 13.

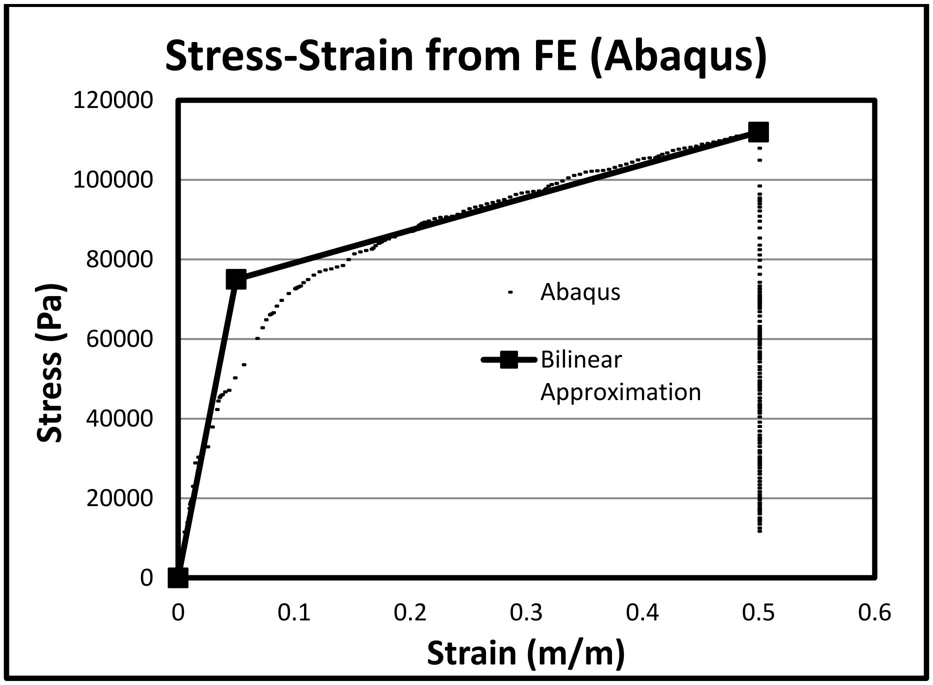


Figure 12: Bilinear Fit to Stress-Strain Results from Abaqus

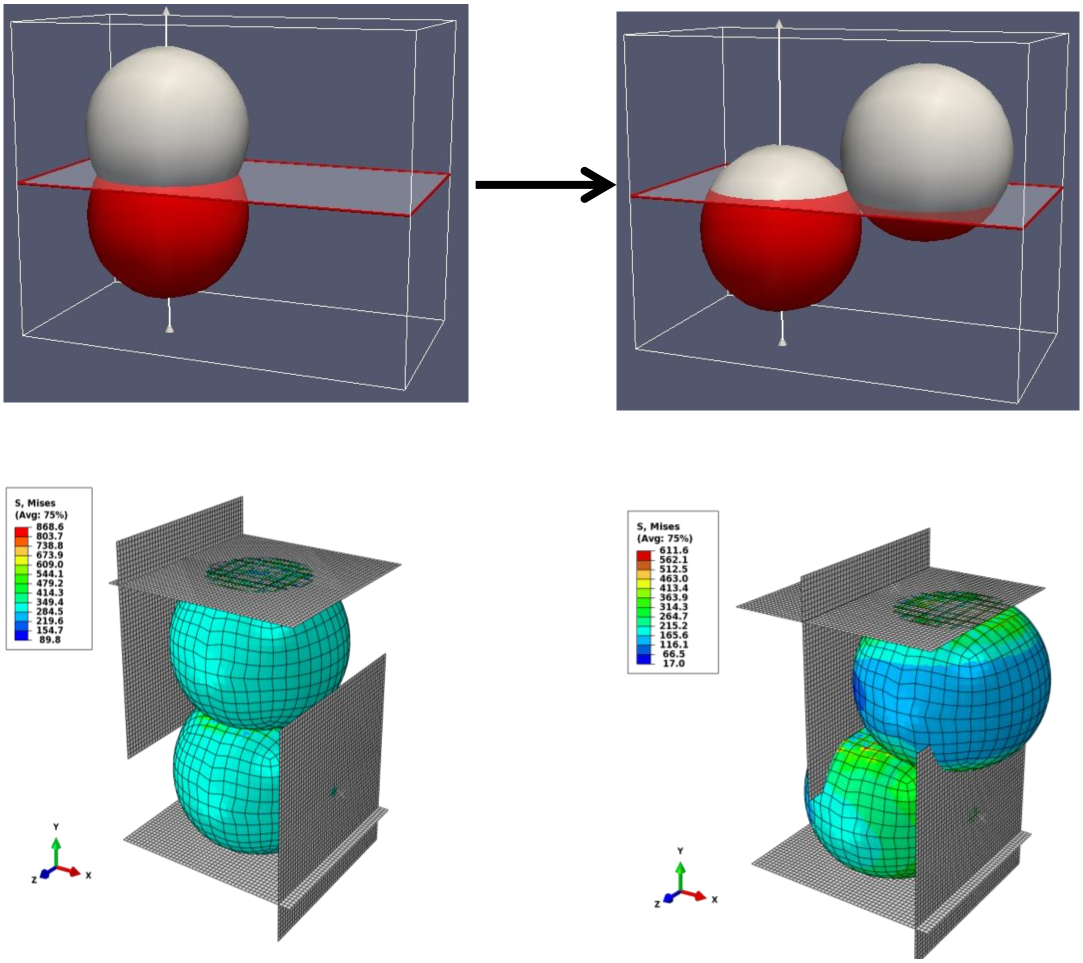


Figure 13: Two-Particle Shear Simulation in ELLIP3D and Abaqus

2.3 Post-processing

All post-processing for the DEM simulation data was achieved by ELLIP3D. The constitutive results were plotted using Excel. Visual three-dimensional particle assemblies were analyzed throughout the simulations using Paraview, an open-source data analysis and visualization software (www.paraview.org). Instructions for building Paraview plugins for ELLIP3D readable files can be found in Appendix D. See Figure 14 for a flow chart of the methodology.

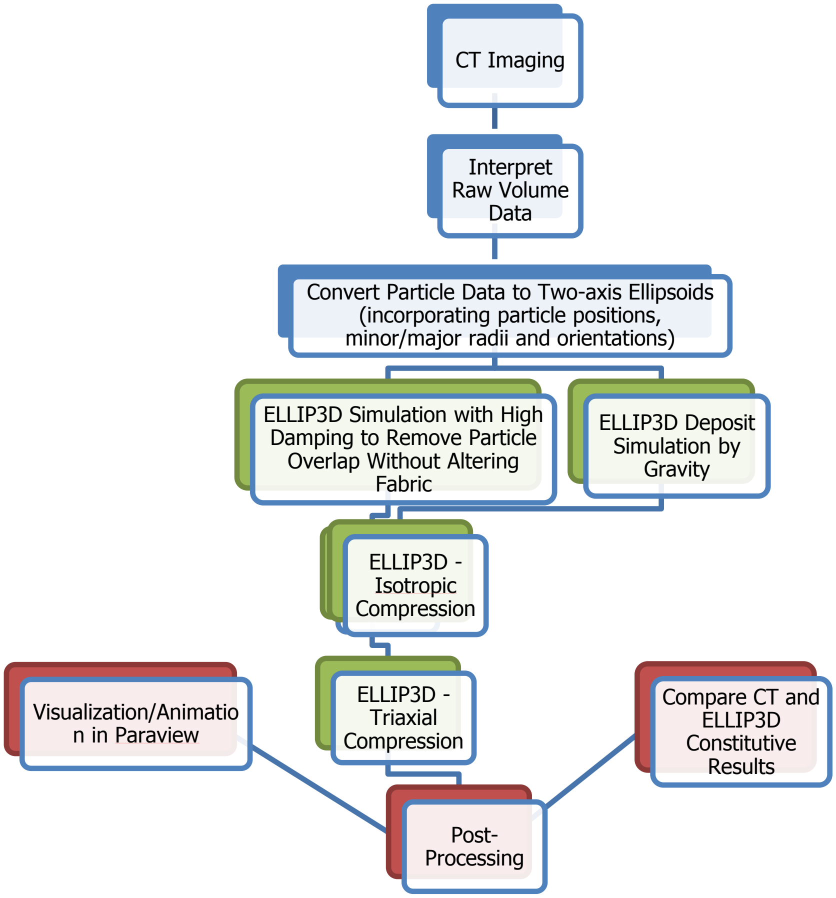


Figure 14: DEM Flow Chart for Simulations of Granular Assemblies

Chapter 3: Results

3.1 *In situ* vs. Gravity Deposition Simulations for Granular Materials in Triaxial Compression

The DEM utilizes a particle-based material modeling approach in order to simulate the macroscopic system behavior. These simulations assess the constitutive results of three-dimensional particle arrangement (fabric), as well as RVE size. In order to assess the accuracy of the ELLIP3D simulations, a comparison to experimental measures will not only enable the calibration of input parameters into ELLIP3D, but also support the validation of the DEM approach.

The experimental results of the principal stress ratio (PSR) vs. axial strain for the ASTM 20/30 and F-75 specimens are shown Figures 15 and 16, respectively. It can be noted that only the initial elastic-plastic portion of the PSR vs. axial strain curve can be achieved with the rigid boundary walls in ELLIP3D. The overall behavior demonstrates that the samples begin to exhibit an increase in PSR to a peak value then soften, in which bulging and shear band formation occur (Alshibli, 2011). Additionally, sharp drops in PSR can be seen in the curves which are the points where the SMT scans were performed. During the imaging, data collection time takes ten minutes and reconstruction time requires five minutes (Alshibli, 2008).

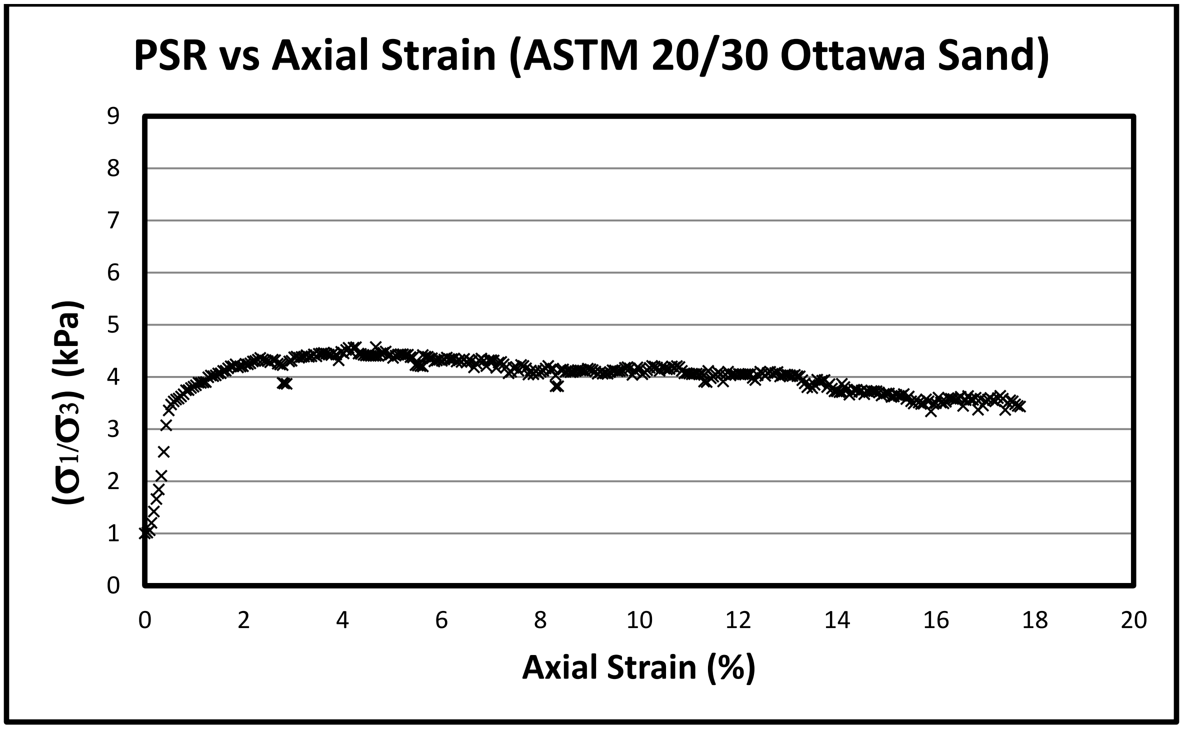


Figure 15: Principal Stress Ratio vs. Axial Strain for ASTM 20/30 Quartz Ottawa Sand (Alshibli, 2011)

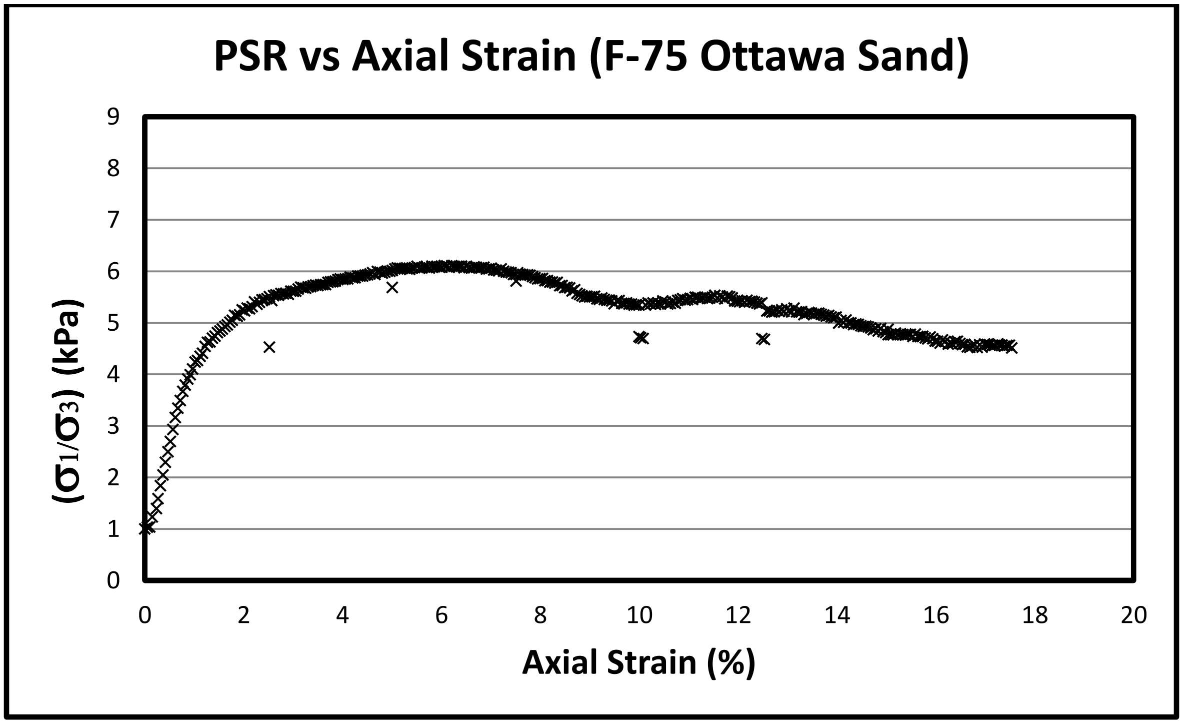
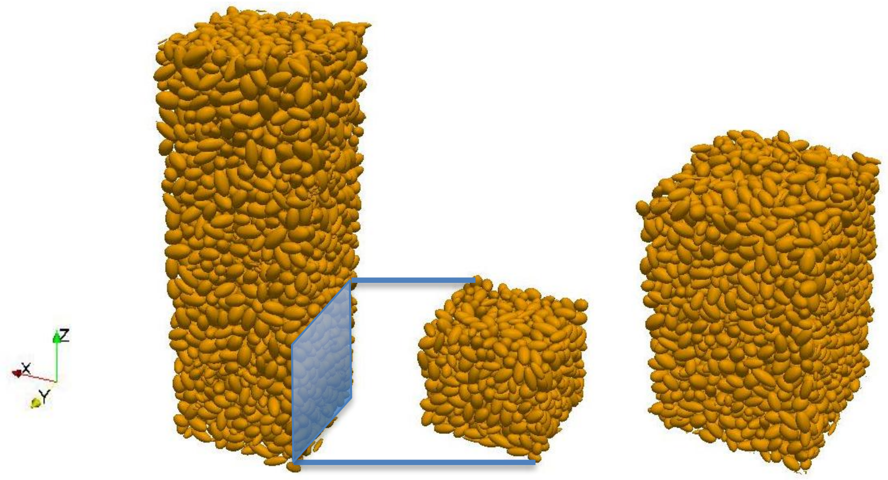


Figure 16: Principal Stress Ratio vs. Axial Strain for F-75 Quartz Ottawa Sand (Alshibli, 2011)

3.1.1 ASTM 20/30 Quartz Ottawa Sand ELLIP3D vs. Experimental Results

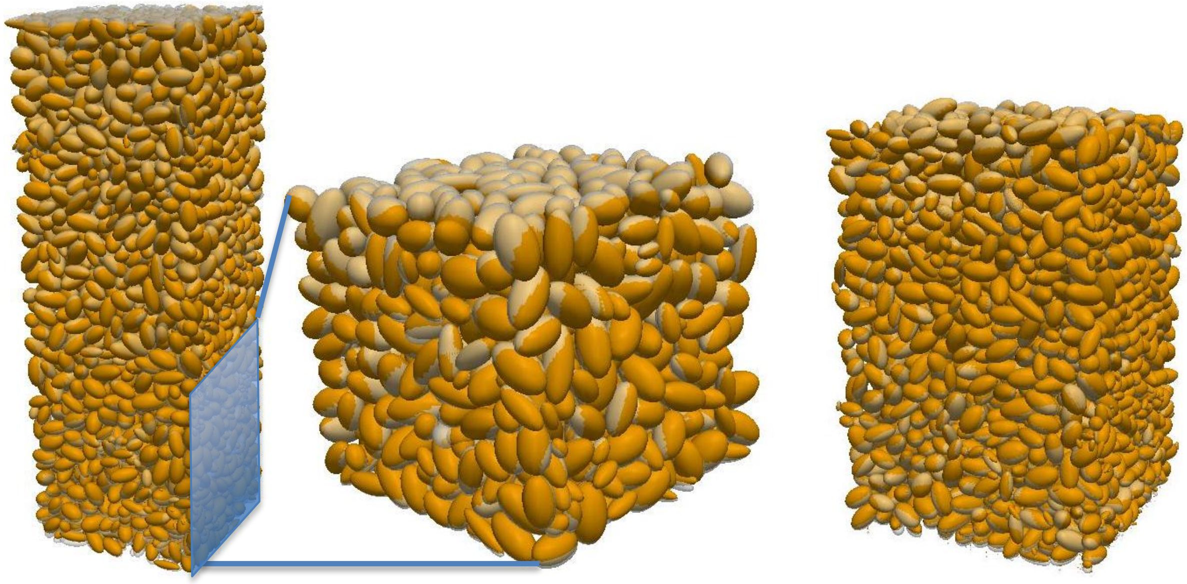
Simulations of conventional triaxial compression tests were performed on the following assemblies of ASTM 20/30 quartz Ottawa sand: 4200 particle boxed RVE which will be called ASTM-Large-RVE, 897 particle boxed RVE trimmed from the same larger RVE which will be denoted as ASTM-Small-RVE, and the large RVE where the fabric is lost by gravity deposition which will be named ASTM-Large-RVE Post-Deposit. Figures 17 (a) – 17 (c) show the boxed RVEs of the ASTM 20/30 quartz Ottawa sand particles after each of the stages preceding the triaxial compression test, i.e., float, isotropic compression and/or gravity deposition followed by isotropic compression. Figures 18 (a) – 18 (c) show the final configurations of the assemblies after triaxial compression with a semi-transparent overlay of the initial particle assembly in order to visually recognize particle displacement. By characterizing the initial and final particle assemblies, it appears that the greatest particle motion occurs at the top and bottom portions of the assemblies, where the boundaries are displacement controlled. A close-up image of the top of the ASTM-Small-RVE in Figure 18 (b) is shown in Figure 19. The initial and final void ratios of the ELLIP3D assemblies during triaxial compression simulations were higher than that in the experiment, e.g., e_0 = 0.78 and e_f = 0.77 at 2% axial strain for the ASTM-Small-RVE as compared to the e_0 = 0.523 in the experiment. A higher void ratio is likely a consequence of the two-axis ellipsoid approximation.

The triaxial compression test simulation parameters are presented in Table 3. The four vertical side walls of the rigid container move outward as the assembly is compressed in order to maintain the prescribed constant confining pressure. These confining pressures on the four walls are given by the sum of the normal forces of the particles divided by the area of the wall. The top and bottom rigid boundaries are displacement controlled at prescribed strain rates shown in Table 3. A much higher axial displacement rate was required in the simulations than that used in experiment. This is due to the necessity of a very small time step in order to maintain stability in terms of particle motion, and using a realistic compression rate would take an unfeasible amount of time to run (approximately 2700 days or 7.4 years running in parallel on 2 or 3 processors for the ASTM-Large-RVE). The results in Figures 20-22 demonstrate that the principal and deviator stress versus strain responses for the large RVE with the soil fabric from the CT imaging to be higher than that of the experiment, which is higher than that of the small RVE and the RVE where the fabric is lost by gravity deposition. The sole difference in input parameters among the simulations is that the Post-Deposit simulation uses a slower compression rate of $7\text{x}10^{-4}$ m/s. This results in a smoother response with less noise than the other two simulations; however, it required longer simulation time (2 weeks). The two-axis ellipsoid particle shapes are crude in approximation to the more angular particles in Figure 6, and thus, a high interparticle friction coefficient is used to account for particle interlocking occurring in the experiment of the sand specimen. It can be presumed that the constitutive results are more sensitive to other factors besides the soil fabric, such as interparticle friction and mass and moment of inertia scaling (M&MoIS), but more analysis is needed. No experimental data were provided for the volumetric strain vs. axial strain plot for comparison to the simulations in Figure 22. Here, negative volumetric strain is assumed to be compaction. The small and large RVEs with fabrics attained from CT imaging did not attain as dense of packing as the Post-Deposit RVE, which justifies the difference compaction observed in Figure 22. The Post-Deposit RVE undergoes initial compression, followed by dilation at low values of volumetric strain which can be seen more clearly in Figure 23; however, this is the typical response that a dry, dense sand with low confining pressure would undergo. This Post-Deposit simulation had initial and final void ratios of e_0 = 0.701 and e_f = 0.740.



(a) (b) (c)

Figure 17: Initial assembly configurations of (a) ASTM-Large-RVE, (b) ASTM-Small-RVE (c) ASTM-Large-RVE Post-Deposit



(a) (b) (c)

Figure 18: Final assembly configurations with a semitransparent overlay of the initial assembly for (a) ASTM-Large-RVE, (b) ASTM-Small-RVE (c) ASTM-Large-RVE Post-Deposit

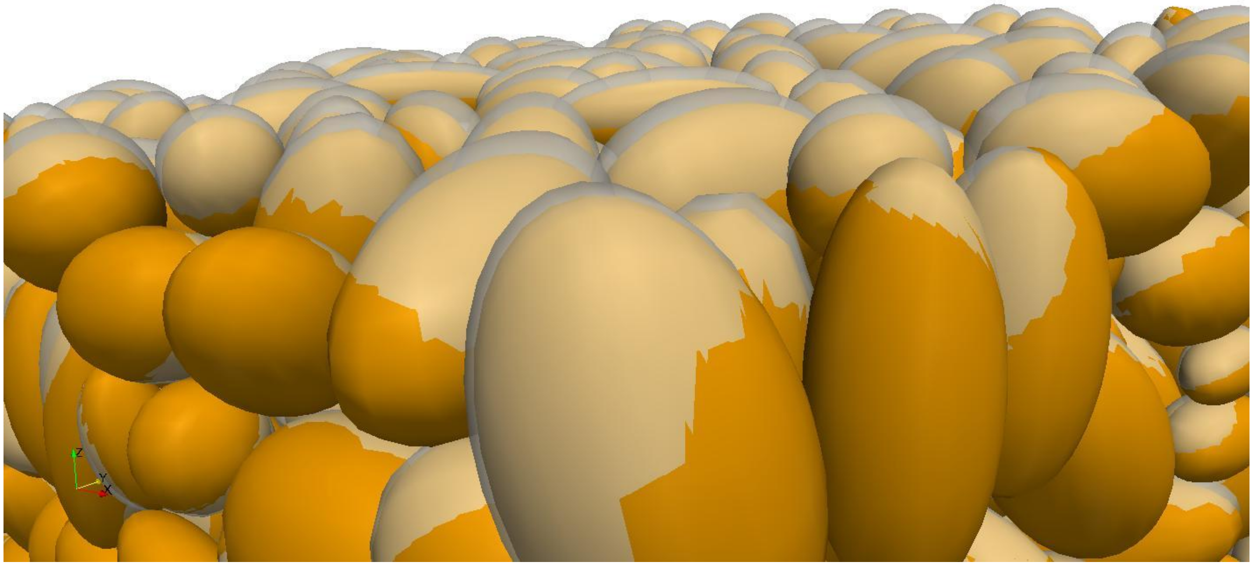


Figure 19: Close-up of the top of the final assembly configurations with a semitransparent overlay of the initial assembly for the ASTM-Small-RVE

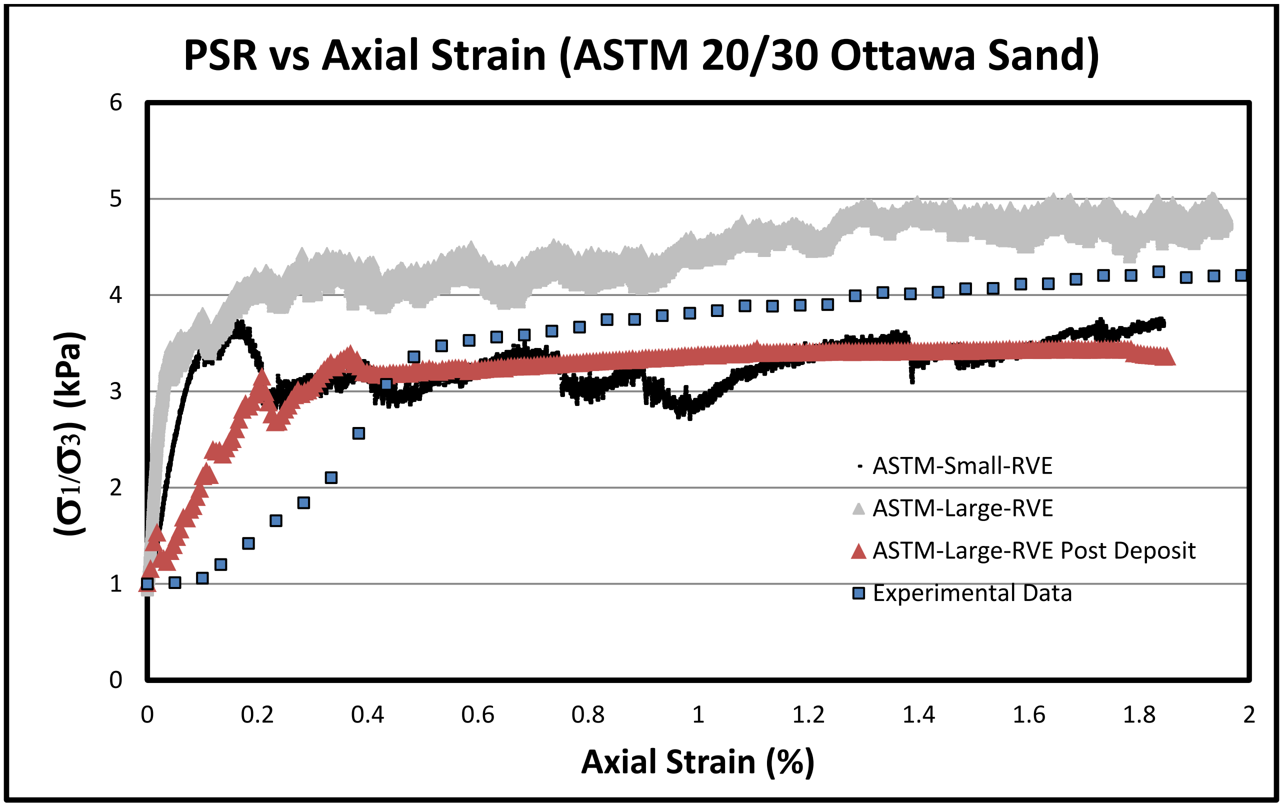


Figure 20: Principal Stress Ratio vs. Axial Strain for ASTM 20/30 Sand in Triaxial Compression

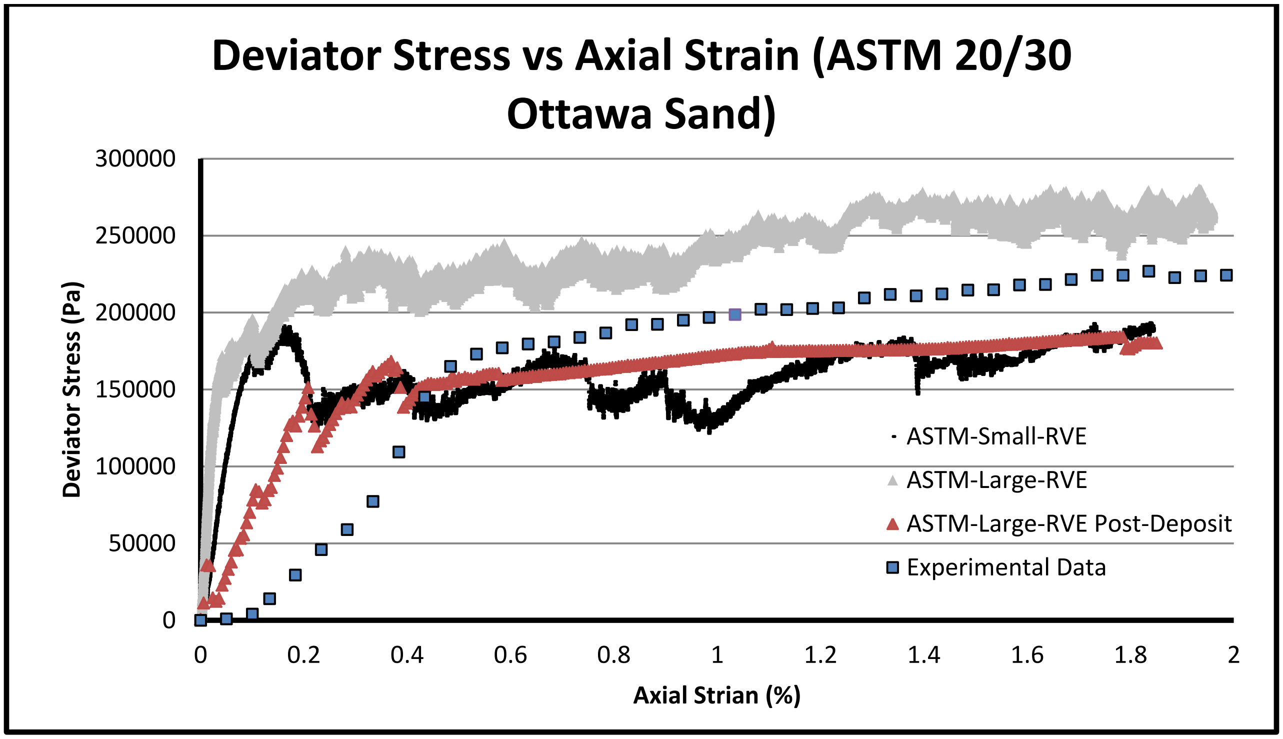


Figure 21: Deviator Stress vs. Axial Strain for ASTM 20/30 Sand in Triaxial Compression

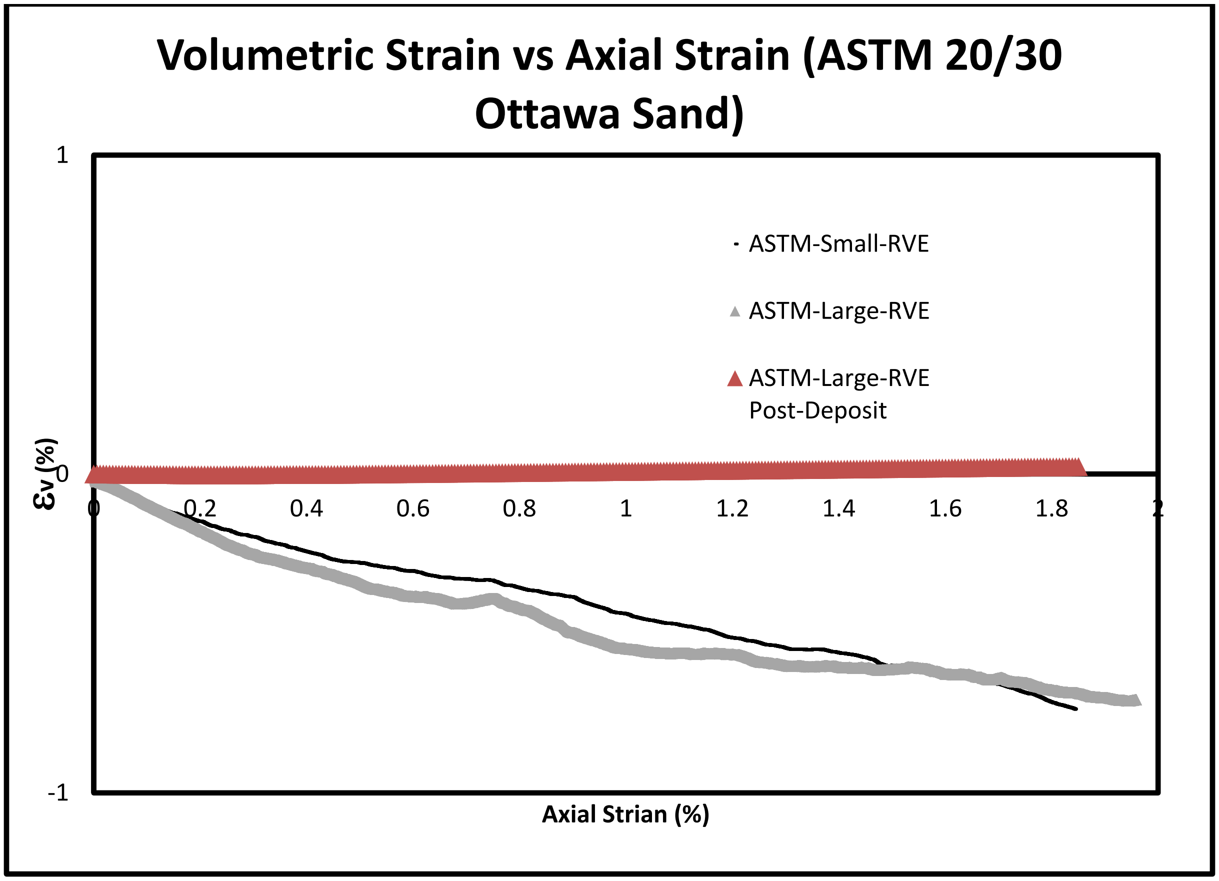


Figure 22: Volumetric Strain vs. Axial Strain for ASTM 20/30 Specimen in Triaxial Compression

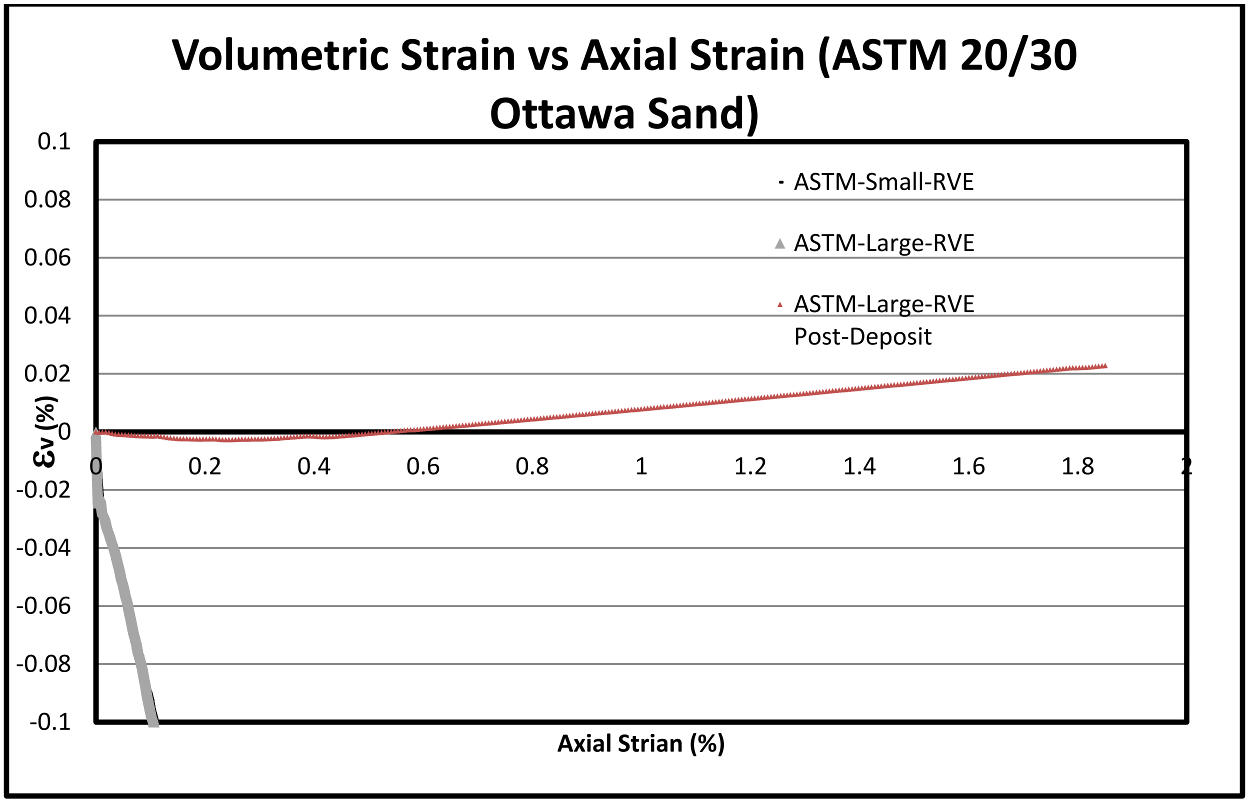


Figure 23: Close-up Volumetric Strain vs. Axial Strain (Same as Figure 22 but with Smaller Y-axis Values)

	ASTM-Large-RVE	ASTM-Small-RVE	ASTM-Large-RVE Post-Deposit
Time Step Size	5.0×10^{-8} s	5.0×10^{-8} s	5.0×10^{-8} s
Mass Scaling	10	10	10
Moment Scaling	10	10	10
Viscous Background Damping on Mass	1.0×10^{4}	1.0×10^{4}	1.0×10^{4}
Viscous Bkrnd. Damp. on Mnt. of Inert.	1.0×10^{4}	1.0×10^{4}	1.0×10^{4}
Contact Damping	0.1	0.1	0.1
Particle-to-Particle Friction	20	20	20
Particle-to-Boundary Friction	20	20	20
Cohesion	0	0	0
Boundary Wall Compression Rate	7.0×10^{-3} m/s	7.0×10^{-3} m/s	7.0×10^{-4} m/s
Boundary Wall Release Rate	7.0×10^{-3} m/s	7.0×10^{-3} m/s	7.0×10^{-4} m/s
Confining Pressure to Achieve	70 kPa	70 kPa	70 kPa

Table 3: ELLIP3D Simulation Parameters for ASTM 20/30 Specimen

3.1.2 F-75 Quartz Ottawa Sand ELLIP3D vs. Experimental Results

Selecting the parameters for the F-75 particle assemblies were particularly challenging. The synthesized particles used in previous ELLIP3D simulations (Yan, 2008) were on average more than one order of magnitude larger than the D_{50} = 0.16 mm diameter F-75 particles. For a realistic gravity deposition simulation, the mass and moment of inertia scaling must be set to a value of one, and there may not be background damping. In view of the fact that the particles are so small, large particle overlaps will develop unless an especially small time step is used, since viscous background damping and mass and moment of inertia scaling should not be used to maintain stability during dynamic simulations. Using an extremely small time step ($1\text{x}10^{-8}$ s) calls for a large number of time steps ($>7\text{x}10^{6}$) in order to complete the simulation. The gravity deposition simulations for the F-75, 2369 particle assembly, ran in parallel using Open MP for about seven days on six Intel Xeon 3.07GHz processors.

The difficulty with compressing these small particles in isotropic compression was discovered with the selection of mass and moment of inertia scaling (M&MoIS). When employing a high value of M&MoIS (such as 1000), the confining pressure would be achieved while observing virtually no particle motion. Utilizing a M&MoIS of 100 would result in a slightly greater particle motion at the boundaries, however, the confining pressure would be achieved before the particles could close in to create a dense packing. This means that a higher particle M&MoIS factor results in greater forces acting on the boundary walls by each particle, which may yield unrealistic results. When running simulations with a M&MoIS of 1, there is too much particle motion as can be demonstrated in Appendix B. Using a M&MoIS of 10 would most likely lead to a stable result while maintaining a low force on the boundary walls in order not to attain the confining pressure prematurely; however, the computation time is the limiting factor. Again, the small size of the particles requires a small time step and low boundary wall compression rate for stability. The largest time step and highest wall compression rate values for the F-75 sample with a M&MoIS of 10 is $5\text{x}10^{-8}$ s and $1\text{x}10^{-4}$ m/s, respectively. Running this simulation to $2\text{x}10^{6}$ time steps required approximately 48 hours to run on 6 processors. To achieve a dense packing through isotropic compression that would experience about 10% strain, the simulation would need to run to approximately $4\text{x}10^{7}$ time steps and could take over 5 weeks to complete. In other words, the DEM is a severely limited by computation time when running simulations of very fine particles without the use of scaling.

Here, we have a comparison of experimental data vs. ELLIP3D simulations for the 2369 particle F-75 quartz Ottawa sand. Previously, it was stated that the very small particles inhibited stable numerical solutions. These plots of PSR vs. axial strain and volumetric strain vs. axial strain were attained from CT imaging with particle size and orientations in their *in situ* state; however, the particle radii are greater than the actual size by a factor of 10. This volumetric scaling is equivalent to designating a M&MoIS of $(10)^3 = 1000$. However, because the contact stiffnesses are nonlinear (Hertz-Mindlin) functions of particle size, the results will not be the same. Thus, it is more important to simulate actual particle sizes and shapes of the granular material using DEM. Figures 24 and 25 along with Table 4 demonstrate the severe sensitivity of M&MoIS and particle-to-particle friction. It can be observed that a small increase in interparticle friction, while keeping other parameters the same, will cause a large increase in principal stress ratios.

The input parameters noted in Table 4 demonstrate the influence of viscous background damping, mass and moment of inertia scaling, interparticle contact damping, and particle-to-particle friction. Viscous background damping can be conceptually thought of as subjecting the particles to a virtual viscous fluid that would reduce the particle motion. A comparison of the simulations of Trial 1 to Trial 2 demonstrates that a reduction in viscous background damping on mass and moment of inertia from 10^5 to 10^3 will enable greater particle motion and reduce the PSR. In DE modeling, interparticle friction reduces the slippage of rounded particles which ultimately helps sustains the rigidity of the macroscopic structure. Trial 3 is simulated with the same parameters as Trial 1 but with an interparticle friction of 1.3 instead of 1.73 and results in a greater initial PSR but lower PSR overall. The parameters implemented into Trial 4 are consistent with Trial 1 but uses a contact damping ratio of 0.1 instead of 0.05. This increase in contact damping also induces a lower PSR by reducing the particle-to-particle and particle-to-boundary forces. Finally, Trial 5 exhibits a reduction in interparticle friction and an increase in contact damping. The greater initial surge in PSR leads to a gradual decrease in PSR for which the macrostructure undergoes strain softening.

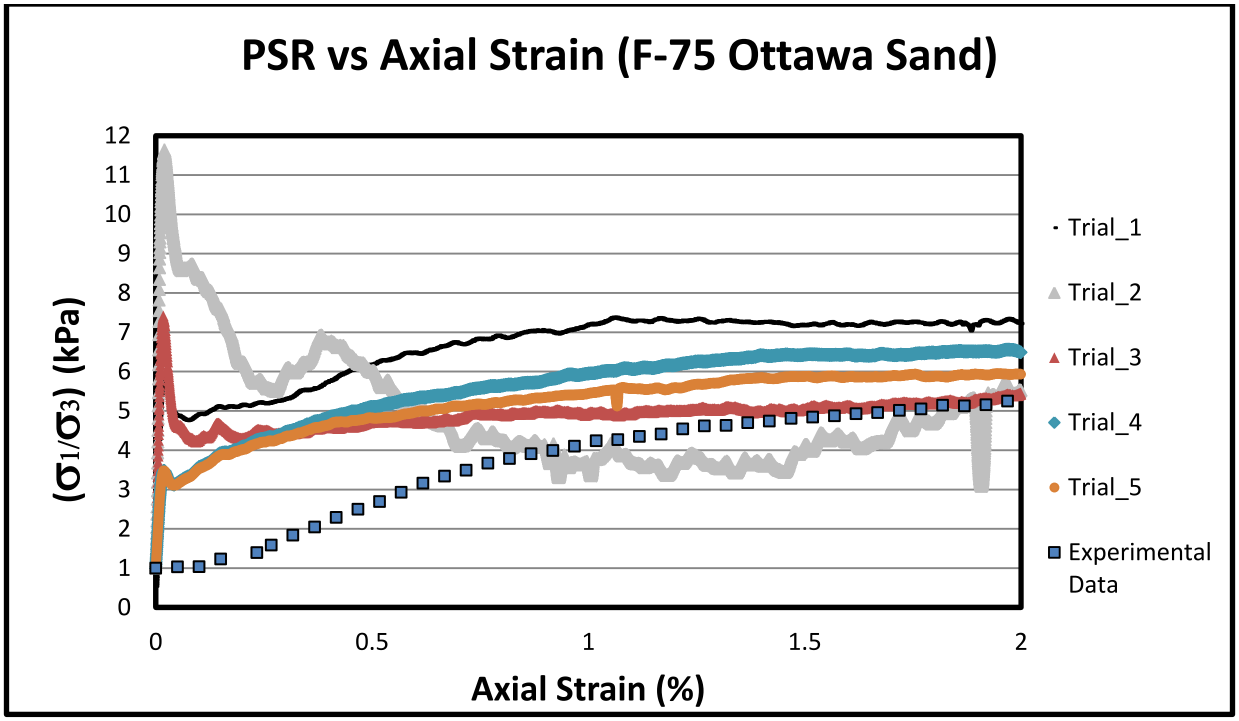


Figure 24: Principal Stress Ratio vs. Axial Strain for F-75 Ottawa Sand in Triaxial Compression

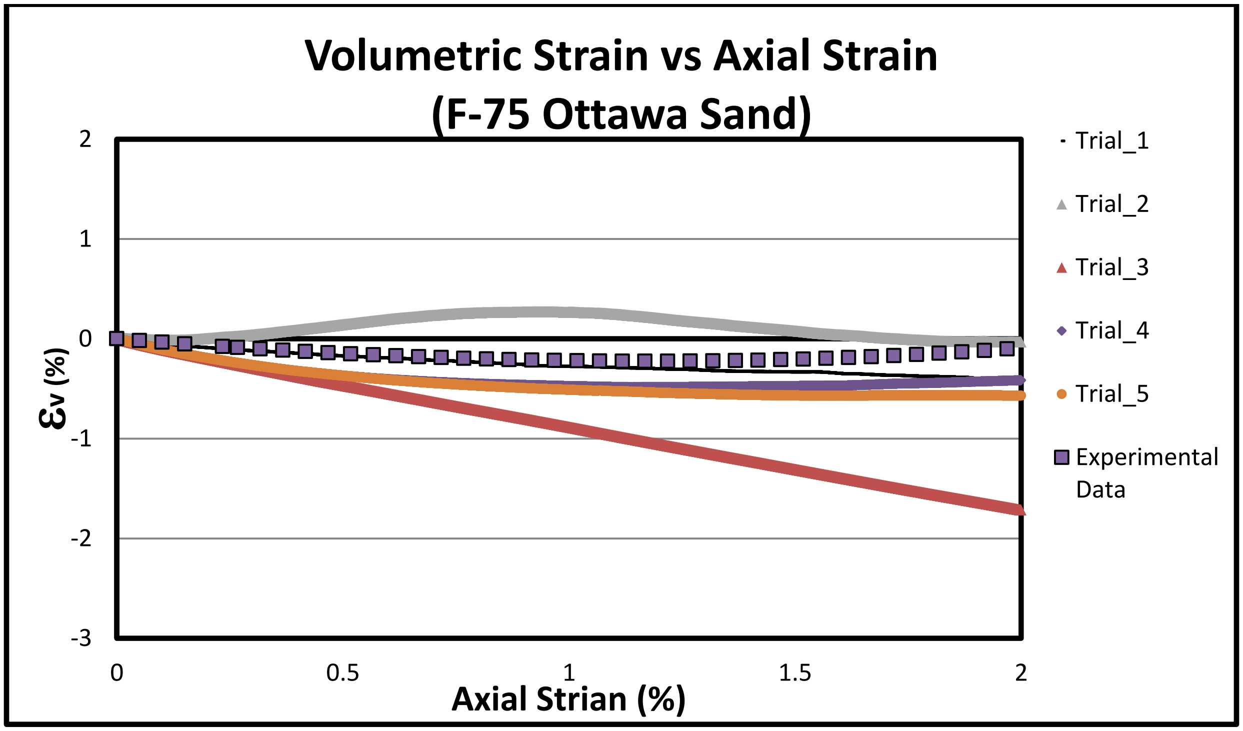


Figure 25: Volumetric Strain vs. Axial Strain for F-75 Ottawa Sand in Triaxial Compression

The volumetric strain vs. axial strain up to 2% strain in Figure 25 shows that the fine sand undergoes a decrease in volumetric strain, then dilation occurs after initial compression. To put this initial drop in perspective, a full volumetric strain vs. axial strain plot of the experimental (physical) results is shown in Figure 26.

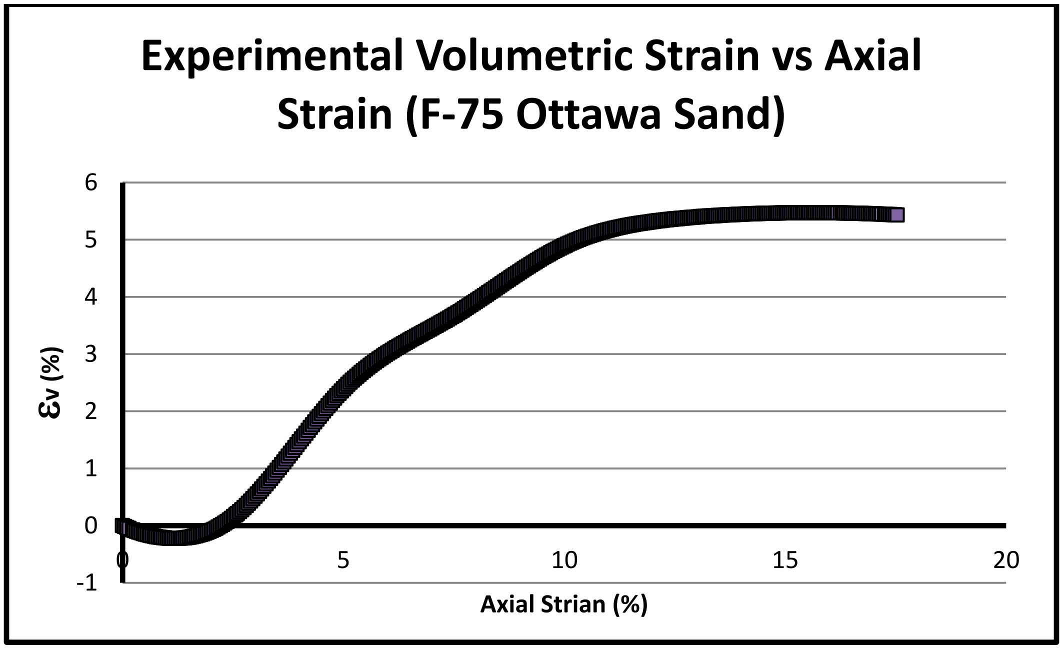


Figure 26: Volumetric Strain vs. Axial Strain for F-75 Ottawa Sand in Triaxial Compression from Experimental Results (Alshibli, 2011)

	Trial_1	**Trial_2**	**Trial_3**	**Trial_4**	**Trial_5**
Time Step Size	3.2×10^{-8} s	3.2×10^{-8} s	3.2×10^{-8} s	3.2×10^{-8} s	3.2×10^{-8} s
Mass Scaling	1.0×10^{5}	1.0×10^{5}	1.0×10^{5}	1.0×10^{5}	1.0×10^{5}
Moment Scaling	1.0×10^{5}	1.0×10^{5}	1.0×10^{5}	1.0×10^{5}	1.0×10^{5}
Viscous Background Damping on Mass	1.0×10^{5}	1.0×10^{3}	1.0×10^{5}	1.0×10^{5}	1.0×10^{5}
Viscous Bkrnd. Damp. on Mnt. of Inert.	1.0×10^{5}	1.0×10^{3}	1.0×10^{5}	1.0×10^{5}	1.0×10^{5}
Contact Damping	0.05	0.05	0.05	0.1	0.1
Particle-to-Particle Friction	1.73	1.73	1.3	1.73	1.3
Particle-to-Boundary Friction	0.5	0.5	0.5	0.5	0.5
Cohesion	0	0	0	0	0
Boundary Wall Compression Rate	7.0×10^{-3} m/s	7.0×10^{-3} m/s	7.0×10^{-3} m/s	7.0×10^{-3} m/s	7.0×10^{-3} m/s
Boundary Wall Release Rate	7.0×10^{-3} m/s	7.0×10^{-3} m/s	7.0×10^{-3} m/s	7.0×10^{-3} m/s	7.0×10^{-3} m/s
Confining Pressure to Achieve	58 kPa	58 kPa	58 kPa	58 kPa	58 kPa

Table 4: ELLIP3D Simulation Parameters for F-75 Specimen

3.2 Compaction of Metallic Powder -- Elasto-Plastic Constitutive Model

The bilinear elasto-plasticity model discussed in Section 2.2 was implemented in ELLIP3D. The calibration of constitutive parameters for the elasto-plasticity model yields the force-displacement relationship for the two-particle compaction and shear simulations as shown in Figure 27 and 28, comparing Abaqus to ELLIP3D results. The FE simulation for the uniaxial compaction was run with a displacement driven top boundary and fixed bottom boundary; the DE simulation employed top and bottom boundary displacements. For the centrically loaded two-particle compaction simulations, the penetration of the particles is equivalent to the axial displacement and these data are plotted in Figure 27. The two-particle shear simulations maintained the bottom particle to be fixed while the top particle was displacement controlled; results are plotted in Figure 28.

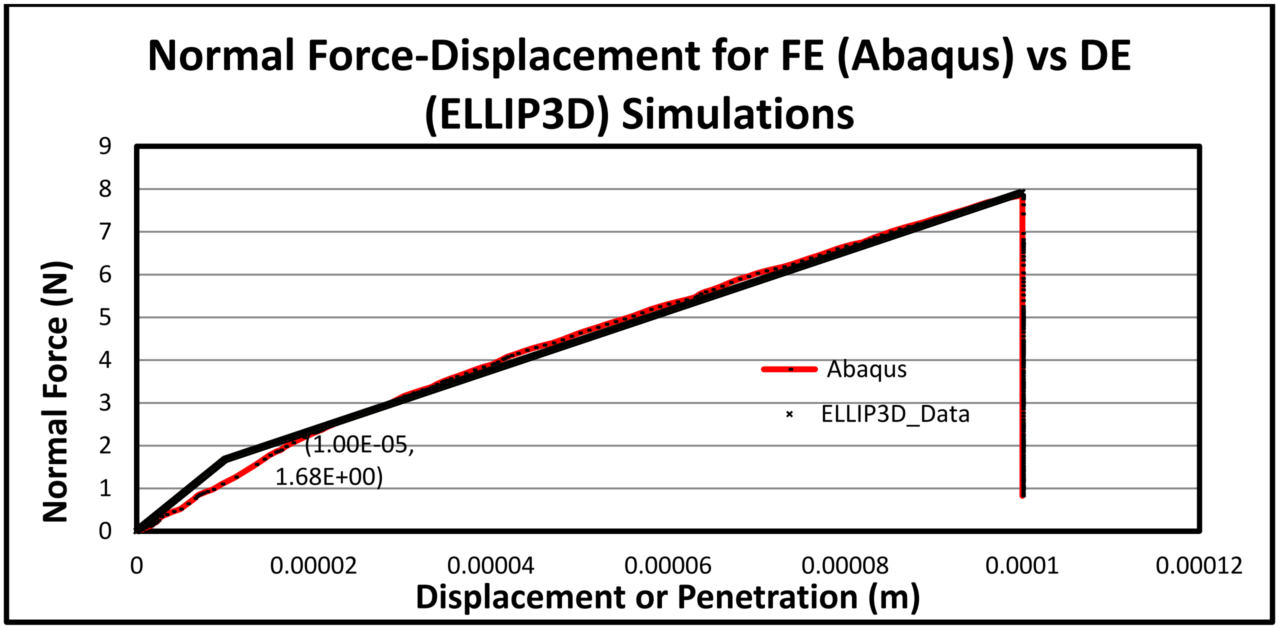


Figure 27: Force-Displacement Plot for Two-Particle Compaction in Abaqus and ELLIP3D

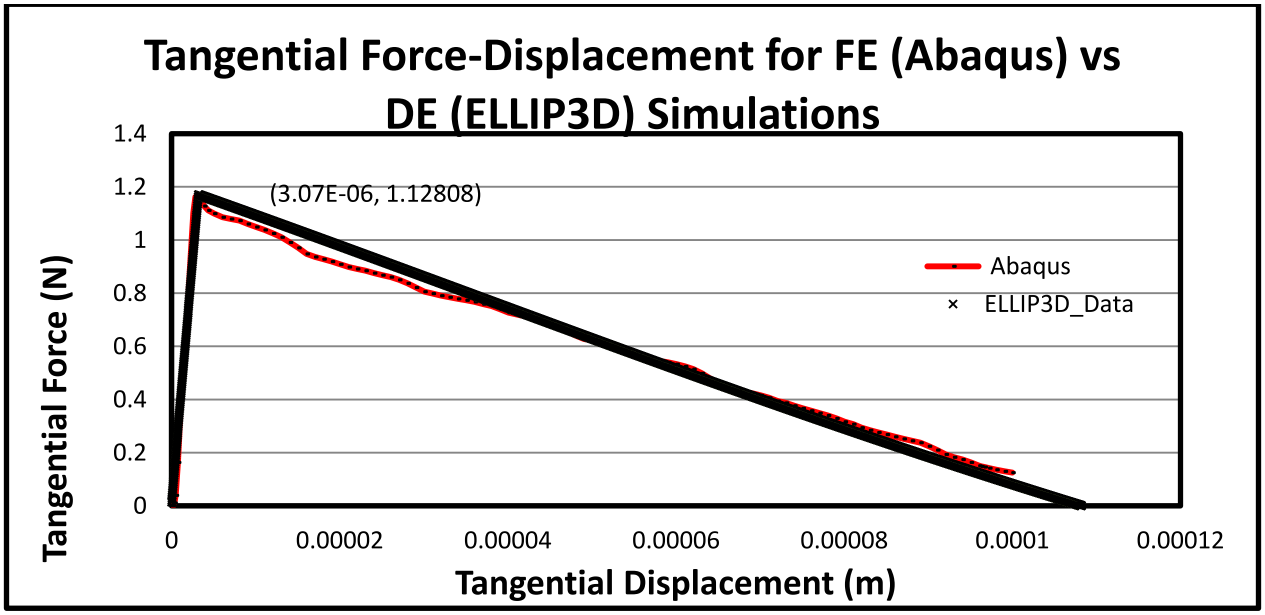


Figure 28: Tangential Force-Displacement Plot for Two-Particle Compaction in Abaqus and ELLIP3D

Chapter 4: Conclusions and Future Work

The DEM simulations require extensive simulation time and computing power to achieve desired parametric accuracy. The simulations of the ASTM 20/30 RVEs demonstrated the sensitivity of constitutive results by the selection of input parameters such as mass and moment of inertia scaling and interparticle friction. As concluded with the small F-75 particle assemblies, stability for small particles is limited by time step size, which induces a large number of time steps and simulation time. Solely preparing the particle assembly for triaxial compression is computationally demanding since the parameters used to maintain stability and decrease time step size, i.e. viscous background damping and mass and moment of inertia scaling, would essentially freeze the particles and generate a large force on the boundary walls, causing the boundary wall confining pressure to be reached prematurely and yielding trivial results. The simulations ran on the two-particle metallic powder compaction tests serve as a precursor to the larger assemblies of metallic powder which enable comparisons of the DE model to the FEA results.

Potential future work that relates to the topics presented in this thesis would include the implementation of cylindrical boundaries into ELLIP3D to simulate full assemblies or cylindrical RVEs. Polar coordinate implementation in ELLIP3D would simplify the analysis. The implementation of particle boundaries (Peters et. al., 2009) to model flexible membranes on triaxial compression tests of granular materials would ideally allow for bulging to occur and possibly even capture shear band formation as evident in Figures 15 and 16. Without flexible membranes, only a small portion of the stress-strain curve can be captured (up to 2% strain). The DEM on its own is computationally expensive for general geotechnical applications; consequently, this work intends to support research involving multiscale, DE-FE coupling efforts. Additionally, utilizing poly-ellipsoids in ELLIP3D for sands of different morphology as opposed to two-axis ellipsoids may significantly reduce the rolling and sliding observed in the triaxial compression simulations (Peters et. al., 2009). This lack of particle interlocking using two-axis ellipsoids was compensated for with a high particle-to-boundary and particle-to-particle friction. Another potential effort could be to track the displacements of selected particles throughout a test, which is a capability of CT imaging as well as ELLIP3D and could contribute to a greater understanding of shear band formation. This may lead to greater insight to shear band failure.

Improvements can also be made to the current macroscopic stress and strain measures. The current stress measure consisting of the summation of forces acting on a boundary wall divided by the area of the boundary wall would be improved through the implementation of a proper overall macroscopic average stress measure for the RVE, which is based on the definition for stress in a granular medium (Christoffersen et. al., 1981):

$$\boldsymbol{\sigma} = \frac{1}{2 \cdot V} \sum_{\varepsilon=1}^{N} (\mathbf{f}^{\varepsilon} \otimes \mathbf{l}^{\varepsilon} + \mathbf{l}^{\varepsilon} \otimes \mathbf{f}^{\varepsilon}) \qquad (4.1)$$

where $\boldsymbol{\sigma}$ stands for the average stress in the considered volume V; ε represents the particle contact; N stands for the total number of contacts in the volume V considered; $\mathbf{f}^{\varepsilon}$ represents the interparticle force at contact ε; $\mathbf{l}^{\varepsilon}$ is the branch vector connecting the centroids of the two-particles interacting at contact ε. Such a theory is proposed because the principal of *noncoaxiality* describes the phenomenon that the directions of the overall macroscopic stress tensor do not directly correspond to the principal directions of the deformation rate. This formulation is derived through the balance of moments and principal of virtual work. It is assumed that moments are not transmitted at contacts and only one of the two contact force pairs is used in the summation in Equation 4.1. A schematic of two particles at contact is shown in Figure 29. A proper macro-scale strain measure is equally as valuable as having a suitable macroscopic stress measure. The current strain measure used in this DEM research is the classical Cauchy Strain ΔL/L, which is the ratio of the total deformation to the current dimension of an RVE. Various authors have proposed methods to link microscopic to macroscopic strains for particle-based mechanics using equivalent continuum methods (Bagi, 1993; Kruyt and Rothenburg, 1996; Kuhn, 1997; Cambou et. al., 2000; Dedecker et. al., 2000; Kruyt, 2003). Duran (2009) explores the validation and verification of various 3D micro-mechanical strain formulations using DEM codes. Their theories aim to account for discrete particle translations through microscale strain tensors and extend it to macroscopic strain theories. Figure 30 illustrates the contact vectors $\mathbf{r}^{i\varepsilon}$ and $\mathbf{r}^{j\varepsilon}$ that connect the centroids of the particles *i* and *j* to the point of contact *ε*. Translations of these contact points can be described by the displacements of the centers of the particles and the rigid-body rotations about their centers as follows:

$$d\boldsymbol{u}^{i\varepsilon} = d\boldsymbol{u}^{i} + \boldsymbol{r}^{i\varepsilon} \times d\boldsymbol{\omega}^{i} \quad \text{and} \quad d\boldsymbol{u}^{j\varepsilon} = d\boldsymbol{u}^{j} + \boldsymbol{r}^{j\varepsilon} \times d\boldsymbol{\omega}^{j} \qquad (4.2)$$

where d$\boldsymbol{\omega}$ represents the rigid body rotation. The displacement increment at contact can be defined by

$$d\boldsymbol{v}^{\varepsilon} = d\boldsymbol{u}^{j\varepsilon} - d\boldsymbol{u}^{i\varepsilon}. \quad (4.3)$$

In order to attain the strain, the gradient of the vector field d$\mathbf{u}$ is denoted as

$$d\boldsymbol{e} = d\boldsymbol{u}/d\boldsymbol{x}. \quad (4.4)$$

Bagi (1993, 1996) proposed a method for 2D and 3D analysis of particles using space cells of a system. Figure 31 shows the general configuration of the 2D triangles that form the resulting space cell boundary used in the formulation. This configuration applies to 3D geometries with the use of tetrahedra instead of triangles. The space cells are created by connecting the centers of neighboring particles, which do not necessarily require contact, and the resulting boundary cell connects the centers of the boundary particles and this union provides the equivalent continuum. This configuration is ultimately employed by taking the summation of the translation gradient d$\mathbf{e}^{C}$ of each cell C over the entire volume (3D) or entire area (2D). For the 2D case, the left translation gradient is determined to be

$$d\boldsymbol{e}^{C} = \frac{1}{A^{C}} \oint d\boldsymbol{u} \cdot \boldsymbol{n}\, dl \qquad \rightarrow \qquad d\boldsymbol{e} = \frac{1}{A} \sum_{C} A^{C}\, d\boldsymbol{e}^{C} \qquad (4.5)$$

and for the 3D case

$$d\boldsymbol{e}^{C} = \frac{1}{V^{C}} \oint d\boldsymbol{u} \cdot \boldsymbol{n}\, dS. \qquad \rightarrow \qquad d\boldsymbol{e} = \frac{1}{V} \sum_{C} V^{C}\, d\boldsymbol{e}^{C} \qquad (4.6)$$

where **n** is the outwards unit normal vector to the surface of space cell C; d**u** is the translation vector of the boundary point; A is defined as the area of space cell C (V is volume for 3D), and the integration is performed along the boundary line l (boundary surface S for 3D). The symmetric part of d$\mathbf{e}^c$ is the microstructural strain tensor and the antisymmetric part is the average rigid body rotation of the cell.

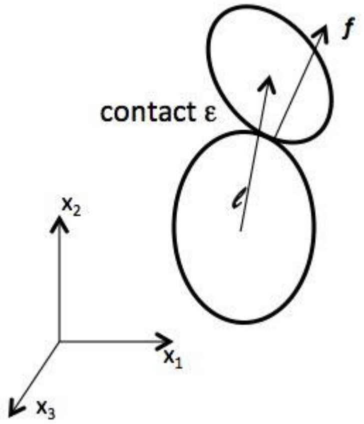


Figure 30: Interparticle force (*f*) and branch vector (*l*) at contact ε

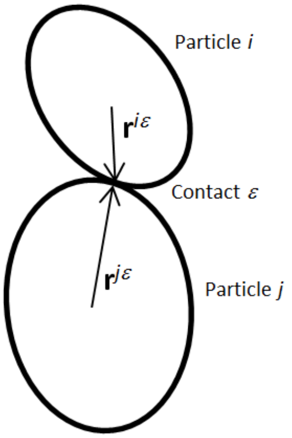


Figure 29: Vectors from Particle Centroids to Point of Contact

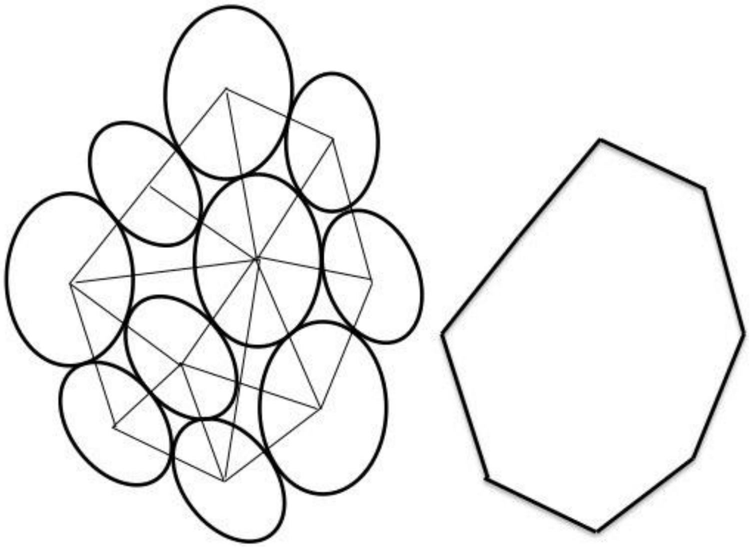


Figure 31: Space Cells and Resulting Boundary Space Cell for 2D Configuration of Particles

In essence, the relations between interparticle forces and macro-level stresses as well as the relation of particle-level displacements and macro-level strains, are intended to increase the accuracy of DEM simulations and ultimately allow researchers to develop micromechanically based constitutive theories. The formulations for calculating stresses and strains need to be modified from those derived by continuum methods since the assumptions, i.e., continuous domains and infinitesimally small points, do not hold true.

References

Alshibli, K., and Hasan, A., 2008, “Applications of Synchrotron Micro-Computed Tomography (SMT) to assess Fabric Evolution of Sheared Granular Materials.” Unpublished.

Alshibli, K., 2011. Personal Communication.

Bagi, K., 2006. “Analysis of microstructural strain tensors for granular assemblies.” International Journal of Solids and Structures, 43: 3166-3184.

Breul, P., Benz, M., Gourves, R., and Saussine, G., 2009, “Penetration test modeling in a coarse granular medium.” AIP Conference Proceedings, 1145: 173-176.

Cambou, B., Chaze, M., Dedecker, F., 2000. “Change of scale in granular materials.” European Journal of Mechanics – A/Solids, 19: 999-1014.

Cundall, P.A., 1971. “A computer model for simulating progressive, large-scale movements in blocky rock systems,” Rock Fractures, Proceedings of the International Symposium of Rock, Mechanics, Nancy, France.12 P, 6 Fig, 4 Ref, Append.

Cundall P.A., and Strack O.D.L., 1979. A discrete numerical model for granular assemblies. Geotechnique, 29: 47–65.

Dedecker, F., Chaze, M., Dubujet, Ph., Cambou, B., 2000. “Specific features of strain in granular materials.” Mech. Cohesive-Frictional Materials. 5: 173-193.

Duran, O., Kruyt N. P., and Luding, S., 2009. “Analysis of three-dimensional micro-mechanical strain formulations for granular materials: Evaluation of accuracy.” International Journal of Solids and Structures, 47: 251-60.

Fischmeister, H.F. and Arzt, E., 1983. “Densification of powders by particle deformation.” Powder Metallurgy, 26: 82-88.

Fischmeister, H.F., Arzt, E., and Olsson, L.R., 1978, “Particle deformation and sliding during compression of spherical powders: a study by quantitative metallurgy.” Powder Metallurgy, 21: 179-187.

Fleck, N.A., 1995. “On the cold compaction of powders.” Journal of the Mechanics and Physics of Solids, 43(9): 1409 – 31.

Fleck, N.A., Kuhn, L.T. and McMeeking, R.M 1992. “Yielding of metal powder bonded by isolated contacts.” Journal of the Mechanics and Physics of Solids, 40: 1139-1162.

Gao, Z., and Zhao, J., 2012. “Constitutive modeling of artificially cemented sand by considering fabric anisotropy.” Computers and Geotechnics, 41: 57-69.

Hertz, H., 1882 Ueber die Beruhrung fester elastischer Korper. “On the fixed elastic body contact.” Journal fur die reine und angewandte Mathematik (Crelle), 92: 156–171.

Hasan, A. and Alshibli, K.A., 2010, “Experimental Assessment of 3D Particle-to-Particle Interaction within Sheared Sand using Synchrotron Microtomorgraphy.” Geotechnique 60 (5): 369-379.

Hammi, Y., 2011. Personal Communication.

Hammi, Y., and Regueiro, R.A., 2011, “Bridging and Coupling Particle to Continuum Length-Scale Mechanics for Simulating Deformation and Flow of Dense Dry Particulate Materials.” NSF Report. Unpublished.

Horner, D.A., Peters, J.F., and Carrillo, A., 2001, “Large Scale Discrete Element Modeling of Vehicle-Soil Interaction.” Journal of Engineering Mechanics, 127 (10): 1027-1032.

Khoei, A.R. and Iranfar, S., 2003. “3D numerical simulation of elasto-plastic behavior in powder compaction process using a quasi-nonlinear technique.” Journal of Materials Processing Technology 143-144: 886-890.

Knuth, M., Hopkins, M., and Cole, D., 2010, “Discrete Element Modeling of Vicksburg Sand and Lunar Simulant.” Earth and Space 2010: Engineering, Science, Construction, and Operations in Challenging Environments. ASCE, 42-48.

Kruyt, N., Rothenburg, L. 1996. “Micromechanical definition of the strain tensor for granular materials.” Journal of Applied Mechanics, 118: 706-711.

Kruyt, N., 2003. “Statics and kinematics of discrete Cosserat-type granular materials.” International Journal of Structures, 40: 511-534.

Kuhn, H.A. and Downey, C.L., 1971, “Deformation characteristics and plasticity theory of sintered powder materials.” International Journal of Powder Metallurgy, 7(1): 15-25.

Kuhn, L.T. and McMeeking, R.M., 1992, “Power law creep of powder bonded by isolated contacts.” International Journal of Mechanical Sciences, 34: 563-573.

Kuhn, M., 1997. “Deformation measures for granular materials.” In: Proc. Mechanics of Deformations and Flow of Particulate Materials, 29: 91-104.

Lu, Y., 2010. "Three-dimensiional DEM modeling of triaxial compression of sands." Geotechnical Special Publication, 200: 220-226.

Mak, J., Chen, .Y, and Sadek, M.A., 2011, "Determining parameters of a discrete element model for soil-tool interaction." Soil & Tillage Research, 118: 117-122.

Martin, C.L., Bouvard, D., and Shima, S., 2003, "Study of particle rearrangement during powder compaction by the Discrete Element Method." Journal of the Mechanics and Physics of Solids, 51: 667-693.

Marshall, J.S., 2009, "Discrete-element modeling of particulate aerosol flows." Journal of Computational Physics, 1541-1561.

Mindlin, R.D., 1949, "Compliance of elastic bodies in contact. Trans." ASME, Journal of Applied Mechanics, 16(3):259–268.

Misra, A., 1995. "Mechanistic Model for Multi-Asperity Contact Between Surfaces." In G. Pande and S. Pietruszczak, editors, Numerical Models in Geomechanics, 95–100. Rotterdam, The Netherlands.

Ng, T., 2009. "Discrete element method simulations of the critical state of granular material." International Journal of Geomechanics, 9(5): 209-216.

Oda, M., 1982. "Fabric tensor for discontinuous geological materials." Soils and Foundations, 22 (4): 96-108.

Oda, M., and Nakayama, H., 1989. "Yield function for soil with anisotropic fabric." Journal of Engineering Mechanics, 115: 89-104.

Peters, J.F., Hopkins, M.A., Kala, R., Wahl, R.E., 2009, "A poly-ellipsoid particle for non-spherical discrete element method," Engineering Computations: International Journal for Computer-Aided Engineering and Software, 26 (6): 645-657.

Pizette, P., Martin, C.L., Delette, G., Sornay, P., and Sans, F., 2010, "Compaction of aggregated ceramic powders: From contact laws to fracture and yield surfaces." Powder Technol., 198: 240–250.

Plassiard, J., Belheine, N., and Donze, F., 2009. A spherical discrete element model: calibration procedure and incremental response. Granular Matter, 11: 293-306.

Regueiro, R.A., Yan, B., 2011, "Coupling discrete elements and micropolar continuum through an overlapping region," Geotechnical Special Publication, Geo-Frontiers: Advances in Geotechnical Engineering, 211: 4283-4292.

Storåkers, B., Fleck, N.A., and McMeeking, R.M., 1999, "The viscoplastic compaction of composite powders." Journal of the Mechanics and Physics of Solids, 47, 785–815.

Thompson, K.E., Willson, C.S., Zhang, W., 2006, "Quantitative computer reconstruction of particulate materials from microtomography images," Powder Technology, 163: 169-182.

Underwood, P., 1983, "Dynamic relaxation," In Belytschko, T. and Hughes, T.J.R., editors, Computational Methods for Transient Analysis, 245-65. North-Holland, Amsterdam.

US Silica Product Data – ASTM 20/30. Berkeley Springs: US Silica Company. PDF.

Wassgren, C., and Sarkar, A., 2008, PSL DEM Lecture 01: Motivation, "http://pharmahub.org/resources/116."

Wellmann, C., Lillie C., and Wriggers P., 2008, "Homogenization of granular material modeled by a three-dimensional Discrete Element Method." Computers and Geotechnics, 35(3): 394-405.

Wilkinson, D. and Ashby, M. F., 1975, "Pressure sintering by power-law creep." Acta Metallurgica, 23: 1277-1285.

Wu, C.Y., Cocks, A.C.F., Gillia, O.T., and Thompson, D.A., 2003. "Experimental and numerical investigations of powder transfer." Physics Today, 138(2-3): 216-228.

Wu, C.Y. and Cocks, A.C.F., 2006. "Numerical and experimental investigations of the flow of powder into a confined space." Mechanics of Materials, 38(4): 304-324.

Yan, B. 2008. 3D Discrete Element Modeling of Granular Materials and its Coupling with Finite Element Method. Diss. University of Colorado, Boulder. Print.

Yan, B., Regueiro, R.A., and Sture, S., 2010, "Three dimensional discrete element modeling of granular materials and its coupling with finite element facets," Engineering Computations, 27 (4): 519-550.

Yan, Y. and Ji, S., 2010, "Discrete element modeling of direct shear tests for a granular material," International Journal for Numerical and Analytical Methods in Geomechanics, 34 (9): 978-990.

Yimsiri, S., and Soga, K., 2010. "DEM analysis of soil fabric effects on behaviour of sand." Geotechnique, 60(6): 483-495.

Yimsiri, S., and Soga, K., 2011. “Effects of soil fabric on behaviors of granular soils: Microscopic modeling.” Computers and Geotechnics, 38: 861-874.

Zeuch, D.H., Grazier, J.M., Arguello, J.G., and Ewsuk, K.G., 2001. Mechanical properties and shear failure surfaces for two alumina powders in triaxial compression. Journal of Material Science, 36(12): 2911-2924.

Appendix A: Additional Figures

The gradation curves in Figure 32 compare the physical data provided by Dr. Alshibli and his group for the ASTM 20/30 sample (US Silica Product Data) used in the experiment to the gradation curve for the computed particles imported into ELLIP3D. Figure 33 exhibits the gradation curves for the experimental and computed particles for the F-75 specimen, respectively.

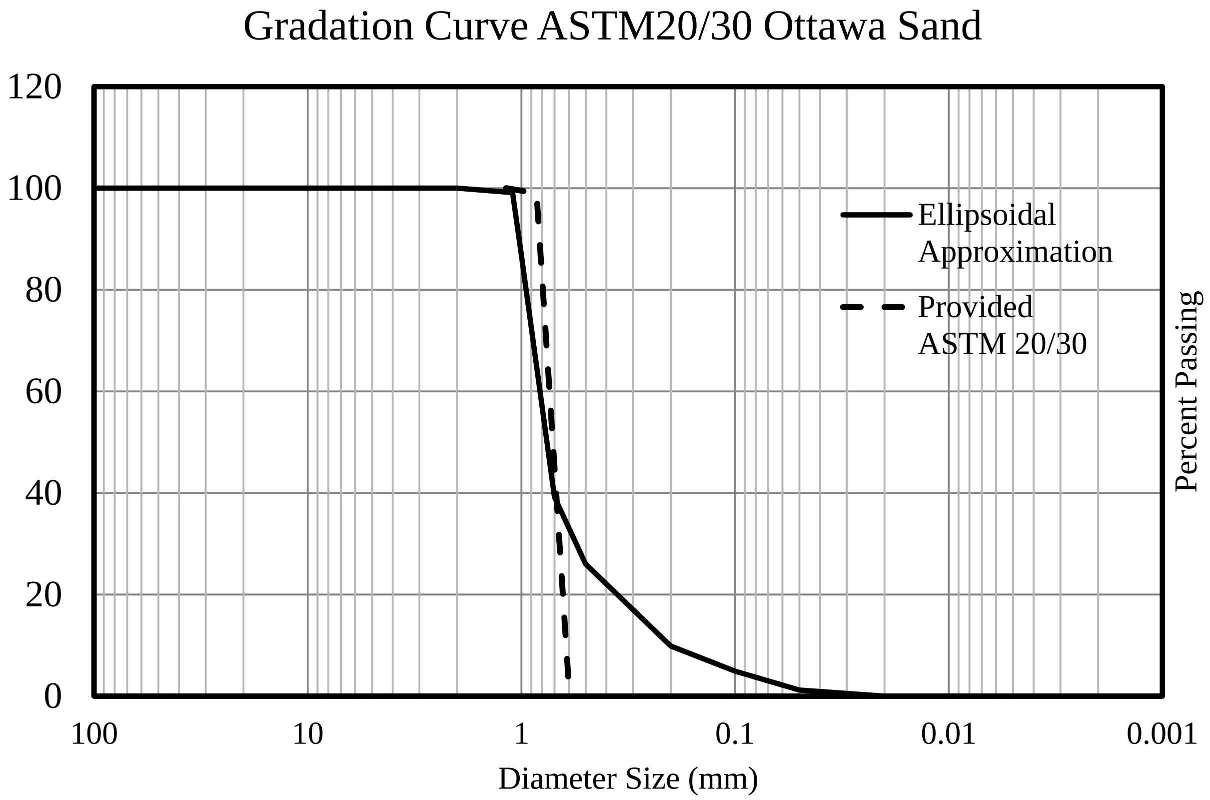


Figure 32: ASTM 20/30 Gradation Curve

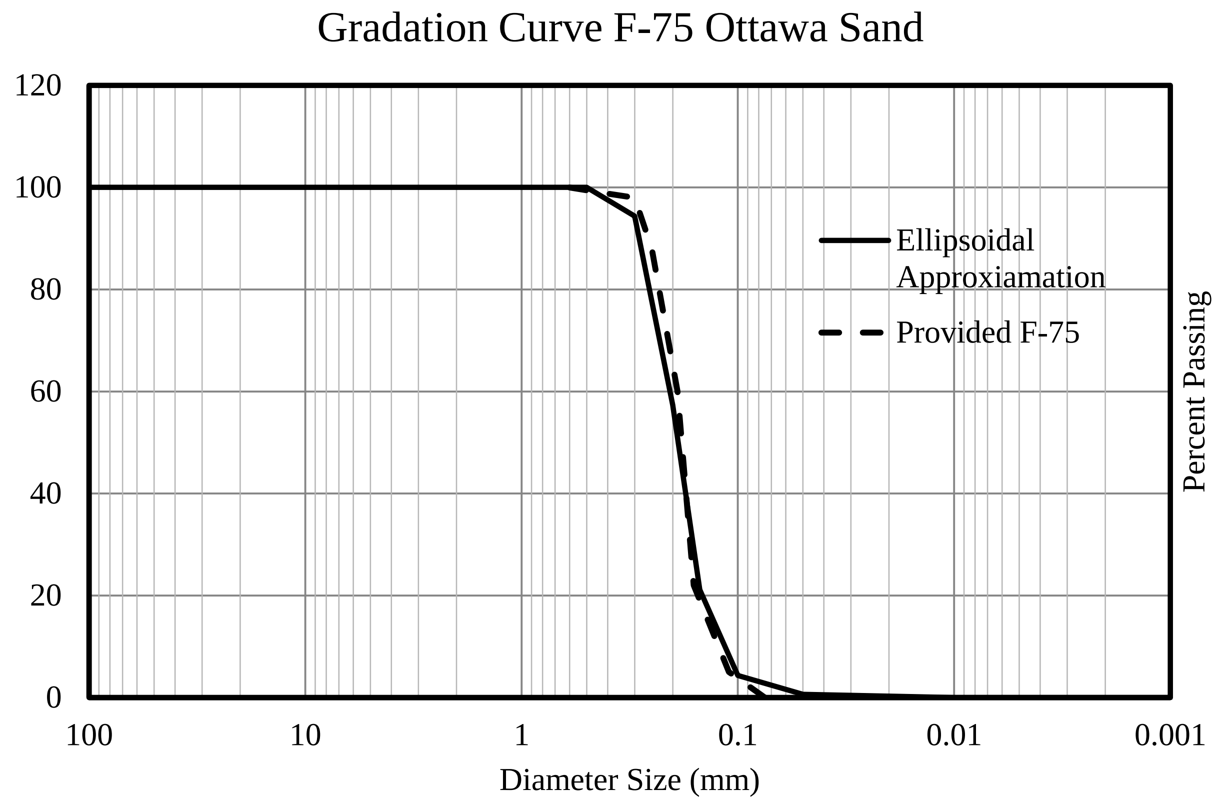


Figure 33: F-75 Gradation Curve

Figure 34 plots the point-wise yield stress vs. plastic strain data that was contributed to the Abaqus particle compaction simulations (Hammi, 2011). These stress-strain parameters utilized in the FE model were calibrated to experimental data on the compaction of nickel powder. The ELLIP3D data was calibrated to the force-displacement constitutive (with assumptions explained in Section 2.2) to the Abaqus model.

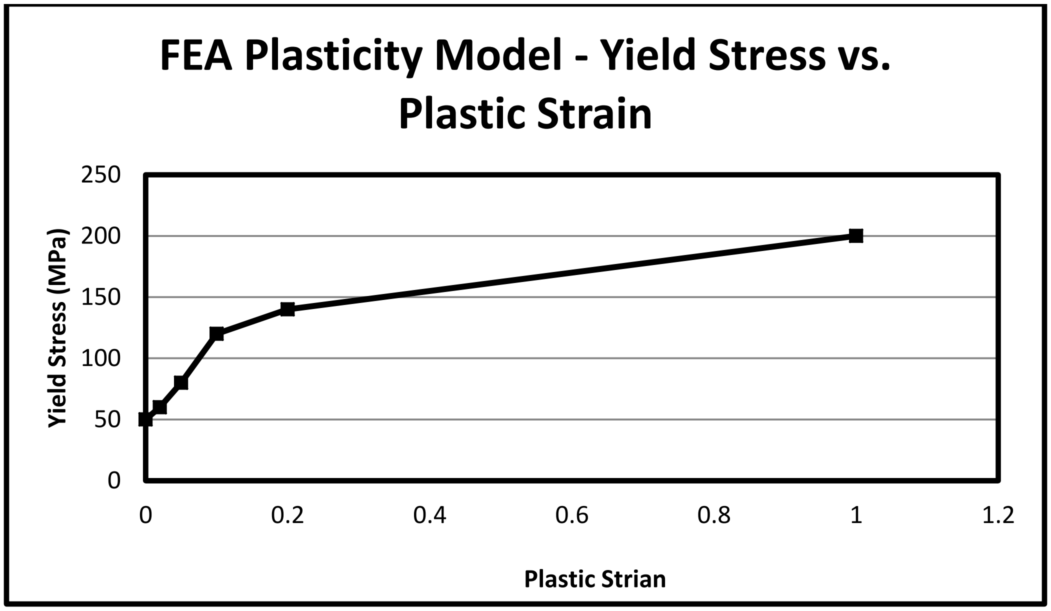


Figure 34: Plasticity Model for FEA from Abaqus

Appendix B: Parametric Studies Examples

The purpose of this appendix is to demonstrate the sensitivity of the input parameters in ELLIP3D. These simulations show the F-75 2369 particle assemblies undergoing isotropic compression, where all six walls compress the sample until the sum of the forces of the particles acting on a given wall, come to equilibrium when achieving a prescribed confining pressure. In Figure 35, the assembly was simulated in isotropic compression with mass and moment of inertia scaling of 1 and the simulation is set to achieve an isotropic confining pressure of 1 kPa. It is apparent that the boundary walls move inwards and outwards (compress and release) but the simulation cannot converge at 1 kPa. The small particles exhibit excessive motion resulting in the noise demonstrated in the plot.

Figure 36 illustrates the same F-75 specimen in isotropic compression but with a higher value of mass and moment of inertia scaling (M&MoIS) and is set to achieve a higher confining pressure, 58 kPa. This also demonstrates that the results are meaningless for such small particles unless the compression rate and time step are small enough to allow the particles to move inwards rather than observing the boundary walls come in and out of contact with the outer particles. At the end of this simulation, ELLIP3D is made to believe that the walls have come to equilibrium, when in fact, the forces on the walls are due to particle-boundary overlap, magnified by M&MoIS, not the macroscopic stress from the particle skeleton interparticle forces.

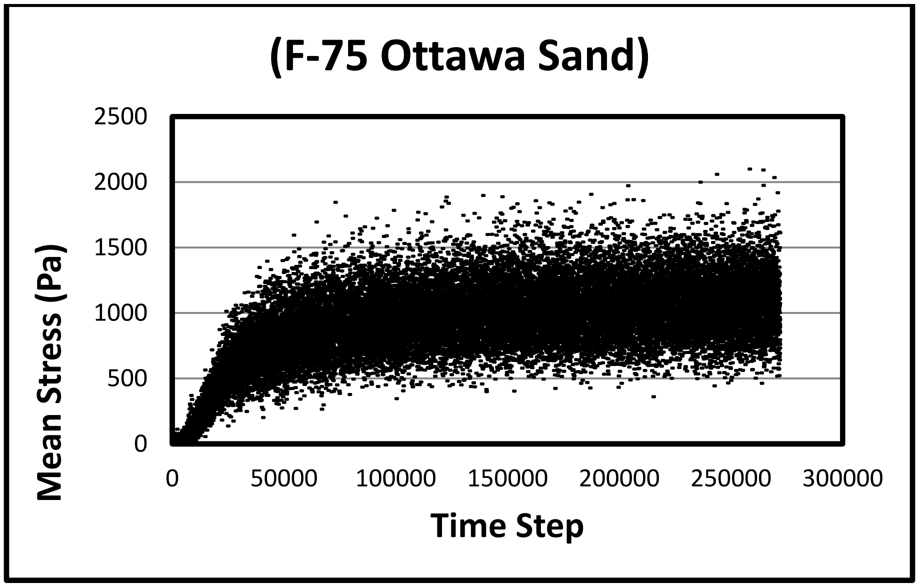


Figure 35: F-75 ELLIP3D Isotropic Compression Simulation with Mass and Moment of Inertia Scaling of 1 to Achieve a Confining Pressure of 1 kPa

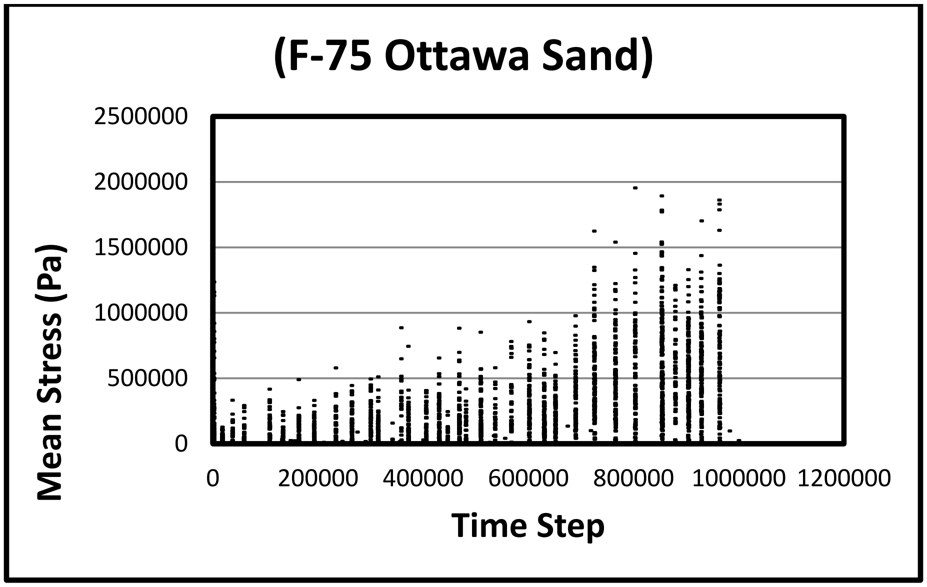


Figure 36: F-75 ELLIP3D Isotropic Compression Simulation with Mass and Moment of Inertia Scaling of 10 to Achieve a Confining Pressure of 58 kPa

	Figure 16	**Figure 17**
Time Step Size	5.0 x 10^{-6} s	5.0 x 10^{-6} s
Mass Scaling	1	10
Moment Scaling	1	10
Gravity Scaling	0	0
Viscous Background Damping on Mass	2/(time step)	2/(time step)
Viscous Background Damping on Mnt of Inert	2/(time step)	2/(time step)
Contact Damping	1	1
Particle-to-Particle Friction	1	1
Particle-to-Boundary Friction	0.5	0.5
Cohesion	0	0
Boundary Wall Compression Rate	1.0 x 10^{-3} m/s	1.0 x 10^{-3} m/s
Boundary Wall Release Rate	1.0 x 10^{-3} m/s	1.0 x 10^{-3} m/s
Confining Pressure to Achieve	1 kPa	58 kPa

Table 5: Parameters for ELLIP3D Isotropic Compression Simulations

Appendix C: ELLIP3D User's Manual

By Yevgeniy Kaufman
Co-authored by Austin Nossokoff

To learn the basics of linux, if you are a true beginner, I recommend the website: *www.ee.surrey.ac.uk/Teaching/Unix*

1.1 Overview:

ELLIP3D: C++ code to simulate Discrete Element Method
Paraview: Open source, data analysis software used for visualization of particles

1.2 Getting Started:

Prior to installation, *development* and *development_benchmark_XML* folders need to be checked out from the *TAHOE cvs* repository (online web storage).

1.2.1 Open your .cshrc file (c-shell text file that allows you to set paths to specific commands you will type in your terminal window) by opening a graphical user interface (GUI) folder or typing the following commands into a terminal:

cd /home/soils/cven/grad/yourname (or the path to where your home directory resides)

to see the contents of the directory, type: *ls*
to see the directory you're currently in, type: *pwd*

to edit your c-shell file, type the following in your terminal window:

```
kedit .cshrc
```

Type the following somewhere into your .cshrc:

```
setenv CVS_RSH ssh
```

The command above is different for Linux machines using bash shells

1.2.2 Checkout the development directories separately (you will need to enter your password, and also accept the ssh connection

cd (to the directory where you want these folders to reside)

mkdir to make directory

```
cvs -d [yourname]@tahoe.colorado.edu:/cvs/private/tahoe checkout development
```

```
cvs -d [yourname]@tahoe.colorado.edu:/cvs/private/tahoe checkout development_benchmark_XML
```

The relevant ELLIP3D and ParaView plugin files are in these folders once checked out.
ELLIP3D:
./development/src/elements/DEM_ellip3d/ellip3d

ParaView plugin:
./development/src/elements/DEM_ellip3d/ParaView/Ellip3dReader

To log onto other computers in the lab from the computer you are logged into, the command is

ssh -X username@computername.colorado.edu, for example, for user Nossokoff log onto quince, it is

```
ssh -X nossokof@quince.colorado.edu
```

If you are logged into soils, just type
ssh quince or the computer desired

To open a GUI in the computer you are logged into while accessing the server externally, the command "konqueror &" will open a window

1.3 Compiling ParaView plugins

Tom Buzbee wrote two plugins that enable the input and output particle files from ELLIP3D to be interpreted by ParaView. Details can be found in Appendix D or in Tom's *Readme* file located in:

./development/src/elements/DEM_ellip3d/ParaView/Ellip3dReader/Readme

Even though ParaView may already be installed on the computer, the only way I was able to load the plugins was to rebuild ParaView, version 3.6.2. Make sure, before generating, to enable shared libraries. Also, make sure that the correct version of QT is in your path, first.

2.0 Running Repository Example Simulations

2.1 Convert Raw Data (Creating particle assembly from .xls file)

Start with directory
./Repository_Example/1_xls_to_elp_conversion/LSU_Data_Initial_Particle_Assembly
This directory contains:
conv.cpp: a code Tom wrote to convert the raw data provided by LSU into a format that is compatible with ellip3d;
ASTM2030_1000x1784_Boxed_RVE.xls: the last sheet in this Excel file contains the 4201-particle assembly data that will be converted to data that can be read by ELLIP3D

2.1.1 Copy and paste the contents (column A-L) from the spreadsheet into a textfile, and call it grain.txt

2.1.2 Compile conv.cpp using the following commands

```
g++ -c conv.cpp
g++ -o convert conv.o
```

This will generate the executable "convert"; then convert the raw data using the following command:

```
./convert grain.txt out.txt
```

The "./" gives the command to run the executable that resides in the current directory that you are in. There is a built-in convert command on the computer, but it is not the correct one for this case

2.1.3 Rename *out.txt* to *particle_input.elp*

Once the ParaView Plugins are loaded, *particle_input.elp* can be opened in paraview

Copy *particle_input.elp* into the directory *2_floating_function*

*Note: The preceding three steps outline the process of creating particle input files for ELLIP3D. The file *filtered.els* contained in the *./LSU_Data_Initial_Particle_Assembly* directory is an alternative particle file that is the same as the *particle_input.elp* file created from the spreadsheet but with all very small particles removed (containing a total of 3563 particles). For this example, I recommend renaming *filtered.els* to *particle_input.elp* and replacing the *particle_input.elp* file in the directory *2_floating_function*. Either will work.

2.2 Creating Wall Boundaries from Raw Data

Open Directory
./Repository_Example/1_xls_to_elp_conversion/LSU_Data_Initial_Boundaries

It contains *BDRY_WALL_input.xls*: this spreadsheet was made to calculate the initial boundary wall particle files based on maximum and minimum coordinate values of particle centroid positions.

Copy highlighted contents from .xls into a text file and name it *boundary_input.elp* (see example of *boundary_input.elp*
in ./Repository_Example/1_xls_to_elp_conversion/XLS_Data if unsuccessful in creating the boundary file)

Copy *boundary_input.elp* into the directories *2_floating_function and 3_isotropic_comp*

2.3 Float Function

Since the particles are approximated as ellipsoids, there is going to be some particle overlap. The Float function allows contact forces to separate particles while a large mass and moment scaling, and large background viscous damping coefficients are applied in order to prevent the particles from blowing up.

The directory *2_floating_function* should contain three files: main.cpp, boundary_input.elp (from 2.2), and particle_input.elp (from 2.1)

Note that the two input files listed in main.cpp: *boundary_input.elp* and *particle_input.elp* can be renamed as long as the match the names of the input files in that directory

To run the simulation:

(1) Copy main.cpp into the ellip3d directory
(2) In a terminal window, go to the /ellip3d directory
(3) Type "*make*" (this compiles ELLIP3D)
(4) Copy "ellip3d" executable into *2_floating_function* directory

After the ellip3d executable is copied, delete all new files created in the ellip3d directory so that it contains its original contents for future simulations. Overwriting when compiling can cause problems if the .d and .o files are still in the ellip3d directory. A quick way to delete the all .d and .o files is to type `rm *.o` and `rm *.d`
However, think twice before typing the rm (remove command) since there is no command to undo it.

Another way is to click on the View tab, go to View Mode, then click Detailed List View and rearrange by date modified. Delete the newest files up until main.cpp, including main.cpp

(5) In a terminal window, open *2_floating_function* directory
(6) Type *ellip3d > & outscreen &*

Go to the directory where you will be running the simulations and type the command

```
./ellip3d > & outscreen &
```

On computers using a bash shell such as soilblast, type nohup first. To run multiple processors on soilblast, type:

```
nohup ./ellip3d 6 > outscreen &
```

Do not run long simulations on soils, run them on scratch (the computer's hard disk space).
To get to the scratch directory,
ssh to a computer such as poplar. (*ssh poplar*)
cd .. (go back a directory) as far as you can go
cd /poplar/scratch
mkdir yourname
and run long simulations here

(7) You can check the progress at any time by typing "tail outscreen"

The simulation might run for several hours (or days), depending on the number of timesteps set.

To quit a simulation, type *top* in a terminal window and find the PID for that job. Then type:

```
kill -9 ####
```

represent the PID and -9 is “without reservations”

After the simulation is complete, type the command *elsadd* to run the executable in that directory in order to add the .els extension to the particle files. Then check the animation in Paraview (Section 2.6). Note: there may possibly not have been overlap to begin with if the *filtered.els* file was used (from 2.1). Copy the final particle assembly file *flo_particle_end* into directory *3_isotropic_comp*
To copy a file from scratch to soils (where your home directory resides) type the following in the terminal window:

```
scp ./file_I_want_to_copy soils:~/directory_I_want_it_in
```

for a directory,

```
scp -r ./directory_I_want_to_copy soils:~/directory_I_want_it_in
```

2.4 Isotropic Compression

The directory 3_*isotropic_comp* should contain three files: main.cpp, boundary_input.elp (from 2.2), and *flo_particle_end* (from 2.3)

Follow the same simulation instructions (steps 1-6) as in 2.3

The simulation could take a few days (depending on the number of timesteps used)

After the simulation is complete, copy the final particle assembly *iso_particle_end* and final boundary *iso_boundary_end* into directory 4_*triaxial_comp.* Check the animation in Paraview (Section 2.6) after running command ./elsadd, and check & plot the constitutive results from the iso_progress file.

2.5 Triaxial Compression

The directory 4_*triaxial_comp* should contain these three files in order to run the triaxial compression test simulations: main.cpp, *iso_particle* (from 2.4), and *iso_boundary* (from 2.4)

Follow the same simulation instructions (steps 1-6) as in 2.3

The simulation could take a few days depending on the number of timesteps selected. The tri_progress file contains stress-strain data. To plot ELLIP3D data versus the experimental data (ASTM.txt) run stress_v_strain_tri_astm.m using Matlab (First remove non-numerical values from the tri_progress file, i.e. first three lines).

2.6 Visualization in ParaView

In the .cshrc file, set your environment variables of ParaView_DIR, CMAKE_DIR, QT,

and LD_LIBRARY_PATH
You can do this by entering the following:

```
setenv ParaView_DIR /home/soils/students/nossokof/Trial2_ParaView/ParaView3/bin/bin
```

(This is where I built Paraview and where the executable is located)

```
setenv CMAKE_DIR /usr/local/cmake
```

(This should be the same for all users)

```
setenv QT /usr/local/qt-4.6.3
```

(This should be the same for all users, although the version of QT may change)

```
setenv LD_LIBRARY_PATH /usr/lib:/usr/local/lib:$QT/lib
```

(This should be the same for all users)

```
set path=( $ParaView_DIR $CMAKE_DIR/bin $QT/bin $HOME/bin . $p)
```

(This should be the same, although there may need to be a /bin after $ParaView_DIR)

Copy the plugins you built for ParaView, from section 1.3, to a directory labeled plugins within the same directory where the paraview executable is located. ParaView searches here for local plugins so you don't have to manage local plugins every time.

2.7 Plotting Data

For triaxial compression, tri_progress contains the constitutive values of interest. Type:

```
kwrite tri_progress
```

when using kwrite, F10 allows you to see the values by turning word wrap off

The data can be viewed and modified in MATLAB or could be saved as "tri_progress.txt" and it can be imported into excel (on a computer running Windows) and plotted there

Alternatively, a Matlab file called stress_v_strain_tri_astm.m located in ./Repository_Example/4_triaxial_comp is set up to interpret the progress files but may need to be adjusted depending on what is to be plotted. Note: the first three lines of the progress file (non-numerical values) need to be deleted for this Matlab script to run

Running DEM Simulations (Particle Deposit / Isotropic Compression / Triaxial Compression):

Ellip3d directory contains the necessary files to create the ellip3d executable (minus the input files noted in main.cpp). Ellip3d executable will run the simulation (Deposit / Isotropic Compression / Triaxial Compression) based on the input parameters in main.cpp and input files noted in main.cpp.

Tom Buzbee's Simple Repository Example:

After checking out development and development_benchmark and getting ellip3d:
./development/src/elements/DEM_ellip3d/ellip3d

See Tom Buzbee's example of Isotropic Compression from 0 kPa to 1 kPa:
development_benchmark_XML/DEM_tom/triaxial/iso-0-1kpa

Additional Notes on Post-Processing:

After the simulations have been run,
(1) Open iso_progress or tri_progress and remove all non-numerical text (first 3 lines)

(2) Copy stress_v_strain.m file (if available) to the directory where the results reside to produce stress strain plot in Matlab

(3) Run stress_v_strain.m in Matlab to produce plot

(4) Open Paraview

(i) Load Plugins (Tools, Manage Plugins):

(ii) Open .els extension files from
*If results do not produce .els files, check assembly.cpp to make sure ".elp" is replaced with ".els" and rerun simulations
Or use the elsadd executable (much easier)

Main.cpp Parameters:

- Mass and moment of inertial scaling give the particles a "heaviness" – this reduces the acceleration of the particles but it also increases the resulting forces on the boundaries
- Viscous damping on mass and moment of inertia acts as a "viscous-like fluid" to resist particle motion
- Control the axial strain by a simple hand calculation knowing the displacement

rate (m/s), timestep size (s), and number of timesteps

Appendix D: Building ParaView Plugins

In order to open ELLIP3D particle assembly files in ParaView, ELLIP3D Reader plugins must be built along with ParaView. The two plugins were written by graduated BS/MS student Tom Buzbee and instructions for building the ELLIP3D Reader plugins can be followed here or in the Readme provided by Tom (access to this readme and necessary files can be found in Section 1.3 in Appendix C.

Paraview is open-source data analysis and visualization software that enables ELLIP3D users to view particle assembly snapshots and play full simulation animations. This is a plugin for ParaView, (www.paraview.org), that will render output files from ELLIP3D.

The plugin is designed to recognize .elp (individual ELLIP3D particle files) and .els (series of ELLIP3D particle files), which enables the user to open them in Paraview. Tom mentioned that he modified ELLIP3D to use the .els extension instead of .elp. This requires a "find-and-replace" on all .elp with .els in assembly.cpp in ELLIP3D. Alternatively, elsadd, an executable written by Dr. Beichuan Yan can be used to add a .els extension to all of the particle files so that they can be opened as a series in Paraview. The latter is much easier, it just requires the command *./elsadd* in the directory that contains the particle files.

.

Paraview cannot be just installed, it must be built manually along with the plugins. The instructions for doing so are as follows:

To build Paraview, follow the link http://paraview.org to obtain the source and follow the instructions on the wiki page: http://paraview.org/Wiki/ParaView:Build_And_Install or the included text files. To build Paraview:

```
cd path/to/paraview_source
mkdir bin
cd bin
ccmake ..
```

```
-> c to configure
-> c to configure (usually need to do it more than once)
-> g to generate makefiles
make
```

If it complains about missing libraries, you probably need to install them separately, i.e., if the commands cmake and ccmake are not found on the computer, email trouble@soils.colorado.edu and they will make them available to the user on all machines in the lab

Building the plugin follows a similar process. First, though, you need to set the ParaView_DIR environment variable to point to the ParaView build directory. The sequence of commands should look like this:

```
export ParaView_DIR="/path/to/paraview_source/bin"
```

(Detailed instructions on setting your path for your .cshrc file can be found in Section 2.6 of Appendix C)

```
cd path/to/Ellip3dReader/Server
mkdir bin
cd bin
ccmake ..
-> c to configure
-> c to configure (usually need to do it more than once)
-> g to generate makefiles
make
```

This generates a .so binary library (or .dylib, .dll, etc. depending on your platform) that you can load into ParaView.

Now you need the client resources. You should be able to use the .bqrc file in the Client directory as-is. You can build it yourself, too. This requires a working Qt installation. If you have one, you can just issue the 'make' command in the Client directory, or email trouble@soils.colorado.edu to install them. This should produce a .bqrc file you can load into ParaView.

Usage:
In Paraview, load the .bqrc and .so files by going to 'Tools > Manage Plugins' menu or automatically when it loads using the PV_PLUGIN_PATH by modifying the environment variable in the .cshrc file. Detailed instructions can be found on the wiki page stated earlier.

To open the .elp or .els particle assembly files, go to 'File > Open' dialog. The .els files should be numbered in sequential order so that the series will show up as a collapsed item in the dialog. Pick the whole group and it will load as an animation. Paraview has many visualization and animation capabilities that can be found on their page or through instructional youtube videos.