Technical Report PAR-10021867-092020-6

Material point method simulations of penetration of sand modeled with tabular, support-vector, and neural network models of elastoplasticity

Biswajit Banerjee
Parresia Research Limited, Auckland, New Zealand
b.banerjee.nz@gmail.com

Rebecca M. Brannon
University of Utah, Salt Lake City, UT, USA
Rebecca.Brannon@utah.edu

David M. Fox
CCDC Army Research Laboratory, Aberdeen Proving Ground, MD, USA
david.m.fox1.civ@mail.mil

Richard A. Regueiro
University of Colorado, Boulder, CO, USA
richard.regueiro@colorado.edu

Tuesday 12th January, 2021

Abstract

Penetration of high-speed projectiles into soils and the effect of explosions in soils on objects above the surface have interested civil and military engineers for decades. With the advent of faster computers and better numerical algorithms, many intractable problems in these domains have become possible to solve. In this paper, we use the Material Point Method (MPM), and a tabular elastoplastic model for soils, to simulate penetration in dry sand. The projection and interpolations operations in MPM, and the corresponding functions and their choice are described. The model is used to simulate the penetration of a punch at 1g and an unexploded ordnance (UXO) projectile into a dry, poorly-graded, concrete sand at 20g in a centrifuge. An interpolated tabular model, a support vector regression (SVR) model, and a multilayer perceptron (MLP) neural network model are used to compute the elastic moduli from identical input data. We find that even if all the components of the elastoplastic model other than the elastic modulus model are kept fixed, small differences in the moduli can lead to different depths of penetration. We also find that elastoplasticity with an SVR elastic modulus model is almost 10 times as computationally expensive as direct tabular interpolation.

1 Introduction

The prediction of the depth of penetration of projectiles into soils has been attempted since at least 1742 (Thompson, 1966). Recent interest in modern methods for the detection and removal of unexploded ordnance in soils has highlighted our incomplete understanding of the processes involved and the need for improved predictive tools (Omidvar, Iskander, and Stephan Bless, 2014; Chung et al., 2017).

Early numerical simulations to tease out important constitutive contributors to the depth of penetration (H. Schreyer and Chiu, 1991) found that the contributions of several variables tended to cancel each other. Schreyer suggested caution while developing theories based on dimensional analysis, such that that developed by (Boguslavskii, Drabkin, and Salman, 1996). Modern testing methods have been able to measure the internal flow fields that develop during impact and penetration events in soils and simulations with peridynamics have produced reasonable matches with experiment (Collins et al., 2011). High-speed experiments by Tanaka et al. (2011) shows that a conical crushed sand region developed below the punch-like projectile. This region can also be observed in photographs from high-speed tests by Borg et al. (2013). Large deviations from the initial direction of impact were observed in experiments and simulations of bullets impacting sand by Børvik, Dey, and Olovsson (2015). Data on larger projectiles at moderate impact velocities were obtained by Omidvar, Malioche, et al. (2015). More recently, the effect of sand density on penetration has been studied experimentally (Chian, Tan, and Sarma, 2017). Rate effects have also been explored experimentally, e.g., Bless et al. (2020), but typically have been ignored in the prediction process.

A large number of experiments at several impact rates were performed by Thompson (1966). Projectile shapes tested included punch-shaped, needle-shaped, and ogive-nosed objects. Though much of these data cannot be used to quantitatively validate simulations because of the differences in material properties of the soil, they do provide a means for qualitative comparison of the punch impact tests discussed in this work.

Most experiments in the literature have involved full-scale tests or scaled-down laboratory tests. Scaling relations have been proposed, often based on ideas developed for impact cratering (K. A. Holsapple and Schmidt, 1982; K. Holsapple et al., 2002), but not always validated under general conditions. Centrifuge experiments (Garnier et al., 2007) were first performed for the impact and penetration problem by Fragaszy and T. A. Taylor (1989) and T. Taylor, Fragaszy, and Ho (1991). However, centrifuge models were not explored further until rockfall studies were performed by Chikatamarla, Laue, and Springman (2006) and dynamically installed anchors were explored by O'Loughlin, Richardson, and Randolph (2009). Centrifuge models for impact and penetration merit further experimental and numerical exploration.

Physics-based continuum models of the penetration process have been proposed in conjunction with many experimental studies, recent examples include Chung et al. (2017) and Dunatunga and Kamrin (2017). Numerical methods ranging from finite elements, meshfree methods (Sheng-Wei, 2017; Mahdavi, Chi, and Atif, 2020), the material point method (Dunatunga and Kamrin, 2017), smoothed particle hydrodynamics, peridynamics (Collins et al., 2011), and discrete element methods (N. Zhang and Evans, 2017) have been explored.

The impact and penetration simulations described in this work have used the material point method to solve the momentum equations in conjunction with the tabular elastoplastic material model described in (Banerjee, Fox, and Regueiro, 2020d). The material model was exercised using tests driven by prescribed deformation gradient histories on single particles as discussed in (Banerjee, 2020). Tabular, support vector, and neural network models for the elastic moduli of the dry, poorly-graded, concrete sand described in Section 3 are compared. These models have been discussed in detail in Banerjee, Fox, and Regueiro (2020d), Banerjee, Fox, and Regueiro (2020c), and Banerjee, Fox, and Regueiro (2020b). The Vaango (Banerjee, 2014) implementation of the Material Point Method (MPM) algorithm and the material model has been used for the simulations. Two test geometries are considered: the impact of a rectangular punch on sand at 1g, and the impact of an unexploded ordnance (UXO) shell into sand contained in a centrifuge at 20g. Validation simulations have not been explored in this work, but can easily be performed using experimental data from the extensive literature on the subject.

We find that even though the choice of elastic material model does not affect the initial penetration dynamics significantly, the final penetration depth can be different even if the material models predict small differences in moduli. We also find that the support vector regression approach is the most computationally expensive of the approaches we have explored in this paper.

This paper is organized as follows. Section 2 discusses the MPM approach and the ideas behind the choice of projection and interpolation functions. An accurate approach for computing the deformation gradient in an explicit time integration scheme is also discussed. The tabular experimental material data and the corresponding interpolated, support vector, and multilayer perceptron models for the elastic moduli are discussed in Section 3. Punch penetration simulations are presented in Section 4 and unexploded ordnance (UXO) simulations are discussed in Section 5. Finally, some concluding remarks are given in Section 6.

Notation

The notation used in this work are as follows. Scalars are represented by normal fonts, e.g., ρ and m. Bold normal symbols, e.g., $\mathbf{b}$ and $\mathbf{G}$, indicate first-order tensors. Second-order tensors are represented by bold italic symbols, e.g., $\boldsymbol{f}$, $\boldsymbol{F}$, and $\boldsymbol{\sigma}$ and their transposes are represented by the superscript T, e.g., $\boldsymbol{F}^T$. Components of tensors are assumes to be given in rectangular Cartesian coordinates and the Einstein summation convention is used unless specified otherwise. The inner product of two first-order tensors is defined as $c = \mathbf{a} \cdot \mathbf{b} = a_i b_i$ while their dyadic (outer) product is defined as $\boldsymbol{c} = \mathbf{a} \otimes \mathbf{b} = a_i b_j$. The inner product of two second-order tensors is given by $c = \boldsymbol{a} : \boldsymbol{b} = a_{ij} b_{ij}$ and their partial inner product is represented as $\boldsymbol{c} = \boldsymbol{a} \cdot \boldsymbol{b} = a_{ij} b_{jk}$. The product of a first-order and a second-order tensor is defined as $\mathbf{c} = \boldsymbol{a} \cdot \mathbf{b} = a_{ij} b_j$ or $\mathbf{d} = \mathbf{a} \cdot \boldsymbol{b} = a_i b_{ij}$. The gradient of a first-order tensor field, $\mathbf{a}(\mathbf{x})$, is defined as $\boldsymbol{G} = \nabla \mathbf{a} = \partial a_i / \partial x_j = G_{ij}$. The divergence of a second-order tensor field, $\boldsymbol{a}(\mathbf{x})$, is defined as $\mathbf{a} = \nabla \cdot \boldsymbol{a} = \partial a_{ij} / \partial x_i$.

2 The Material Point Method

The following discussion of aspects of the material point method of relevance to impact simulations is largely based on an exposition by Brannon.[1] For a detailed explanation of various aspects of the theory and algorithm, please consult the Vaango theory manual (Banerjee, 2014).

2.1 Weak form

The spatial form of the governing equations for the balance of linear momentum is

$$\nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b} = \rho \mathbf{a} \quad \forall \quad \mathbf{x} \in \Omega \tag{1}$$

[1] R.M. Brannon, 2015, A primer on the material point method. Personal Communication.

where $\boldsymbol{\sigma}(\mathbf{x})$ is the Cauchy stress, ρ is the mass density, $\mathbf{b}$ is the body force per unit mass, and $\mathbf{a}$ is the acceleration. The current configuration of the physical body is represented by Ω. The Material Point Method (MPM), similar to standard finite element methods, is based on a weak form of (1) (Sulsky, A. Chen, and H. L. Schreyer, 1994):

$$\int_{\Omega} \mathbf{w} \cdot (\nabla \cdot \boldsymbol{\sigma})\, \mathrm{dV} + \int_{\Omega} \rho \mathbf{w} \cdot \mathbf{b}\, \mathrm{dV} = \int_{\Omega} \rho \mathbf{w} \cdot \mathbf{a}\, \mathrm{dV} \tag{2}$$

where $\mathbf{w}$ is an arbitrary, vector-valued, test function. After integration by parts, we get the spatial weak formulation cast in terms of spatial integrals:

$$\int_{\partial\Omega} \mathbf{w} \cdot \mathbf{t}\, \mathrm{dA} - \int_{\Omega} \boldsymbol{\sigma} : (\nabla \mathbf{w})\, \mathrm{dV} + \int_{\Omega} \rho \mathbf{w} \cdot \mathbf{b}\, \mathrm{dV} - \int_{\Omega} \rho \mathbf{w} \cdot \mathbf{a}\, \mathrm{dV} = 0 \tag{3}$$

where $\mathbf{t} := \boldsymbol{\sigma} \cdot \mathbf{n}$ is the surface traction and $\mathbf{n}$ is the outward unit normal to the surface $\partial\Omega$.

Because it was originally developed to model large deformations of history-dependent materials, MPM typically uses an updated Lagrangian formulation. Equation (3) can be converted directly into an updated Lagrangian form that is cast in terms of integrals and gradients with respect to the configurations at the start of each time step. [2]

$$\int_{\partial\Omega_n} B\, \mathbf{w} \cdot \mathbf{t}\, \mathrm{dA}_n - \int_{\Omega_n} J_n(\boldsymbol{\sigma} \cdot \boldsymbol{f}^{-T}) : (\nabla_n \mathbf{w})\, \mathrm{dV}_n + \int_{\Omega_n} \rho_n \mathbf{w} \cdot \mathbf{b}\, \mathrm{dV}_n - \int_{\Omega} \rho_n \mathbf{w} \cdot \mathbf{a}\, \mathrm{dV}_n = 0 \tag{4}$$

where the subscript n indicates the configuration at the beginning of the time step t_n, ∇_n is the gradient with respect to the updated Lagrangian position vector $\mathbf{x}_n$, $\boldsymbol{f}$ is the incremental deformation gradient, defined as

$$\boldsymbol{F} = \boldsymbol{f} \cdot \boldsymbol{F}_n \quad ; \quad \boldsymbol{f} := \frac{\partial \mathbf{x}}{\partial \mathbf{x}_n} \tag{5}$$

where $\boldsymbol{F}$ is the total deformation gradient, and $J_n := \det(\boldsymbol{f})$. The quantity B is the incremental areal stretch given by

$$B := \left\| (J_n \boldsymbol{f}^{-T}) \cdot \mathbf{n}_n \right\| \tag{6}$$

where $\mathbf{n}_n$ is the outward unit normal to the surface at time t_n.

2.2 Spatial discretization

Equation (4) is discretized in MPM in a manner similar to finite elements. The test function $\mathbf{w}$ and the displacement field $\mathbf{u}$ (and consequently the velocity $\mathbf{v}$ and the acceleration $\mathbf{a}$) are expressed as linear combinations of basis functions $S_i(\mathbf{x}_n)$ such that

$$\begin{aligned} \mathbf{w}(\mathbf{x}_n, t) &= \sum_i \mathbf{w}_i(t) S_i(\mathbf{x}_n)\,, \quad \mathbf{u}(\mathbf{x}_n, t) = \sum_j \mathbf{u}_j(t) S_j(\mathbf{x}_n) \implies \\ \mathbf{v}(\mathbf{x}_n, t) &= \sum_j \mathbf{v}_j(t) S_j(\mathbf{x}_n)\,, \quad \mathbf{a}(\mathbf{x}_n, t) = \sum_j \mathbf{a}_j(t) S_j(\mathbf{x}_n)\,. \end{aligned} \tag{7}$$

However, in contrast to finite elements, the basis functions are defined with respect to a background grid that tessellates the computational domain (including the body) rather than a grid that tessellates

[2] R.M. Brannon, 2015, A primer on the material point method. Personal Communication.

only the body. The basis functions are chosen such that they satisfy a partition of unity and linear completeness. Note that the Kronecker interpolation property is not required, i.e., $S_i(\mathbf{x}_{n,j}) \neq \delta_{ij}$ ($\mathbf{x}_{n,j}$ indicates the j-th grid node). [3]

If we use the same basis functions for the test function and the trial displacement field, we get the discretized grid node equations

$$\mathbf{f}_i^{\text{ext}} + \mathbf{f}_i^{\text{int}} + \mathbf{f}_i^{\text{body}} = m_i \mathbf{a}_i \tag{8}$$

where

$$\begin{aligned} \mathbf{f}_i^{\text{ext}} &:= \int_{\partial\Omega_n} B\, S_i\, \mathbf{t}\, \text{dA}_n \;,\;\; \mathbf{f}_i^{\text{int}} := -\int_{\Omega_n} J_n (\boldsymbol{\sigma} \cdot \boldsymbol{f}^{-T}) \cdot \mathbf{G}_{n,i}\, \text{dV}_n \;, \\ \mathbf{f}_i^{\text{body}} &:= \int_{\Omega_n} \rho_n\, S_i\, \mathbf{b}\, \text{dV}_n \;,\;\; m_i := \int_{\Omega_n} \rho_n\, S_i\, \text{dV}_n \,. \end{aligned} \tag{9}$$

In the above, the basis function gradient is defined as

$$\mathbf{G}_{n,i} := \frac{dS(\mathbf{x}_n)}{d\mathbf{x}_n} = \nabla_n S_i \,. \tag{10}$$

Note that the quantities B, $\boldsymbol{f}$, $\mathbf{t}$, $\mathbf{b}$, and $\boldsymbol{\sigma}$ may vary over both space and time over a time step and a midpoint time integrator must evaluate these quantities at the middle of the time step.

Evaluation of the grid node integrals in (9) requires information carried by the Lagrangian MPM particles. For the external force term, we have [4]

$$\mathbf{f}_i^{\text{ext}} = \int_{\partial\Omega_n} B\, S_i\, \mathbf{t}\, \text{dA}_n = \int_{\partial\Omega} S_i\, \mathbf{t}\, \text{dA} = \sum_{p=1}^{n_p} \int_{\partial\Omega_p \cap \partial\Omega} S_i\, \mathbf{t}\, \text{dA} =: \sum_{p=1}^{n_p} \overline{A^p} \langle \mathbf{f}_{ip}^{\text{ext}} \rangle \tag{11}$$

where $\overline{A^p} := A_{\partial\Omega_p \cap \partial\Omega}$ is the area of intersection between the particle surface (area A^p) and the domain boundary, and

$$\langle \mathbf{f}_{ip}^{\text{ext}} \rangle := \frac{1}{A^p} \int_{\partial\Omega_p \cap \partial\Omega} S_i\, \mathbf{t}\, \text{dA} \,. \tag{12}$$

For the internal force term,

$$\mathbf{f}_i^{\text{int}} = -\int_{\Omega_n} J_n (\boldsymbol{\sigma} \cdot \boldsymbol{f}^{-T}) \cdot \mathbf{G}_{n,i}\, \text{dV}_n = -\int_{\Omega} \boldsymbol{\sigma} \cdot \mathbf{G}_i\, \text{dV} = -\sum_{p=1}^{n_p} \int_{\Omega_p \cap \Omega} \boldsymbol{\sigma} \cdot \mathbf{G}_i\, \text{dV} =: \sum_{p=1}^{n_p} \overline{V^p} \langle \mathbf{f}_{ip}^{\text{int}} \rangle \tag{13}$$

where $\overline{V^p} := V_{\Omega_p \cap \Omega}$ is the volume of intersection of the particle and the domain, and

$$\langle \mathbf{f}_{ip}^{\text{int}} \rangle := -\frac{1}{V^p} \int_{\Omega_p \cap \Omega} \boldsymbol{\sigma} \cdot \mathbf{G}_i\, \text{dV} \,. \tag{14}$$

The body force term is discretized as

$$\mathbf{f}_i^{\text{body}} = \int_{\Omega_n} \rho_n\, S_i\, \mathbf{b}\, \text{dV}_n = \int_{\Omega} \rho\, S_i\, \mathbf{b}\, \text{dV} = \sum_{p=1}^{n_p} \int_{\Omega_p \cap \Omega} \rho\, S_i\, \mathbf{b}\, \text{dV} =: \sum_{p=1}^{n_p} \overline{V^p} \langle \mathbf{f}_{ip}^{\text{body}} \rangle \tag{15}$$

[3] R.M. Brannon, 2015, A primer on the material point method. Personal Communication.

[4] R.M. Brannon, 2015, A primer on the material point method. Personal Communication.

where

$$\langle \mathbf{f}_{ip}^{\text{body}} \rangle := \frac{1}{V^p} \int_{\Omega_p \cap \Omega} \rho \, S_i \, \mathbf{b} \, \mathrm{dV} \,. \tag{16}$$

Finally, for the lumped nodal mass, we have

$$m_i = \int_{\Omega_n} \rho_n \, S_i \, \mathrm{dV}_n = \int_{\Omega} \rho \, S_i \, \mathrm{dV} = \sum_{p=1}^{n_p} \int_{\Omega_p \cap \Omega} \rho \, S_i \, \mathrm{dV} =: \sum_{p=1}^{n_p} \overline{V^p} \langle m_{ip} \rangle \tag{17}$$

where

$$\langle m_{ip} \rangle := \frac{1}{V^p} \int_{\Omega_p \cap \Omega} \rho \, S_i \, \mathrm{dV} \,. \tag{18}$$

2.3 Moving information from grid to particles and back

Various versions of MPM have been proposed in the literature (Sulsky, Zhou, and H. L. Schreyer, 1995; S. G. Bardenhagen and Kober, 2004; Sadeghirad, Rebecca M Brannon, and Burghardt, 2011), some of which have been incorporated into the Vaango software (Banerjee, 2014). [5] These are distinguished by the choice made in evaluating the averages in (12), (14), (16), and (18). In general, these averages can be expressed as

$$\langle F_{ip} \rangle := \frac{1}{V^p} \int_{\Omega_p \cap \Omega} F_i(\mathbf{x}) \, \mathrm{dV} \,. \tag{19}$$

Typical MPM approximations of these averages take the form

$$\langle F_{ip} \rangle \approx \frac{1}{V^{p,\star}} \int_{\Omega_p^\star} F_i^\star(\mathbf{x}) \, \chi^{p,\star}(\mathbf{x}) \, \mathrm{dV} \quad \text{where} \quad V^{p,\star} := \int_{\Omega_p^\star} \chi^{p,\star}(\mathbf{x}) \, \mathrm{dV} \tag{20}$$

and χ is a particle indicator function. Note that while Ω_p tessellate the body, the domains $\Omega_p^\star$ do not necessarily do so. In typical MPM implementations, the approximate function $F_i^\star$ is almost always used instead of the exact F_i.

We define $\langle S_{ip} \rangle$ as the average of the i-th grid basis function over the p-th particle:

$$\langle S_{ip} \rangle := \frac{1}{V^p} \int_{\Omega_p \cap \Omega} S_i(\mathbf{x}) \, \mathrm{dV} \approx \frac{1}{V^{p,\star}} \int_{\Omega_p^\star} S_i^\star(\mathbf{x}) \, \chi^{p,\star}(\mathbf{x}) \, \mathrm{dV} \tag{21}$$

where V^p is the volume of the MPM particle Ω_p. For traditional MPM (Sulsky, Zhou, and H. L. Schreyer, 1995), the functions, S_i, have the form shown in Figure 1 while the characteristic functions, $\chi^{p,\star}$ are Dirac delta functions.

For the GIMP approach (S. G. Bardenhagen and Kober, 2004) used in this work, we have

$$\begin{aligned}
\langle \mathbf{f}_{ip}^{\text{ext}} \rangle &\approx A_n^p \, \langle S_{ip} \rangle_A \, \mathbf{t}^p \quad \text{where} \quad \langle S_{ip} \rangle_A := \frac{1}{A_n^p} \int_{\partial\Omega_p \cap \partial\Omega} S_i \, \mathrm{dA}_n \\
\langle \mathbf{f}_{ip}^{\text{int}} \rangle &\approx -V_n^p \boldsymbol{\sigma}^p \cdot \langle \mathbf{G}_{ip} \rangle \quad \text{where} \quad \langle \mathbf{G}_{ip} \rangle := \frac{1}{V_n^p} \int_{\Omega_p \cap \Omega} \mathbf{G}_{n,i} \, \mathrm{dV}_n \\
\langle \mathbf{f}_{ip}^{\text{body}} \rangle &\approx m^p \langle S_{ip} \rangle \mathbf{b}^p \quad \text{where} \quad \langle S_{ip} \rangle := \frac{1}{V_n^p} \int_{\Omega_p \cap \Omega} S_i \, \mathrm{dV}_n \\
\langle m_{ip} \rangle &\approx m^p \langle S_{ip} \rangle
\end{aligned} \tag{22}$$

[5] A detailed exposition of the MPM theory and algorithms can be found in the Vaango theory manual (Banerjee, 2014).

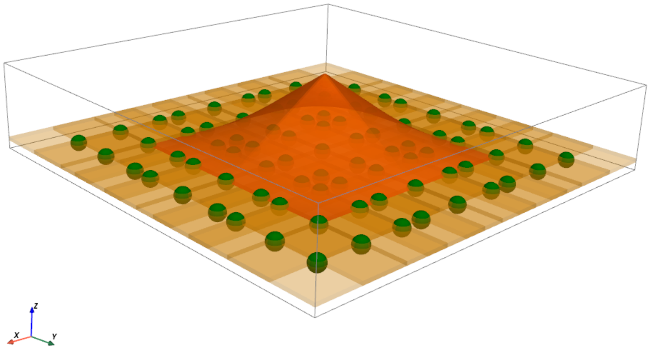
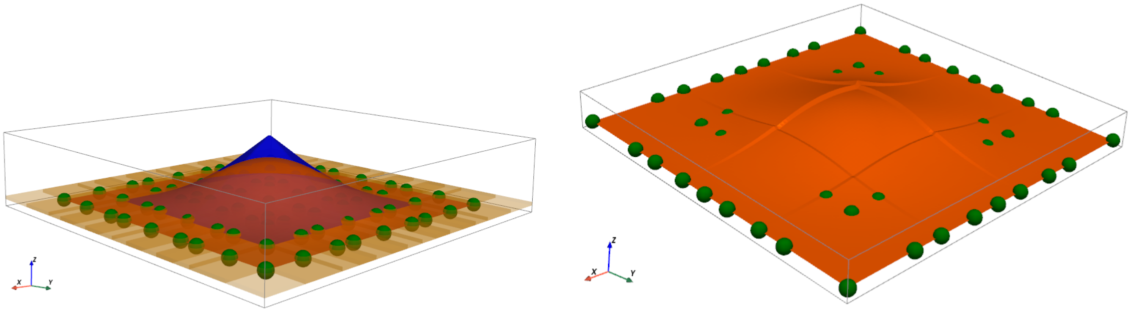


Figure 1 – *Linear grid node shape functions, $S_i(\mathbf{x})$, for traditional MPM in two dimensions. Balls indicate grid node locations.*

where A_n^p, V_n^p are the surface area and volume of particle p at the beginning of the time step, $\mathbf{t}^p$ is the traction applied on the particle, $\boldsymbol{\sigma}^p$ is the Cauchy stress in the particle, $\mathbf{b}^p$ is the particle body force, and m^p is the mass of the particle. Note that particle field quantities are actually averages over the particle volume. The GIMP shape functions are compared to traditional shape function in Figure 2(a). The characteristic functions, $\chi^{p,\star}$, are square and the effective shape function, S_{ip}, has the form shown in Figure 2(b).



(a) Two-dimensional GIMP shape functions, $S_i(\mathbf{x})$, compared to those for traditional MPM (blue).

(b) Two-dimensional GIMP effective shape functions, $S_i \chi^p$.

Figure 2 – *GIMP grid shape functions and effective grid-particle shape functions.*

After the discretized balance of linear momentum has been solved on the grid, the next step is to move the updated field quantities back to the particles. Let f_i is the nodal values of a field on the grid. We assume that

$$f(\mathbf{x}) = \sum_i f_i S_i(\mathbf{x}) \,. \tag{23}$$

The value of the field at a particle p is assumed to be the weighted average of the field over the particle domain:

$$f^p(\mathbf{x}^p) = \frac{1}{W^p} \int_{\Omega_p \cap \Omega} f(\mathbf{x}) w^p(\mathbf{x}) \, \mathrm{dV} \quad \text{where} \quad W^p := \int_{\Omega_p \cap \Omega} w^p(\mathbf{x}) \, \mathrm{dV} \,. \tag{24}$$

The weighting function w^p can be chosen to be one, in which case the particle value is just a average over the particle volume V^p. In our MPM approach, the average in (24) is approximated as

$$f^p \approx \frac{1}{W^{p,\star}} \int_{\Omega_p \cap \Omega} f(\mathbf{x}) w^{p,\star}(\mathbf{x})\, \mathrm{dV} \quad \text{where} \quad W^{p,\star} := \int_{\Omega_p \cap \Omega} w^{p,\star}(\mathbf{x})\, \mathrm{dV} \tag{25}$$

with the weighting function defined as

$$w^{p,\star} := w^p(\mathbf{x}) \chi^{p,*} \,. \tag{26}$$

The particle averaging function $\chi^{p,*}$ is required to have compact support only in a neighborhood containing particle p. These particle neighborhoods do not typically tessellate the body because a perfect description of Ω_p is not generally available. Combining (23) and (25), we have

$$f^p \approx \sum_i f_i \frac{1}{W^{p,\star}} \int_{\Omega_p \cap \Omega} S_i(\mathbf{x}) w^{p,\star}(\mathbf{x})\, \mathrm{dV} = \sum_i f_i \phi_{ip} \tag{27}$$

where the weighted average ϕ_{ip} is defined as

$$\phi_{ip} := \frac{1}{W^{p,\star}} \int_{\Omega_p \cap \Omega} S_i(\mathbf{x}) w^{p,\star}(\mathbf{x})\, \mathrm{dV} \,. \tag{28}$$

Note that we may choose the weighting function to be $w^{p,\star} = 1$ in which case $\phi_{ip} = \langle S_{ip} \rangle$ (see (21)).

The inverse operation of moving particle information to the grid may be interpreted as a least squares minimization problem. Recall from (23) that the field over the body can be expressed as

$$f(\mathbf{x}) = \sum_i f_i S_i(\mathbf{x}) \,. \tag{29}$$

An alternative particle-based representation of the field is

$$f(\mathbf{x}) = \sum_p f^p \chi^p(\mathbf{x}) \tag{30}$$

where χ^p is the particle characteristic function that typically has the form

$$\chi^p(\mathbf{x}) = \begin{cases} 1 & \text{if} \quad \mathbf{x} \in \Omega_p \\ 0 & \text{otherwise} \,. \end{cases} \tag{31}$$

We would like to minimize the weighted square error between the two representations:

$$e = \int_\Omega \left\| \sum_i f_i S_i(\mathbf{x}) - \sum_p f^p \chi^p(\mathbf{x}) \right\|^2 \eta(\mathbf{x})\, \mathrm{dV} \tag{32}$$

where $\eta(\mathbf{x})$ is a weighting function satisfying

$$\int_\Omega \eta(\mathbf{x})\, \mathrm{dV} = 1 \,. \tag{33}$$

Given particle field values f^p, we can minimize e with respect to f_i to get

$$\sum_j \mu_{ij} f_j = \sum_p \nu_{ip} f^p \tag{34}$$

where, using (31),

$$\mu_{ij} = \int_{\Omega} S_i(\mathbf{x}) S_j(\mathbf{x}) \eta(\mathbf{X})\, \mathrm{dV} \quad \text{and} \quad \nu_{ip} = \int_{\Omega_p \cap \Omega} S_i(\mathbf{x}) \eta(\mathbf{X})\, \mathrm{dV}\,. \tag{35}$$

Using a lumped approximation for μ_{ij} and assuming that the weight function $\eta(\mathbf{x})$ is a normalized form of $w^p(\mathbf{x})$, we get the particle-to-node mapping

$$f_i = \sum_p f^p \psi_{pi} \quad \text{where} \quad \psi_{pi} := \frac{V^p \phi_{ip}}{\sum_p V^p \phi_{ip}}\,. \tag{36}$$

Alternatively, we can write the above equation as

$$f_i \sum_p V^p \phi_{ip} = \sum_p f^p V^p \phi_{ip}\,. \tag{37}$$

Let us define two extensive nodal quantities, the nodal volume (V_i) and the nodal mass (m_i), as

$$V_i := \sum_p V^p \langle S_{ip} \rangle \quad \text{and} \quad m_i := \sum_p m^p \langle S_{ip} \rangle\,. \tag{38}$$

If we consider the choice $w^{p,\star}(\mathbf{x}) = 1$ in which case $\phi_{ip} = \langle S_{ip} \rangle$, equation (37) can be written as

$$f_i V_i = \sum_p f^p V^p \langle S_{ip} \rangle\,. \tag{39}$$

This is the volume-weighted mapping from particles to the grid that is used in many MPM algorithms. Alternatively, if we choose $\phi_{ip} = \rho^p \langle S_{ip} \rangle$ where the mass density ρ^p is assumed to be constant over a particle, we have

$$f_i \sum_p V^p \rho^p \langle S_{ip} \rangle = \sum_p f^p V^p \rho^p \langle S_{ip} \rangle \tag{40}$$

or

$$f_i m_i = \sum_p f^p m^p \langle S_{ip} \rangle\,. \tag{41}$$

This mapping is the mass-weighted form that is also used extensively in MPM algorithms. In our implementation, we use a volume-weighted map for the momentum per unit volume $(\rho\mathbf{v})$ and, implicitly, a mass-weighted map for the velocity. A mass-weighted map is also used for the Cauchy stress, though it may be argued that a volume-weighted map is more appropriate for the stress because of its units of energy per volume.

2.4 Time integration

In this work we use an explicit time integrator:

$$\mathbf{x}_i(t) = \mathbf{x}_{n,i} + \mathbf{v}_{n,i}\,(t - t_n) + \tfrac{1}{2}\mathbf{a}_{n,i}\,(t - t_n)^2\,, \quad \mathbf{v}_i(t) = \mathbf{v}_{n,i} + \mathbf{a}_{n,i}\,(t - t_n)\,. \tag{42}$$

Once these grid quantities are known, an end-of-step mapping operation is needed in MPM to map fields to a new grid. In MPM, the body Ω is discretized into n_p non-overlapping particles that tessellate the body. Grid values are first transferred to MPM particles and then remapped to the new grid. Accelerations are mapped from the grid to particles using

$$\mathbf{a}_n^p = \sum_i \mathbf{a}_{n,i} \langle S_{ip} \rangle \tag{43}$$

where $\langle S_{ip} \rangle$ is the average of the i-th grid basis function over the p-th particle:

$$\langle S_{ip} \rangle := \frac{1}{V^p} \int_{\Omega_p \cap \Omega} S_i(\mathbf{x}) \, \mathrm{dV} \tag{44}$$

where V^p is the volume of the MPM particle Ω_p. The particle positions and velocities are subsequently updated using

$$\mathbf{v}^p_{n+1} = \mathbf{v}^p_n + \mathbf{a}^p_n \Delta t \, , \quad \mathbf{x}^p_{n+1} = \mathbf{x}^p_n + \mathbf{v}^p_n \Delta t + \tfrac{1}{2} \mathbf{a}^p_n \Delta t^2 \, . \tag{45}$$

A significant component of any MPM implementation is the nonlinear constitutive model that maps deformations to stresses ($\boldsymbol{\sigma}^p$) at each particle. These models typically require the spatial velocity gradient ($\boldsymbol{l}$) and the deformation gradient ($\boldsymbol{F}$), defined as

$$\boldsymbol{l} := \frac{\partial \mathbf{v}}{\partial \mathbf{x}} = \frac{\partial \mathbf{v}}{\partial \mathbf{x}_n} \cdot \frac{\partial \mathbf{x}_n}{\partial \mathbf{x}} = \dot{\boldsymbol{f}} \cdot \boldsymbol{f}^{-1} \quad \text{and} \quad \boldsymbol{F} := \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \frac{\partial \mathbf{x}}{\partial \mathbf{x}_n} \cdot \frac{\partial \mathbf{x}_n}{\partial \mathbf{X}} = \boldsymbol{f} \cdot \boldsymbol{F}_n \tag{46}$$

where $\mathbf{x}$ is the current position of a material point and $\mathbf{X}$ is the position of the material point at time $t = 0$. Using the Galerkin expansion (7) for position vectors on the grid:

$$\mathbf{x}(\mathbf{x}_n, t) = \sum_i \mathbf{x}_i(t) S_i(\mathbf{x}_n) \tag{47}$$

and substituting into (5) (using (10)), we have

$$\boldsymbol{f} = \sum_i \mathbf{x}_i \otimes \frac{\partial S_i}{\partial \mathbf{x}_n} = \sum_i \mathbf{x}_i \otimes \mathbf{G}_{n,i} \, , \quad \dot{\boldsymbol{f}} = \sum_i \mathbf{v}_i \otimes \mathbf{G}_{n,i} \, , \quad \ddot{\boldsymbol{f}} = \sum_i \mathbf{a}_i \otimes \mathbf{G}_{n,i} \, . \tag{48}$$

At $t = t_n$, we then have

$$\boldsymbol{f}_n = \sum_i \mathbf{x}_{n,i} \otimes \mathbf{G}_{n,i} = \boldsymbol{I} \, , \quad \dot{\boldsymbol{f}}_n = \sum_i \mathbf{v}_{n,i} \otimes \mathbf{G}_{n,i} \, , \quad \ddot{\boldsymbol{f}}_n = \sum_i \mathbf{a}_{n,i} \otimes \mathbf{G}_{n,i} \, . \tag{49}$$

Also, using the explicit integration scheme in (42), we have

$$\begin{aligned} \sum_i \mathbf{x}_i(t) \otimes \mathbf{G}_{n,i} &= \sum_i \mathbf{x}_{n,i} \otimes \mathbf{G}_{n,i} + \sum_i \mathbf{v}_{n,i} \otimes \mathbf{G}_{n,i} \, (t - t_n) + \tfrac{1}{2} \sum_i \mathbf{a}_{n,i} \otimes \mathbf{G}_{n,i} \, (t - t_n)^2 \\ \sum_i \mathbf{v}_i(t) \otimes \mathbf{G}_{n,i} &= \sum_i \mathbf{v}_{n,i} \otimes \mathbf{G}_{n,i} + \sum_i \mathbf{a}_{n,i} \otimes \mathbf{G}_{n,i} \, (t - t_n) \, . \end{aligned} \tag{50}$$

Using (49) in the above leads to

$$\begin{aligned} \sum_i \mathbf{x}_i \otimes \mathbf{G}_{n,i} = \boldsymbol{f}(t) &= \boldsymbol{I} + \dot{\boldsymbol{f}}_n \, (t - t_n) + \tfrac{1}{2} \ddot{\boldsymbol{f}}_n \, (t - t_n)^2 \\ \dot{\boldsymbol{f}}(t) &= \dot{\boldsymbol{f}}_n + \sum_i \ddot{\boldsymbol{f}}_n \, (t - t_n) \, , \quad \ddot{\boldsymbol{f}}(t) = \dot{\boldsymbol{f}}_n \, . \end{aligned} \tag{51}$$

We may now compute the deformation gradient increment at any $t \in [t_n, t_{n+1}]$, e.g.,

$$\boldsymbol{f}_{n+\frac{1}{2}} = \sum_i \mathbf{x}_{n+\frac{1}{2},i} \otimes \mathbf{G}_{n,i} \, , \quad \boldsymbol{f}_{n+1} = \sum_i \mathbf{x}_{n+1,i} \otimes \mathbf{G}_{n,i} \tag{52}$$

where

$$\mathbf{x}_{n+\frac{1}{2},i} = \mathbf{x}_{n,i} + \mathbf{v}_{n,i}\frac{\Delta t}{2} + \mathbf{a}_{n,i}\frac{\Delta t^2}{8} ~,~~ \mathbf{x}_{n+1,i} = \mathbf{x}_{n,i} + \mathbf{v}_{n,i}\Delta t + \mathbf{a}_{n,i}\frac{\Delta t^2}{2} ~. \tag{53}$$

Since the spatial velocity gradient and the deformation gradient are evaluated at MPM particles, the above functions must be evaluated as a weighted average over the particle domain. Then we can write

$$\boldsymbol{f}^p_n = \boldsymbol{I} ~,~~ \boldsymbol{f}^p_{n+\frac{1}{2}} = \sum_i \mathbf{x}_{n+\frac{1}{2},i}\langle \mathbf{G}_{n,ip}\rangle ~,~~ \boldsymbol{f}^p_{n+1} = \sum_i \mathbf{x}_{n+1,i}\langle \mathbf{G}_{n,ip}\rangle \tag{54}$$

and

$$\dot{\boldsymbol{f}}^p_n = \sum_i \mathbf{v}_{n,i}\langle \mathbf{G}_{n,ip}\rangle ~,~~ \dot{\boldsymbol{f}}^p_{n+\frac{1}{2}} = \sum_i \mathbf{v}_{n+\frac{1}{2},i}\langle \mathbf{G}_{n,ip}\rangle ~,~~ \dot{\boldsymbol{f}}^p_{n+1} = \sum_i \mathbf{v}_{n+1,i}\langle \mathbf{G}_{n,ip}\rangle ~. \tag{55}$$

These averaged incremental particle deformation gradients and their rates are related to the particle velocity and deformation gradients via

$$\boldsymbol{l}^p_n = \dot{\boldsymbol{f}}^p_n \cdot (\boldsymbol{f}^p_n)^{-1} ~,~~ \boldsymbol{l}^p_{n+\frac{1}{2}} = \dot{\boldsymbol{f}}^p_{n+\frac{1}{2}} \cdot (\boldsymbol{f}^p_{n+\frac{1}{2}})^{-1} ~,~~ \boldsymbol{l}^p_{n+1} = \dot{\boldsymbol{f}}^p_{n+1} \cdot (\boldsymbol{f}^p_{n+1})^{-1} \tag{56}$$

and

$$\boldsymbol{F}^p_n = \boldsymbol{f}^p_n \cdot \boldsymbol{F}^p_n ~,~~ \boldsymbol{F}^p_{n+\frac{1}{2}} = \boldsymbol{f}^p_{n+\frac{1}{2}} \cdot \boldsymbol{F}^p_n ~,~~ \boldsymbol{F}^p_{n+1} = \boldsymbol{f}^p_{n+1} \cdot \boldsymbol{F}^p_n ~. \tag{57}$$

The particle volumes may then be computed from the deformation gradients using

$$V^p_n = \det(\boldsymbol{F}^p_n)\,V^p_0 ~,~~ V^p_{n+\frac{1}{2}} = \det(\boldsymbol{F}^p_{n+\frac{1}{2}})\,V^p_0 ~,~~ V^p_{n+1} = \det(\boldsymbol{F}^p_{n+1})\,V^p_0 \tag{58}$$

where V^p_0 is the particle volume at $t = 0$. Note that for the incremental rate-form constitutive models in this work we use particle values at $t = t_n + \frac{1}{2}\Delta t$ to compute the rate of deformation tensor.

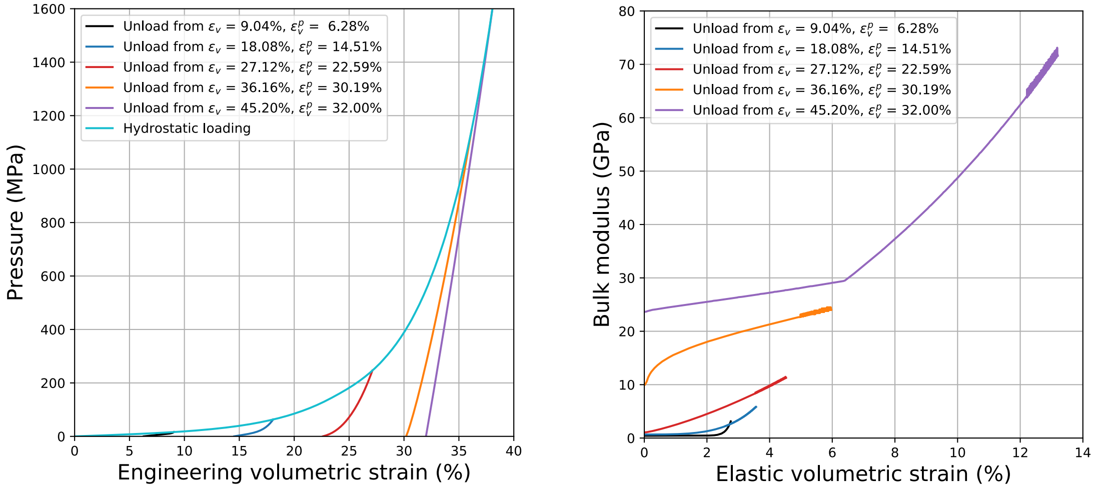
3 Experimental data

The experimental data used in this work as a test-bed for the modeling process are for a dry, poorly-graded, concrete sand described by Fox et al. (2014) and tested at the University of Maryland.[6] Further details on the particular data set used in this work can be found in Banerjee, Fox, and Regueiro (2020c). Compressive stress and strain states have been assigned positive values while tensile states are assigned negative values.

Hydrostatic loading-unloading data for the concrete sand are presented in the form of pressure (p) as a function of the total volumetric strain (ε_v) in Figure 3(a). The loading curve, shown in cyan, is used to fit a crush-curve model. The unloading curves are used to fit a bulk modulus model that depends on the volumetric plastic strain. Tangents to the unloading curves represent the bulk modulus and have been plotted in Figure 3(b). Observe that there is a strong dependence of the bulk modulus (K) on both the elastic (ε_v^e) and plastic (ε_v^p) volumetric strains.

The crush-curve extracted from the hydrostatic compression data is shown in Figure 4(a). Since the term "crush-curve" more commonly refers to the change in porosity as a function of pressure, we show that curve in Figure 4(b). For the crush-curve, the porosity (ϕ) has been computed using $\phi = p_3 - \varepsilon_p^v$ where ε_p^v is the volumetric plastic strain and $p_3 = 0.325$ is the volumetric plastic strain at which all pores have been crushed (R. M. Brannon et al., 2015).

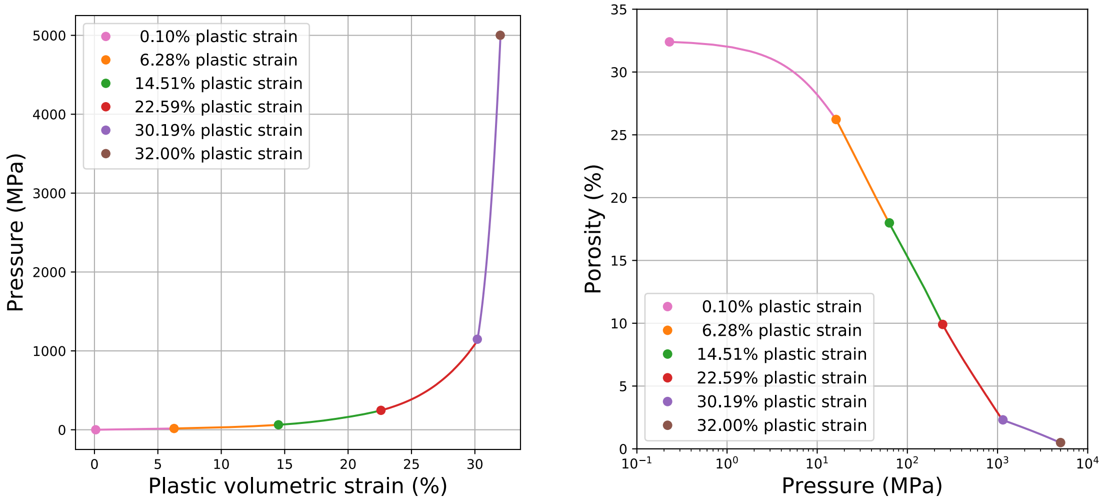
[6] Stephen Akers, 2018, Private communication, CCDC Army Research Laboratory, Aberdeen Proving Ground, MD, USA



(a) Loading-unloading curves

(b) Tangent bulk modulus curves

Figure 3 – *Hydrostatic loading-unloading data and unloading bulk moduli for dry poorly-graded concrete sand.*



(a) Pressure crush-curve.

(b) Porosity crush-curve.

Figure 4 – *Crush-curves for dry poorly-graded concrete sand.*

The yield function data for the concrete sand intersects the deviatoric stress axis in the compressive half-space, at an abscissa of approximately 5 kPa, indicating that a purely elastic response is not possible when the sand was loaded from zero strain in either tension or compression. To ensure robustness of our simulations, the yield data were shifted to the tensile regime by 5 kPa to provide a small amount of cohesion. A plot of the yield function in p-q space is shown in Figure 5. Here p is the mean stress, defined as $p = 1/3\,\text{tr}(\boldsymbol{\sigma})$, q is the effective deviatoric yield stress given by $q = \sqrt{3J_2}$ where $J_2 = 1/2\,\mathbf{s} : \mathbf{s}$, $\boldsymbol{\sigma}$ is the Cauchy stress, $\mathbf{s} = \boldsymbol{\sigma} - p\boldsymbol{I}$ is the deviatoric part of the Cauchy stress, and $\boldsymbol{I}$ is the second-order identity tensor. Note that the existence of a crush curve indicates that nonlinear bulk moduli are not sufficient to model the observed yield response and a compression cap model is essential.

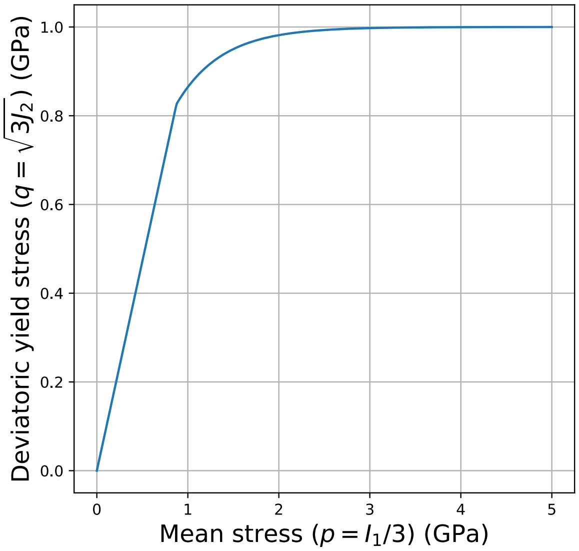


Figure 5 – *Yield function for dry poorly-graded concrete sand.*

In the simulations described in this paper, tabular models are used for the crush curve and yield function as described in Banerjee, Fox, and Regueiro (2020d) with linear interpolation of the data (Banerjee, Fox, and Regueiro, 2020a). The purely tabular approach is compared with models where elastic moduli are computed using support vector regression (SVR) (Banerjee, Fox, and Regueiro, 2020c) and multilayer perceptron (MLP) neural network models (Banerjee, Fox, and Regueiro, 2020b). Further detail can be found in the deformation-driven simulations in Banerjee (2020).

3.1 Tabular model

The bulk modulus vs. pressure data shown in Figure 6 are interpolated to determine the value of the tangent bulk and shear moduli at a given value of mean stress and volumetric plastic strain. Derivatives of the bulk modulus with respect to plastic strain are computed using a central difference scheme using two nearby values of plastic strain that differ by 2.0×10^{-6}. A constant Poisson's ratio of 0.189 is used to compute the shear modulus from the interpolated bulk modulus.

The data are read is as a JSON file (Crockford, 2006). The format of the file is described in Banerjee (2014).

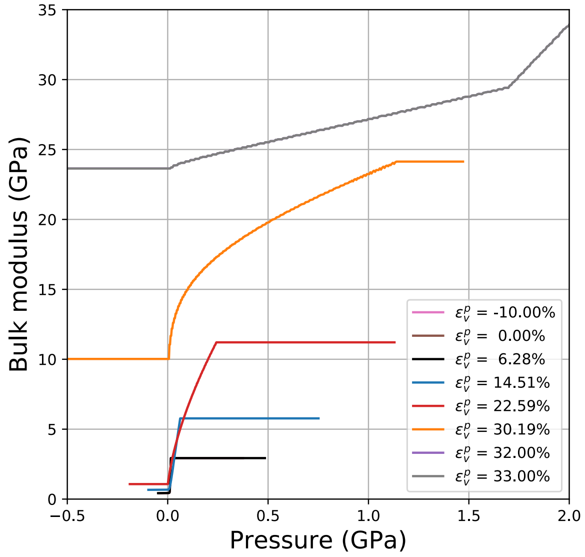
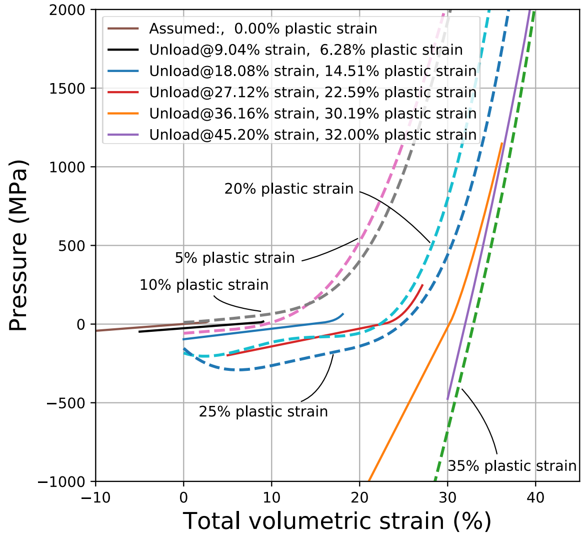


Figure 6 – *Bulk modulus table expressed as a function of pressure.*

3.2 Support vector regression model

The support vector regression (SVR) model used in these simulations is the pressure vs. total volumetric strain fits to the experimental data compute with a value of $C = 10$ and $\epsilon = 0.001$ as described in Banerjee, Fox, and Regueiro (2020c). Pressure curves predicted by the model are shown in Figure 7. Bulk modulus are extracted from the predicted pressures by analytical differentiation of the pressure SVR model.



$C = 10$, $\epsilon = 0.001$.

Figure 7 – *Support vector regression predictions of pressure as a function of the total volumetric strain.*

The SVR model is also read into Vaango as a JSON file as described in Banerjee (2014).

3.3 Multiplayer perceptron neural network model

A dense, multilayer perceptron (MLP) model is used to model the elastic modulus data as described in Banerjee, Fox, and Regueiro (2020b). The three-layer model of the bulk modulus as a function of elastic volumetric strain is used in this paper. The model containing 64 neurons with ReLU activation in layer 1, 32 neurons with ReLU activation in layer 2, and 32 neurons with ReLU activation in layer 3. The model was trained for 800 epochs with batch sizes of 32. Predictions from that model are shown in Figure 8. Note that this model uses a linear extrapolation of the experimental pressure curves before computing the bulk modulus instead of assuming constant bulk moduli outside the range of the experimental data. The fitted model is saved as a HDF5 file (Koranne, 2011) by Tensorflow 2.0 (Abadi et al., 2016) and read into Vaango using the HDF5 library.

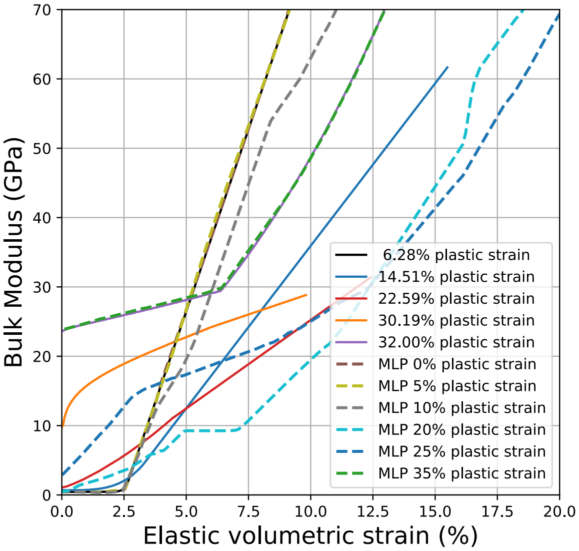


Figure 8 – *Multilayer perceptron neural network predictions of bulk modulus as a function of the elastic volumetric strain.*

4 Rectangular punch penetration

Plane strain simulations were performed of a rectangular steel punch 30 cm × 20 cm impacting the dry, poorly-graded, concrete sand discussed in Section 3. The sand was modeled as contained in a box of dimension 100 cm × 50 cm, with reflecting boundaries. The density of the soil was 1050 kg/m^3. Contact between the punch and the sand was modeled using a frictional contact algorithm (S. Bardenhagen et al., 2001) with a friction coefficient of 0.1. The computational domain had dimensions 100 cm × 80 cm and was discretized using 100 × 80 grid cells, partitioned into four patches for parallel computation. The initial velocity of the punch was 200 m/s. Spatial interpolation was carried out using the GIMP approach and explicit time integration was used, with a timestep multiplier of 0.05. Artificial viscosity was used to damp the simulations but no damage models were activated. A gravity body force was applied in the direction of punch motion based on an acceleration of 1g = 9.81 m/s^2.

Three models for the elastic moduli of the sand have been compared in this section. The first used a tabular interpolation model (Table) as described in (Banerjee, Fox, and Regueiro, 2020a; Banerjee, 2020). The second used a support vector regression model (SVR) for the elastic moduli (Banerjee, Fox, and Regueiro, 2020c; Banerjee, 2020). The third model for the elastic moduli used a multilayer perceptron (MLP) neural net (Banerjee, Fox, and Regueiro, 2020b; Banerjee, 2020). Figure 9 shows the penetration depths predicted by the three models. All three models predict almost identical penetration depths, with the SVR model predicting the deepest crater and the MLP model predicting the shallowest one. The profiles of the ejecta are also slightly different. The MLP model is the most computationally expensive of the three, and the Table model takes the least time to run.

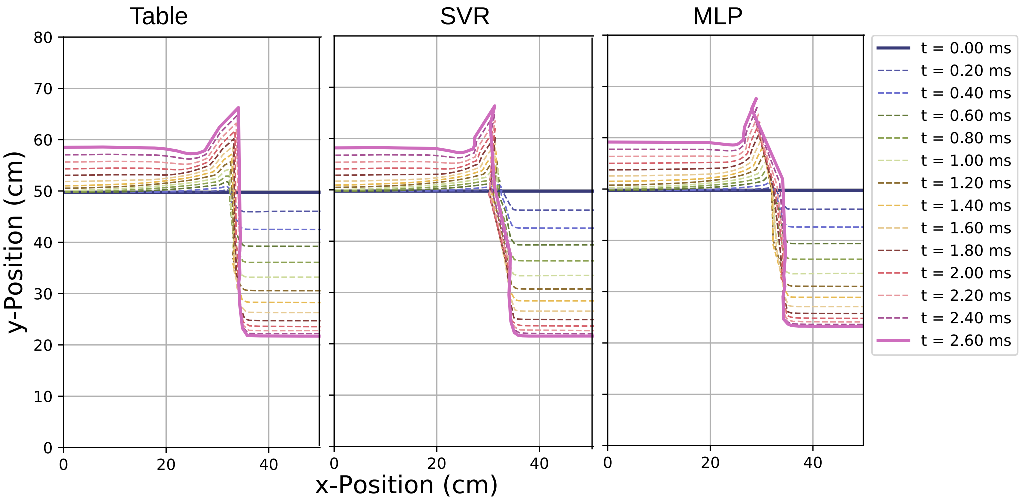


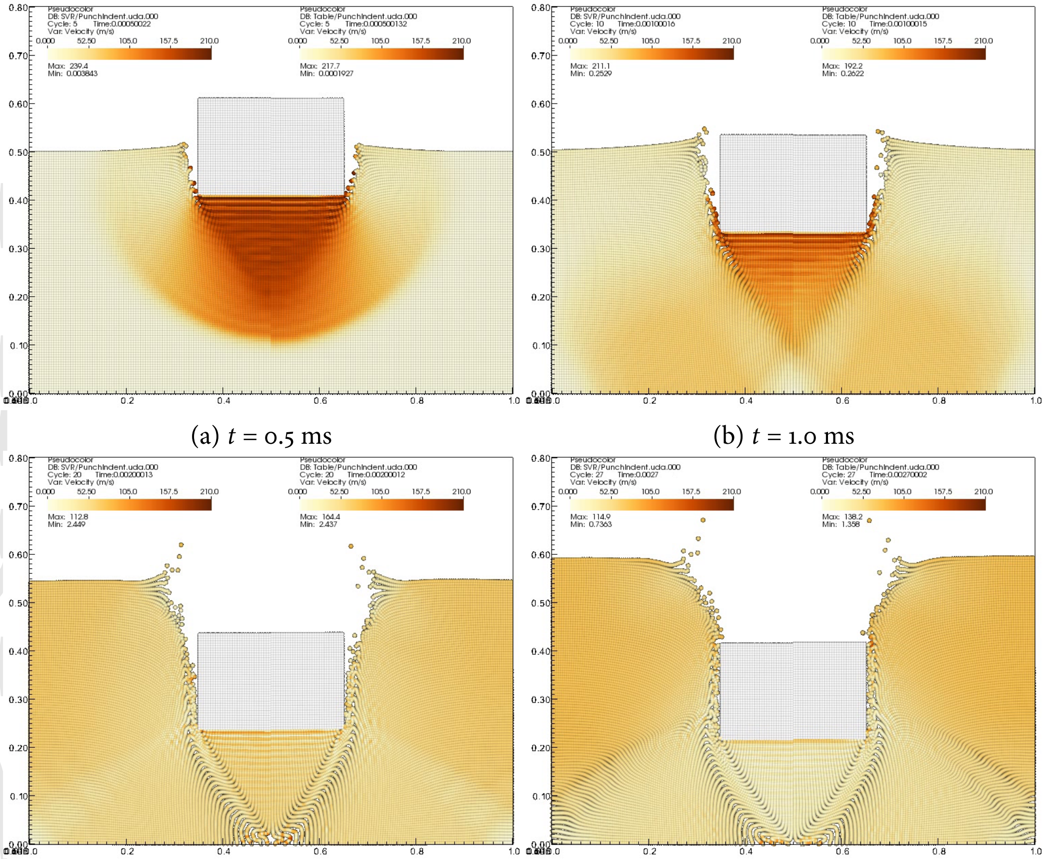
Figure 9 – *Penetration depths predicted by the Table, SVR, and MLP models for a punch impacting a dry poorly-graded concrete sand at 200 m/s.*

It is informative to examine the evolution of various state variables in the sand during the impact and penetration event. We first compare the predictions of the SVR and Table models, followed by the MLP and Table models.

4.1 SVR vs. Table

The evolution of velocity in the sand is shown in Figure 10. Initial velocities predicted by the Table model are marginally faster than those from the SVR model. A plug is formed at approximately 1 ms. Within 2 ms of impact, the sand has slowed down the punch significantly, after which the majority of the energy is used to impart momentum to the ejecta and the soil above the punch. The SVR model predicts a slightly larger depth of penetration than the Table model.

Figure 11 plots the evolution of bulk modulus in the dry sand with colors assigned on a logarithmic scale. Even though the initial bulk modulus is higher in the SVR material and, after 0.5 ms, the bulk modulus of the material in the triangular plug is larger on average for the SVR model than for the



(a) $t = 0.5$ ms

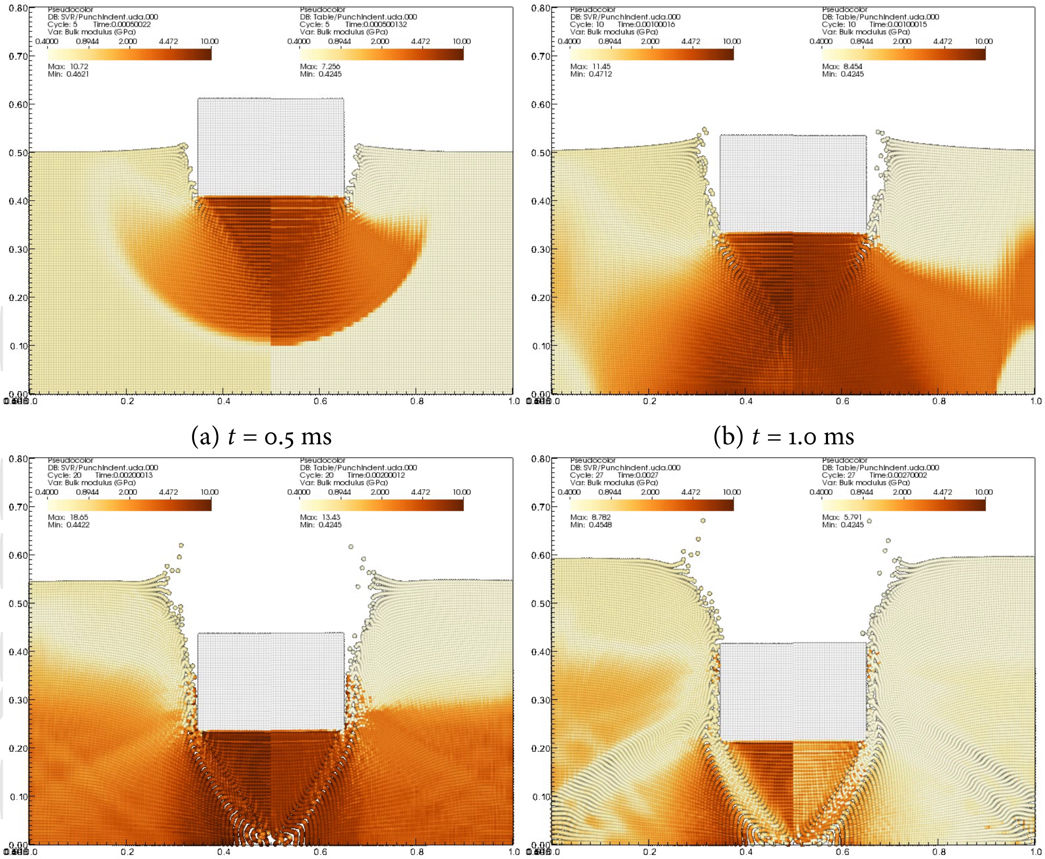
(b) $t = 1.0$ ms

(c) $t = 2.0$ ms

(d) $t = 2.7$ ms

Figure 10 – *Evolution of the particle velocity magnitudes during the punch impact event. The* SVR *predictions are on the left half of each image while the* Table *predictions are on the right half.*

Table model. However, outside the plug region, the bulk modulus is lower in the SVR model. This allows the deformation wave to move faster in the Table material. The bulk modulus increases faster in the Table material as observed from the snapshot at 1 ms. At 2.7 ms, the bulk modulus of the plug from the SVR model is visibly higher than that from the Table model.



(a) $t = 0.5$ ms

(b) $t = 1.0$ ms

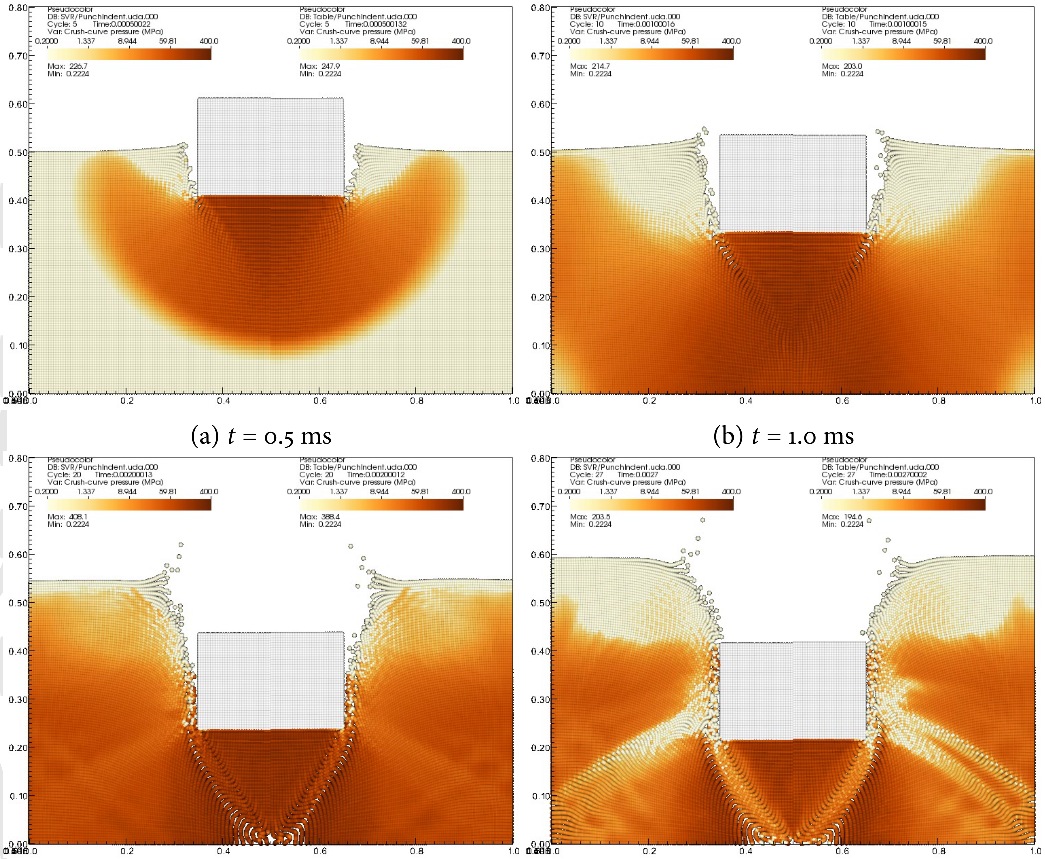
(c) $t = 2.0$ ms

(d) $t = 2.7$ ms

Figure 11 – *Evolution of the particle bulk moduli during the punch impact event. The SVR predictions are on the left half of each image while the Table predictions are on the right half.*

The evolution of the position of the apex of the pressure cap is shown in Figure 12 (using a logarithmic color scale). Even at 0.5 ms after impact, the cap pressure is observed to have evolved to approximately 9 MPa over a large region of the sand. Cap expansion is experienced by all particles except a few near the surface and in high tension/shear regions by 2 ms. As the release wave propagates through the sand, the material experiences tension, and the cap pressure drops off to its original value 2.7 ms after impact. It is at this stage that differences between the SVR and the Table models become apparent.

The mean stress, $p = \text{tr}(\boldsymbol{\sigma})$, predicted by the two models in presented in Figure 13. Interestingly, the evolution of the mean stress is almost identical for the SVR and the Table models. The same is observed for the effective deviatoric stress, $q = \sqrt{3J_2}$, shown in Figure 14. However, the values



(a) t = 0.5 ms

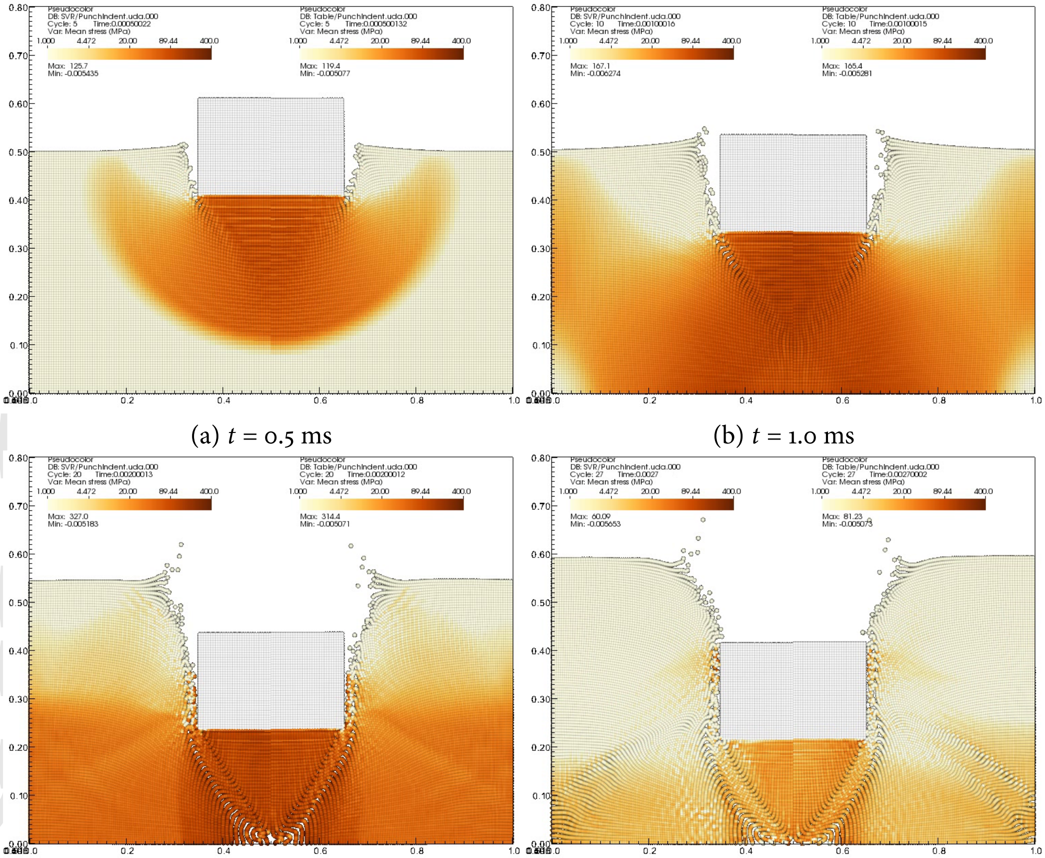
(b) t = 1.0 ms

(c) t = 2.0 ms

(d) t = 2.7 ms

Figure 12 – *Evolution of the particle cap pressure during the punch impact event. The* SVR *predictions are on the left half of each image while the* Table *predictions are on the right half.*

predicted by the SVR model are slightly higher than those from the Table model.



(a) $t = 0.5$ ms

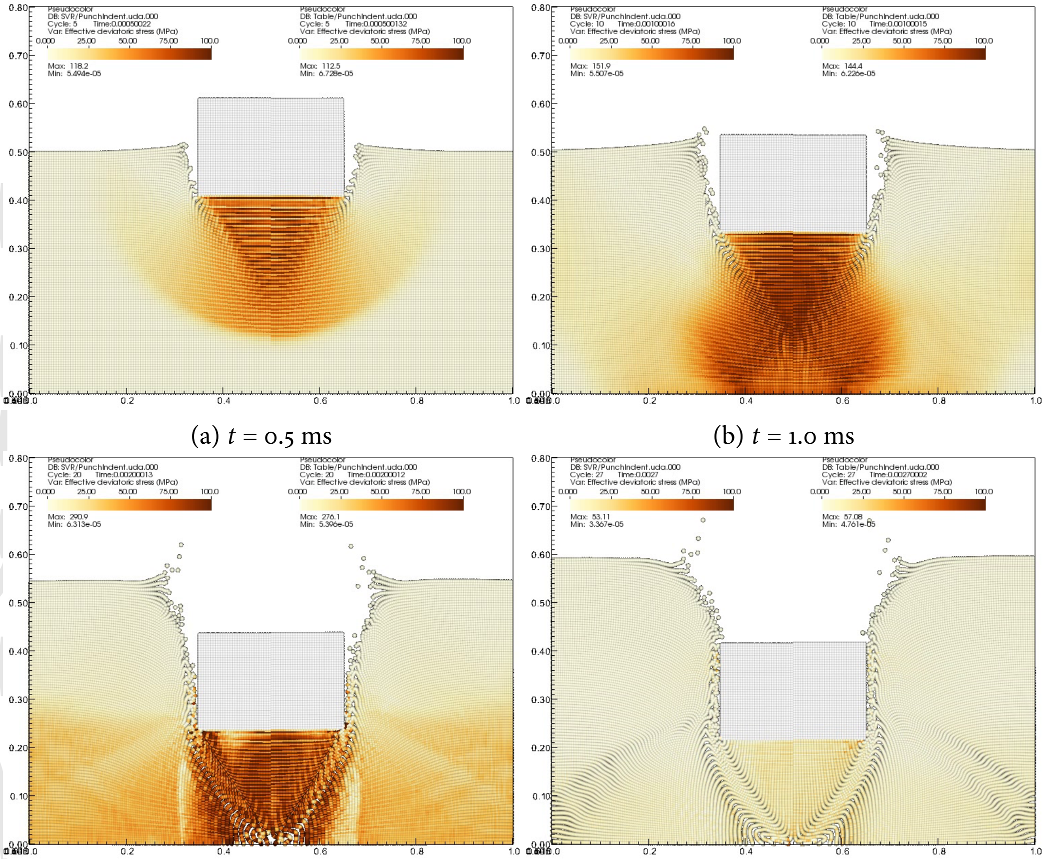
(b) $t = 1.0$ ms

(c) $t = 2.0$ ms

(d) $t = 2.7$ ms

Figure 13 – *Evolution of the particle mean stress during the punch impact event. The SVR predictions are on the left half of each image while the Table predictions are on the right half.*

Figure 15 shows the evolution of the volumetric elastic strain. The values predicted by the SVR models are higher than those from the Table model. At 2.7 ms, the SVR model shows that loading is continuing because of the existence of relatively large elastic strains, which the Table model has unloaded to a larger extent. Plastic volumetric strains are depicted in Figure 16. We observe that tensile volumetric plastic strains develop in the ejecta and regions close to the surface. However, the plastic strains predicted by both the models are nearly identical though there are differences in the shear bands at 2.7 ms.



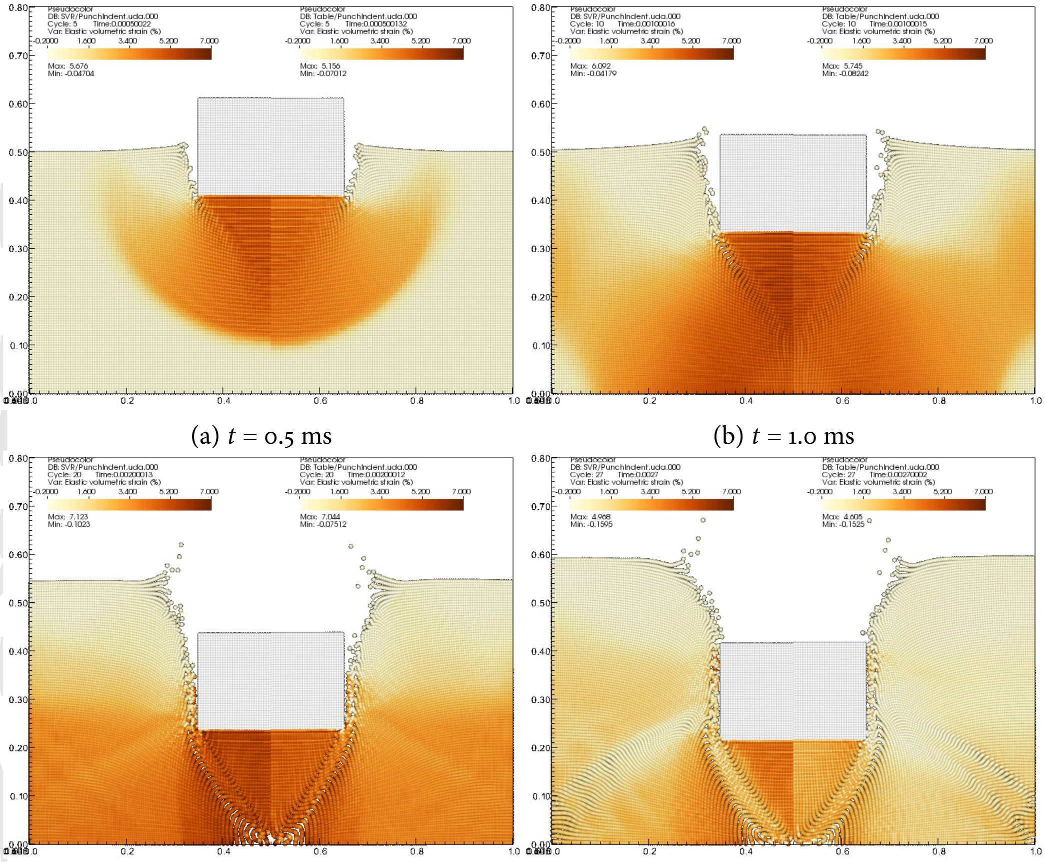
(a) t = 0.5 ms

(b) t = 1.0 ms

(c) t = 2.0 ms

(d) t = 2.7 ms

Figure 14 – *Evolution of the particle effective deviatoric stress during the punch impact event. The SVR predictions are on the left half of each image while the Table predictions are on the right half.*



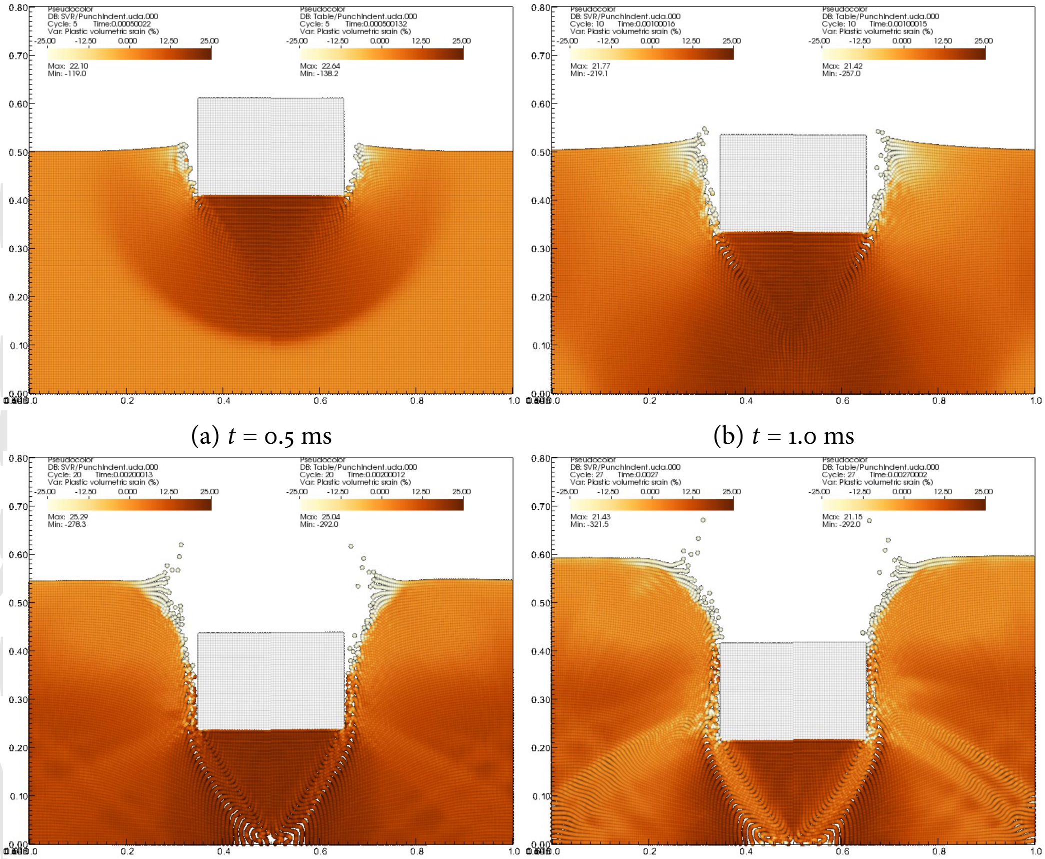
(a) t = 0.5 ms

(b) t = 1.0 ms

(c) t = 2.0 ms

(d) t = 2.7 ms

Figure 15 – *Evolution of the particle elastic volumetric strain during the punch impact event. The SVR predictions are on the left half of each image while the Table predictions are on the right half.*



(a) t = 0.5 ms

(b) t = 1.0 ms

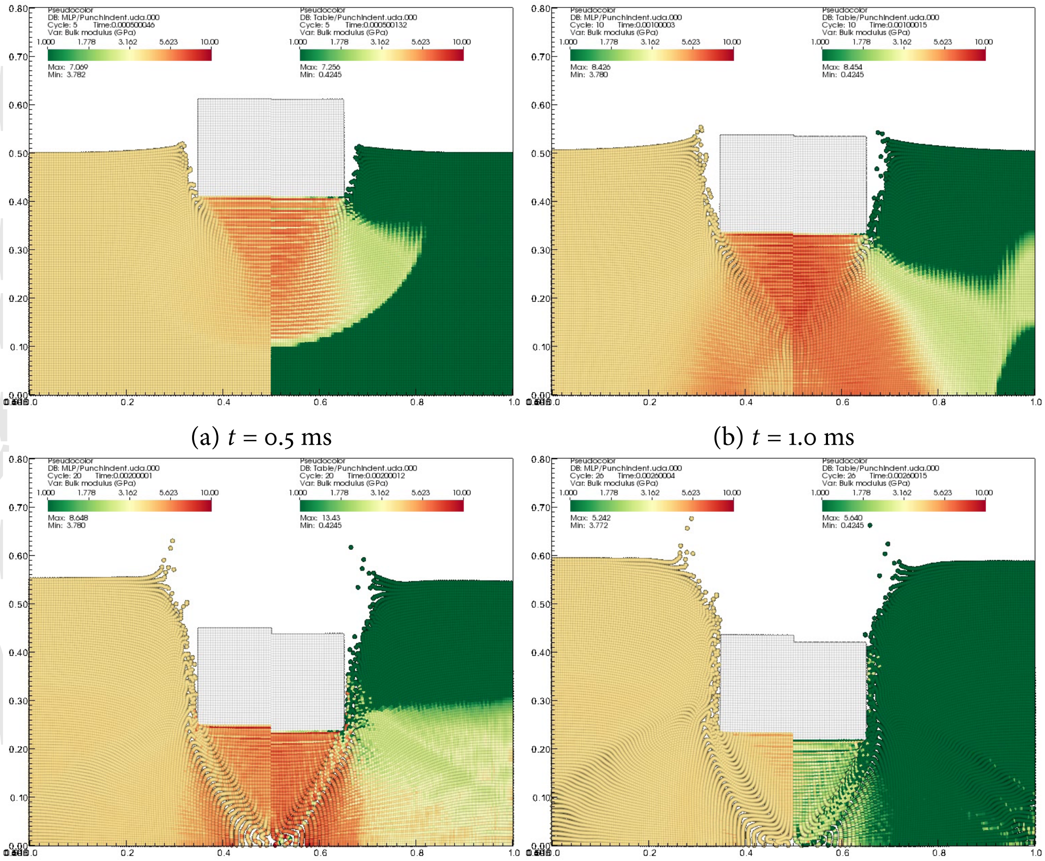
(c) t = 2.0 ms

(d) t = 2.7 ms

Figure 16 – *Evolution of the particle plastic volumetric strain during the punch impact event. The* SVR *predictions are on the left half of each image while the* Table *predictions are on the right half.*

4.2 MLP vs. Table

Comparisons of the predictions from the MLP model are compared with those of the Table model in this section. Figure 17 show the bulk modulus in the dry sand as the impact event progresses, with colors on a logarithmic scale. The initial bulk modulus is significantly higher in the MLP material. The bulk moduli of the MLP and Table materials become compared in the triangular plug after 0.5 ms, and a region in the Table sand even exhibits a higher bulk modulus than the MLP sand at 1 ms and 2 ms. However, the release wave leads to a reduction of the bulk modulus as the material goes into tension at 2.6 ms, the Table sand again reaches a state with bulk moduli lower than the MLP model. Note that the bulk moduli in the MLP sand outside the plug remain remarkably uniform through the event.



(a) t = 0.5 ms

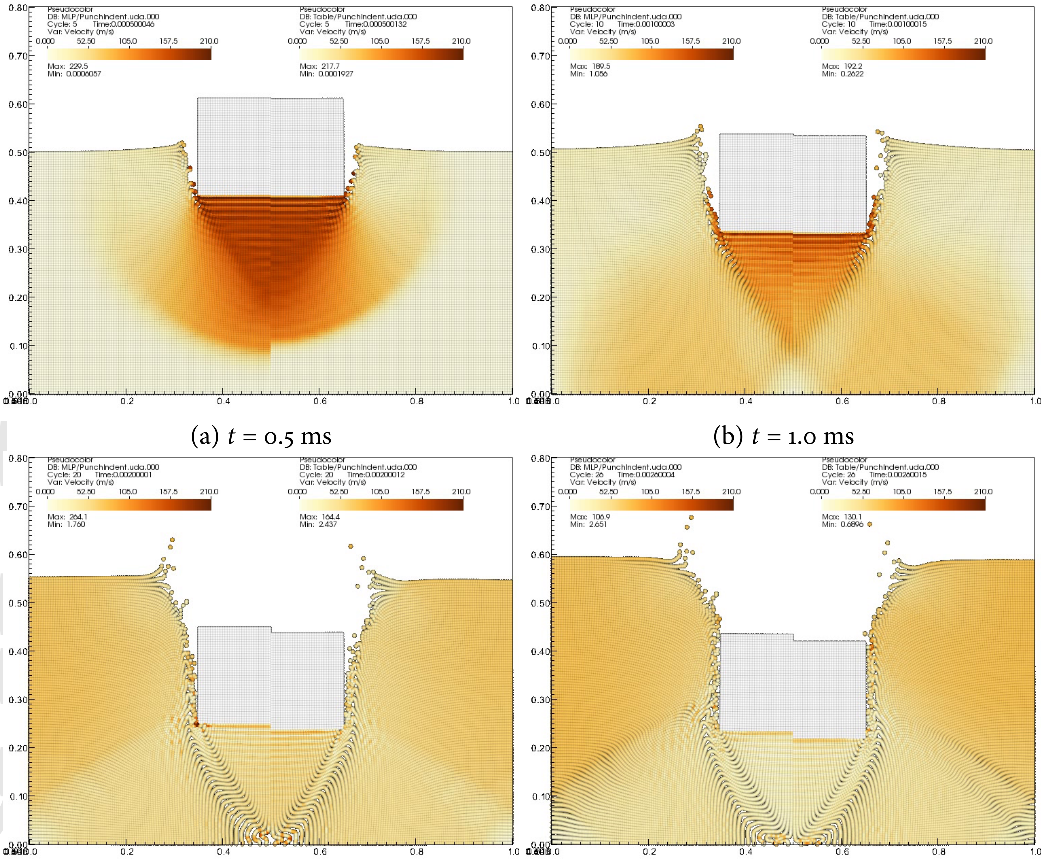
(b) t = 1.0 ms

(c) t = 2.0 ms

(d) t = 2.6 ms

Figure 17 – *Evolution of the particle bulk moduli during the punch impact event. The MLP predictions are on the left half of each image while the Table predictions are on the right half.*

The MLP model predicts a smaller depth of penetration than the Table model. The evolution of velocity in the sand is shown in Figure 18. Initial velocities predicted by the MLP model are faster than those from the Table model. The final velocity distributions predicted by the two models are close to each other.



(a) $t = 0.5$ ms

(b) $t = 1.0$ ms

(c) $t = 2.0$ ms

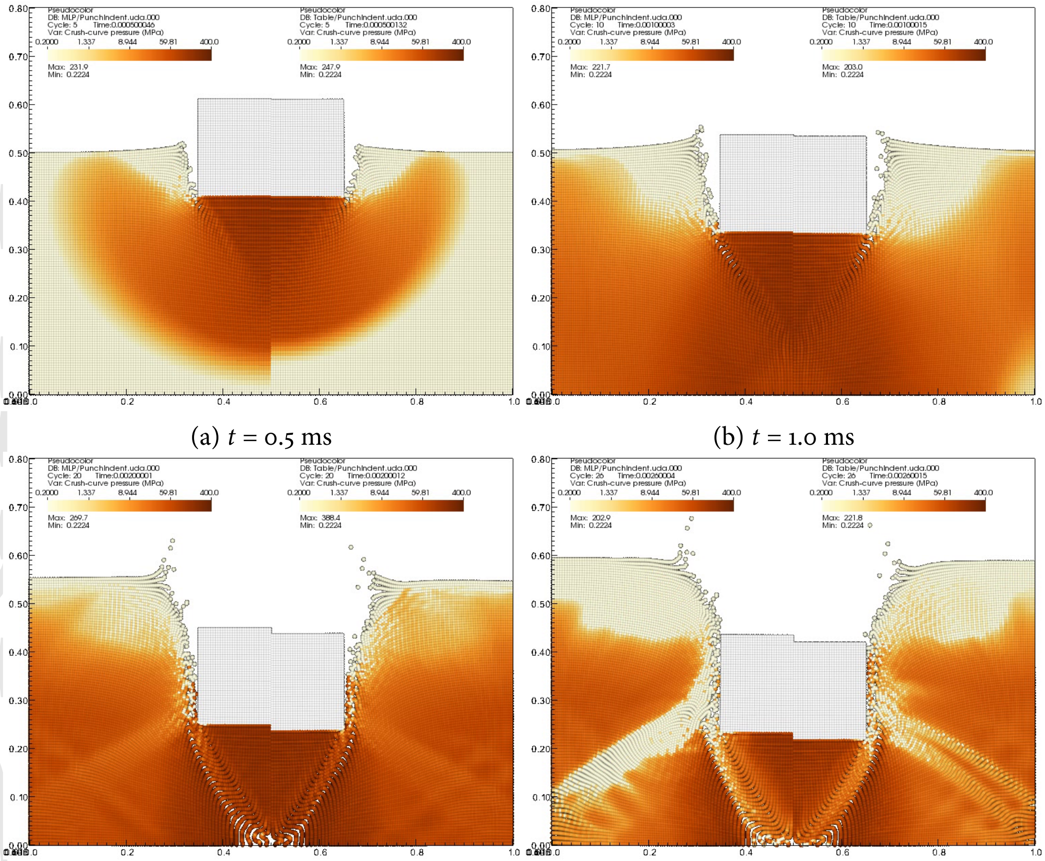
(d) $t = 2.6$ ms

Figure 18 – *Evolution of the particle velocity magnitudes during the punch impact event. The* MLP *predictions are on the left half of each image while the* Table *predictions are on the right half.*

The evolution of the cap pressure is compared in Figure 19. Initially, a faster increase in the cap pressure is predicted by the MLP model. However, as the impact event progresses, the cap pressures in the two materials become comparable. A peak value of around 400 MPa is observed in the Table sand at 2 ms. Interestingly, the MLP model predicts a peak value of only around 270 MPa at that time.

Mean stress evolution in the two materials is shown in Figure 20. The initial wave speed in the MLP material is faster because of the larger bulk modulus. At 1 ms, both materials have approximately the same peak pressures (180 MPa in MLP vs. 166 in Table). However, at 2 ms, the unloading process in already underway in the MLP material (peak pressure of 156 MPa) while the Table material is still loading (peak pressure 315 MPa). At 2.6 ms, both materials have unloaded significantly except for the plug region below the punch.

The effective deviatoric stress is shown in Figure 21. Initial shear wave speeds are also higher in the



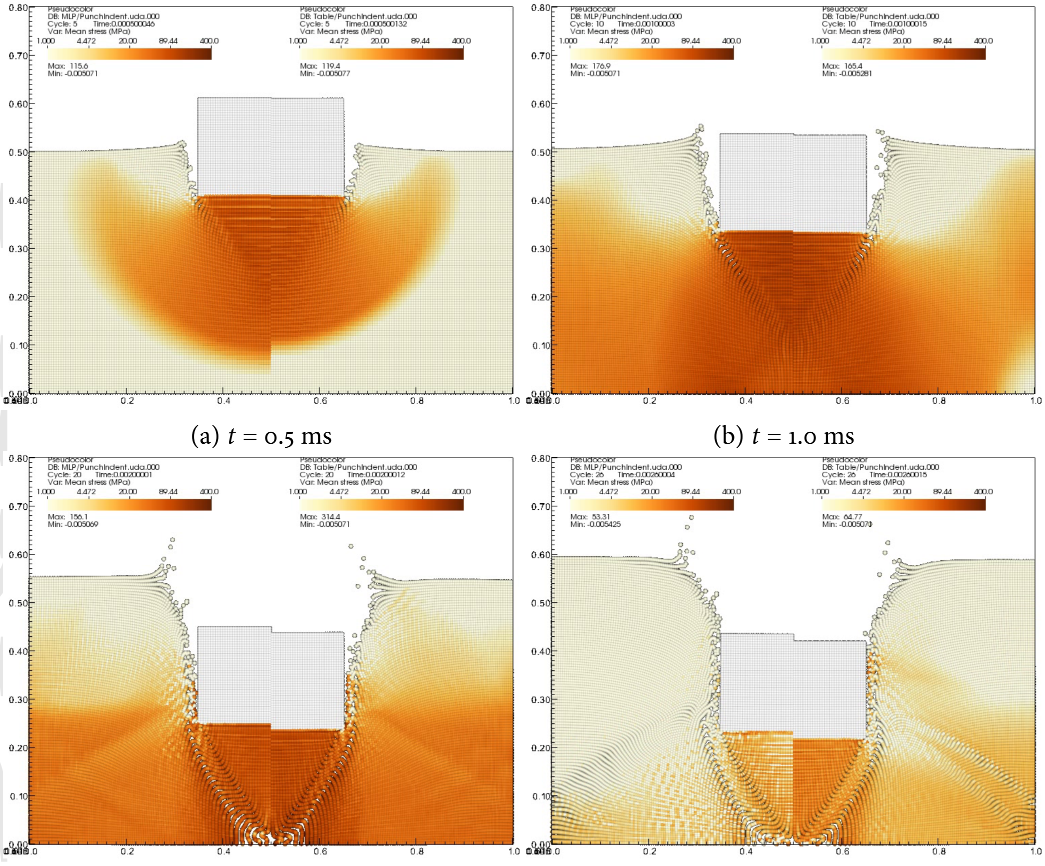
(a) t = 0.5 ms

(b) t = 1.0 ms

(c) t = 2.0 ms

(d) t = 2.6 ms

Figure 19 – *Evolution of the particle cap pressure during the punch impact event. The* MLP *predictions are on the left half of each image while the* Table *predictions are on the right half.*



(a) t = 0.5 ms

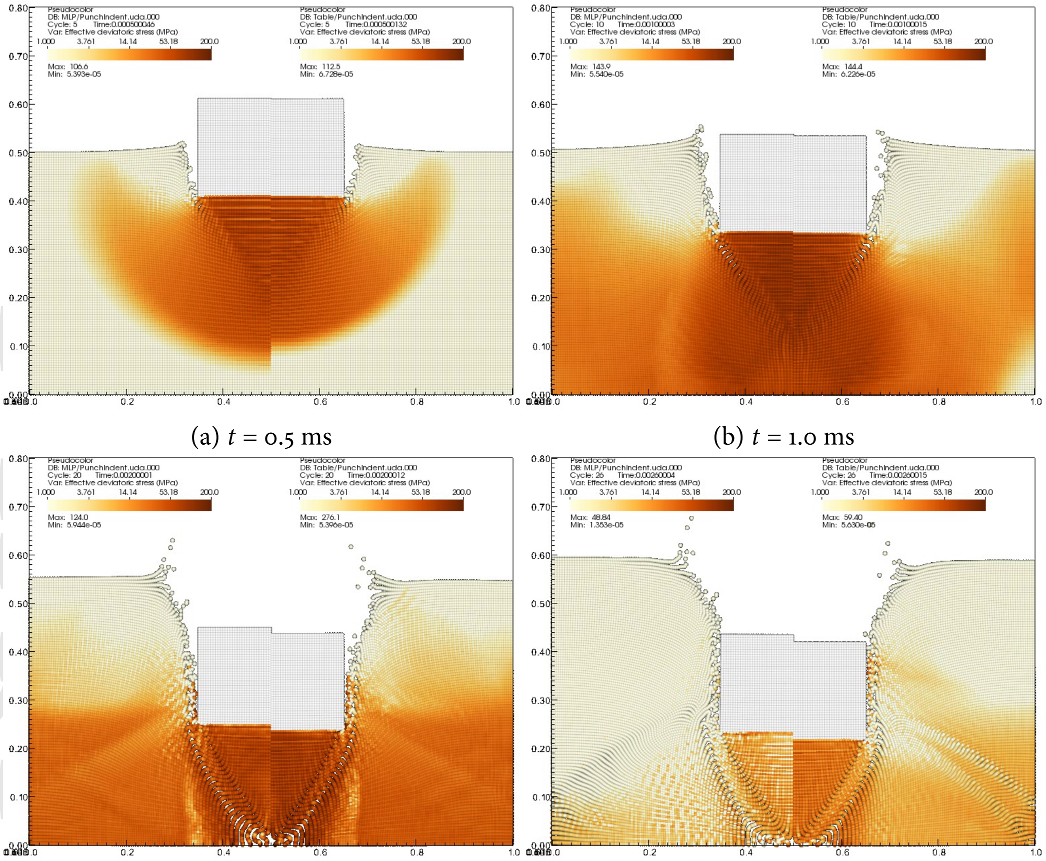
(b) t = 1.0 ms

(c) t = 2.0 ms

(d) t = 2.6 ms

Figure 20 – *Evolution of the particle mean stress during the punch impact event. The* MLP *predictions are on the left half of each image while the* Table *predictions are on the right half.*

MLP material. At 1 ms, effective stresses in the two materials are comparable with peak values of around 145 MPa. At 2 ms, the MLP material has unloaded significantly more than the Table material as can be seen by the peak pressures of 124 MPa and 276 MPa, respectively. At 2.6 ms, both materials have unloaded further and exhibit peak deviatoric stresses of 50 MPa and 60 MPa, respectively.



(a) $t = 0.5$ ms

(b) $t = 1.0$ ms

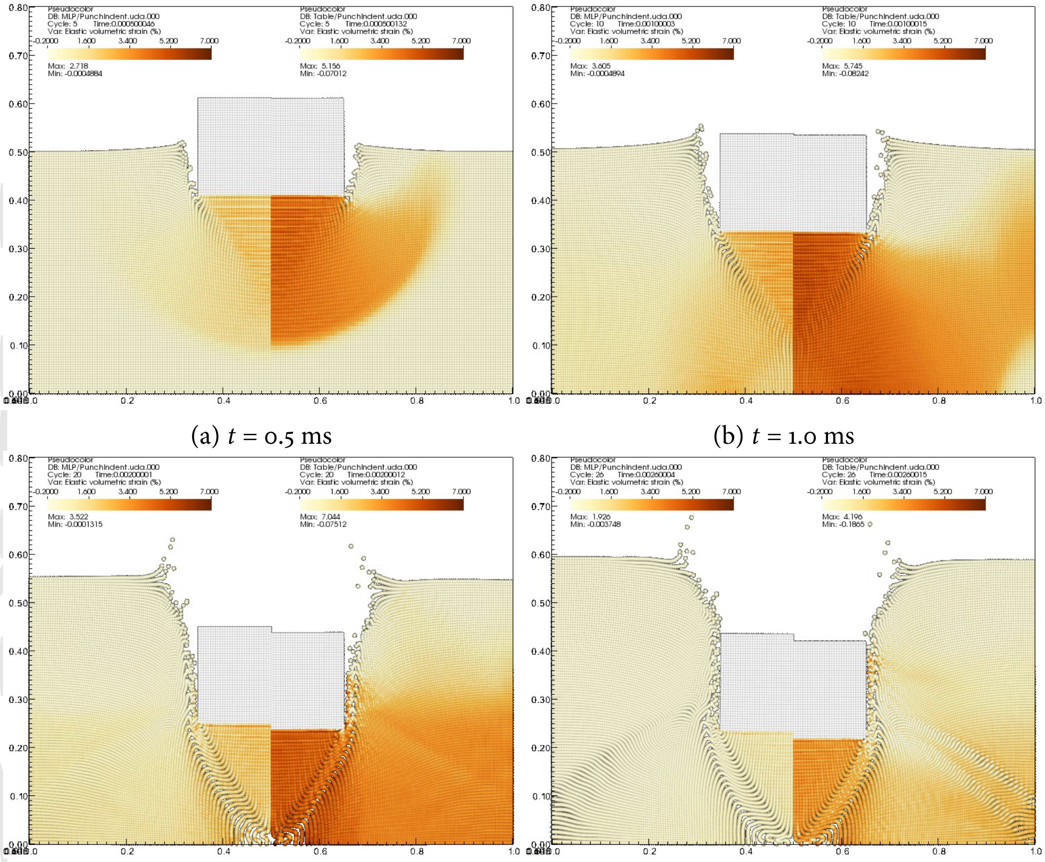
(c) $t = 2.0$ ms

(d) $t = 2.6$ ms

Figure 21 – *Evolution of the particle effective deviatoric stress during the punch impact event. The MLP predictions are on the left half of each image while the Table predictions are on the right half.*

Figure 22 shows the evolution of the volumetric elastic strain in the two materials. As expected, the elastic strains in the MLP material are approximately 50% less than the Table predictions. By 2.6 ms, the MLP material has unloaded almost completely from compression and is largely in the tensile regime, while the Table material continues to be compressed.

On the other hand, plastic volumetric strains are almost identical in the two materials, as seen in Figure 23. The predicted plastic strains from the MLP , SVR , and Table models are all within a few percent of each other.



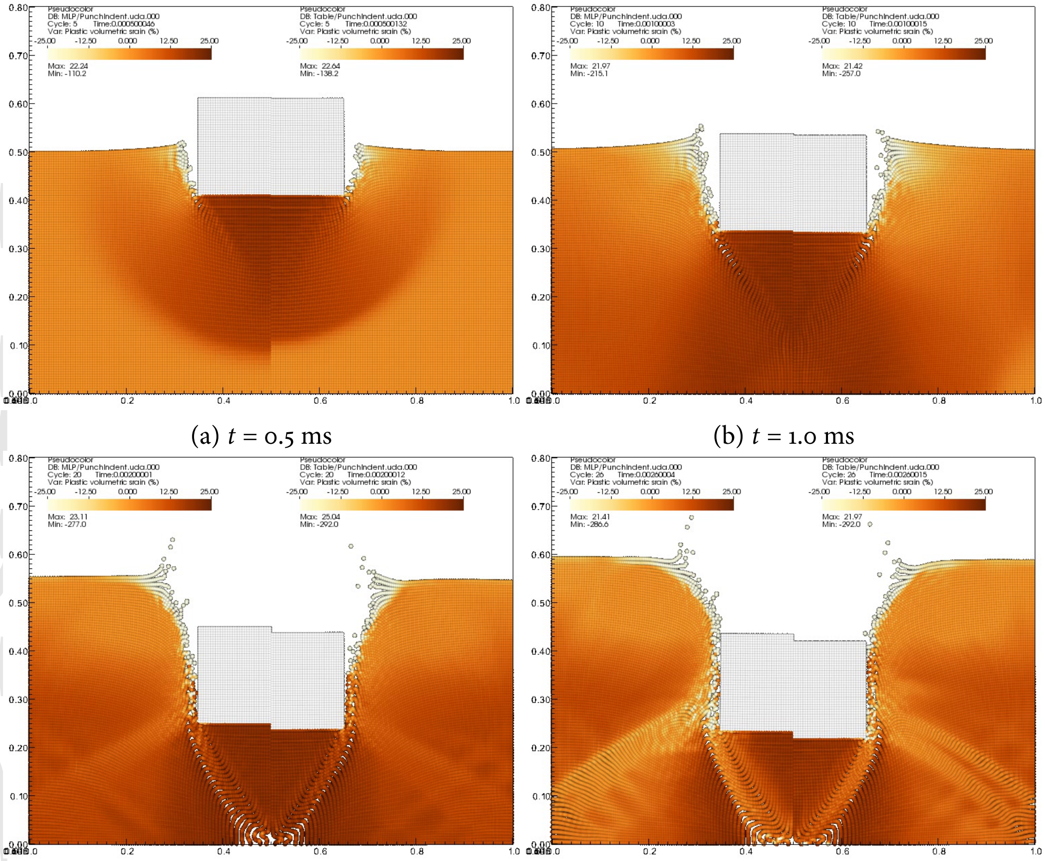
(a) t = 0.5 ms

(b) t = 1.0 ms

(c) t = 2.0 ms

(d) t = 2.6 ms

Figure 22 – *Evolution of the particle elastic volumetric strain during the punch impact event. The* MLP *predictions are on the left half of each image while the* Table *predictions are on the right half.*



(a) $t = 0.5$ ms

(b) $t = 1.0$ ms

(c) $t = 2.0$ ms

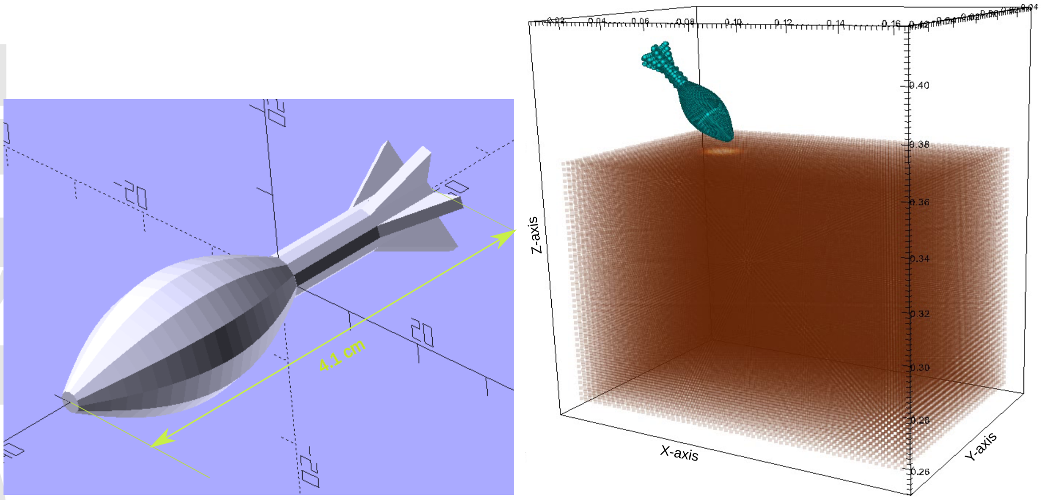
(d) $t = 2.6$ ms

Figure 23 – *Evolution of the particle plastic volumetric strain during the punch impact event. The* MLP *predictions are on the left half of each image while the* Table *predictions are on the right half.*

5 Unexploded ordnance in a centrifuge

In these simulations, we model the entry of a 4.1 cm long unexploded ordnance (UXO) projectile into dry sand contained in a centrifuge bucket rotating at 6 radians/s. The centrifuge arm is 7 m long, leading to an acceleration of 20g at the bottom of the bucket. Standard scaling laws for length imply that the real world projectile being modeled is 82 cm long. The projectile is initially oriented at 45° to the vertical. The projectile density is 5000 kg/m^3, and its bulk and shear moduli are 140 GPa and 8 GPa, respectively. The initial axial velocity of the projectile is 283 m/s. Figure 24(a) shows the geometry of the projectile.

The bucket is modeled with rigid walls and floor, and filled with sand to a height of 10.7 cm. For computational convenience, a section of the bucket 15 cm long and 10 cm thick is modeled. Friction between the projectile and the sand is modeled with a friction coefficient of 0.1. The initial sand density is 1624 kg/m^3 ad the Poisson's ratio is 0.189. Figure 24(b) shows the initial configuration of the projectile and the sand. The material models for the sand are identical to those discussed in Section 4.



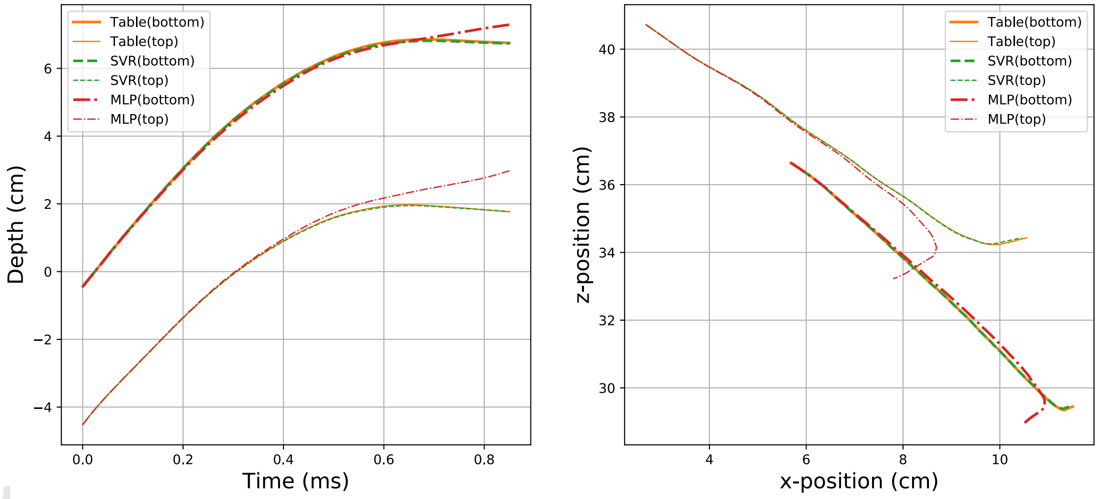
(a) UXO projectile geometry

(b) Sand in centrifuge with UXO

Figure 24 – *Geometry of the UXO projectile and the sand in the centrifuge bucket.*

The depth of penetration as function of time, and the change in orientation of the projectile, has been plotted in Figures 25(a) and (b), respectively. The positions of the top and bottom of the projectile are shown, with the top depicted with thinner lines. From the depth vs. time plots in Figure 25(a) we see that the Table and SVR models predict almost identical depths of penetration. The projectile decelerates rapidly after ≈500 μs, and then bounces back (depth of penetration decreases). Differences in depth between the top and the bottom of the projectile are due to the rotation of the projectile as it decelerates. However, the MLP model predicts a slightly different behavior after 700 μs. For that model, the projectile sinks deeper into the sand at an orientation closer to the initial orientation than the Table and SVR models.

The change in orientation of the projectile can be observed more clearly from Figure 25(b). That



(a) Depth of penetration

(b) Projectile orientation

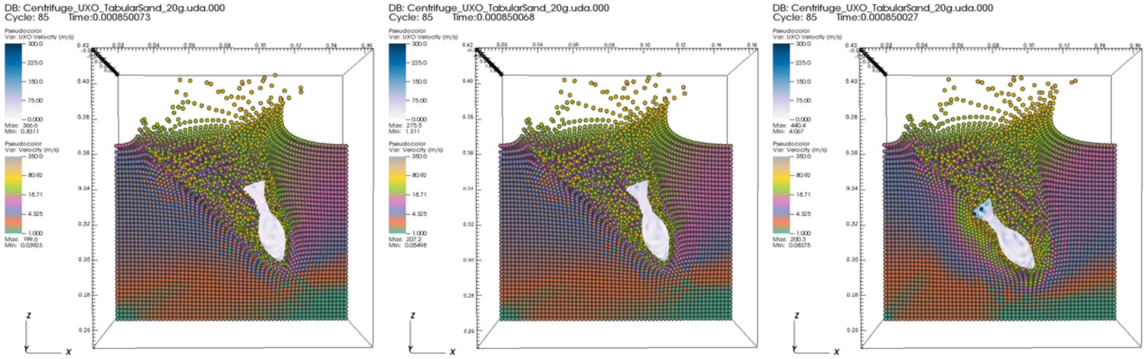
Figure 25 – *Evolution of projectile positions during the impact event.*

figure shows plots of the coordinates of the top and the bottom of the projectile in the plane of impact (x-z). For the Table and SVR model, the final orientation is around 75° from the horizontal. Small differences between the predictions of the Table and SVR models become apparent only after most of the momentum has been transferred from the projectile to the sand. It is also worth noting that there is a few mm motion of the projectile in the y-direction, partly due to gravity. In the case of the MLP model, the final orientation of the projectile is around 55° from the horizontal, and the projectile appears to fall back and sink into the sand, due to the artificial gravity induced by the centrifuge motion, after its initial momentum has been exhausted.

A more direct visualization of the positions and orientations of the UXO projectile can be observed from Figure 26. The images show the velocity magnitudes in the projectile and the sand 850μs after impact. The velocity field and projectile orientations are nearly identical in the Table and SVR predictions. The white color of the velocity field of the projectile indicates that its velocity is close to zero in all three simulations. However, in the MLP predictions, the projectile appears to be sinking into the sand, and that motion generates a different velocity field on the side where the projectile touches the crater wall.

One observation from these simulations is that the material model does not appear to affect the predicted depth during the phase where the motion is dominated by the initial momentum of the projectile. It is only after the projectile loses most of its momentum that the differences between the three material models exhibit themselves as differences in the predicted depth. This suggests that rate effects may not be important for high-speed penetration simulations of granular materials.

Plots of the mean stress in the soil particles at three snapshots in time (t = 200 μs, 400 μs, and 600 μs) for the three models are shown in Figure 27. To highlight stress differences, a logarithmic scale has been used to color the sand particles. Though the initial wave speed is faster in the MLP model (see the snapshots at 200 μs), as the projectile moves through the soil, the mean stress distribution is similar for the three models. The shape and orientation of the crater is also very similar for the three



(a) Table model (b) SVR model (b) MLP model

Figure 26 – *Orientation and depth of the UXO projectile at t = 850 μs after impact. Soil particles have been colored based on velocity magnitude.*

models. Though the Table and SVR models predict almost identical mean stresses, slight differences can be observed between the SVR and MLP models near the projectile tip (green and yellow regions) where the pressure varies between 50 MPa and 80 MPa.

The evolution of the effective deviatoric stress, $q = \sqrt{3J_2}$, can be observed from Figure 28. The initial effective stress near the tip of the projectile is around 100 MPa, and the shear wave speed is larger in the MLP model. At 400 μs after impact, all three models appear to produce the same effective stress field with marginal differences observed in the MLP model. Differences between the SVR and MLP model appear after around 600 μs. Stresses appear to relax in the sand above the projectile as it sinks into the sand due to the artificial gravity caused by the centrifuge motion.

It is also informative to examine the evolution of the volumetric plastic strain in the soil during the impact event, as shown in Figure 29. At 200 μs, the volumetric plastic strain in the sand at the tip of the projectile is around 17% for all three models, with the SVR model predicting slightly larger values than the Table model and the MLP model predicting the largest values. There is a region (red) with around 7% strain and a larger region (blue) with approximately 3% strain. The high-shear region forming the surface of the crater has large tensile volumetric plastic strains. Note that these strains are nonphysical but can be modeled with an appropriate disaggregation model.

At 400 μs, the red 7% plastic strain region is larger for the MLP model, and the peak plastic strain is around 14% for that model while it is around 12% for the Table and MLP models. At 600 μ, the Table and MLP models predict volumetric plastic strain peaks of around 13.5% while the corresponding MLP value is 16.5%. The location of the red (7% strain) region also shifts in the MLP simulation to the side of the crater bottom. Further study is needed to determine the causes that contribute to that difference between the Table (or SVR) model and the MLP model.

5.1 Timing and memory usage

Since the Table and SVR models produce almost identical results, we can use the time taken and memory used in the simulations to select the preferable model. The simulations in this work were performed on a four-core Intel i7-4770 CPU with a peak clock speed of 3.4 GHz, 256K of L2 cache, and 8192K of L3 cache. Vaango , compiled with gcc 7.5.0 and O3 optimization activated, was used

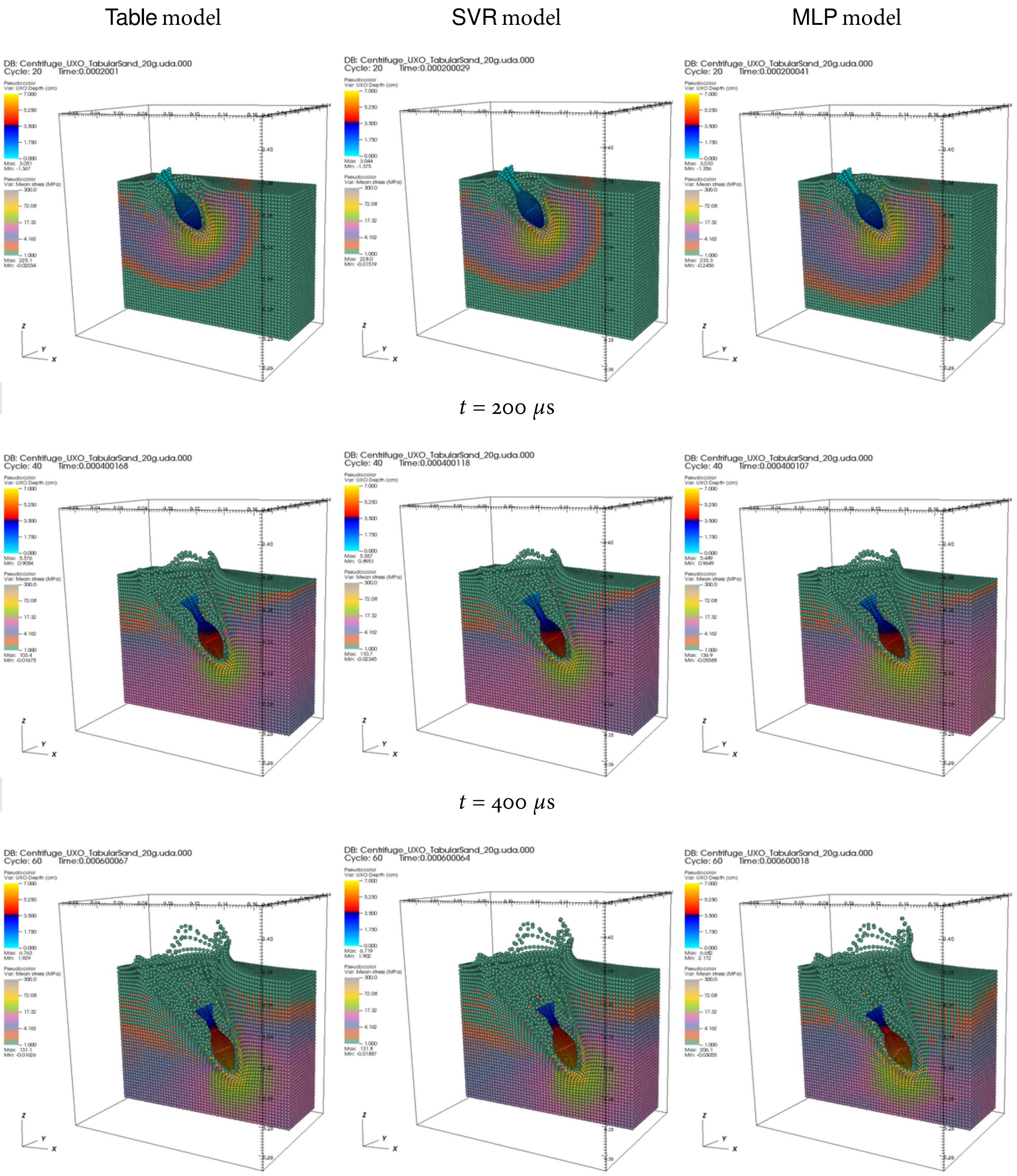


Table model
SVR model
MLP model
DB: Centrifuge_UXO_TabularSand_20g.uda.000
Cycle: 20 Time:0.0002001
Cycle: 20 Time:0.000200029
Cycle: 20 Time:0.000200041
Cycle: 40 Time:0.000400168
Cycle: 40 Time:0.000400118
Cycle: 40 Time:0.000400107
Cycle: 60 Time:0.000600067
Cycle: 60 Time:0.000600064
Cycle: 60 Time:0.000600018
Pseudocolor
Var: UXO Depth (cm)
7.000
5.250
3.500
1.750
0.000
Pseudocolor
Var: Mean stress (MPa)
300.0
72.08
17.32
4.162
1.000
Z
Y
X

t = 200 *μ*s

t = 400 *μ*s

t = 600 *μ*s

Figure 27 – *Mean stress evolution predicted for the* Table, SVR, *and* MLP *models during UXO penetration into a dry, poorly-graded, concrete sand.*

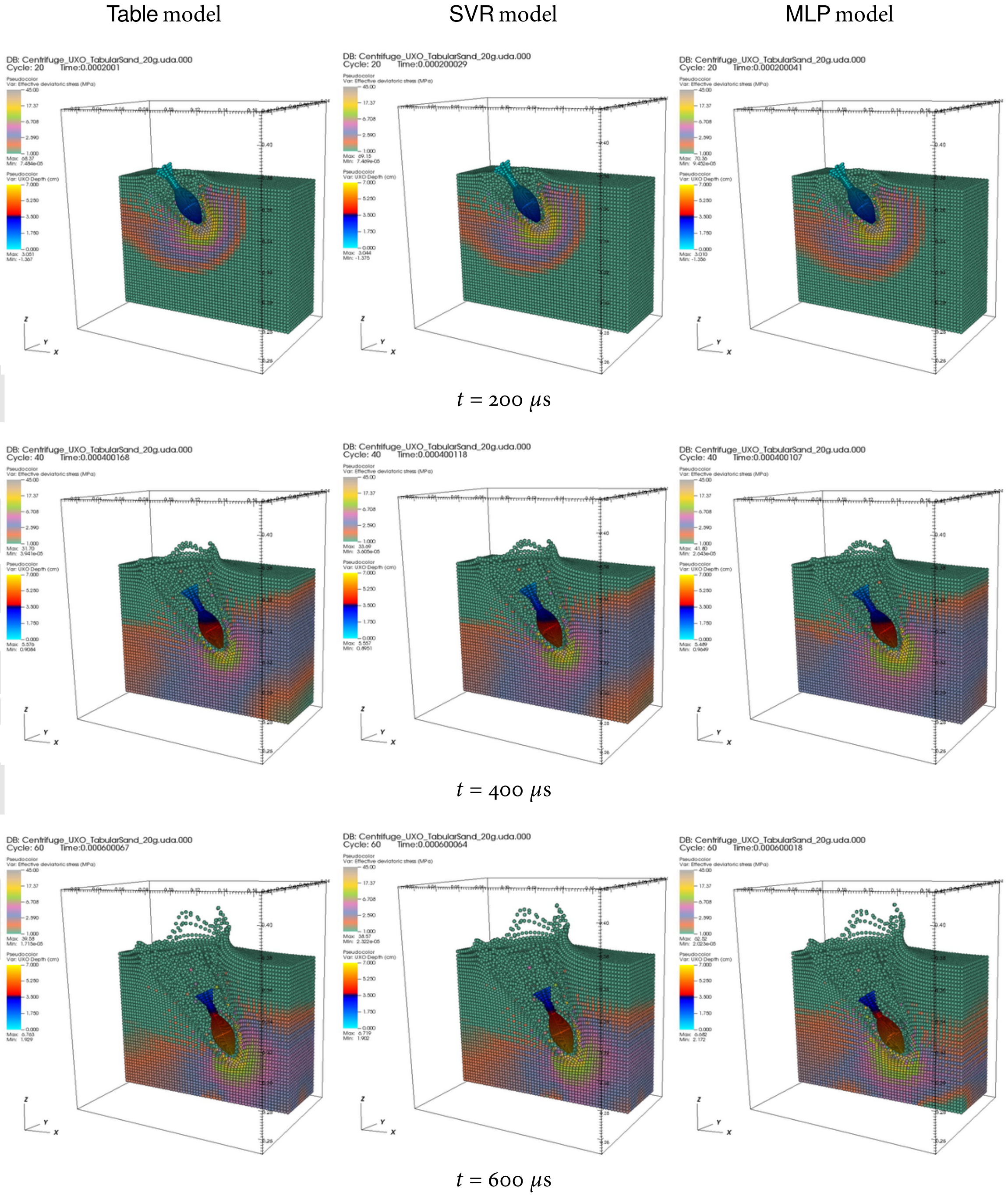


Table model
SVR model
MLP model
DB: Centrifuge_UXO_TabularSand_20g.uda.000
Cycle: 20 Time:0.0002001
Cycle: 20 Time:0.000200029
Cycle: 20 Time:0.000200041
Pseudocolor
Var: Effective deviatoric stress (MPa)
Var: UXO Depth (cm)
t = 200 μs
Cycle: 40 Time:0.000400168
Cycle: 40 Time:0.000400118
Cycle: 40 Time:0.000400107
t = 400 μs
Cycle: 60 Time:0.000600067
Cycle: 60 Time:0.000600064
Cycle: 60 Time:0.000600018
t = 600 μs

Figure 28 – *Evolution of the effective deviatoric stress ($q = \sqrt{3J_2}$) predicted for the Table, SVR, and MLP models during UXO penetration into a dry, poorly-graded, concrete soil.*

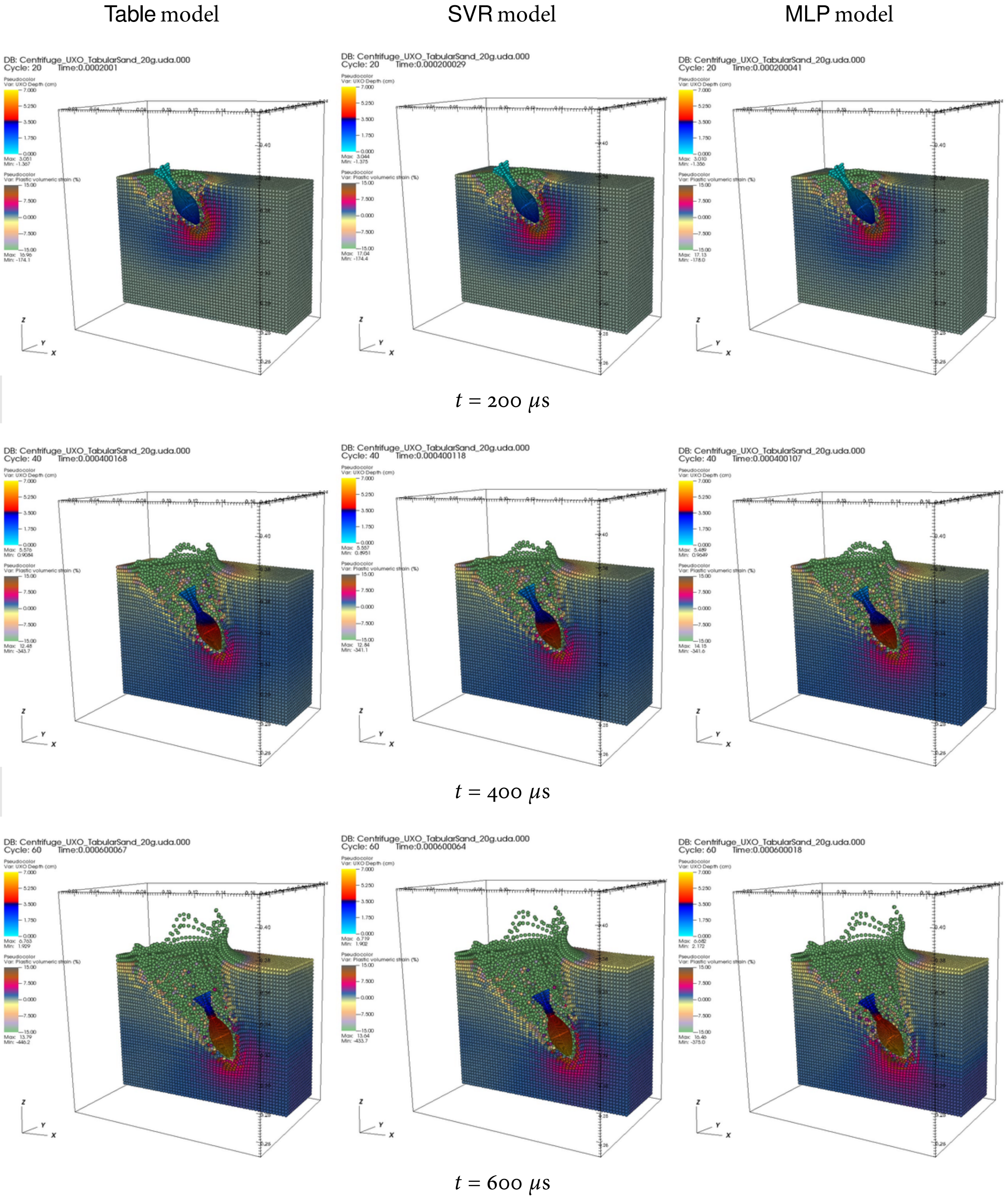


Table model
SVR model
MLP model
DB: Centrifuge_UXO_TabularSand_20g.uda.000
Cycle: 20 Time:0.0002001
Cycle: 20 Time:0.000200029
Cycle: 20 Time:0.000200041
Pseudocolor
Var: UXO Depth (cm)
Pseudocolor
Var: Plastic volumetric strain (%)
z
y
x
Cycle: 40 Time:0.000400168
Cycle: 40 Time:0.000400118
Cycle: 40 Time:0.000400107
Cycle: 60 Time:0.000600067
Cycle: 60 Time:0.000600064
Cycle: 60 Time:0.000600018

$t = 200\ \mu s$

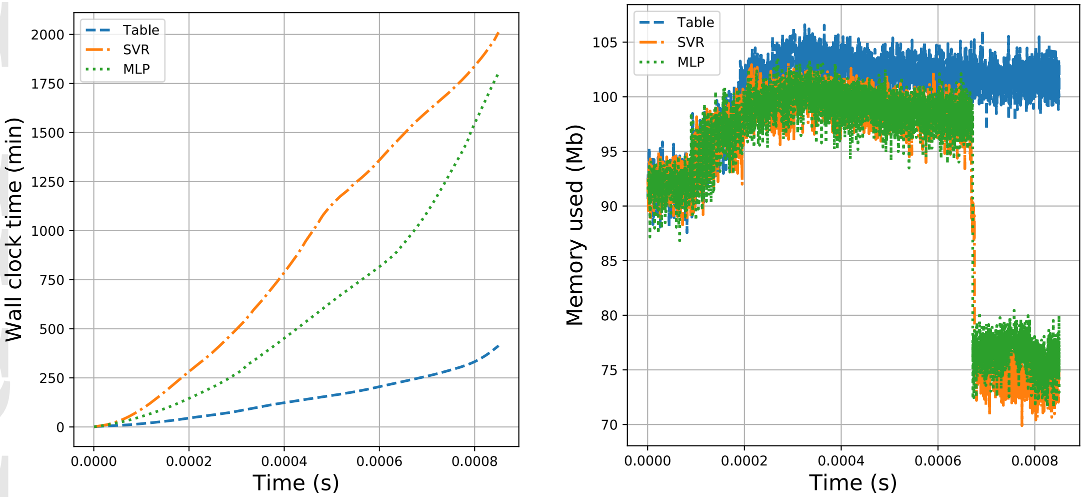
$t = 400\ \mu s$

$t = 600\ \mu s$

Figure 29 – *Volumetric plastic strain evolution predicted for the Table, SVR, and MLP models during UXO penetration into a dry, poorly-graded, concrete soil.*

to run the simulations. The code was run on the four cores in parallel using mpirun .

Timing and memory usage data for the three simulations are plotted in Figure 30. From the time taken for the simulation to run in Figure 30(a), we observe that the Table model is the fastest while the SVR model is the slowest (slower than Table by almost an order of magnitude). Though the MLP model is faster than the SVR model, it is still considerably slower than the Table model, suggesting that the Table model should be used whenever possible. The memory usage, in Figure 30(b), shows an increase as the yield surface expands in particles around the projectile until it reaches an approximately steady value of around 100 Mb. The SVR and MLP models require a smaller amount of memory than the Table model. There is a sharp drop in the memory requirement for these models after the particle velocity drops to its minimum value. These results indicate that the suspected memory leak discussed in Banerjee (2020) is probably just an artifact of the prescribed deformation path.



(a) Total simulation time.

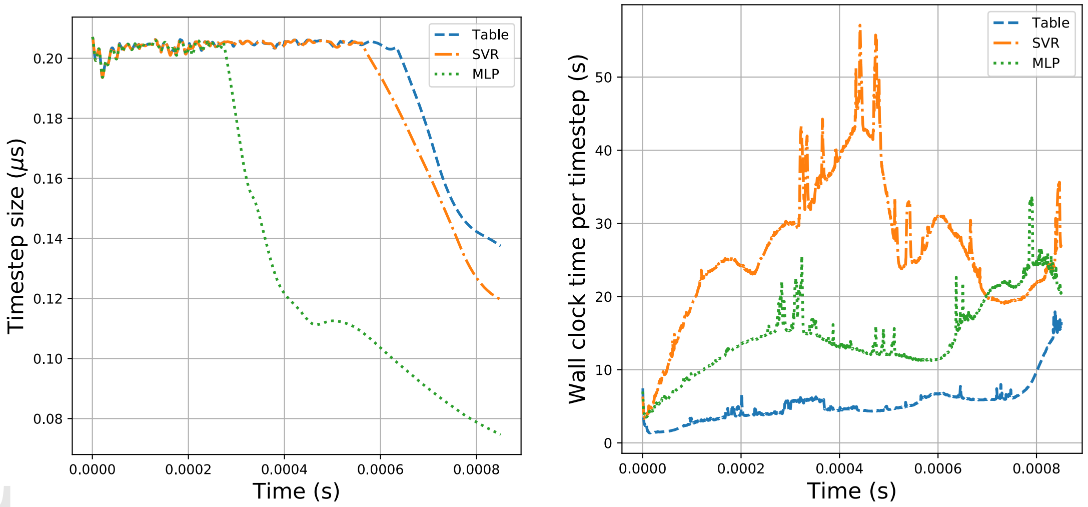
(b) Memory usage.

Figure 30 – *Timing and memory usage for the three UXO simulations.*

The timestep size and the time taken to run each timestep of the simulation are shown in Figures 31(a) and (b), respectively. The Table and SVR models use identical timesteps for a large fraction of the simulation. However, the MLP model uses a smaller timestep starting at around 300 μs. Even though the MLP model uses a smaller timestep than the SVR model, the use of ReLU activation leads to a more efficient evaluation of the model as seen in the time per timestep plot in Figure 31(b). The Table model is again the most efficient of the three.

6 Concluding remarks

In this paper, we have explained the choice of functions used in the interpolation and projection of the Material Point Method (MPM), and provided a more accurate way of integrating the velocity gradient than used in most implementations of the method. In conjunction with the material stress update algorithm discussed in Banerjee, Fox, and Regueiro (2020d), this method provides a way of simulating sand and other granular materials when the input data are provided in tabular form. The



(a) Timestep size (b) Time per timestep

Figure 31 – *Timestep size and time per timestep for the UXO simulations.*

single-particle tests described in Banerjee (2020) had examined the response of three types of model for the elastic moduli: an interpolated model (Table), a support vector regression model (SVR), and a multilayer perceptron (MLP) neural network model. In this paper, we have compared MPM predictions for the three models in the context of a two-dimensional punch impact problem and a three-dimensional unexploded ordnance (UXO) penetration problem.

Our results show that, because the large compressive stresses observed in Banerjee (2020) are not attained in the MPM simulations, the differences between the predictions of the three models are not very large. However, differences do exist. The MLP model predicts a lower depth of penetration for the punch simulation test than the Table or SVR models. On the other hand, the same material model predicts a larger depth of penetration and a different final orientation in the UXO test than the Table and SVR models. The Table model is the least computationally expensive while the SVR model takes the most time to compute. The MLP model uses ReLU activation units and consumes less computational resource than the SVR model. Tabular cap plasticity models are more expensive than closed-form models because standard root-finding algorithms cannot be used in the return mapping process.

In general, we have observed that differences in material models become more relevant after the initial momentum of the projectile has largely been transferred to the target. The response up to that stage can be predicted using simple models provided the impact velocity is large enough. However, the final stage of the settlement of the projectile depends strongly on the response of the target material at low strain rates. This observation is important for the efficient recovery of unexploded ordnance projectiles.

Acknowledgements

This research has been partially funded by the US Office of Naval Research PTE Federal award number N00014-17-1-2704.

References

Abadi, Martín et al. (2016). "Tensorflow: A system for large-scale machine learning". In: *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283 (cit. on p. 15).

Banerjee, B. (2014). *Vaango User Manual: parallel simulation of high strain-rate solid mechanics with the Material Point Method*. https://github.com/bbanerjee/ParSim/tree/master/Vaango (cit. on pp. 3, 6, 13, 15).

— (2020). *Comparison of tabular, support-vector, and neural network models for granular elastoplasticity*. Tech. rep. PAR-10021867-092020-5. Parresia Research Limited. DOI: 10.13140/RG.2.2.36835.99364 (cit. on pp. 2, 13, 16, 37, 38).

Banerjee, B., D. M. Fox, and R. A. Regueiro (2020a). *Interpolating tabular data for granular material models*. Tech. rep. PAR-10021867-092020-3. Parresia Research Limited (cit. on pp. 13, 16).

— (2020b). *Multilayer perceptron neural networks as multi-variable material models*. Tech. rep. PAR-10021867-092020-2. Parresia Research Limited (cit. on pp. 3, 13, 15, 16).

— (2020c). *Support vector regression for fitting multi-variable material models*. Tech. rep. PAR-10021867-092020-1. Parresia Research Limited (cit. on pp. 3, 11, 13, 14, 16).

— (2020d). *Tabular models for high pressure and high strain-rate granular plasticity*. Tech. rep. PAR-10021867-092020-4. Parresia Research Limited (cit. on pp. 2, 3, 13, 37).

Bardenhagen, S. G. and E. M. Kober (2004). "The Generalized Interpolation Material Point Method". In: *Comp. Model. Eng. Sci.* 5.6, pp. 477–495 (cit. on p. 6).

Bardenhagen, S.G. et al. (2001). "An Improved Contact Algorithm for the Material Point Method and Application to Stress Propagation in Granular Material". In: *Computer Modeling in Engineering and Sciences* 2, pp. 509–522 (cit. on p. 15).

Bless, S et al. (2020). "Rate dependent penetration of dry sand". In: *AIP Conference Proceedings*. Vol. 2272. 1. AIP Publishing LLC, p. 110001 (cit. on p. 2).

Boguslavskii, Y., S Drabkin, and A Salman (1996). "Analysis of vertical projectile penetration in granular soils". In: *Journal of Physics D: Applied Physics* 29.3, p. 905 (cit. on p. 2).

Borg, JP et al. (2013). "In situ velocity and stress characterization of a projectile penetrating a sand target: experimental measurements and continuum simulations". In: *International Journal of Impact Engineering* 51, pp. 23–35 (cit. on p. 2).

Børvik, Tore, Sumita Dey, and L Olovsson (2015). "Penetration of granular materials by small-arms bullets". In: *International Journal of Impact Engineering* 75, pp. 123–139 (cit. on p. 2).

Brannon, R. M. et al. (2015). "KAYENTA: Theory and User's Guide". In: *Sandia National Laboratories report SAND2015-0803* (cit. on p. 11).

Chian, Siau Chen, Beng Chye Vincent Tan, and Anand Sarma (2017). "Projectile penetration into sand: relative density of sand and projectile nose shape and mass". In: *International journal of impact engineering* 103, pp. 29–37 (cit. on p. 2).

Chikatamarla, Ravikiran, Jan Laue, and Sarah M Springman (2006). "Centrifuge scaling laws for guided free fall events including rockfalls". In: *International Journal of Physical Modelling in Geotechnics* 6.2, pp. 15–26 (cit. on p. 2).

Chung, Jae H et al. (2017). *Physics based Prediction of Unexploded Ordnance Penetration in Granular Materials*. Tech. rep. University of Florida, Gainesville, United States (cit. on p. 2).

Collins, AL et al. (2011). "The effect of rod nose shape on the internal flow fields during the ballistic penetration of sand". In: *International Journal of Impact Engineering* 38.12, pp. 951–963 (cit. on p. 2).

Crockford, Douglas (2006). "JSON: The fat-free alternative to XML". In: *http://www. json. org/xml. html* (cit. on p. 13).

Dunatunga, Sachith and Ken Kamrin (2017). "Continuum modeling of projectile impact and penetration in dry granular media". In: *Journal of the Mechanics and Physics of Solids* 100, pp. 45–60 (cit. on p. 2).

Fox, D. M. et al. (2014). "The effects of air filled voids and water content on the momentum transferred from a shallow buried explosive to a rigid target". In: *International Journal of Impact Engineering* 69, pp. 182–193 (cit. on p. 11).

Fragaszy, Richard J and Teresa A Taylor (1989). *Centrifuge modeling of projectile penetration in granular soils*. Tech. rep. ESL-TR-88-76. Engineering, Services Laboratory, Air Force Enegineering, and Services Center, Tyndall Air Force Base, FL (cit. on p. 2).

Garnier, Jacques et al. (2007). "Catalogue of scaling laws and similitude questions in geotechnical centrifuge modelling". In: *International Journal of Physical Modelling in Geotechnics* 7.3, pp. 01–23 (cit. on p. 2).

Holsapple, K. A. and R. M. Schmidt (1982). "On the scaling of crater dimensions: 2. Impact processes". In: *Journal of Geophysical Research: Solid Earth* 87.B3, pp. 1849–1870 (cit. on p. 2).

Holsapple, K et al. (2002). "Asteroid impacts: Laboratory experiments and scaling laws". In: *Asteroids III* 1, pp. 443–462 (cit. on p. 2).

Koranne, Sandeep (2011). "Hierarchical data format 5: HDF5". In: *Handbook of Open Source Tools*. Springer, pp. 191–200 (cit. on p. 15).

Mahdavi, Ashkan, Sheng-Wei Chi, and Mohammed M Atif (2020). "A two-field semi-Lagrangian reproducing kernel model for impact and penetration simulation into geo-materials". In: *Computational Particle Mechanics* 7.2, pp. 351–364 (cit. on p. 2).

O'Loughlin, Conleth D, Mark D Richardson, and Mark F Randolph (2009). "Centrifuge tests on dynamically installed anchors". In: *International conference on offshore mechanics and arctic engineering*. Vol. 43475, pp. 391–399 (cit. on p. 2).

Omidvar, Mehdi, Magued Iskander, and Stephan Bless (2014). "Response of granular media to rapid penetration". In: *International Journal of Impact Engineering* 66, pp. 60–82 (cit. on p. 2).

Omidvar, Mehdi, Jeanne Doreau Malioche, et al. (2015). "Phenomenology of rapid projectile penetration into granular soils". In: *International Journal of Impact Engineering* 85, pp. 146–160 (cit. on p. 2).

Sadeghirad, Alireza, Rebecca M Brannon, and Jeff Burghardt (2011). "A convected particle domain interpolation technique to extend applicability of the material point method for problems involving massive deformations". In: *International Journal for numerical methods in Engineering* 86.12, pp. 1435–1456 (cit. on p. 6).

Schreyer, HL and CP Chiu (1991). "The effects on penetrator deceleration of material features exhibited in soils". In: *International journal of solids and structures* 27.10, pp. 1327–1346 (cit. on p. 2).

Sheng-Wei, Chi (2017). *Meshfree Modeling of Munitions Penetration in Soils*. Tech. rep. SERDP Project MR-2628. University of Illinois, Chicago Chicago United States (cit. on p. 2).

Sulsky, D., A. Chen, and H. L. Schreyer (1994). "A particle method for history-dependent materials". In: *Computer Methods in Applied Mechanics and Engineering* 118, pp. 179–196 (cit. on p. 4).

Sulsky, D., S. Zhou, and H. L. Schreyer (1995). "Application of a particle-in-cell method to solid mechanics". In: *Computer Physics Communications* 87, pp. 236–252 (cit. on p. 6).

Tanaka, Koichi et al. (2011). *Phenomenological studies of the response of granular and geological media to high-speed (Mach 1-5) projectiles*. Tech. rep. AOARD-104115. Chubu Univ. (Japan), Dept. of Mechanical Engineering (cit. on p. 2).

Taylor, Teresa, Richard J Fragaszy, and Carlton L Ho (1991). "Projectile penetration in granular soils". In: *Journal of Geotechnical Engineering* 117.4, pp. 658–672 (cit. on p. 2).

Thompson, LJ (1966). *Dynamic penetration of selected projectiles into particulate media*. Tech. rep. SC-RR-66-376. Sandia Corp, Albuquerque, NM (cit. on pp. 1, 2).

Zhang, Nan and T Matthew Evans (2017). "Offshore anchor penetration in sands—Granular simulations". In: *Geotechnical Frontiers 2017*, pp. 132–142 (cit. on p. 2).