An Implicit Particle-in-Cell Method
for Granular Materials

S. J. Cummins and J. U. Brackbill

Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545
E-mail: scummins@lanl.gov, jub@lanl.gov

An implicit-in-time method for granular materials is described. The method combines the material point method, a first-order contact algorithm, and a Newton–Krylov equation solver to give improved energy conservation, stabilization of the finite-grid-instability, and the correct description of collisions between grains.
© 2002 Elsevier Science (USA)

Key Words: granular flow; finite-difference methods; implicit methods, quadratic conserving; nonlinear stability; Newton–Krylov.

1. INTRODUCTION

Granular material is a common yet interesting and complex form of matter that has engaged the attention of a wide range of disciplines [26]. Physicists and mathematicians are fascinated by simple experiments, such as sand piles, which reveal new phenomena, such as self-organized criticality and stress-bridging, where large fluctuations are a striking feature [23, 24]. Engineers seek to extend continuum mechanics to granular flow to provide practical descriptions of industrial processes, for example, to predict discharge rates from hoppers [10].

Numerical models have been helpful in dealing with the complexities of granular flow and have revealed some of the underlying physics. Among the wide variety of models that have been used are the q model, with which the statistics of force fluctuations in bead packs under static loading are studied [21], and a simple frustrated, Ising lattice gas model [22], which yields a force distribution similar to the q model. Contact dynamics' simulations assume that the individual grains are perfectly rigid and interact through exact Coulombian friction, which introduces a nonlocal character to momentum transfers that requires that all kinematic constraints be simultaneously accounted for in the solution of the dynamics [40]. Other models include Monte Carlo models [25], cellular automata [2], and granular dynamics' simulations using the distinct element method (DEM) [3]. The

force law for interactions among grains in DEM may take various forms, including a hard core potential [29], or a contact law derived from Hertz–Mindlin theory [16]. Finally, there are methods which seek to model the material properties of the grains realistically [38, 44, 45]. These models resolve grain scale lengths, allow grains to deform, and solve continuum equations in which contact and friction have been regularized.

Grain scale models include realistic constitutive models for the individual grains and describe grain deformation under load. There are Eulerian [45] and Lagrangean [38] models, which differ in the way they treat interactions between grains. In the Eulerian models, grain interactions are treated using a mixture theory in which continuity of either strain or stress is assumed [46]. In Lagrangean models, as in the material point method (MPM) based on the particle-in-cell method (PIC), grain interactions are modeled by a contact algorithm. In MPM, regularization is provided by shape functions, which are associated with each particle and provide interpolation of grid data.

MPM combines a Lagrangean description, using material points, with an Eulerian grid. The material points resolve edges, support a multiple valued velocity so that sliding can be modeled, eliminate the numerical diffusion that would ordinarily occur as a grain moves, and allow history-dependent effects such as plastic strain to be recorded. The Eulerian grid supports efficient computation of interactions among the material points. Contacts between grains are computed with a model based on the immersed boundary method [15], which forbids interpenetration, but which allows grains to separate, to slide past one another, and to bond [38].

MPM has been used to study the complex behavior of granular samples with realistic material properties under applied compression and shear. Under compression [39], static loading results in stress bridging in polydisperse samples, and dynamic loading results in a two-wave structure with stress fingering propagating ahead of a normal compaction wave. In polydisperse samples with a stiff viscoelastic binder, stress fingering is suppressed. Under shear [37], inclined columns of highly stressed particles form, dissipate, and re-form with increasing strain, similarly to experimental observations [19] and theoretical studies [16]. Sheared samples exhibit very large stress fluctuations, with an exponential probability distribution for stresses above the mean as reported by Liu *et al.* and Radjai *et al.* [4, 21], but limited by the plastic yield stress of the material being modeled. There is a interest in introducing even more realistic material properties into MPM, especially to model damage due to brittle fracture at high strain rates or large strains.

Here, an implicit formulation of the material point method is considered [1, 8, 32]. Previous versions of MPM advance the solution in time by means of a leapfrog algorithm. These work well for relatively high strain rates. However, for quasi-static granular problems implicit methods are potentially more efficient [12]. A robust and efficient MPM has many applications in the study of the response of granular materials to low strain rates, such as experiments on nonlinear stiffening [13] or shear cell experiments [19]. An important byproduct of the implicit formulation is the elimination of aliasing instabilities and a more faithful representation of the physics. This is confirmed by the analysis and numerical experiments in this paper.

The solution of the implicit MPM model presents some special challenges. Some of these come from the physics of the model, which incorporates a model for contact among grains with friction. The standard Coulomb friction model includes discontinuities that require unconventional mathematics to describe them [40]. Other challenges are a consequence of

the material point description of the material, in which important data is distributed among a very large number of particles.

A Newton–Krylov (NK) algorithm is applied to the solution of the implicit MPM model [41]. In the past decade, Newton–Krylov methods have been applied to a wide variety of challenging problems; see, for example, [53, 54, 59]. A primary advantage of the Newton–Krylov implementation is that it can be matrix-free [42]; the matrix is required only in the form of a matrix–vector product. The application of the solver is therefore straightforward. It is easy to incorporate into the solution of large systems of nonlinear equations, and it requires only the evaluation of the remaining residual error in the solution at each iterative step. In practice, however, preconditioning the linear problem is essential for the overall efficiency of the algorithm and the choice of preconditioner must be tailored to the requirements of each application. A notable trend in the literature is the use of multigrid methods as preconditioners [43]. Our experience with a multigrid preconditioner is reported.

In the following sections, the granular flow model is reviewed, the implicit formulation of MPM is described and the linear and nonlinear stability of the implicit MPM method is analyzed. Next, the application of the Newton–Krylov technique is described, which encompasses the evaluation of the nonlinear residual and preconditioning. The results of numerical experiments for a single grain, an assembly of grains under compression, and a monodisperse assembly of grains that is subjected to large shear deformation are presented and discussed.

2. A MODEL FOR GRANULAR MATERIAL WITH DEFORMABLE GRAINS

Granular material is modeled as a collection of deformable grains [38]. The dynamics of each grain, g, are described by the standard equations of continuum mechanics, which include mass continuity, stress and strain evolution, and momentum equations,

$$\frac{d\rho_g}{dt} = -\rho_g \nabla \cdot \mathbf{v}_g, \tag{1}$$

$$\frac{d\sigma_g}{dt} = \mathbf{T}_g : \frac{d\mathbf{e}_g}{dt}, \tag{2}$$

$$\frac{d\mathbf{e}_g}{dt} = \frac{1}{2}\left(\nabla \mathbf{v}_g + \nabla \mathbf{v}_g^T\right). \tag{3}$$

$$\rho_g \frac{d\mathbf{v}_g}{dt} = \nabla \cdot \sigma_g + \sum_{g'} \rho_g \mathbf{f}_{gg'}, \tag{4}$$

where ρ_g is the density, $\mathbf{v}_g$ is the velocity, σ_g is the stress, $\mathbf{T}_g$ is the tangent modulus, $\mathbf{e}_g$ is the strain, and $\mathbf{f}_{gg'}$ is the specific contact force exerted on grain g by grain g'.

The grains are modeled as isotropic, linearly elastic-plastic materials, and the strain is decomposed into elastic and plastic parts,

$$\mathbf{e} = \mathbf{e}_e + \mathbf{e}_{pl}. \tag{5}$$

As is described in detail in Sulsky *et al.* [8], for computational convenience the plastic strain is computed using a radial return rule, or normal to the yield surface in stress space

for isotropic materials,

$$\frac{d\mathbf{e}_{pl}}{dt} = \frac{d\lambda}{dt} \frac{\sigma_g}{|\sigma_g|}. \tag{6}$$

The parameter, λ, is monotone increasing and is calculated to satisfy a consistency condition which requires that the stress equal the current yield stress, $|\sigma_g| = |\sigma_{yield}|$, as computed from a strain hardening model.

When the grains are held together by a binder, the binder is modeled as viscoplastic material [36, 39].

In the absence of friction or bonding, the contact force prevents interpenetration of grains but allows them to move apart freely. When contact between two or more grains occurs, interpenetration will not occur if the velocity for each grain at the point of contact, $\mathbf{x}_c$, satisfies the inequality,

$$\hat{\mathbf{n}}_g \cdot (\mathbf{v}_g - \mathbf{v}_{g'}) \leq 0, \tag{7}$$

where $\hat{\mathbf{n}}_g$ is the outward directed surface normal for grain g. In general, this is a very expensive constraint to satisfy, because the computational effort scales as the square of the number of grains. A constraint whose cost scales as the number of grains is derived by multiplying Eq. (7) by $\rho_{g'}$ and summing over g',

$$\hat{\mathbf{n}}_g \cdot (\mathbf{v}_g - \mathbf{v}) \leq 0, \tag{8}$$

where $\mathbf{v}$ is the Fabre-averaged velocity at the point of contact,

$$\mathbf{v} = \frac{\sum_g \rho_g \mathbf{v}_g}{\sum_g \rho_g}. \tag{9}$$

The constraints given by Eqs. (7) and (8) are equivalent when two grains are in contact. We will assume no more than two grains are in contact at any given time and replace $\mathbf{f}_{g,g'}$ by $\mathbf{f}_g = \sum_{g'} \mathbf{f}_{g,g'}$. The summation includes all grains in contact at $\mathbf{x}_c$. Imposed strain boundary conditions are imposed through the contact force by specifying $\mathbf{v}$ on the boundary.

It is assumed that the unit normals for grains in contact satisfy

$$\hat{\mathbf{n}}_g \cdot \hat{\mathbf{n}}_{g'} = -1, \quad g \neq g'. \tag{10}$$

It is required that

$$\rho_g \mathbf{f}_{gg'} + \rho_{g'} \mathbf{f}_{g'g} = 0 \tag{11}$$

for any g and g', so that momentum is conserved. More complex contact forces include friction and bonding. Coulomb friction adds a component to $\mathbf{f}_g$, in the plane formed by the relative velocity and the surface normal that acts to resist sliding at the point of contact. Bonding prevents all relative motion. Debonding occurs if the normal traction exceeds the bond strength, σ_{bond},

$$\hat{\mathbf{n}}_g \cdot \sigma_g \cdot \hat{\mathbf{n}}_g \geq \sigma_{bond} \tag{12}$$

and Eq. (7) is satisfied.

3. AN IMPLICIT MPM METHOD FOR GRANULAR FLOW

The material point method is a particle-in-cell method with many of the similarities to the fluid-implicit-particle method (FLIP) [35]. Like FLIP, MPM combines two meshes, one a material or Lagrangean mesh defined by a distribution of mass or material points, the other a logically rectangular computation grid, which covers the computational domain. On these two grids, the equations of motion are integrated in two phases. In the first phase, the dynamical equations are solved in Lagrangean form with both the material and computation grids moving with the local fluid velocity. In the second phase, a new computation grid is defined, and the particles, which are stationary in this phase, are relocated on the new grid. The advantages of a dual material-point/computation-grid representation are the Galilean invariance of the numerical equations, the absence of numerical diffusion, the interface tracking afforded by the material points, and the ease with which history-dependent materials can be modeled.

There are also significant differences between FLIP and MPM as well. In FLIP, the particle data is interpolated to the computation grid using particle shape functions, then differentiated. In MPM, the particle shape functions are differentiated, then the derivatives are used to interpolate particle data to the computation grid [8]. FLIP's approach requires that the conjugacy relations needed for conservation be provided by operators on the grid [6] and allows considerable freedom in choosing smooth interpolation formulae for some variables and not for others. MPM's approach requires a consistent choice for all interpolation formulae, because the conjugacy relations must be satisfied by the derivatives of the particle shape functions [8]. (Further information on this issue can be found in Birdsall and Langdon [9, pp. 213–232].) Since consistency is most easily satisfied using low-order b-splines, MPM typically uses linear b-splines.

The most pertinent difference between MPM and FLIP for the present discussion is that FLIP and its progeny have all used time-implicit difference equations, while MPM uses an explicit leapfrog algorithm. In this section, we review MPM and outline an implicit formulation. Our discussion focuses on the first or Lagrangean phase, during which the grid and particle variables are updated.

3.1. Defining the Solution Space of MPM

In MPM, the material data is carried on Lagrangean mass points or particles. To each mass point p, there is assigned essential information, such as the position, $\mathbf{x}_p$, mass, m_p, velocity, $\mathbf{v}_p$, stress σ_p, and volume, V_p. The material points move through a logically rectangular grid of dimension d, whose vertices are located at $\mathbf{x}_i$. Each cell of the rectangular grid is mapped onto a unit square or cube, $\xi^l \in [0, 1], l = 1 \ldots d$. The mapping we choose is a bilinear (trilinear) mapping using b-splines, s, of linear order. The mapping is invariant as the computation grid moves with the flow,

$$\frac{ds(\xi)}{dt} = 0, \tag{13}$$

and maps cell edges on the logically rectangular grid onto cell edges on the logical grid. The b-splines are positive functions with bounded support. For interpolation in more than

one variable, a tensor product of b-splines is used,

$$s(\xi) = \prod_{l=1}^{d} s(\xi^l). \tag{14}$$

At the beginning of the first phase of each time step, the logical coordinates, ξ^l are defined within each cell by the mapping between physical and logical coordinates,

$$\mathbf{x}(\xi) = \sum_j \mathbf{x}_j s(\xi - \xi_j). \tag{15}$$

Because $ds/dt = 0$ in the first phase, the velocity, the acceleration, and the specific contact force are interpolated from the grid by

$$\mathbf{v}_g(\xi) = \sum_j \mathbf{v}_{gj} s(\xi - \xi_j), \tag{16}$$

$$\frac{d\mathbf{v}(\xi)}{dt} = \sum_j \frac{d\mathbf{v}_j}{dt} s(\xi - \xi_j). \tag{17}$$

$$\mathbf{f}(\xi) = \sum_j \mathbf{f}_j s(\xi - \xi_j). \tag{18}$$

The rate of strain experienced by each mass point is calculated by differentiating the velocity,

$$\frac{d\mathbf{e}_p}{dt} = \frac{1}{2} \sum_i [\nabla_{pi}\mathbf{v}_i + (\nabla_{pi}\mathbf{v}_i)^T], \tag{19}$$

where ∇_{pi} gives the value of the derivative of nodal data at the particle,

$$\nabla_{pi} = \sum_{\alpha=1}^{d} \nabla\xi^\alpha \frac{\partial}{\partial\xi^\alpha} s\left(\xi^\alpha - \xi_i^\alpha\right)\Big|_{\xi^\alpha = \xi_p^\alpha}. \tag{20}$$

The mass density, $\rho(\mathbf{x})$, and stress are given by

$$\rho(\mathbf{x}) = \sum_p m_p \delta(\mathbf{x} - \mathbf{x}_p), \tag{21}$$

$$\sigma(\mathbf{x}) = \sum_p \sigma_p V_p \delta(\mathbf{x} - \mathbf{x}_p), \tag{22}$$

where

$$\delta(\mathbf{x} - \mathbf{x}_p) \equiv \prod_{l=1}^{d} \delta\left(x_i^l - x_p^l\right). \tag{23}$$

3.2. Spatial Differencing with MPM

A formulation of MPM in weak form is given in [8], but we will follow the development given in Burgess *et al.* [48] to derive an approximation for spatial derivatives. We begin by

calculating the integral of the inner product of the velocity and the momentum equation, Eq. (4),

$$\int_D \mathbf{v} \cdot \left[\rho \frac{d\mathbf{v}}{dt} - \nabla \cdot \sigma - \rho \mathbf{f} \right] d^3 x. \tag{24}$$

The first term in the integral can be written, using Eqs. (16), (17), and (21), and integrating

$$\frac{dK}{dt} = \sum_{i,j} m_{ij} \mathbf{v}_i \cdot \frac{d\mathbf{v}_j}{dt}, \tag{25}$$

$$m_{ij} = \sum_p m_p s(\xi_i - \xi_p) s(\xi_j - \xi_p). \tag{26}$$

Since the mass matrix, m_{ij}, is a positive, symmetric matrix, the kinetic energy in Eq. (25) is positive.

Following [48], the node velocity is computed from the particle momenta by inverting the mass matrix,

$$\sum_j m_{ij} \mathbf{v}_j = \sum_p m_p \mathbf{v}_p s(\xi_i - \xi_p). \tag{27}$$

If one compares the kinetic energy, K, with the mass-point kinetic energy,

$$k = \sum_p \frac{1}{2} m_p \mathbf{v}_p^2, \tag{28}$$

it is easily shown that $K \leq k$. However, $dK/dt = dk/dt$, as is easily verified from Eqs. (17) and (27).

The material stress in Eq. (24) becomes

$$\int_D d^3 x' \mathbf{v} \cdot \nabla \cdot \sigma = \int_D d^3 x' \sum_i \mathbf{v}_i s(\xi' - \xi_i) \cdot \nabla_{x'} \cdot \sum_p \sigma_p V_p \delta(\mathbf{x}' - \mathbf{x}_p). \tag{29}$$

The integral appearing in this equation,

$$\int_D d^3 x' s(\xi' - \xi_i) \nabla_{x'} \delta(\mathbf{x}' - \mathbf{x}_p) = \nabla_x \xi^\alpha |_{x=x_p} \frac{\partial}{\partial \xi^\alpha} s(\xi^\alpha - \xi_i^\alpha)|_{\xi^\alpha = \xi_p^\alpha} = -\nabla_{pi}, \tag{30}$$

gives the value of the derivative of a particle variable at a grid node. The minus sign occurs because this operator is conjugate to the operator defined in Eq. (20).

For the contact force, we use Eqs. (16) and (18) to derive

$$\int_D d^3 x' \mathbf{v} \cdot \rho \mathbf{f} = \sum_{i,j} \mathbf{v}_i \cdot m_{ij} \mathbf{f}_j. \tag{31}$$

Combining Eqs. (25), (29), and (31), Eq. (24) can be written

$$\sum_i \mathbf{v}_i \cdot \left[\sum_j m_{ij} \frac{d\mathbf{v}_j}{dt} + \sum_p \nabla_{pi} \cdot \sigma_p V_p - \sum_j m_{ij} \mathbf{f}_j \right] = 0. \tag{32}$$

Since this equation holds for arbitrary $\mathbf{v}_i$, the term in brackets must be identically zero. This is the grid momentum equation,

$$\sum_j m_{ij} \frac{d\mathbf{v}_j}{dt} = -\sum_p \nabla_{pi} \cdot \sigma_p V_p + \sum_j m_{ij} \mathbf{f}_j. \tag{33}$$

The particle stress is calculated by evaluating Eq. (2) for each mass point using Eq. (19),

$$\frac{d\sigma_p}{dt} = \mathbf{T} : \frac{d\mathbf{e}_p}{dt}. \tag{34}$$

3.3. Implicit (and Explicit) MPM

The complete implicit (and explicit) difference equations one uses to advance the solution from t to $t + \Delta t$ are listed in this section. (For simplicity, we take Δt to be constant, even though it is not required by the analysis or in practice.)

A computational time step begins with the calculation of the natural coordinates of each particle. These coordinates are calculated from Eq. (15).

$$\mathbf{x}_p^0 = \sum_j \mathbf{x}_j^0 s\left(\xi_p^0 - \xi_j^0\right), \tag{35}$$

where the superscript 0 refers to the data at the beginning of any computation cycle, and 1 at the end.

Next, one computes the grid velocity, $\mathbf{v}_i^0$, from the particle data.

$$\sum_j m_{ij}^0 \mathbf{v}_j^0 = \sum_p m_p \mathbf{v}_p^0 s\left(\xi_i^0 - \xi_p^0\right). \tag{36}$$

Usually, a lumped mass matrix, m_i, is substituted for m_{ij},

$$m_i = \sum_j m_{ij}. \tag{37}$$

This introduces dissipation so that $k^1 - k^0 < K^1 - K^0$, but eliminates the need to invert a mass matrix [35].

The implicit form of the rate of strain, particle volume, stress evolution, and momentum equations is

$$\mathbf{e}_p^1 - \mathbf{e}_p^0 = \frac{1}{2} \sum_i \left[\nabla_{pi} \mathbf{v}_i^\theta + \nabla_{pi} \mathbf{v}_i^{\theta T}\right] \Delta t, \tag{38}$$

$$V_p^1 - V_p^0 = -V_p^0 \sum_i \left[\nabla_{pi} \cdot \mathbf{v}_i^\theta\right] \Delta t, \tag{39}$$

$$\sigma_p^1 - \sigma_p^0 = \mathbf{T} : \left(\mathbf{e}_p^1 - \mathbf{e}_p^0\right), \tag{40}$$

$$m_i \frac{\mathbf{v}_i^1 - \mathbf{v}_i^0}{\Delta t} = -\sum_p \nabla_{pi} \cdot \left(\sigma_p^\theta V_p^\theta\right) + m_i \mathbf{f}_i^\theta, \tag{41}$$

where $\mathbf{f}_i$ is the value of the specific force of constraint at vertex i, and $\frac{1}{2} \leq \theta \leq 1$.

Finally, one updates the particle positions and velocities from the grid solution,

$$\mathbf{x}_p^1 - \mathbf{x}_p^0 = \sum_i \mathbf{v}_i^\theta s \left(\xi_i^0 - \xi_p^0 \right) \Delta t, \tag{42}$$

$$\mathbf{v}_p^1 - \mathbf{v}_p^0 = \sum_i \left(\mathbf{v}_i^1 - \mathbf{v}_i^0 \right) s \left(\xi_i^0 - \xi_p^0 \right). \tag{43}$$

An implicit solution requires that Eqs. (38) through (41) be solved self-consistently.

One can compare the implicit θ scheme above with the explicit leapfrog scheme used in previous versions of the MPM granular flow model [38]. Corresponding to the implicit continuity equation, Eq. (39), the explicit continuity equation is

$$V_p^{1/2} - V_p^{-1/2} = -V_p^{-1/2} \sum_i \left[\nabla_{pi} \cdot \mathbf{v}_i^0 \right] \Delta t. \tag{44}$$

Corresponding to the implicit stress, Eq. (40), the explicit leapfrog stress equation is

$$\sigma^{1/2} - \sigma^{-1/2} = \mathbf{T} : \left(\mathbf{e}^{1/2} - \mathbf{e}^{-1/2} \right), \tag{45}$$

where, corresponding to Eq. (38), the explicit equation for the strain increment is

$$\mathbf{e}^{1/2} - \mathbf{e}^{-1/2} = \frac{1}{2} (\nabla \mathbf{v}^0 + \nabla \mathbf{v}^{0T}) \Delta t. \tag{46}$$

Finally, corresponding to the implicit momentum equation, Eq. (41), the explicit momentum equation is

$$m_i \frac{\mathbf{v}_i^1 - \mathbf{v}_i^0}{\Delta t} = -\sum_p \nabla_{pi} \cdot \left(\sigma_p^{1/2} V_p^{1/2} \right) + m_i \mathbf{f}_i^0. \tag{47}$$

3.4. The Contact Force

Points of contact between grains are detected by comparing the grain and center of mass velocities. Since $\mathbf{v}_{gi} \neq \mathbf{v}_i$ only when more than one grain contributes to the center of mass velocity at a grid node, every point where this condition holds is a contact point where two grains overlap. The outward-directed normal to the grain's surface is given by the gradient of the density,

$$\mathbf{n}_g = -\nabla \rho_g. \tag{48}$$

(Even though the density may vary in the interior of a grain, interior normals cannot be contact points, so density is a suitable characteristic function.)

At points of contact, the constraint which prevents interpenetration, Eq. (8), is replaced by

$$\hat{\mathbf{n}}_g \cdot \left(\tilde{\mathbf{v}}_g^\phi - \tilde{\mathbf{v}}^\phi \right) \leq 0, \tag{49}$$

where $\phi = 1$ for the leapfrog scheme, $\phi = \theta$ for the implicit scheme, and $\tilde{\mathbf{v}}_g$ is the solution to Eq. (47) or (41) with $\mathbf{f} = 0$. One notes that $\mathbf{v}^\phi$, the center-of-mass velocity at $(n + \phi)\Delta t$,

does not depend on $\mathbf{f}_g$ because of momentum conservation, Eq. (11). That is,

$$\mathbf{v}_i^\phi = \tilde{\mathbf{v}}_i^\phi = \frac{\sum_g m_{gi} \tilde{\mathbf{v}}_{gi}^\phi}{\sum_g m_{gi}}. \tag{50}$$

When Eq. (49) is not satisfied, the specific contact force $\mathbf{f}_i$ in Eq. (47) for the leapfrog scheme, or Eq. (41) for the implicit scheme, is calculated by requiring equality in Eq. (49), with the result

$$\hat{\mathbf{n}}_g \cdot \mathbf{f}_g \phi \Delta t = -\hat{\mathbf{n}}_g \cdot \left(\tilde{\mathbf{v}}_g^\phi - \tilde{\mathbf{v}}^\phi \right). \tag{51}$$

With Coulomb friction, the contact force becomes

$$\mathbf{f}_g \phi \Delta t = -(\hat{\mathbf{n}}_g + \mu'(\hat{\mathbf{n}}_g \times \hat{\omega}_g))\hat{\mathbf{n}}_g \cdot \left(\tilde{\mathbf{v}}_g^\phi - \tilde{\mathbf{v}}^\phi \right), \tag{52}$$

where

$$\hat{\omega}_g = \frac{\hat{\mathbf{n}}_g \times \left(\tilde{\mathbf{v}}_g^\phi - \tilde{\mathbf{v}}^\phi \right)}{\left| \hat{\mathbf{n}}_g \times \left(\tilde{\mathbf{v}}_g^\phi - \tilde{\mathbf{v}}^\phi \right) \right|}, \tag{53}$$

and

$$\mu' = \min_g \left[\mu, \frac{\left| \hat{\omega}_g \times \left(\tilde{\mathbf{v}}_g^\phi - \tilde{\mathbf{v}}^\phi \right) \right|}{\left| \hat{\omega}_g \cdot \left(\tilde{\mathbf{v}}_g^\phi - \tilde{\mathbf{v}}^\phi \right) \right|} \right], \tag{54}$$

as described in [38].

The contact force is volume weighted at points of contact, with weight,

$$V_{fi} = \frac{\sum_p V_p s(\xi_i - \xi_p)}{V_i}, \tag{55}$$

where V_i is the grid volume at node i, and contributions from all grains are included in the sum over material points. The volume fraction at node i, V_{fi}, is equal to 1 in the interior of a grain, and 0 outside unless two grains overlap. At contact points, V_{fi} varies in proportion to the degree of overlap. As grains separate by a full cell, V_{fi} decreases to 0. As two grains approach, V_{fi} increases through 1 as mass points defining the edge of the grain touch, at which point each grain contributes 1/2 to the weight. When material points interpenetrate, the volume fraction increases to more than 1 causing the grains to repel one another. Volume weighting results in a first-order accurate interaction force [47]. However, it also results in a smaller effective grain size with the same distribution of material points as with the previous version of the contact force without volume weighting [38].

4. THE STABILITY OF IMPLICIT AND EXPLICIT TIME DIFFERENCE APPROXIMATIONS FOR THE GRANULAR FLOW MODEL

The linear and nonlinear stability properties of the implicit MPM equations are examined and compared with the explicit leapfrog MPM equations. The principal results are that standard linear stability analysis shows that implicit differencing in time eliminates the

usual Courant stability limit, but that both explicit and implicit MPM are linearly unstable to the finite grid instability, even for $\Delta t = 0$. Nonlinear stability analysis shows that implicit MPM is stable in the L_2 norm, but that explicit MPM is not. In this analysis, the kinetic and strain energies are the appropriate choice for the L_2 norm.

A well-known method that is stable in the L_2 norm for $\Delta t = 0$ is the marker-and-cell (MAC) scheme for the solution of the incompressible flow equations method with space-centered, forward-time differencing of convection terms. Lilly shows that MAC is mass, momentum, and kinetic energy conserving when $\Delta t = 0$, and that instabilities caused by aliasing will not occur in quadratic conserving schemes such as MAC [20]. He also notes that mass, momentum, and energy conservation is exact with $\Delta t > 0$ if one uses second-order implicit time differencing ($\theta = 1/2$). (For other references and a general discussion of conservation in staggered mesh schemes, see [27].)

4.1. Neumann Stability

Standard Neumann analysis of the linearized equations (in 1D) gives the stability properties of the leapfrog and implicit θ schemes. Assuming plane wave solutions of the form $u(x, t) = \sum_{k,\omega} u_{k,\omega} e^{i(kx - \omega t)}$, where k is the wave number and ω the frequency, the dispersion equation for the leapfrog scheme is given by

$$\sin\left(\frac{\omega \Delta t}{2}\right) = (\pm ks\,\Delta t), \tag{56}$$

where

$$\frac{\Omega \Delta t}{2} = \sin\left(\frac{\omega \Delta t}{2}\right) \tag{57}$$

and $s = \sqrt{\frac{T}{\rho_0}}$ is the shear wave speed. When $ks\,\Delta t \geq 1$, a complex conjugate pair of roots is found, one of which has a negative imaginary part yielding an exponentially growing or unstable mode. For smaller values of Δt, both roots are real and the leapfrog algorithm is stable.

The corresponding dispersion relation for the θ scheme is

$$\Omega = \pm ks\left(1 - i\left(\theta - \frac{1}{2}\right)\frac{\Omega \Delta t}{2}\right), \tag{58}$$

where

$$\frac{\Omega \Delta t}{2} = \tan\left(\frac{\omega \Delta t}{2}\right). \tag{59}$$

When $\theta = \frac{1}{2}$, in which case the differencing is second-order accurate, the solutions are always real and bounded by the Nyquist frequency,

$$\frac{-\pi}{\Delta t} \leq \omega \leq \frac{\pi}{\Delta t}. \tag{60}$$

For $\theta > 1/2$, the imaginary part of Ω is always negative,

$$\mathrm{Im}[\Omega] = \frac{-\left(\theta - \frac{1}{2}\right)ks}{\left(1 + \left(\theta - \frac{1}{2}\right)^2 k^2 s^2\right)}, \tag{61}$$

and the solutions are therefore damped. Thus, the implicit differencing is unconditionally stable, and dissipative for $\theta > \frac{1}{2}$.

4.2. Finite Grid Instability

A PIC calculation typically has many more degrees of freedom than a finite-difference calculation with the same number of grid points, because there are usually several mass points in every cell. In MPM, for example, where every particle has its unique velocity, the grid velocity is restricted to a linear variation between grid points. Consequently, the particle and grid kinetic energies may be very different, even though their momenta are identical.

As is well known, modulations of the particle density which have the same amplitudes at grid points cannot be distinguished when the interactions among particles on the grid are calculated.

Let us consider a simple problem in one dimension. A beam of particles with uniform spacing moves with uniform speed U_0 in the x direction. The y velocity component is perturbed, resulting in a perturbed stress, and a shear wave propagating in the x direction. The implicit in time, Lagrangean equations of motion are

$$\frac{\sigma_p^1 - \sigma_p^0}{\Delta t} + \lambda \sum_j \frac{\partial s(x_j - x)}{\partial x}\bigg|_{x=x_p} v_j^{1/2} = 0 \tag{62}$$

$$\rho_0 \left[\frac{v^1 - v^0}{\Delta t}\right] + \sum_p \frac{\partial s(x - x_p)}{\partial x}\bigg|_{x=x_j} \sigma_p^{1/2} = 0. \tag{63}$$

Substitute a plane wave solution as above in the laboratory frame,

$$-i(\Omega - k_q U_0)\sigma(k_q) + \lambda i k_q s(k_q)v(k_q) = 0 \tag{64}$$

$$-i(\Omega - k_q U_0)\rho_0 v(k_q) + \sum_r i k_r \sigma(k_r)s(k_r) = 0, \tag{65}$$

where $k_q(k_r)$ is the sum of a principal wave number and its qth (rth) harmonic and Ω is given by Eq. (59),

$$k_q = k + q\left(\frac{2\pi}{\Delta x}\right), \quad -\frac{\pi}{\Delta x} \le k \le \frac{\pi}{\Delta x}. \tag{66}$$

A linear dispersion relation is derived by eliminating v in the equations above,

$$\sigma(k_q)[\Omega - k_q U_0]^2 + k_q \lambda s(k_q) \sum_r k_r \sigma(k_r)s(k_r) = 0. \tag{67}$$

Multiply this equation by $i k_q s(k_q)$, and sum over q to derive

$$1 - \frac{\lambda}{\rho_0} \sum_q \frac{k_q^2 s^2(k_q)}{[\Omega - k_q U_0]^2} = 0. \tag{68}$$

This is exactly the same form derived for FLIP in [34] and yields complex roots with exponentially growing modes. The finite-grid instability will occur at all wave numbers in the principal interval for values of $U_0/\sqrt{\frac{\lambda}{\rho_0}} < 0.4$ (when s is a linear b-spline).

Although the only dependence on Δt in Eq. (68) is contained in Ω, and that acts only to shift the resonances, it is noted in [34, p. 491] that, "Even with relatively low-order interpolation, the ringing instability appears not to grow with implicit differencing in time." However, more than implicit time differencing seems to be required for complete suppression of the finite-grid instability. For example, the instability is absent in implicit plasma simulations only over a range of time steps [5]. In the next section, we examine energy conservation with the MPM granular flow model, in preparation for a nonlinear stability analysis in the following section.

4.3. Energy Conservation

Energy conservation is not only an important measure of the quality of the solution of a numerical calculation, but also of its stability. In PIC plasma simulations, for example, there is a strong identification of nonphysical heating with the finite grid instability [9, p. 176]. In this section, we derive expressions for the kinetic and strain energies, as well as closed form expressions for the various dissipation terms.

The total energy, E, is the sum of the kinetic energy,

$$K = \int_D \frac{1}{2} \rho \mathbf{v}^2 \, dV \tag{69}$$

and strain energies,

$$S = \int_D \frac{1}{2} \mathbf{e} : \mathbf{T} : \mathbf{e} \, dV. \tag{70}$$

The tangent modulus has symmetry properties such that the strain energy is quadratic in the strain [33].

For the implicit θ scheme, we derive an energy integral directly from the equations of motion. (There will be contributions from boundary terms, for example, if the boundary strain or stress is imposed, but we will neglect these and assume energy should be a constant of the motion.) The change in kinetic energy per time step (for both the particle and grid kinetic energies) is given by

$$K^1 - K^0 = k^1 - k^0 = \frac{1}{2} \int_{D^1} \rho^1 (\mathbf{v}^1)^2 \, dV - \frac{1}{2} \int_{D^0} \rho^0 (\mathbf{v}^0)^2 \, dV. \tag{71}$$

(As noted in Section 3.2, the changes in the particle and grid kinetic energies are equal; $k^1 - k^0 = K^1 - K^0$.) From the volume evolution equation, Eq. (39), the domain growth is

$$\int_{D^0} \nabla \cdot \mathbf{v}^\theta \, \Delta t \, dV = \int_{D^1} dV - \int_{D^0} dV, \tag{72}$$

which allows one to write

$$k^1 - k^0 = \frac{1}{2} \int_{D^0} \rho^0 ((\mathbf{v}^1)^2 - (\mathbf{v}^0)^2) \, dV. \tag{73}$$

One then substitutes the MPM momentum equation, Eq. (41), for the term in brackets in Eq. (32), with the result,

$$E^1 - E^0 = 0 = \sum_i \mathbf{v}_i^{\frac{1}{2}} \cdot \sum_j m_{ij} \left(\mathbf{v}_j^1 - \mathbf{v}_j^0 \right) + \sum_i \mathbf{v}_i^\theta \cdot \left[\sum_p \nabla_{pi} \cdot \sigma_p^\theta V_p - \sum_j m_{ij} \mathbf{f}_j \right] \Delta t$$

$$+ \sum_{i,j} \left(\theta - \frac{1}{2} \right) \left(\mathbf{v}_i^1 - \mathbf{v}_i^0 \right) \cdot m_{ij} \left(\mathbf{v}_j^1 - \mathbf{v}_j^0 \right). \tag{74}$$

The last term on the RHS is the error that results from substituting $\mathbf{v}^\theta$ for $\mathbf{v}^{1/2}$ in the second term. The first term on the RHS is the change in kinetic energy, $k^1 - k^0$.

The second term is the work done by material stress. The strain energy increment is calculated by inverting the order of summation;

$$\mathcal{S}^1 - \mathcal{S}^0 = \sum_p \sigma_p^\theta : \sum_i \nabla_{pi} \mathbf{v}_i^\theta \Delta t. \tag{75}$$

$\mathcal{S}$ contains a positive strain energy and a second error term, which we now show. Denoting $\mathcal{S}$ for each particle by $\int$, one finds

$$\int^1 - \int^0 = (\theta \mathbf{e}^1 + (1 - \theta) \mathbf{e}^0) : \mathbf{T} : (\mathbf{e}^1 - \mathbf{e}^0), \tag{76}$$

where the subscript p is understood. The total strain is obtained by integrating the strain increments from $t = 0$ to $t = n\Delta t$,

$$\int^n - \int^0 = \sum_{l=1}^n \frac{1}{2} (\mathbf{e}^l + \mathbf{e}^{l-1}) : \mathbf{T} : (\mathbf{e}^l - \mathbf{e}^{l-1}) + \left(\theta - \frac{1}{2} \right) \sum_{l=1}^n (\mathbf{e}^l - \mathbf{e}^{l-1}) : \mathbf{T} : (\mathbf{e}^l - \mathbf{e}^{l-1}). \tag{77}$$

The second term on the RHS is an error that decreases the strain energy density each time step. The first term telescopes to give the particle strain energy,

$$s^n - s^0 = \frac{1}{2} \sum_p \mathbf{e}_p^n : \mathbf{T} : \mathbf{e}_p^n, \tag{78}$$

assuming $\mathbf{e}_p(t = 0) = 0$.

We now consider the contribution of work done by the contact force to Eq. (32). First, consider the normal component of the contact force. The work done by this component is

$$\sum_{g,i,j} \mathbf{v}_{gi}^\theta \cdot \hat{\mathbf{n}}_g \hat{\mathbf{n}}_g \cdot m_{gij} \mathbf{f}_{gj} \theta \Delta t = \sum_{g,i,j} \left[(\mathbf{v}_g^\theta - \tilde{\mathbf{v}}^\theta) + \tilde{\mathbf{v}}^\theta \right]_i \cdot \hat{\mathbf{n}}_g \hat{\mathbf{n}}_g \cdot m_{gij} \mathbf{f}_{gj} \theta \Delta t = 0. \tag{79}$$

The work done by the normal component of the contact force is zero as a consequence of Eqs. (11) and (51).

The frictional term acts in the direction of the unit tangent vector,

$$\hat{\mathbf{n}}_g \times \hat{\omega}_g = \frac{\hat{\mathbf{t}}_g \hat{\mathbf{t}}_g \cdot (\tilde{\mathbf{v}}_g^\theta - \tilde{\mathbf{v}}^\theta)}{\left| \tilde{\mathbf{v}}_g^\theta - \tilde{\mathbf{v}}^\theta \right|}. \tag{80}$$

The work done by the frictional contact force each time step is

$$\mathcal{F} = \sum_{g,i,j} \mathbf{v}_g^\theta \cdot \hat{\mathbf{t}}_g \hat{\mathbf{t}}_g \cdot m_{gij} \mathbf{f}_{gj} \theta \Delta t = -\sum_{g,i,j} m_{gij} \mu' \mathbf{v}_{gj}^\theta \cdot \hat{\mathbf{t}}_g \hat{\mathbf{t}}_g \cdot \left(\tilde{\mathbf{v}}_{gj}^\theta - \tilde{\mathbf{v}}_j^\theta \right). \tag{81}$$

The effective friction coefficient is calculated so that the velocity does not reverse sign under the action of friction, $(\mathbf{v}_g^\theta \cdot \hat{\mathbf{t}}_g)(\tilde{\mathbf{v}}_g^\theta \cdot \hat{\mathbf{t}}_g) \geq 0$. As above, adding and subtracting $\tilde{\mathbf{v}}^\theta$ and using Eq. (9), one finds the work done by frictional contact to be dissipative,

$$\mathcal{F} = -\sum_{g,i,j} \mu'\left(\mathbf{v}_{gi}^\theta - \tilde{\mathbf{v}}_i^\theta\right) \cdot \hat{\mathbf{t}}_g \hat{\mathbf{t}}_g \cdot m_{gij}\left(\tilde{\mathbf{v}}_{gj}^\theta - \tilde{\mathbf{v}}_j^\theta\right) \leq 0. \tag{82}$$

The work done by friction, above, and by plastic deformation is not recoverable. Thus, the contribution of plasticity to the total energy must be dissipative on physical grounds and the numerical solutions must reproduce this. Differencing Eq. (6), yields the contribution from plasticity, $\mathcal{P}$, which is given by

$$\mathcal{P} = -\sum_p \sigma_p^{n+\theta-1} : \left(\mathbf{e}_{p,pl}^n - \mathbf{e}_{p,pl}^{n-1}\right) V_p = -\sum_p \left(\lambda_p^n - \lambda_p^{n-1}\right)\frac{\sigma_p^{n+\theta-1} : \sigma_p^{n+\theta-1}}{\left|\sigma_p^{n+\theta-1}\right|} V_p \leq 0 \tag{83}$$

and is dissipative since λ is monotone increasing.

Viscoelastic materials can be analyzed similarly, and one finds that the viscosity contributes a dissipative term to the total energy equation.

The lumped mass matrix introduces a decrement in kinetic energy each time step, $\mathcal{L}$, given by

$$\mathcal{L} = \frac{1}{2}\sum_p m_p\left[\left(\mathbf{v}_p^n - \mathbf{v}_p^{n-1}\right)^2 - \sum_i s(\xi_i - \xi_p)\left(\mathbf{v}_i^n - \mathbf{v}_i^{n-1}\right)^2\right], \tag{84}$$

which is negative definite [35]. Including contributions from friction, plasticity, and the interpolation error due to lumping the mass matrix, the total energy, $E = k + s$, decreases each time step by an amount given by

$$E^n = E^{n-1} - \left(\theta - \frac{1}{2}\right)\sum_p\left[m_p\left(\mathbf{v}_p^n - \mathbf{v}_p^{n-1}\right)^2\right.$$
$$\left. + \left(\mathbf{e}_p^n - \mathbf{e}_p^{n-1}\right):\mathbf{T}:\left(\mathbf{e}_p^n - \mathbf{e}_p^{n-1}\right)\right] + \mathcal{F} + \mathcal{P} + \mathcal{L} \tag{85}$$

for $\theta \geq \frac{1}{2}$. Of course, exact energy conservation could be achieved in the usual way by storing an internal energy, to which all dissipation terms would be added in each computation step as heat sources. (Note also that the total energy is unchanged through the second phase of each time step, since it is calculated from particle data.)

4.4. Stability Analysis Using the Energy Method

The energy method "requires that one devise a norm for the solution vector which one can then show increases by a factor no greater than $1 + O(\Delta t)$ at each time step. This implies stability in this norm and the argument is then usually rounded off by demonstrating the equivalence of the norm to the L_2 norm. The reason for the name is that in certain fairly simple cases the physical energy of the system provides such a norm" [31, p. 132]. (A general review of the stability of difference methods is given by Thomée [14].)

The energy method for the stability of finite difference equations is described in Chapter 6 of Richtmyer and Morton [31, pp. 131–151]. The authors comment that energy analysis

will "in general obtain a sufficient condition for stability. ...Thus, to some extent the method is complementary to normal mode analysis which tends to yield necessary stability conditions." [31, p. 133]. (For example, a time step half the Courant limit is sufficient for stability, but unnecessary.) In the case of linear, isotropic elasticity, the energy of the system provides an appropriate norm. In this section we use energy stability analysis on the MPM equations to show that the implicit θ scheme is stable in the L_2 norm.

As is well known, the leapfrog algorithm conserves energy to high order in Δt. Mazur argues that leapfrog conservation is even better than commonly believed, because much of the apparent error is due to the interpolation one uses to construct an energy from variables that are defined at different time levels [30]. That is, if one wishes to compute E^n or $E^{n+1/2}$, one must interpolate either the velocity or the strain between time levels. With sufficiently high-order interpolation, the leapfrog algorithm can be shown to conserve energy in general to at least $O(\Delta t^3)$. However, even though total energy is conserved to high order, the leapfrog algorithm cannot be shown to be stable in the L_2 norm. Consider, for example, leapfrog solutions of the simple harmonic oscillator problem, for which the the energy integrals are either

$$E^n = \frac{1}{2}\left[m(\mathbf{v}^n)^2 + \mathcal{K}\mathbf{x}^{n-\frac{1}{2}} \cdot \mathbf{x}^{n+\frac{1}{2}}\right] \tag{86}$$

or

$$E^{n+\frac{1}{2}} = \frac{1}{2}\left[m\mathbf{v}^n \cdot \mathbf{v}^{n+1} + \mathcal{K}\mathbf{x}^{n+\frac{1}{2}} \cdot \mathbf{x}^{n+\frac{1}{2}}\right]. \tag{87}$$

These energies are each constant for all n and equal to each other. However, even though this energy is a constant of the motion, neither the kinetic nor the strain energy is bounded. For example, in a time oscillatory solution with angular frequency $\omega = \pi/\Delta t$, the solution alternates in sign on successive time steps. For this mode, the inner product of the strain in Eq. (86), and the velocity in Eq. (87), can be negative. Thus, neither the kinetic energy in Eq. (86) nor the strain energy in Eq. (87) is bounded by the initial total energy. (Of course, negative energies will not occur in this simple case unless $\sqrt{\frac{\mathcal{K}}{m}}\Delta t$ is comparable to π. However, there is substantial evidence from fluid dynamics calculations [28] and plasma simulations [9] using leapfrog time advancement that more complex systems do develop high frequency noise, especially as Δt approaches the explicit stability limit. As noted earlier, energy growth is a signature of the finite-grid instability in plasma simulations. Implicit plasma simulations are not as hospitable to the accumulation of energy in high frequency modes and hence are much less prone to the finite-grid instability even though energy is not strictly conserved [5].)

Numerical tests seem to confirm that leapfrog solutions are not bounded in the L_2 norm. A catastrophic, nonlinear instability is observed with the leapfrog algorithm, which seems to occur more frequently when the solution oscillates about zero [11]. The instability is attributed to aliasing error, which tends to accumulate energy at the shortest scales.

With the implicit θ scheme, one can make strong statements about the L_2 norm of the solution using the results of Section 4.3.

Both the kinetic, Eq. (28), and strain energies, Eq. (78), are positive or zero at every time step n,

$$k^n, \quad s^n \geq 0. \tag{88}$$

The total energy at step n is bounded by the total energy at step 0,

$$E^n \leq E^0. \tag{89}$$

(In fact, for $\theta > \frac{1}{2}$ and with the lumped mass matrix, the total energy decreases each time step by an amount given by Eq. (85). Therefore, the kinetic and strain energies are both bounded by E^0,

$$k^n, \quad s^n \leq E^0, \tag{90}$$

and the implicit θ scheme is stable in the L_2 norm independent of the time step.)

5. A FULLY IMPLICIT ALGORITHM USING THE NEWTON–KRYLOV TECHNIQUE

The matrix-free Newton–Krylov method [51] and [52] is becoming an increasingly popular tool to solve fully coupled transient and steady-state problems. It has been applied successfully to a wide range of applications such as combustion [54], phase change [53], and plasma physics [59], among others.

In this study, an inexact, matrix-free, Newton–Krylov technique is used to solve the fully implicit, nonlinear equations of motion for granular flows with intergranular contact. This implicit integration algorithm is in contrast to that of Benson [12], who implemented an implicit Eulerian formulation based on mixture theory by solving a quasi-Newton problem. It is also in contrast to the work of Guilkey and Weiss [55], who solved a linearized implicit form of the MPM equations (with intergranular contact neglected) by inverting a stiffness matrix.

The nonlinear system of equations is represented by the vector

$$\mathbf{F}(\mathbf{v}) = [F_x(\mathbf{v}), F_y(\mathbf{v})]. \tag{91}$$

Here F_x represents a concatenation of all x residual lists associated with each grain $g \epsilon [1, N]$,

$$F_x = \left[F_x^1, \ldots, F_x^g, \ldots, F_x^N \right], \tag{92}$$

where N is the number of grains. For a given grain g, its residual list F_x^g contains x component residuals at all nodes i belonging to g, $i \epsilon [1, N_g]$,

$$F_x^g = \left[F_{x_1}^g, \ldots, F_{x_i}^g, \ldots, F_{x_{N_g}}^g \right], \tag{93}$$

where N_g is the number of nodes belonging to grain g. Similarly for the y component residuals,

$$F_y = \left[F_y^1, \ldots, F_y^g, \ldots, F_y^N \right], \tag{94}$$

where

$$F_y^g = \left[F_{y_1}^g, \ldots, F_{y_i}^g, \ldots, F_{y_{N_g}}^g \right]. \tag{95}$$

All residual components $F_{x_i}^g$ and $F_{y_i}^g$ are dimensionally identical.

Typically, one uses the NK method to find a single-valued vector field. In our application, a multivalued velocity field is required since relative motion between the grains must be simulated. When grains come into close contact, a node i can be shared between more than one grain. Therefore, there will be more than one velocity field associated with i. Without a contact force, the grains would be decoupled and the NK technique could conceivably be applied to individual grains. However, the contact force couples the grains, and one must solve for a full multivalued velocity field in the NK technique. The nonlinear residual vector $\mathbf{F}$ must therefore contain the residuals for nodes i belonging to each grain g, for all grains.

The solution of the fully coupled, nonlinear problem is a concatenated velocity field $\mathbf{v}^{k+1}$, which satisfies

$$\frac{\|\mathbf{F}(\mathbf{v}^{k+1})\|_2}{F_{scale}} < \text{tol}, \tag{96}$$

where tol is a chosen tolerance, which in this paper is consistently 10^{-6}, and F_{scale} is a scaling factor that is problem dependent. Using Newton's method, the roots of this equation are calculated by solving the following linear system

$$\mathbf{J}^k \delta \mathbf{v}^k = -\mathbf{F}(\mathbf{v}^k), \tag{97}$$

$$\mathbf{v}^{k+1} = \mathbf{v}^k + \delta \mathbf{v}^k, \tag{98}$$

where $\mathbf{J}^k$ is the Jacobian matrix whose element (i, j) is

$$J_{i,j}^k = \frac{\partial F_i}{\partial v_j^k}. \tag{99}$$

In the nonlinear granular flow equation set, the frictional component of the contact model and the plastic constitutive model are both governed by nonlinear functions (see Section 2); in addition the plastic constitutive model is also nonanalytic [8]. The Jacobian matrix for this system will therefore be difficult to invert. One of the primary advantages of employing the matrix-free NK technique to granular flows is that the Jacobian does not need to be formed (hence the term "matrix-free"). This occurs because a pertinent choice of the linear solver used in Eq. (97) requires the Jacobian only in the form of a matrix–vector product. One such solver is the generalized minimal residual (GMRES) algorithm [56]. GMRES constructs the solution of Eq. (97) from a basis of Krylov vectors $(\mathbf{r_0}, \mathbf{Jr_0}, \ldots, \mathbf{J}^{l-1}\mathbf{r_0})$, where the number of GMRES iterations is l and $\mathbf{r_0}$ is the initial linear residual,

$$\mathbf{r_0} = -\mathbf{F}(\mathbf{v}^k) - \mathbf{J}^k \delta \mathbf{v}^0, \tag{100}$$

constructed from an initial guess $\delta \mathbf{v}^0$. The matrix–vector product we use is the following second-order approximation to Eq. (99),

$$\mathbf{J}^k \mathbf{r} = \frac{\mathbf{F}(\mathbf{v}^k + \epsilon \mathbf{r}) - \mathbf{F}(\mathbf{v}^k)}{\epsilon}. \tag{101}$$

The choice of perturbation parameter ϵ is heuristic and affects convergence considerably. An ϵ too large produces an inaccurate approximation to the local gradient, particularly if

the residual has variations on a scale smaller than ϵ. However, an ϵ too small results in local numerical noise severely depleting the accuracy of the local gradient calculation. In this study,

$$\epsilon = 0.001 \times \frac{1}{N} \|\mathbf{r}\|_2 \sum |\mathbf{v}^k| \tag{102}$$

is consistently employed.

At each Newton step, the linear problem is solved only to a tolerance that is proportional to the current nonlinear residual,

$$\mathbf{J}^k \delta \mathbf{v} < -\gamma \mathbf{F}^k, \tag{103}$$

hence the term "inexact." This tolerance is employed so that time is not spent needlessly solving the linear problem when the local gradient is far away from the root of the nonlinear residual. For this study

$$\gamma = 0.001. \tag{104}$$

6. THE NONLINEAR RESIDUAL FUNCTION

For a given grain, the θ-centered nonlinear residual at node i is

$$\mathbf{F}_i^k = m_i^n \left(\frac{\mathbf{v}_i^{n+\theta,k} - \mathbf{v}_i^{n,k}}{\theta \Delta t} \right) - \left[\sum_p \nabla_{pi} \cdot \left(\sigma_p^{n+\theta,k} V_p^{n+\theta,k} \right) + m_i^n \mathbf{f}_i^{n+\theta,k} \right] \tag{105}$$

at the kth Newton iteration. For notational convenience, the grain subscript g has been suppressed and the following scaling

$$F_{scale} = V_{char} \sum_p m_p \tag{106}$$

is employed, where V_{char} is the characteristic velocity of the system.

The residual is mass weighted to prevent stiff behavior in the residual. Specifically, not mass weighting the residual results in a division by the mass of the internal force term. At grain edges, the nodal mass can be orders of magnitude smaller than in the interior. The small mass occurs when a material point moves into a new element at a grain edge (the shape function used to interpolate particle masses tends to zero as the interpolating distance approaches the element width). The corresponding internal force calculation at these nodes will not, however, tend to zero as the gradient shape function is piecewise constant throughout the element [8]. The internal force and nodal mass do not scale here. A small change in internal force (due to small velocity changes) will result in large changes in the residual at the nodes in question, if the residual is not mass weighted.

The disparity in scale between the vertex mass and internal force term can result in a matrix system that is not diagonally dominant for large time steps. In these cases, the linear problem is poorly conditioned and the linear iteration converges slowly. To prevent this poor conditioning, a FLIP [35] implementation of the internal force term was also constructed. In this implementation, cell-centered velocity gradients are calculated by applying a second-order gradient function to nodal velocities. The cell-centered gradients are quadratically

interpolated to particles. The constitutive stress model is applied to produce particle stresses which are then quadratically interpolated back to cell centers. The nodal internal force is then calculated by applying the second-order gradient function to cell-centered stresses. The higher order interpolation in the internal force calculation is found to provide better conditioning in the implicit algorithm (for the single rebounding grain problem in Section 8.3 the total number of linear iterations was halved) and more robust results for the explicit algorithm. The stencil is broader, however, and produces errors in shearing near the grain edges. Therefore the MPM implementation is consistently used in this work.

7. PRECONDITIONING

The overall efficiency of a Newton–Krylov solver is highly dependent on the choice of preconditioner used in the solution of the linear system. A preconditioner is indeed almost mandatory in a Newton–Krylov implementation. In most NK applications in the literature, an approximate matrix is constructed and either incomplete LU (ILU) factorizations [54] or smoothing techniques such as weighted Jacobi (WJ) or symmetric successive overrelaxation (SSOR) used in the preconditioner [57]. In these cases, the Newton–Krylov implementation is not strictly "matrix-free" as a matrix that approximates the Jacobian is constructed and inverted. Recently, the emphasis has been on the use of multilevel preconditioners, for which the number of iterations is only weakly dependent on the number of grid points [58].

In this study, the linear system is right preconditioned so that the system to be solved is

$$\mathbf{J}^k \mathbf{M}^{-1} (\mathbf{M} \delta \mathbf{v}^k) = -\mathbf{F}(\mathbf{v}^k). \tag{107}$$

The solution is formed from the following basis of l Krylov vectors

$$\delta \mathbf{v}^k \approx \sum_{q=0}^{q=l} \beta_q \mathbf{M}_q^{-1} \mathbf{V}_q, \tag{108}$$

where $\mathbf{V}_q = q$th Krylov vector $= (\mathbf{J}^k \mathbf{M}_q^{-1}) \mathbf{V}_{q-1}$,

$$\mathbf{J}^k \mathbf{M}_q^{-1} \mathbf{V}_q = \frac{\mathbf{F}(\mathbf{v}^k + \epsilon \mathbf{M}_q^{-1} \mathbf{V}_q) - \mathbf{F}(\mathbf{v}^k)}{\epsilon}. \tag{109}$$

To evaluate the preconditioned Krylov vector,

$$\mathbf{y}_q \approx \mathbf{M}_q^{-1} \mathbf{V}_q, \tag{110}$$

a multilevel preconditioner that uses a matrix-free operator on each level is implemented. On a given level, the matrix-free operator is constructed by employing the second-order matrix–vector product given in Eq. (101) in conjunction with a fixed number of weighted Jacobi iterations. This means that the implementation is truly matrix-free and a knowledge of the Newton residual at the given level is the only requirement to implement the preconditioner operator. To employ the preconditioner operator on coarse levels, coarse particles and coarse shape functions are recursively constructed. The coarse shape functions are identical in shape to those on the fine level but have compact support that scales with the coarse

level mesh size. Coarser level shape functions can be formulated by extending the compact support to scale with the new coarse level mesh size.

A coarse particle is constructed using the following rules:

1. It occupies one fine level cell only and its position in that cell is the center-of-mass position of the fine level particles.

2. Its mass is the sum of the fine level particle masses in that cell. (Therefore the total mass is an invariant for each level.)

3. Its volume is the sum of the fine level particle volumes in that cell.

4. Its velocity is the center-of-mass velocity of the fine level particle velocities. (Therefore total momentum is an invariant for each level.)

5. Its stress tensor is the volume-weighted stress tensors of the fine level particles in that cell.

Coarser level particles can then be recursively constructed by considering the previous level's particles. Figure 1 displays a schematic of the construction of coarse level particles from the previous finer level particles. Figure 2 shows the particle positions and vertex mass contours produced for a single grain on a 32×32 mesh with four levels used in the preconditioner. In particular, one will note that the support of the vertex mass expands as the levels coarsen, due to the increased size of support of the shape functions.

These constructs are then used to evaluate a Newton residual at all levels—in the same way as on the finest level. In this way, the process of calculating the preconditioner operator is identical for all levels, the only difference being the particles and shape functions used in this calculation. In the following sections, the WJ preconditioner operator and the multilevel algorithm will be outlined further.
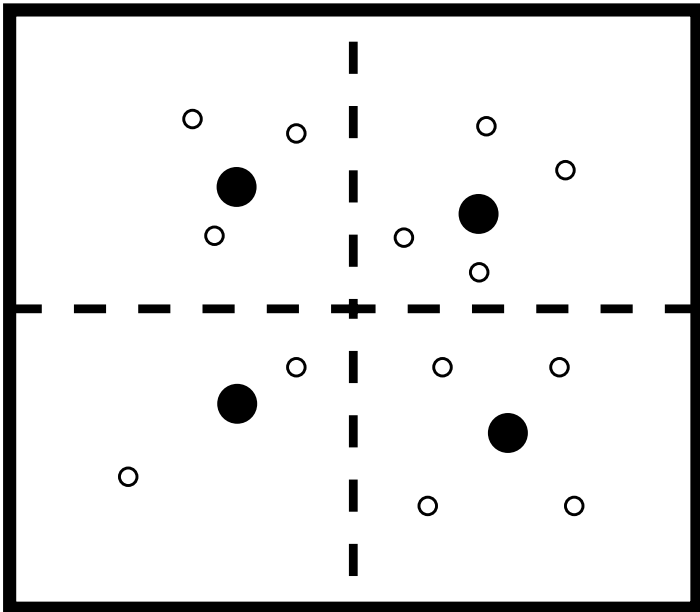

FIG. 1. Schematic showing how the coarse level particle and grid structure are generated from the fine level. The dark dots and dark lines indicate the coarse level particles and grid structure. The open dots and dashed lines indicate the fine level particles and grid structure.
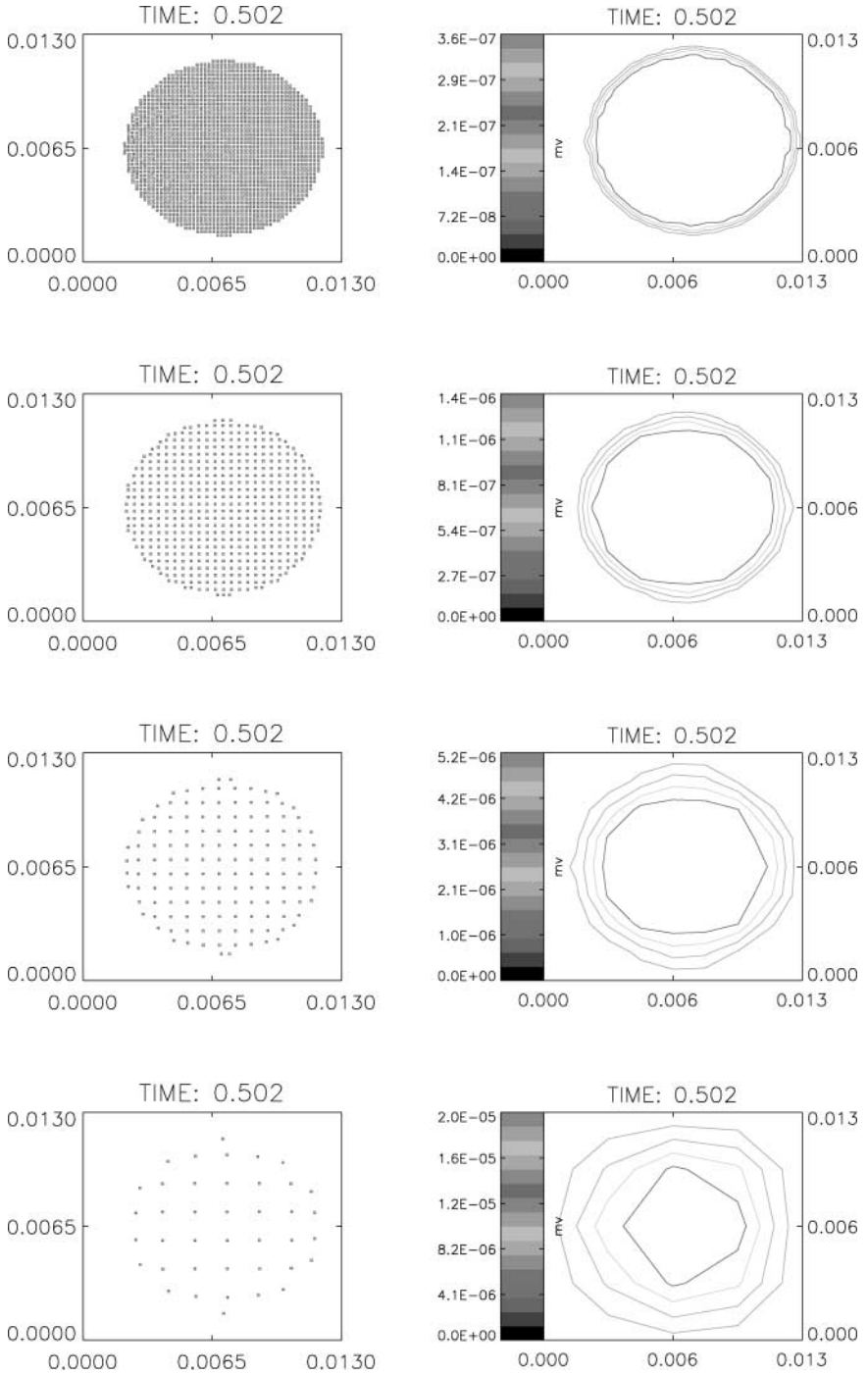
FIG. 2. 32×32 rebounding elastic grain simulation with multilevel preconditioner. This figure shows the particle positions and vertex mass contours at $t = 0.5$ for four levels in the multilevel preconditioner. The levels range from a 32×32 grid structure to a 4×4 grid structure.

7.1. Weighted Jacobi Operator

To solve the system approximately,

$$\mathbf{y} \approx \mathbf{M}^{-1}\mathbf{V}, \tag{111}$$

the following WJ smoother is employed

$$\mathbf{y}^s = \mathbf{y}^{s-1} + \frac{w}{\mathbf{D}}(\mathbf{V} - \mathbf{M}\mathbf{y}^{s-1}), \tag{112}$$

where $w = 0.67$, s is the smoothing iteration, $\mathbf{D}$ is the diagonal vector of the matrix $\mathbf{M}$, and the following matrix-free approximation

$$\mathbf{M}\mathbf{y} = \frac{\tilde{\mathbf{F}}(\mathbf{v}^k + \epsilon(\mathbf{y})) - \tilde{\mathbf{F}}(\mathbf{v}^k)}{\epsilon} \tag{113}$$

is used. In this study two sweeps of the WJ operator are employed on a given level.

For the purpose of efficiency, we assume the Fabre-averaged velocity is constant throughout the preconditioning operation. This is a reasonable assumption given that it is a mass weighting of the surrounding grain velocities. A large change in a grain's velocity would typically be associated with a small mass and thus, the effect of this change on the Fabre-averaged velocity is lessened since it is weighted by the small mass. The advantage of this assumption is that the internal force and contact force can be calculated in one pass of the grid rather than two. A simplified form of the Newton residual, ($\tilde{\mathbf{F}}$ in Eq. (113)), is therefore employed in the calculation of $\mathbf{M}$,

$$\tilde{\mathbf{F}}(\mathbf{v}_i^{n+\theta,k}) = m_i^n \left(\frac{\mathbf{v}_i^{n+\theta,k} - \mathbf{v}_i^{n,k}}{\theta \, \Delta t} \right) - \left[\sum_p \nabla_{pi} \cdot \left(\sigma_p^{n+\theta,k} V_p^{n+\theta,k} \right) + m_i^n \tilde{\mathbf{f}}_i^{n+\theta,k} \right], \tag{114}$$

where $\tilde{\mathbf{f}}$ denotes the contact force with a constant Fabre-averaged velocity throughout the preconditioner.

In order to retain the matrix-free approximation, each element in the diagonal is effectively calculated by passing in a unit vector into Eq. (113). Thus element a in diagonal $\mathbf{D}$ is

$$\mathbf{D}_a = \mathbf{M}\mathbf{e}_a, \tag{115}$$

where $\mathbf{e}_a$ is the zero vector with a unit entry at element a. For a two-dimensional system with n nodes, this operation requires $2n$ Newton residual calculations (for two dimensions, $\mathbf{D}$ has two components, $[D_x, D_y]$, per node and thus is a vector of length $2n$). For large systems, this approach quickly becomes prohibitive. Instead, the work required in the diagonal calculation is reduced by recognizing the sparsity of $\mathbf{M}$ due to the compact nature of the MPM equations in the calculation of $\tilde{\mathbf{F}}$. The domain of influence of a node (i, j) used in the calculation of a two-dimensional MPM Newton residual totals nine cells. To calculate one of the diagonal components, say D_x at node (i, j), one then only has to pass in an input vector that zeroes D_y at node (i, j) and D_x and D_y at the surrounding eight nodes. The diagonal $\mathbf{D}$ can therefore be calculated using 18 rather than $2n$ calls to the matrix-free approximation.

7.2. Multilevel Treatment

In this study, a defect-correction V-cycle multilevel preconditioner [60] is investigated to evaluate the preconditioned Krylov vector

$$\mathbf{y}_q \approx \mathbf{M}_q^{-1} \mathbf{V}_q. \tag{116}$$

On each grid a coarse approximation to the system in Eq. (111) is solved using two sweeps of WJ. For all levels, the matrix-free approximation is retained. On the downward component of the V-cycle, the error in this solution is restricted to the next coarse level (which then becomes the source term for the system on the next level). On the upward component of the V-cycle, the error is prolonged to a finer level and added to the existing solution. The restriction operator employs mass-weighted quadratic interpolation while the prolongation operator uses bilinear interpolation. The implementation of the V-cycle is standard. The error on a coarse grid l ($l > 1$) is

$$\mathbf{E}^l \approx \mathbf{s}_l - \mathbf{M}_q^l \mathbf{y}^l, \tag{117}$$

where $\mathbf{s}_l$ is the source term (the restricted error from the previous finer level) and $\mathbf{y}_l$ is the solution on grid level l after two WJ sweeps and

$$\mathbf{M}_q^l \mathbf{y}^l = \frac{\tilde{\mathbf{F}}^l(\mathbf{v}_l^n + \epsilon \mathbf{y}^l) - \tilde{\mathbf{F}}^l(\mathbf{v}_l^n)}{\epsilon}, \tag{118}$$

where $\tilde{\mathbf{F}}^l$ refers to the Newton residual calculated on grid l. This calculation uses the constructed particles and shape functions associated with grid l. The nodal velocities at grid level l, $\mathbf{v}_l^n$ are calculated from grid l particle velocities using the shape functions associated with l. As velocities at coarse grids are not updated during the Newton iteration, the matrix-free approximation invokes a perturbation from velocity $\mathbf{v}_l^n$ instead of $\mathbf{v}_l^k$.

7.3. Multilevel Results

The multilevel preconditioner is tested on the single impacting grain calculation discussed in Section 8.3. The simulation is run to a time $t = 0.5$, enough time to allow for one impact with the wall. Simulations were run on a 16×16 mesh, a 32×32 mesh, and a 64×64 mesh. The results for the 64×64 mesh are shown in Table I and are representative of the behavior for the other two resolutions. Table I provides the required total Newton iterations, total linear iterations, and average linear iterations per Newton iteration for preconditioners with varying levels.

For all meshes, the single-level preconditioner makes an obvious difference. Without a preconditioner, the average number of linear iterations per Newton iteration is 10. This occurs because for these results, a maximum number of 10 linear iterations per Newton iteration is allowed. (That is, a maximum of 10 Krylov vectors is permitted in the GMRES solver. If linear convergence is not achieved, solution of the linear problem ceases and a new Newton residual is calculated; see Section 8.2). Thus, without the preconditioner, the linear problem is never solved to within the specified tolerance. Employing a single-level preconditioner results in this average reducing to ≈ 4. In addition, the number of Newton iterations decreases for all three resolutions.

TABLE I

Multilevel WJ Preconditioner Statistics for the Single Grain Simulation on a 64 × 64 Grid

Levels	Newton iterations	Linear iterations	Linear its./Newton it.
No preconditioner	4249	42490	10.00
1 Level	1724	6727	3.900
2 Levels	1711	4590	2.683
3 Levels	1695	4587	2.706
4 Levels	1717	4596	2.677
5 Levels	1691	4442	2.627
6 Levels	1696	4452	2.625

Note. The simulation runs to a time $t = 0.50$.

For multilevel preconditioners, the results are somewhat disappointing. For all meshes, little improvement is seen for preconditioners with more than two levels. One of the problems with the use of a multilevel preconditioner in this context is that the representation of the contact force is self-similar with mesh spacing. Therefore as the mesh coarsens the distance over which the contact force act increases (it scales with the mesh size) and hence the proportion of the grain under contact increases. Indeed for a coarse enough grid (i.e., 1×1 grid), the single grain will always be in contact with the boundaries. This suggests that the character of the matrix is different at different levels (i.e., it may incorporate the contact function at a particular coarse level but not at finer levels) and therefore the error added back to finer levels is misrepresentative.

While use of the multilevel preconditioner is not disastrous (indeed the number of linear iterations per Newton iteration is limited and nearly mesh independent, as reported in [58]), it has currently not provided a positive contribution given the overhead associated with use of the multilevels.

8. RESULTS

8.1. Time Step Restriction

For the implicit calculations, the time step is governed by a limitation on the maximum strain incurred in a time step. The idea behind the constraint is to ensure that an undisturbed stress state will not evolve to a plastic yielded state in one time step and therefore to ensure the stress is close to the yield surface immediately before the onset of plasticity. For all calculations, the time step is limited such that the maximum strain is $<1\%$,

$$\Delta t_{imp} < \frac{0.01}{\left(\frac{de}{dt}\right)_{max}}, \tag{119}$$

where

$$\left(\frac{de}{dt}\right)_{max} = \sqrt{\frac{1}{4}[\nabla \mathbf{v} + \nabla \mathbf{v}^T] : [\nabla \mathbf{v} + \nabla \mathbf{v}^T]}. \tag{120}$$

For the plastic granular materials in this paper, a strain of 1% will result in a stress increment of $\approx 0.25 \times \sigma_y$, where σ_y is the plastic yield stress. While this strain restriction is relevant

only to plastic materials, we apply it to simulations of purely elastic materials as well. For materials with different yield properties the maximum strain limit will need to alter.

For the explicit calculations, the Courant stability restriction is applied to the time step to ensure linear stability (see Section 4). The maximum propagation wave speed is

$$s = \sqrt{\frac{\lambda + 4/3\mu}{\rho}} \tag{121}$$

and therefore the explicit time step is restricted to

$$\Delta t_{exp} < \frac{\alpha \Delta x}{s}. \tag{122}$$

A conservative CFL stability limit of $\alpha = 0.25$ gives better energy conservation with the explicit leapfrog method (which conserves total energy to at least $O(\Delta t^3)$). Hence for the explicit calculations,

$$\Delta t_{exp} < \min\left(\frac{0.01}{\left(\frac{de}{dt}\right)_{max}}, \frac{\alpha \Delta x}{s}\right). \tag{123}$$

8.2. Exiting a Newton–Krylov Iteration

The linear systems solved in this study become poorly conditioned for time steps $\Delta t > 10 \times \Delta t_{exp}$. The use of the preconditioner aids in the efficiency of the solution for $\Delta t < 10 \times \Delta t_{exp}$ but does not increase the radius of convergence. The reason for this is currently unclear. To cater to these situations, a maximum number of Krylov vectors in the GMRES solver is specified. If linear convergence is not achieved after k iterations (that is, with a solution containing k Krylov basis vectors), the GMRES iteration ceases and the resulting Newton residual is calculated.

If convergence of the nonlinear problem has not been achieved after a maximum number of Newton iterations, the time step is halved and the Newton–Krylov iteration started again.

8.3. Single Elastic Grain

The problem of an elastic, impacting grain is examined using the implicit and explicit MPM formulations. Table II provides the physical properties of the grain and the simulation properties. The physical properties are chosen so that the grain is representative of the material high melting explosive (HMX) used in plastic bonded explosives (PBX). The difference here is that the grain behaves elastically and hence is assumed to have an infinite yield stress. Table III provides the specifications of the implicit formulation of the problem. Figure 3 shows the initial physical configuration.

Figure 4 reveals the grain positions for the implicit formulation. Figure 5 shows the final grain position using the explicit formulation with a CFL stability limit of $\alpha = 0.25$ and $\alpha = 1.0$, respectively. The results show that as the simulation evolves, the edges of the grain separate from the main body. The grain spallation is more catastrophic when the CFL limit is increased to $\alpha = 1.0$. Spallation is completely absent in the implicit results, corresponding to an apparent absence of the finite-grid instability with an implicit formulation. In addition, the energetics of contact are accurately computed because the internal forces are evaluated

TABLE II
Specifications of the Single Elastic Grain Problem

Resolution	17×17
Δx	8.118×10^{-6} m
Δy	8.118×10^{-6} m
Maximum particles per cell	4
Simulation time	2.00
Δt_{MAX}	$2.5 \times \Delta t_{exp}$
Bulk modulus λ	11.36×10^9 Pa
Shear modulus μ	7.48×10^9 Pa
Density ρ	1.9×10^3 kg/m³
U_o	-1.0×10^2 m/s
V_o	0.0 m/s

TABLE III
Specifications for the Implicit Simulation of the Single Elastic Grain Problem

Newton tolerance	10^{-6}
ϵ	10^{-3}
γ	10^{-3}
F_{scale}	$U_o \times \sum_p^{Np} m_p$
Maximum Newton iterations	20
Maximum linear iterations	5
Preconditioner	None
θ	0.5
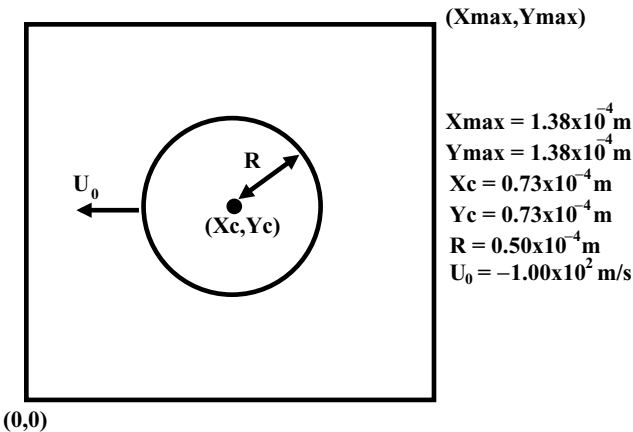
FIG. 3. Schematic showing the initial configuration of the rebounding elastic grain problem.
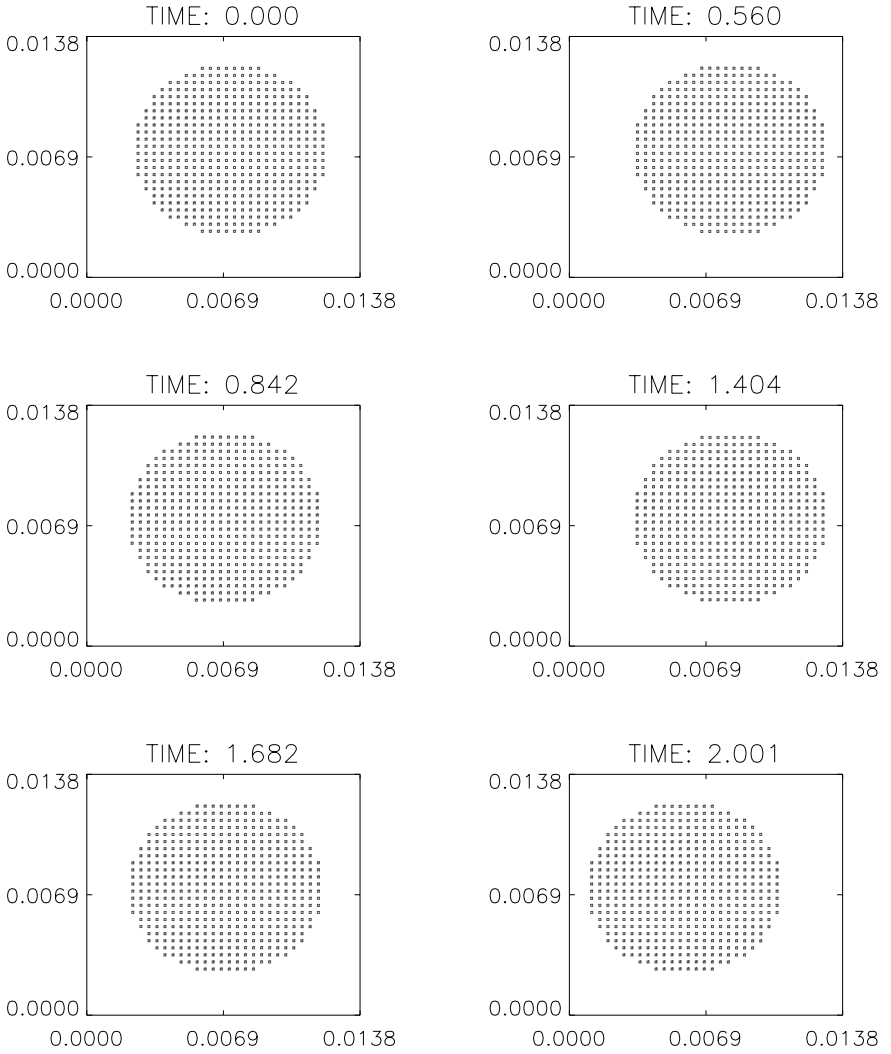
FIG. 4. Grain positions in the elastic rebounding grain problem using the implicit MPM formulation.
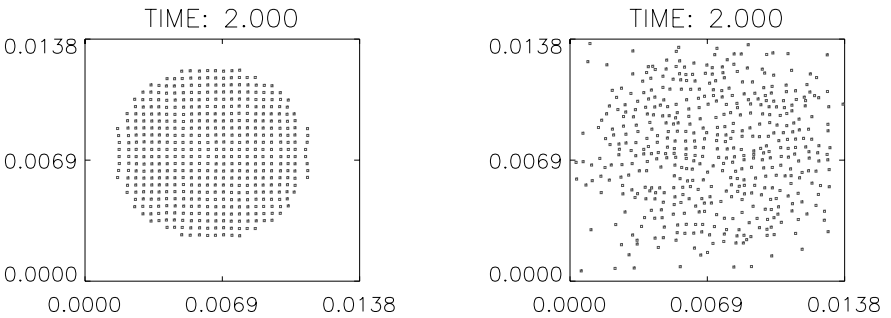
FIG. 5. Grain positions at time $t = 2.00$ in the elastic rebounding grain problem using the explicit MPM formulation with a CFL stability limit of $\alpha = 0.25$ and $\alpha = 1.00$.

using an estimate of the time-averaged velocity (averaged over the integration period). The contact force responds to this internal force which will then alter the estimate of the time-averaged velocity. The process continues until the solution relaxes to an equilibrium between the internal force, the contact force, and the time-averaged acceleration (i.e., the solution converges to $\mathbf{F} = 0$).

In contrast, the explicit formulation calculates internal forces based only on the velocity at the start of the time step. This velocity can significantly alter during the integration period, causing the internal force estimate to be inaccurate during the integration period. There is no guarantee of a balance between forces and the time-averaged acceleration. The problem is magnified further where grains cross cell boundaries. In these cases the mass is small here, and large accelerations result during the time step. Using the velocity at the start of the time step is clearly in error.

Plots of total energy, elastic energy, and momentum transfer for both implicit and explicit cases are presented in Fig. 6. For the explicit leapfrog, the total energy grows by an order of magnitude. The primary contributor to the total energy growth in the explicit case is the elastic energy, as shown in Fig. 6. This energy accumulates significantly for the explicit case, after each wall contact and even during simple advection. An inelastic deformation is occurring after each wall contact and residual stresses remain. The effect of the inelastic collision is seen in the explicit momentum plot in Fig. 6, which shows a spurious increase in momentum after each interaction. The energy growth during the simple advection stage is likely due to the finite-grid instability, which is not suppressed by the explicit leapfrog algorithm. The explicit leapfrog algorithm is not strictly energy conserving; it conserves energy to $0(\Delta t^3)$, and it is not stable in the L_2 norm, as shown in Section 4.4. Instabilities will be suppressed only to truncation error and thus, can persist.

In contrast to the explicit leapfrog scheme, the total energy grows to a maximum variation of $\approx 7\%$ in the implicit scheme. Energy conservation is closely approximated, in keeping with the analysis in Sections 4.4 and 4.3. The finite-grid instability, observed in the explicit case, does not exist in the implicit formulation (using $\theta = \frac{1}{2}$) due to the strict constraint of total energy conservation. (Instabilities will be suppressed to round off error rather than truncation error). In [34] it is speculated that large time steps compared with the CFL limit are required to dampen the instability growth. This appears to be incorrect. Our calculations with the stability time step constraint in Eq. (123) with $\alpha = 0.25$ and $\alpha = 0.10$ are stable and energy conserving and show that a large time step is unnecessary for stability.

A small amount of elastic energy accumulates after each wall contact for the implicit scheme. The accumulation is, however, approximately two orders of magnitude less than that seen in the explicit case. Reducing the Newton tolerance in the nonlinear solution reduces this growth in elastic energy. The implicit momentum plot reveals an overall dissipation in kinetic energy which is due to the spatial error wrought in transferring information between grid and particles using a lumped mass matrix (see Section 3.3).

To gauge the accuracy of the impact time, the explicit and implicit results were compared for the problem of an impacting rectangular block (there is no analytical solution for the propagation speed of elastic waves in an impacting circular grain). The effective propagation speeds s (see Eq. (121)) were measured and compared to analytical results for the impacting rectangular block. After the first wall interaction a 3% error was incurred for both implicit and explicit formulations. Thereafter the momentum results were of a character similar to those shown in Fig. 6.
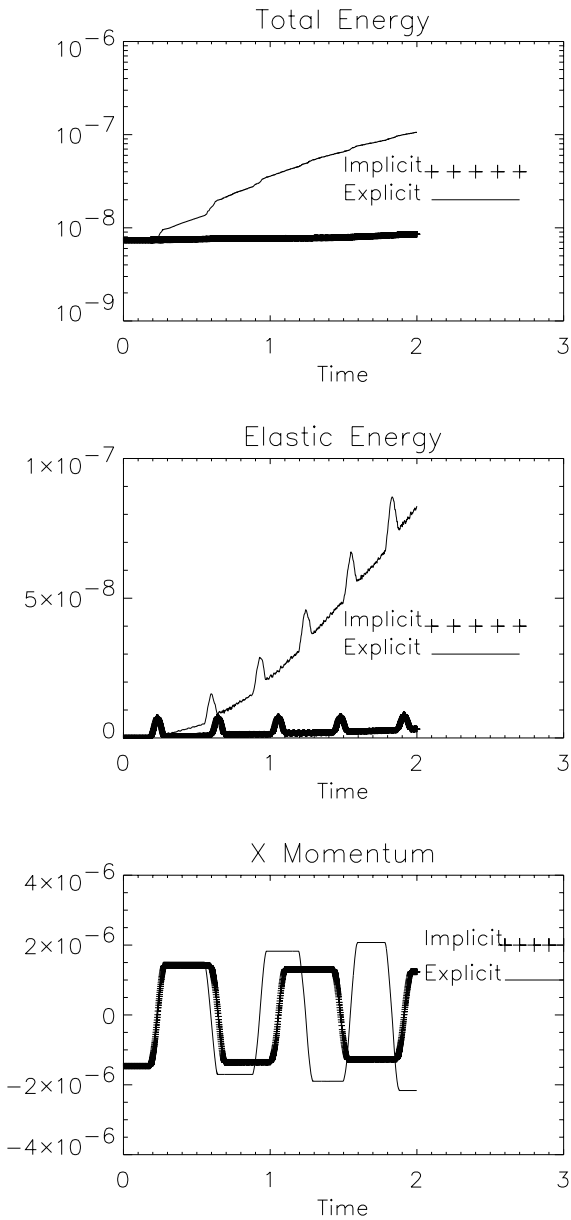
Total Energy

Elastic Energy

X Momentum

FIG. 6. Total energy, elastic energy, and x-momentum comparisons for the implicit and explicit formulations for the rebounding elastic grain problem.

8.4. Compressed Grains

The second problem examined is the dynamic loading of an assembly of grains. The motivation behind the simulation of this problem comes from the experimental work of Rossmanith and Shukla [49]. Here, investigations of dynamic wave propagation in granular media were undertaken in order to study impact wave propagation and load transfer in granular materials. Detonators were used to dynamically load the disks and photoelasticity was employed to visualize the stress distribution. The grains were fabricated from a photoelastic

TABLE IV
Specifications of the Compressed Grain Problem

Resolution	36×176
Δx	3.125×10^{-4} m
Δy	3.125×10^{-4} m
Maximum particles per cell	4
Simulation time	60.00
Δt_{MAX}	$10.0 \times \Delta t_{exp}$
Bulk modulus λ	4.786×10^9 Pa
Shear modulus μ	2.893×10^9 Pa
Density ρ	1.9×10^3 kg/m³
U_o	0.0
V_o	-1.00×10^1 m/s
Loading time	$0.12\ \mu$s

material, which when subjected to a state of stress, produced optical interference fringes. These fringes are contours of the difference in principal stresses.

Recently, Bardenhagen *et al.* [50] employed the explicit MPM formulation to simulate the stress wave propagation through an assembly of disks. The results were compared to photoelastic experiments like those in [49] but where a Hopkinson bar, rather than a detonator, was employed as the dynamic loading mechanism. This allowed for a more easily applied boundary condition (a velocity step function) rather than a variable forcing. The explicit MPM formulation was found to reproduce the character of the stress wave propagation closely although some calibration was required due to the uncertainty of the experimental material properties.

In this example, an assembly of five disks is dynamically loaded using a velocity step function. The purpose of this example is to highlight the extended range of applicability and robustness of the implicit formulation over the explicit formulation due to the suppression of the finite-grid instability. Table IV provides the specifications of the problem and Table V provides the specifications of the implicit formulation of the problem. Figure 7 shows the initial physical configuration. The disks are separated from the left and right boundaries by a distance of $2\Delta x$.

TABLE V
Specifications for the Implicit Simulation of the
Compressed Grain Problem

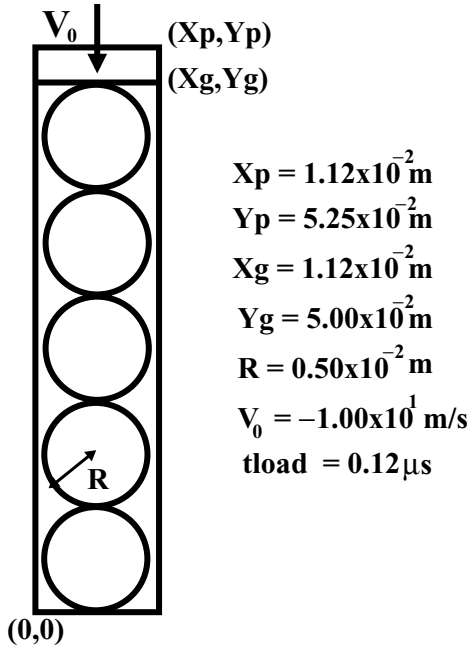| Newton tolerance	10^{-6}
ϵ	10^{-3}
γ	10^{-3}
F_{scale}	$V_o \times \sum_p^{Np} m_p$
Maximum Newton iterations	5
Maximum linear iterations	10
Preconditioner	WJ
Levels	1
Relaxation sweeps	2
θ	0.5

FIG. 7. Schematic showing the initial configuration of the grain compression problem.

The shear and bulk moduli in Table IV were chosen such that they produced the same P_{wave} and S_{wave} speeds reported in [49] for a density $\rho = 1.9 \times 10^3$ kg/m^3. The P_{wave} speed is

$$P_{wave} = \sqrt{\frac{\lambda + 4/3\mu}{\rho}} = 2.133 \times 10^3 \text{ m/s} \tag{124}$$

and the S_{wave} speed is

$$S_{wave} = \sqrt{\mu/\rho} = 1.234 \times 10^3 \text{ m/s}. \tag{125}$$

Hence λ and μ can be calculated. The loading time was chosen so that it corresponded to the same fractional component of the simulation time as in [49].

Figures 8 and 9 display the contours of the fringe patterns of the in-plane principal stress differences. Fringes are generated by taking the cosine of the difference in in-plane principal stress,

$$F_r = \cos\left(\frac{2\pi \times f_{no}(\sigma_1 - \sigma_2)}{\max(\sigma_1 - \sigma_2)}\right). \tag{126}$$

The explicit and implicit calculations display similar average propagation speeds, with the P-wave speed $\approx 2.5 \times 10^3$ m/s. However, as the explicit calculation proceeds, the fringe patterns are dominated by high frequency noise; this is a further example of the finite-grid instability discussed in Section 8.3. The total energy grows as seen in Fig. 10. In contrast, the implicit formulation does not exhibit this high frequency noise and total energy is conserved to within 1%. (The total energy dissipation seen in this figure occurs during the transfer of information from the particles to the grid.)
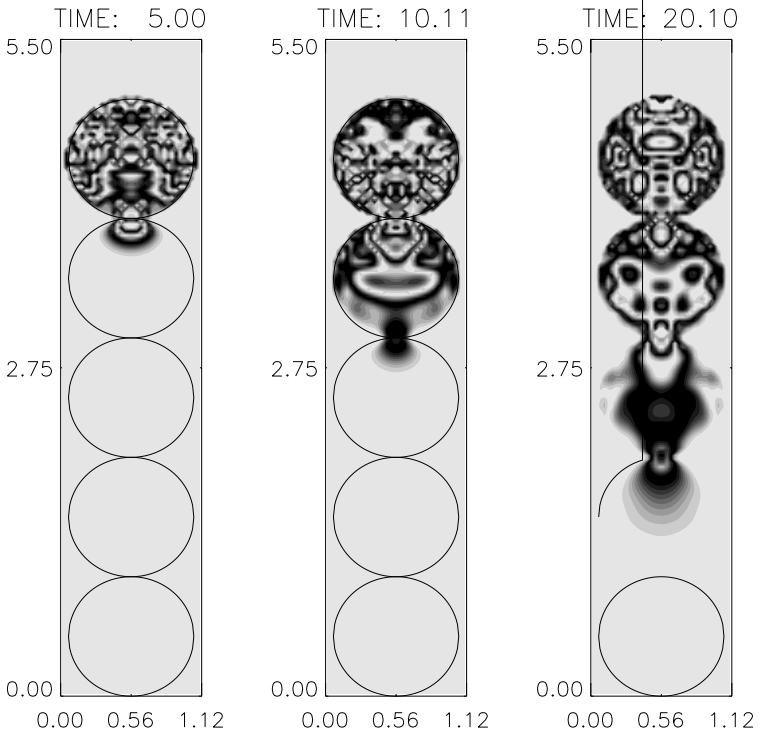
FIG. 8. Fringe plots of the principal stress differences for the compressed grain problem using the implicit MPM formulation.
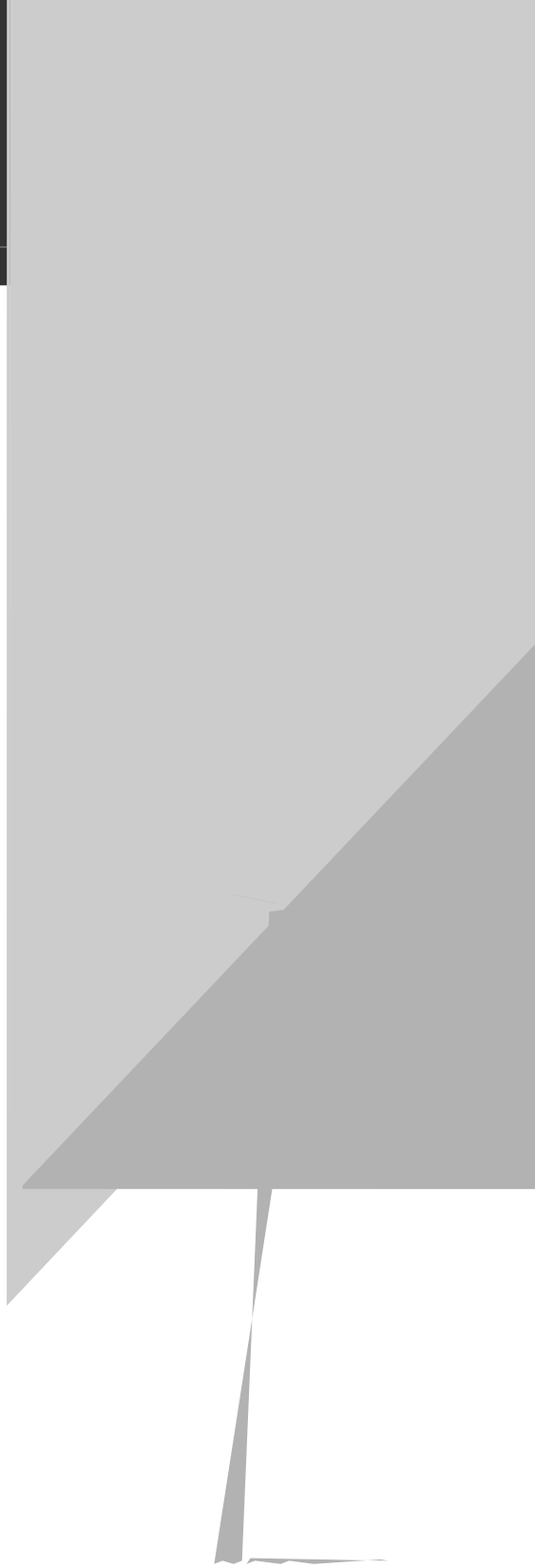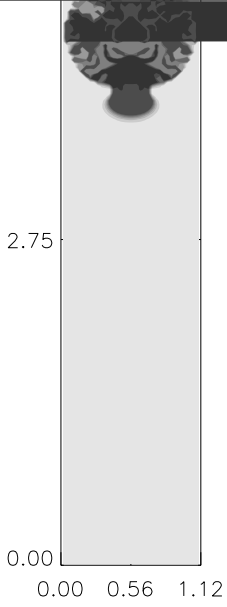
FIG. 9. Fringe plots of the principal stress differences for the compressed grain problem using the explicit MPM formulation. The CFL stability limit is $\alpha = 0.25$.
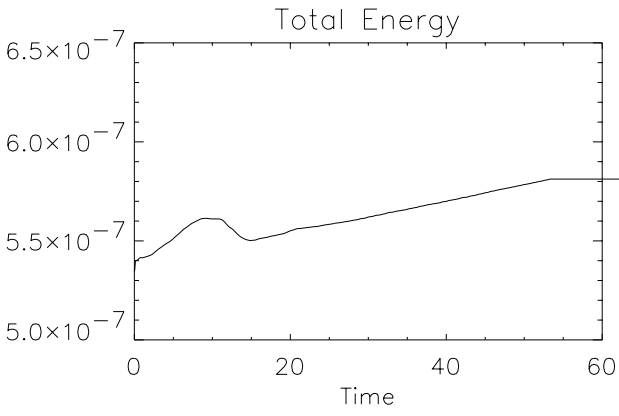
Total Energy

Pa

109
×
48
7

TABLE VII
Specifications for the Implicit Simulation of the
Monodispersed Grain Shearing Problem

Newton tolerance	10^{-6}
ϵ	10^{-3}
γ	10^{-3}
F_{scale}	$V \times \sum_{p}^{Np} m_p$
Maximum Newton iterations	5
Maximum linear iterations	5
Preconditioner	WJ
Levels	1
Relaxation sweeps	2
θ	0.5

the specifications of the implicit formulation of the problem. Figure 11 shows the initial physical configuration. Shear is simulated by attaching grains to the left and right boundary and the right boundary is moved with speed V. Periodic boundary conditions are imposed on the top and bottom boundaries. The simulation is run to a time ($t = 40.0 \ \mu$s) such that the average shear strain exceeds 300%.

Figures 12 and 13 display the grain positions and the principal stress differences for the implicit and explicit cases at various times. Light shading in the principal stress plots indicates larger values of principal stress differences and hence the existence of stress chains. Comparing the explicit calculations with those in [37] reveals the disorder propagates more slowly using the volume-weighted contact force. By time $t = 40 \ \mu$s (which corresponds to an average shear strain of 330%), approximately one third of the structure is still regular near the center. Without the volume weighting, the overall packing order is lost by time $t = 35 \ \mu$s (which corresponds to an average shear strain of 284%). The volume-weighted contact force allows a denser row packing before the onset of row buckling. Therefore the disorder from the left takes longer to initiate and propagate through to the center. As noted in [37], as the disorder propagates toward the center, the force chain density reduces and a small number of primary load paths result.
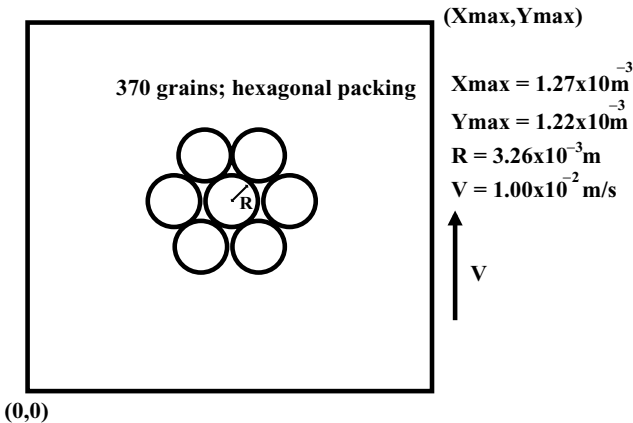

FIG. 11. Schematic showing the initial configuration of the monodisperse shearing problem.
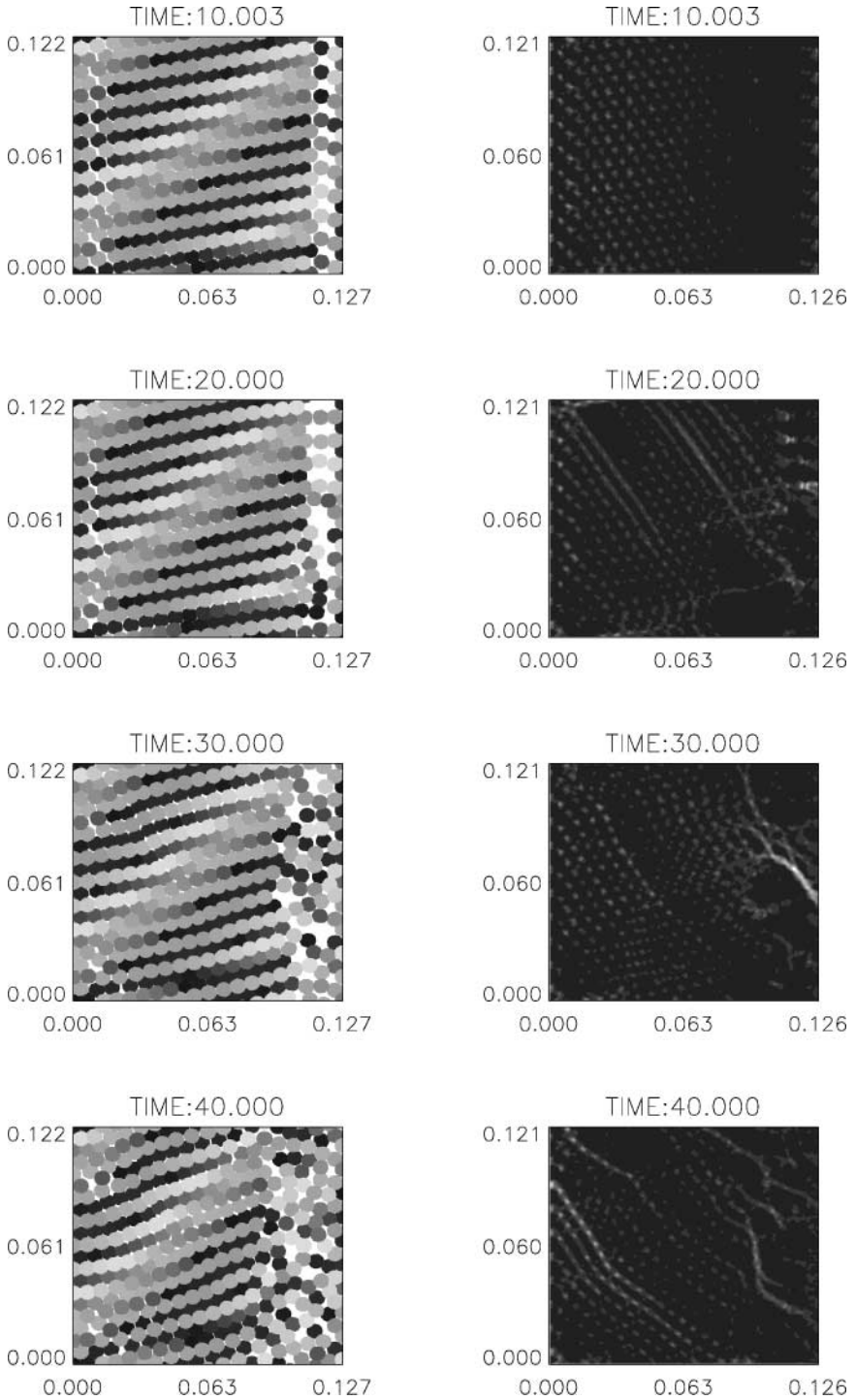
FIG. 12. Particle positions and principal stress differences in the monodisperse shearing problem using the implicit MPM formulation.
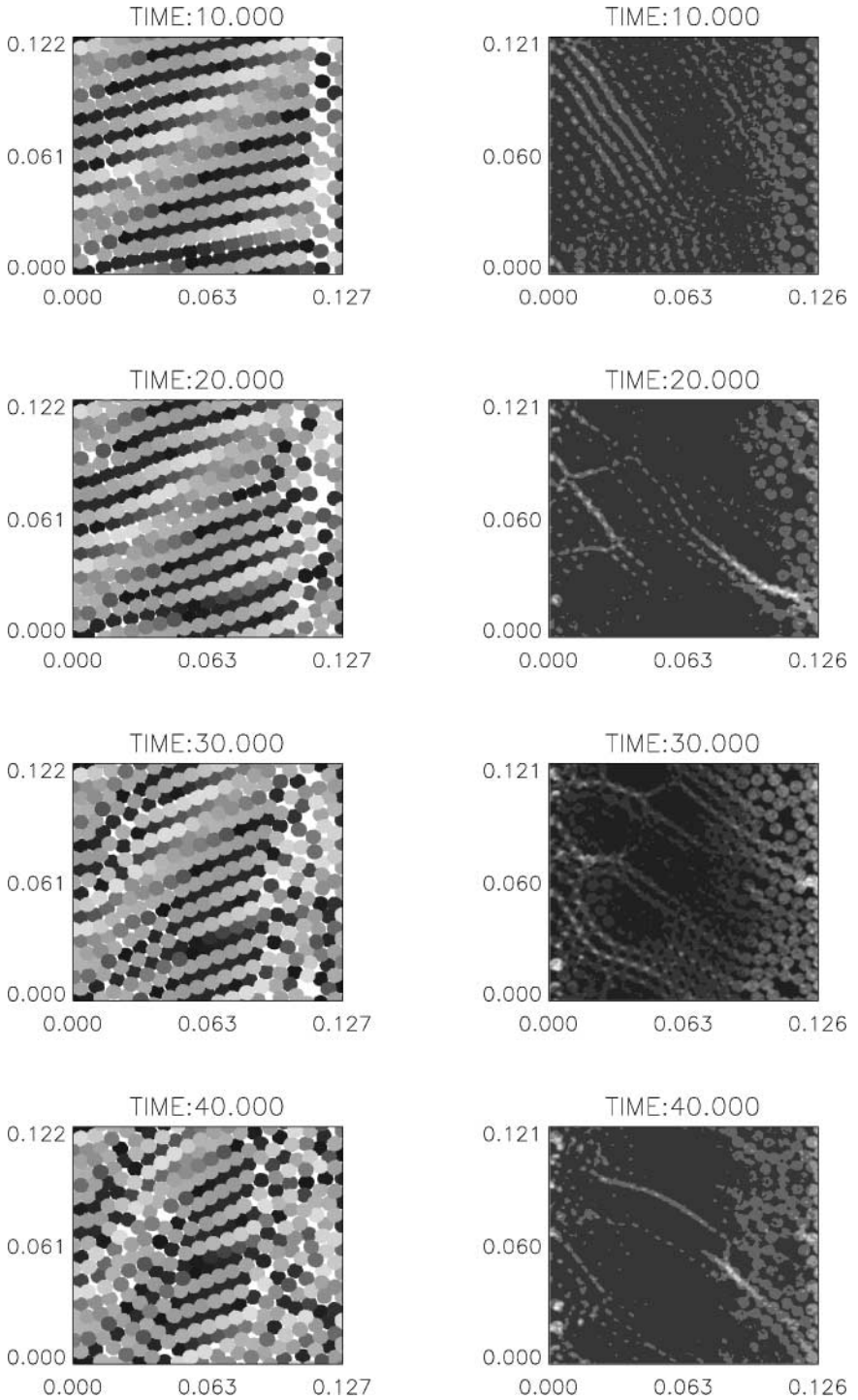
FIG. 13. Particle positions and principal stress differences in the monodisperse shearing problem using the explicit MPM formulation.

In the implicit results, the disorder propagates even more slowly than the explicit calcu-lation, particularly from the left boundary. It is postulated that the more rapid development of disorder seen in the explicit calculations is a result of the lack of energy conservation associated with the explicit leapfrog algorithm. The results of the rebounding elastic grain in Section 8.3 revealed spurious elastic energy, residual stresses, and momentum production after each wall collision for the explicit calculation. Strict energy conservation imposed by the implicit algorithm prevents these spurious productions, and a longer time is required to reach a similarly disordered state.

Plots of total energy and elastic energy for both the implicit and explicit simulations are shown in Fig. 14. As with the rebounding grain problem, the total energy grows by two orders of magnitude more for the explicit calculation than the implicit calculation. Elastic energy is the primary contributor to the energy growth in the explicit simulation. It continually grows indicating the existence of spurious residual grain stresses after collisions. The energy dissipated by plasticity ensures the explicit calculation remains robust without grain spallation. In the implicit calculation, the elastic energy does not grow monotonically but oscillates as grains collide and rebound. The main contribution to the total energy is kinetic energy.
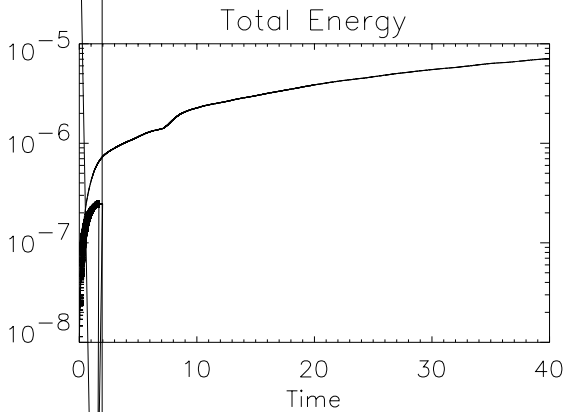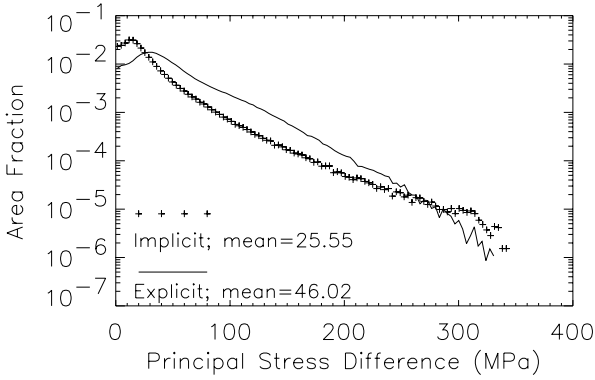
FIG. 15. Statistics of the principal stress differences for the implicit and explicit MPM formulations for the monodisperse shearing problem.

The existence of residual stresses in the explicit simulation is verified in the histogram plot in Fig. 15. This plot reveals the fractional area within which the computed principal stress difference lies within a small interval. Data is taken from times $t > 10.0\,\mu$s after which the initial order is disrupted. The histogram shows that the explicit algorithm consistently produces more grains with principal stress differences >30 MPa. As a result, the mean principal stress difference is nearly twice that produced from the implicit calculation. As noted in [37], the explicit distribution is exponential for principal stress differences larger than the mean. The implicit distribution deviates from an exponential distribution at small values of the stress, most likely because grains on the left are not yet strained, as a steady state has not been reached yet. Both distributions are bounded by $\Delta\sigma \leq 2/\sqrt{3}\sigma_y = 346$ MPa.

9. CONCLUSIONS

A fully implicit formulation of MPM has been developed and implemented using a matrix-free Newton–Krylov algorithm. The algorithm is ideally suited to the nonlinear and nonanalytic characteristics of MPM that result from grain plasticity and intergranular contact and will easily accommodate improved physics, such as grain fracture and intergranular binding.

An important result of this work is the elimination of the finite grid instability by the implicit formulation, and a demonstration of the importance of stability in the L_2 norm. Rigorous energy conservation results in a more robust and physical method, which gives high quality results for a wider range of strain rates than before. Limits on the time step, which prevent values of strain that result in the yield stress for real materials being exceeded, impose constraints on the maximum time step that can be used. However, the implicit algorithm with time steps that exceed the Courant stability limit yields more accuracy than can be achieved with the explicit algorithm. This occurs even when the explicit scheme uses time steps much smaller than the Courant limit.

Computational results show that simple diagonal preconditioners converge as rapidly as more complex multigrid preconditioners for time steps where convergence is reliable. One explanation for this may be that the class of preconditioners considered so far fail to expand the region of convergence of the nonlinear, Newton iteration. This places a high value on

efficient implementation with the present preconditioner and provides strong incentives for the development of more effective preconditioners in the future.

ACKNOWLEDGMENTS

REFERENCES

1. D. Sulsky, Z. Chen, and H. L. Schreyer, A particle method for history-dependent materials, *Comput. Meth. Appl. Mech. Eng.* **118**, 179 (1994).

2. A. Mehta, Real sandpiles: Dilatency, hysteresis and cooperative dynamics, *Physica A* **186**, 121 (1992).

3. P. A. Cundall and O. D. L. Strack, Discrete numerical model for granular assemblies, *Geotechnique* **29**, 47 (1979).

4. Farhang Radjai, Michel Jean, Jean-Jacques Moreau, and Stéphane Roux, Force distributions in dense two-dimensional granular systems, *Phys. Rev. Lett.* **77**, 274 (1996).

5. J. U. Brackbill and D. W. Forslund, Simulation of low-frequency electromagnetic phenomena in plasmas, in *Multiple Time Scales*, edited by J. U. Brackbill and B. I. Cohen (Academic Press, San Diego, 1985), p. 271.

6. D. Sulsky and J. U. Brackbill, A numerical method for suspension flow, *J. Comput. Phys.* **96**, 339 (1991).

7. J. U. Brackbill, FLIP MHD: A particle-in-cell method for magnetohydrodynamics, *J. Comput. Phys.* **96**, 163 (1991).

8. D. Sulsky, S.-J. Zhou, and H. L. Schreyer, Application of a particle-in-cell method to solid mechanics, *Comput. Phys. Commun.* **87**, 236 (1995).

9. Charles K. Birdsall and A. Bruce Langdon, *Plasma Physics via Computer Simulation* (McGraw–Hill, New York, 1985).

10. R. M. Nedderman, U. Tuzun, S. B. Savage, and G. T. Houlsby, The flow of granular materials—I: Discharge rates from hoppers, *Chem Eng. Sci.* **37**, 1597 (1982).

11. B. Fornberg, On the instability of leap-frog and Crank–Nicholson approximations of a nonlinear partial differential equation, *Math. Comput.* **27**, 47 (1973).

12. David J. Benson, An implicit multi-material Eulerian formulation, *Int. J. Numer. Meth. Eng.* **48**, 475 (2000).

13. W. L. Elban and M. A. Chiarito, Quasi-static compaction study of coarse HMX explosive, *Powder Tech.* **46**, 191 (1986).

14. Vidar Thomée, From finite differences to finite elements. A short history of numerical analysis of partial differential equations, *J. Comput. Appl. Math.* **128**, 1 (2001).

15. C. S. Peskin, Flow patterns around heart valves: A numerical method, *J. Comput. Phys.* **10**, 252 (1972).

16. J. K. Morgan and M. S. Boettcher, Numerical simulations of granular shear zones using the distinct element method: 1. Shear zone kinematics and the micromechanics of localization, *J. Geophys. Res.* **104**, 2703 (1999).

17. J. K. Morgan, Numerical simulation of granular shear zones using the distinct element method: 2. Effects of particle size distribution and interparticle friction on mechanical behavior, *J. Geophys. Res.* **104**, 2721 (1999).

18. Francis H. Harlow and J. Eddie Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, *Phys. Fluids* **8**, 2182 (1965).

19. D. W. Howell, R. P. Behringer, and C. T. Veje, Fluctuations in granular media, *Chaos* **9**, 559 (1999).

20. Douglas K. Lilly, On the computational stability of numerical solutions of time-dependent non-linear geophysical fluid dynamics problems, *Mon. Weather Rev.* **93**, 11 (1965).

21. C.-h. Liu, S. R. Nagel, D. A. Schechter, S. N. Coppersmith, S. Majumdar, O. Narayan, and T. A. Witten, Force fluctuations in bead packs, *Science* **269**, 513 (1995).

22. Mario Nicodemi, Antonio Coniglio, and Hans Herrmann, Compaction and force propagation in granular packings, *Physica A* **240**, 405 (1997).

23. L. P. Kadanoff, S. R. Nagel, L. Wu, and S. M. Zhou, Scaling and universality in avalanches, *Phys. Rev. A* **39**, 6524 (1989).

24. B. Miller, C. O'Hern, and R. P. Behringer, Stress fluctuations for continuously sheared granular materials, *Phys. Rev. Lett.* **77**, 3110 (1996).

25. P. Meakin and R. Jullien, 3-dimensional model for particle-size segregation by shaking, *Phys. Rev. Lett.* **69**, 640 (1992).

26. H. M. Jaeger, S. R. Nagel, and R. P. Behringer, Granular solids, liquids, and gases, *Rev. Mod. Phys.* **68**, 1259 (1996).

27. Blair Perot, Conservation properties of unstructured staggered mesh schemes, *J. Comput. Phys.* **159**, 58 (2000).

28. J. M. Sanz-Serna, Studies in numerical nonlinear instability. 1. Why do leapfrog schemes go unstable? *SIAM J. Sci. Stat. Comput.* **6**, 923 (1985).

29. O. J. Schwarz, Y. Horie, and M. Shearer, Discrete element investigation of stress fluctuations in granular flow at high strain rates, *Phys. Rev. E* **57**, 2053 (1998).

30. Alexey K. Mazur, Common molecular dynamics algorithms revisited: Accuracy and optimal time steps of Stormer-leapfrog integrators, *J. Comput. Phys.* **136**, 354 (1997).

31. Robert D. Richtmyer and K. W. Morton, *Difference Methods for Initial-Value Problems* (Wiley, New York, 1957).

32. D. Sulsky and H. L. Schreyer, Axisymmetric form of the material point method with applications to upsetting and Taylor impact problems, *Comput. Meth. Appl. Mech. Eng.* **139**, 409 (1996).

33. A. J. M. Spencer, *Continuum Mechanics* (Wiley, New York, 1980).

34. J. U. Brackbill, The ringing instability in particle-in-cell calculations of low-speed flow, *J. Comput. Phys.* **75**, 469 (1988).

35. J. U. Brackbill and H. M. Ruppel, FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions, *J. Comput. Phys.* **65**, 314 (1986).

36. S. G. Bardenhagen, M. G. Stout, and G. T. Gray, Three-dimensional, finite-deformation, viscoplastic constitutive models for polymeric materials, *Mech. Mat.* **25**, 235 (1997).

37. S. G. Bardenhagen, J. U. Brackbill, and D. Sulsky, Numerical study of stress distribution in sheared granular material in two dimensions, *Phys. Rev. E* **62**, 3882 (2000).

38. S. G. Bardenhagen, J. U. Brackbill, and D. Sulsky, The material-point method for granular materials, *Comput. Meth. Appl. Mech. Eng.* **187**, 529 (2000).

39. S. G. Bardenhagen and J. U. Brackbill, Dynamic stress bridging in granular material, *J. Appl. Phys.* **83**, 5732 (1998).

40. David E. Stewart, Rigid-body dynamics with friction and impact, *SIAM Rev.* **42**, 3 (2000).

41. P. N. Brown and Y. Saad, Convergence theory of nonlinear Newton–Krylov algorithms, *SIAM J. Optim.* **4**, 297 (1994).

42. P. R. McHugh and D. A. Knoll, Comparison of standard and matrix-free implementations of several Newton–Krylov solvers, *AIAA J.* **34**, 2394 (1994).

43. D. A. Knoll and W. J. Rider, A multigrid preconditioned Newton–Krylov method, *SIAM J. Sci. Comput.* **21**, 691 (1999).

44. R. L. Williamson, Parametric studies of dynamic powder consolidation using a particle-level numerical model, *J. Appl. Phys.* **68**, 1287 (1990).

45. D. J. Benson, W. Tong, and G. Ravichandran, An analysis by direct numerical simulation of the effects of particle morphology on the shock compaction of copper powder, *Model. Sim. Mat. Sci. Eng.* **2**, 535 (1994).

46. D. J. Benson, A mixture theory for contact in multi-material Eulerian formulations, *Comput. Meth. Appl. Mech. Eng.* **140**, 59 (1997).

47. E. A. Fadlun, R. Verzicco, P. Orlandi, and J. MohdYusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, *J. Comput. Phys.* **161**, 35 (2000).

48. D. Burgess, D. Sulsky, and J. U. Brackbill, Mass matrix formulation of the FLIP Particle-in-Cell method, *J. Comput. Phys.* **103**, 1 (1992).

49. H. P. Rossmanith and A. Shukla, Photoelastic investigation of dynamic load transfer in granular media, *Acta Mech.* **42**, 211 (1982).

50. S. G. Bardenhagen, J. E. Guilkey, K. M. Roessig, J. U. Brackbill, W. M. Witzel, and J. C. Foster, An improved contact algorithm for the material point method and application to stress propagation in granular materials, private communication.

51. T. Chan and K. Jackson, Nonlinearly preconditioned Krylov subspace methods for discrete Newton algorithms, *SIAM J. Sci. Stat. Comput.* **5**, 533 (1984).

52. P. Brown and Y. Saad, Hybrid Krylov methods for nonlinear systems of equations, *SIAM J. Sci. Stat. Comput.* **11**, 450 (1990).

53. D. Knoll, D. Kothe, and B. Lally, A new nonlinear solution method for phase-change problems, *Numer. Heat Transfer Part B—Fundamentals* **35**, 439 (1999).

54. D. Knoll and P. McHugh, Newton–Krylov methods for low-mach-number compressible combustion, *AIAA J.* **34**, 961 (1996).

55. J. E. Guilkey and J. A. Weiss, An implicit time integration strategy for use with the material point method, computational fluid and solid mechanics, in *Proceedings of the First MIT Conference on Computational Fluid and Solid Mechanics, June 12–15, 2001, Cambridge, MA*, edited by K. J. Bathe (Elsevier, Amsterdam, 2001).

56. Youcef Saad and Martin H. Schultz, GMRES: A generalised minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.* **7**, 856 (1986).

57. D. A. Knoll, W. J. Rider, and G. L. Olsen, An efficient nonlinear solution method for non-equilibrium radiation diffusion, *J. Quant. Spectrosc. Radiative Transfer* **63**, 15 (1999).

58. D. A. Knoll and W. J. Rider, A multigrid preconditioned Newton–Krylov method, *SIAM J. Sci. Comput.* **21**, 691 (1999).

59. L. Chacon, D. C. Barnes, D. A. Knoll, and G. H. Miley, An implicit energy-conservative 2D Fokker–Planck algorithm: II. Jacobian-free Newton–Krylov solver, *J. Comput. Phys.* **157**, 654 (2000).

60. A. Brandt, Multilevel adaptive solution to boundary value problems, *Math. Comput.* **31**, 333 (1977).