Using Nesterov's method to accelerate multibody dynamics with friction and contact

Using Nesterov's Method to Accelerate Multibody Dynamics with Friction and Contact

HAMMAD MAZHAR*, TOBY HEYN*† and DAN NEGRUT*
*Department of Mechanical Engineering, University of Wisconsin-Madison, Madison, WI
†Epic Systems, Verona, WI
and
ALESSANDRO TASORA‡
‡Department of Industrial Engineering, University of Parma, Parma, Italy

We present a solution method that, compared to the traditional Gauss-Seidel approach, reduces the time required to simulate the dynamics of large systems of rigid bodies interacting through frictional contact by one to two orders of magnitude. Unlike Gauss-Seidel, it can be easily parallelized, which allows for the physics-based simulation of systems with millions of bodies. The proposed Accelerated Projected Gradient Descent (APGD) method relies on an approach by Nesterov in which a quadratic optimization problem with conic constraints is solved at each simulation time step to recover the normal and friction forces present in the system. The APGD method is validated against experimental data, compared in terms of speed of convergence and solution time with the Gauss-Seidel and Jacobi methods, and demonstrated in conjunction with snow modeling, bulldozer dynamics, and several benchmark tests that highlight the interplay between the friction and cohesion forces.

Categories and Subject Descriptors: G.1.0 [**Mathematics of Computing**]: Numerical Analysis—*Numerical algorithms*; I.6.8 [**Simulation And Modeling**]: Types of Simulation—*Parallel*

General Terms: Algorithms, Performance

Additional Key Words and Phrases: rigid body dynamics, friction, contact, physics-based simulation

ACM Reference Format:

H. Mazhar, T. D. Heyn, A. Tasora and D. Negrut. Using Nesterov's Method to Accelerate Multibody Dynamics with Friction and Contact.

The authors acknowledge an Army Research Office award W911NF-12-1-0395 and a National Science Foundation award CMMI-GOALI–1362583.
e-mail: hmazhar@wisc.edu, tdheyn@gmail.com, tasora@ied.unipr.it, negrut@wisc.edu.

DOI
http://doi.acm.org/

1. INTRODUCTION

The literature on the topic of frictional contact in multibody dynamics is vast, yet at a high level most of the existing methods can be grouped into two categories. First, and most common, are approaches that use a penalty, or regularization, approach in handling contact between bodies. Second, there are approaches built around a methodology that extends the concept of equations of motion to include differential inclusions [Filippov 1967]. For frictional contact, this methodology leads to differential variational inequality (DVI) problems, which upon discretization assume the form of mathematical programs with equilibrium and complementarity constraints.

Despite being used in the context of rigid body dynamics, the penalty method (PM) assumes the bodies deform just slightly at the contact point. Employing the finite element method to characterize this deformation would incur a stiff computational cost. Therefore, at each time step, the deformation of the bodies is approximated during the collision detection stage of the solution by relating it to the amount of interpenetration between the otherwise rigid bodies. The process of defining a surrogate deformation of the contacting bodies leads to a plethora of algorithms for handling frictional contact in the penalty method. Although the geometry of the bodies in contact might be overly complex, it is customary to combine the surrogate deformation to the Hertzian theory for sphere-to-sphere or sphere-to-plane contact, see for instance [Johnson 1987], in order to yield a methodology to produce the contact force. With the contact force in hand, various approaches have been proposed to produce a friction force at the point of contact. This regularization of the contact discontinuity comes at the price of (i) a difficult and somewhat ad-hoc process of identifying model parameters, (ii) small integration time steps limited in size on numerical stability grounds, and (iii) a perceived randomness in the contact force. For (i), there are model parameters, e.g.: stiffness, damping, creep, etc., that are not always constant over time and might depend on the geometry of the bodies in contact, the strategy adopted to compute the surrogate deformation, and even the choice of time-discretization scheme. Choosing the values of these parameters is not straightforward and it is one aspect that challenges the predictive attribute of the penalty method. Nonetheless, this has been the approach of choice in the vast majority of the numerical studies of the dynamics of large rigid body systems, see for instance [Cundall 1971; Cundall and Strack 1979; Cundall 1988; Jaeger et al. 1996; Brilliantov et al. 1996; Vu-Quoc and Zhang 1999; Vu-Quoc et al. 2004; Luding 2005; Pöschel and Schwager 2005]. Finally, in terms of (iii), the randomness can be tied back to the undesired influence that the

integration step-size has on the surrogate deformation at the next time step, see discussion in [Kaufman and Pai 2012].

A wealth of methodologies approach the frictional contact problem within the DVI framework. Methods proposed in [Bertails-Descoubes et al. 2011; Daviet et al. 2011; Acary et al. 2011] use a nonsmooth Newton approach to solve a fixed point iteration problem. The solution of a frictional contact induced nonlinear complementarity problem (NCP) is pursued in [Daviet et al. 2011] by means of a Newton method applied in conjunction with an appropriately defined Fischer-Burmeister complementarity function, an approach that is backed up as a fail-safe by a more robust albeit costlier local solver [Bonnefon and Daviet 2011]. The DVI approach has been modified beyond sliding friction to model rolling and spinning friction by introducing new constraints for each contact, see for instance [Tasora and Anitescu 2013]. A mass splitting procedure was embedded into a DVI formulation to improve numerical stability and reduce jitter for piles and stacks of objects in [Tonge et al. 2012]. The overall solution methodology adopted herein shares several traits with the one proposed in [Kaufman and Pai 2012]: both impose complementarity between normal force and the distance gap instead of bringing into the discussion the relative velocities at the contact point. Furthermore, like in [Kaufman and Pai 2012], symplectic integration is considered for numerical discretization of the equations of motion. While here the discretization is based on a half-implicit Euler method, [Kaufman and Pai 2012] present an approach that accommodates a spectrum of formulas: implicit midpoint, Newmark, and an implicit-explicit formula due to [Kane et al. 1999]. In handling impact and constraint drift the approach of [Kaufman and Pai 2012] is more versatile compared to the our approach which in the current implementation handles inelastic collision and relies on a Baumgarte-style penalty to prevent constraint violation or drift. However, on the upside, the approach herein handles friction and is guaranteed to produce a global solution for systems with friction and contact.

This paper concentrates solely on the DVI-based approach for the following reasons: (a1) it allows for large integration times steps, (b1) it handles the stick-slip conditions in an effective way, (c1) it integrates well into the rigid-body model embraced in formulating the equations of motion, and (d1) it can be easily modified to account for cohesion. The DVI approach is not without its own issues: e.g., (a2) it displays a lack of uniqueness in the frictional contact force distribution as the rigid body model in conjunction with the classical dry-friction model of Da Vinci-Amontons-Coulomb can lead to an under constrained problem [Delannay et al. 2007]; (b2) it calls for the solution of a large optimization problem that poses scalability challenges; and (c2) unless special measures are taken, see for [Smith et al. 2012], most DVI approaches render impact phenomena as zero coefficient of restitution events; i.e., the impacts are plastic.

This paper is concerned with (b2). It proposes a method aimed at speeding up the solution of the optimization problem that represents the manifest computational bottleneck of the DVI approach. Beyond being instrumental in the context of the frictional-contact solution approach embraced herein, the method can be used in numerous other graphics-specific applications: for handling fluid-solid interaction problems [Bodin et al. 2012], in an approach like the one proposed in [Smith et al. 2012] to handle friction, contact, and impact phenomena [Smith et al. 2012], or in a mass splitting framework aimed at increasing robustness [Tonge et al. 2012]. Broadly speaking, the method proposed can replace Gauss-Seidel or Jacobi-based solutions in variational approaches to handling friction and contact where a global complementarity problem can be equivalently formulated as a constrained minimization problem.

The paper is organized as follows. Section 2 poses the multibody dynamics problem and outlines the DVI-based solution process. This discussion provides the backdrop for the APGD solution method introduced in Section 3. In Section 4 we validate the APGD method against experimental data and compare its efficiency with that of the Gauss-Seidel approach and other established methods that build on the Mosek and PATH solvers [Andersen and Andersen 2000; Andersen et al. 2003; Dirkse and Ferris 1995]. This section also demonstrates the performance of the APGD method in conjunction with snow/clay modeling, bulldozer dynamics, and several benchmark tests that highlight the interplay between friction and cohesion. The discussion in Section 5 places the APGD method and the overall solution approach in the frictional contact numerical solution landscape familiar to the computer graphics audience. Section 6 summarizes directions of future work and our contributions, which are identified as being (1) the APGD method for DVI–based handling of frictional contact in rigid multibody dynamics, and (2) an expeditious way of adding cohesion into the contact model.

2. BACKGROUND

Herein, the set of generalized coordinates used to position and orient a rigid body j in the 3D Euclidean space are $\mathbf{r}_j \in \mathbb{R}^3$ and $\epsilon_j \in \mathbb{R}^4$ [Haug 1989]. The former provides the absolute position of the center of mass of body j, while the latter represents a set of Euler parameters (quaternions) that characterize body orientation in a global reference frame. The set of generalized coordinates for a system of n_b bodies works out to be $\mathbf{q} = \left[\mathbf{r}_1^T, \epsilon_1^T, \ \ldots, \mathbf{r}_{n_b}^T, \epsilon_{n_b}^T\right]^T \in \mathbb{R}^{7n_b}$ and their time derivatives $\dot{\mathbf{q}} = \left[\dot{\mathbf{r}}_1^T, \dot{\epsilon}_1^T, \ldots, \dot{\mathbf{r}}_{n_b}^T, \dot{\epsilon}_{n_b}^T\right]^T \in \mathbb{R}^{7n_b}$. Rather than using $\dot{\mathbf{q}}$ to pose the Newton-Euler equations of motion, the array $\mathbf{v} = \left[\dot{\mathbf{r}}_1^T, \bar{\omega}_1^T, \ldots, \dot{\mathbf{r}}_{n_b}^T, \bar{\omega}_{n_b}^T\right]^T \in \mathbb{R}^{6n_b}$ is used since it leads to: (i) a smaller problem; and (ii) a constant, symmetric and positive definite mass matrix. There is a simple linear transformation that for each body B relates its angular velocity expressed in the body-fixed reference frame, $\bar{\omega}_B$, to the time derivatives of the Euler parameters $\dot{\epsilon}_B$. Specifically, $\bar{\omega}_B = 2\mathbf{G}(\epsilon_B)\dot{\epsilon}_B$, where the entries in the matrix $\mathbf{G} \in \mathbb{R}^{3\times 4}$ depend linearly on the Euler parameters ϵ_B [Haug 1989]. Defining the block diagonal matrix $\mathbf{L}(\mathbf{q}) \equiv \text{diag}\left[\mathbf{I}_{3\times 3}, \frac{1}{2}\mathbf{G}^T(\epsilon_1), \ldots, \mathbf{I}_{3\times 3}, \frac{1}{2}\mathbf{G}^T(\epsilon_{n_b})\right] \in \mathbb{R}^{7n_b\times 6n_b}$, where $\mathbf{I}_{3\times 3}$ is the identity matrix, yields $\dot{\mathbf{q}} = \mathbf{L}(\mathbf{q})\mathbf{v}$.

2.1 Modeling Aspects

Consider the contact between two bodies A and B represented in Fig. 1. Assuming that the body geometries are regular at the contact point, the contact point along with the shared tangent plane are used to define two local reference frames, one for each body. For body A, the normal $\mathbf{n}_{i,A}$ at contact point i is chosen to be perpendicular on the tangent plane at the shared contact point and to point towards the exterior of body A. Two mutually perpendicular unit vectors $\mathbf{u}_{i,A}$ and $\mathbf{w}_{i,A}$ are chosen to define a right-hand local reference frame associated with contact i on body A. A similar sequence of steps is followed to define a local reference frame for body B based on $\mathbf{n}_{i,B}$, $\mathbf{u}_{i,B}$, $\mathbf{w}_{i,B} \in \mathbb{R}^3$. The Lagrange multiplier $\widehat{\gamma}$ associated with contact i is used to pose a complementarity condition in relation to the gap (distance) Φ between bodies A and B: $0 \leq \widehat{\gamma}_{i,n} \perp \Phi_i(\mathbf{q}) \geq 0$. When the bodies in contact have a smooth convex geometry, producing the gap function is straightforward. For complex and/or nonconvex geometries defining $\Phi_i(\mathbf{q})$ might pose difficulties [Anitescu et al. 1996; Flickinger et al. 2013], an issue not addressed herein.

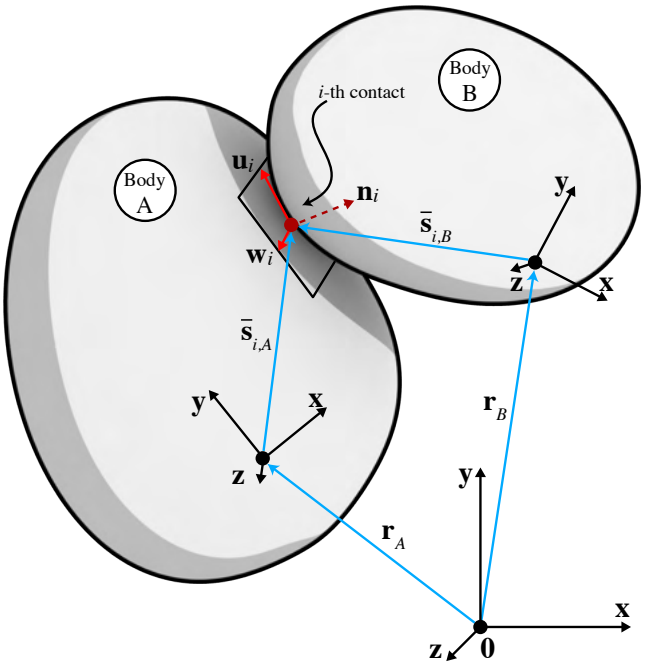

Fig. 1. Contact i between two bodies $A, B \in \{1, 2, \ldots, n_b\}$.

In what follows, by convention, $\boldsymbol{\alpha}_i \equiv \boldsymbol{\alpha}_{i,A}$, for $\alpha \in \{\mathbf{n}, \mathbf{u}, \mathbf{w}\}$. The force associated with contact i can then be decomposed into the normal component, $\boldsymbol{F}_{i,N} = \hat{\gamma}_{i,n}\boldsymbol{n}_i$, and the tangential component, $\boldsymbol{F}_{i,T} = \hat{\gamma}_{i,u}\boldsymbol{u}_i + \hat{\gamma}_{i,w}\boldsymbol{w}_i$, where the multipliers $\hat{\gamma}_{i,n} > 0$, $\hat{\gamma}_{i,u}$, and $\hat{\gamma}_{i,w}$ represent the magnitude of the force in each direction. The friction forces are assumed to satisfy the Coulomb dry-friction model, which can be expressed as [Stewart and Trinkle 1996; Stewart 2000]

$$\begin{aligned} \sqrt{\hat{\gamma}_{i,u}^2 + \hat{\gamma}_{i,w}^2}, &\leq \mu_i \hat{\gamma}_{i,n} \\ \|\boldsymbol{v}_{i,T}\| \left(\sqrt{\hat{\gamma}_{i,u}^2 + \hat{\gamma}_{i,w}^2} - \mu_i \hat{\gamma}_{i,n} \right) &= 0, \\ \langle \boldsymbol{F}_{i,T}, \boldsymbol{v}_{i,T} \rangle &= -\|\boldsymbol{F}_{i,T}\| \|\boldsymbol{v}_{i,T}\| \,, \end{aligned} \tag{1}$$

where $\boldsymbol{v}_{i,T}$ represents the relative tangential velocity between bodies A and B at the point of contact. These equations represent the first order Karush-Kuhn-Tucker optimality condition for the following optimization problem in two dummy variables $y, z \in \mathbb{R}$:

$$(\widehat{\gamma}_{i,u}, \widehat{\gamma}_{i,w}) = \underset{\sqrt{y^2+z^2} \leq \mu_i \widehat{\gamma}_{i,n}}{\operatorname{argmin}} \quad \mathbf{v}_{i,T}^T \left(y\mathbf{u}_i + z\mathbf{w}_i \right). \tag{2}$$

The force at the i^{th} contact point can be expressed as $\boldsymbol{F}_i = \boldsymbol{F}_{i,N} + \boldsymbol{F}_{i,T} = \hat{\gamma}_{i,n}\boldsymbol{n}_i + \hat{\gamma}_{i,u}\boldsymbol{u}_i + \hat{\gamma}_{i,w}\boldsymbol{w}_i \in \Upsilon_i$, where Υ_i is a 3D cone of slope $\tan^{-1}\mu_i$, i.e., $\Upsilon_i = \{[x, y, z]^T \in \mathbb{R}^3 | \sqrt{y^2 + z^2} \leq \mu_i x\}$, oriented along $\boldsymbol{n}_i$ and with its tip at the contact point.

The Newton-Euler equations of motion [Stewart and Trinkle 1996] then assume the following expression:

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{L}(\mathbf{q})\mathbf{v} \\ \mathbf{M}\dot{\mathbf{v}} &= \mathbf{f}(t, \mathbf{q}, \mathbf{v}) \\ &+ \sum_{i \in \mathcal{A}(\mathbf{q}, \delta)} \left(\widehat{\gamma}_{i,n}\, \mathbf{D}_{i,n} + \widehat{\gamma}_{i,u}\, \mathbf{D}_{i,u} + \widehat{\gamma}_{i,w}\, \mathbf{D}_{i,w} \right) \\ i \in \mathcal{A}(\mathbf{q}(t), \delta) \;&:\; 0 \leq \widehat{\gamma}_{i,n} \;\perp\; \Phi_i(\mathbf{q}) \geq 0 \\ (\widehat{\gamma}_{i,u}, \widehat{\gamma}_{i,w}) &= \underset{\sqrt{y^2+z^2} \leq \mu_i \widehat{\gamma}_{i,n}}{\operatorname{argmin}} \quad \mathbf{v}^T \left(y\, \mathbf{D}_{i,u} + z\, \mathbf{D}_{i,w} \right). \end{aligned} \tag{3}$$

The tangent space generators $\mathbf{D}_i = [\mathbf{D}_{i,n}, \mathbf{D}_{i,u}, \mathbf{D}_{i,w}] \in \mathbb{R}^{6n_b \times 3}$ are defined as

$$\begin{aligned} \mathbf{D}_i^T = \big[\mathbf{0} \;\; \ldots \;\; -\mathbf{A}_{i,p}^T \;\; \mathbf{A}_{i,p}^T \mathbf{A}_A \tilde{\bar{\mathbf{s}}}_{i,A} \;\; \mathbf{0} \;\; \ldots \;\; \mathbf{0} \\ \mathbf{A}_{i,p}^T \;\; -\mathbf{A}_{i,p}^T \mathbf{A}_B \tilde{\bar{\mathbf{s}}}_{i,B} \;\; \ldots \;\; \mathbf{0} \big], \end{aligned} \tag{4}$$

where $\mathbf{A}_{i,p} = [\mathbf{n}_i, \mathbf{u}_i, \mathbf{w}_i] \in \mathbb{R}^{3\times3}$ is the orientation matrix associated with contact i; $\mathbf{A}_A = \mathbf{A}(\epsilon_A)$ and $\mathbf{A}_B = \mathbf{A}(\epsilon_B)$ are the rotation matrices of bodies A and B respectively; and the vectors $\bar{\mathbf{s}}_{i,A}$ and $\bar{\mathbf{s}}_{i,B}$ represent the contact point positions in body-relative coordinates as illustrated in Fig. 1. Finally, the set of active and potential unilateral constraints is denoted by $\mathcal{A}(\mathbf{q}, \delta)$ and is defined based on the bodies that are mutually less than a distance δ apart.

2.2 Numerical Solution Methodology

The numerical solution methodology for the aforementioned DVI problem is built around the following two decisions: ($D1$) following the approach proposed in [Stewart and Trinkle 1996], a symplectic half implicit Euler methods is used to discretize the dynamics; and ($D2$) a zero gap nonpenetration condition between bodies in mutual contact is enforced at the new time step $t^{(l+1)}$. Given a consistent position $\mathbf{q}^{(l)}$ and velocity $\mathbf{v}^{(l)}$ at time $t^{(l)}$, the numerical solution at $t^{(l+1)} = t^{(l)} + h$ is obtained by solving the following mathematical programming problem with complementarity and equilibrium constraints:

$$\begin{aligned} \mathbf{M}(\mathbf{v}^{(l+1)} - \mathbf{v}^{(l)}) &= h\mathbf{f} \\ &+ \sum_{i \in \mathcal{A}(q^{(l)}, \delta)} \left(\gamma_{i,n}\, \mathbf{D}_{i,n} + \gamma_{i,u}\, \mathbf{D}_{i,u} + \gamma_{i,w}\, \mathbf{D}_{i,w} \right) \end{aligned} \tag{5}$$

$$i \in \mathcal{A}(q^{(l)}, \delta) : 0 \leq \frac{1}{h}\Phi_i(\mathbf{q}^{(l)}) + \mathbf{D}_{i,n}^T \mathbf{v}^{(l+1)} \perp \gamma_n^i \geq 0 \tag{6}$$

$$(\gamma_{i,u}, \gamma_{i,w}) = \underset{\sqrt{y^2+z^2} \leq \mu_i \gamma_{i,n}}{\operatorname{argmin}} \quad \mathbf{v}^T \left(y\, \mathbf{D}_{i,u} + z\, \mathbf{D}_{i,w} \right) \tag{7}$$

$$\mathbf{q}^{(l+1)} = \mathbf{q}^{(l)} + h\mathbf{L}(\mathbf{q}^{(l)})\mathbf{v}^{(l+1)}. \tag{8}$$

Here, $\gamma_{i,s}$ represents a constraint impulse associated with contact i: $\gamma_{i,s} = h\widehat{\gamma}_{i,s}$, for $s = n, u, w$. The superscript $(l+1)$ on γ_s was dropped for notational brevity. All forces acting on the system except the frictional contact forces are evaluated at time $t^{(l)}$ and denoted by $\mathbf{f} \equiv \mathbf{f}(t^{(l)}, \mathbf{q}^{(l)}, \mathbf{v}^{(l)})$. The term $\frac{1}{h}\Phi_i(\mathbf{q}^{(l)})$ achieves constraint stabilization by eliminating any penetration within one time-step.

For large models with millions of contacts, no effective methods are available for solving the numerical problem in Eqs. (5)–(7). This observation motivated a third decision ($D3$), which was to recast the aforementioned numerical problem into a more amenable

one by a convexification of the NCP; i.e., by relaxing the complementarity condition in Eq. (6) [Anitescu and Hart 2004] to

$$\begin{aligned} i \in \mathcal{A}(q^{(l)}, \delta) : 0 \leq \frac{1}{h}\Phi_i(\mathbf{q}^{(l)}) + \mathbf{D}_{i,n}^T \mathbf{v}^{(l+1)} \\ -\mu_i \sqrt{(\mathbf{v}^T \mathbf{D}_{i,u})^2 + (\mathbf{v}^T \mathbf{D}_{i,w})^2} \perp \gamma_n^i \geq 0 \ . \end{aligned} \tag{9}$$

Owing to this relaxation, the resulting set of equations become a cone complementarity problem (CCP). Specifically, solving for $\mathbf{v}^{(l+1)}$ from Eq. (5) and plugging its expression in Eq. (9) yields a mathematical programming problem with complementarity constraints formulated exclusively in the set of Lagrange multipliers γ [Anitescu 2006]. To pose the CCP, the following notation is used: the number of contacts in $\mathcal{A}(\mathbf{q}, \delta)$ is n_c; $\mathbf{D} \equiv [\mathbf{D}_1, \cdots, \mathbf{D}_{n_c}] \in \mathbb{R}^{6n_b \times 3n_c}$ is the generalized contact transformation matrix; $\mathbf{D}_i \equiv [\mathbf{D}_{i,n}, \mathbf{D}_{i,v}, \mathbf{D}_{i,w}] \in \mathbb{R}^{6n_b \times 3}$ is the contact transformation matrix associated with contact $i \in \mathcal{A}(q^{(l)}, \delta)$; $\mathbf{r}_i \equiv \mathbf{b}_i + \mathbf{D}_i^T \mathbf{M}^{-1}\mathbf{f} \in \mathbb{R}^3$ is the generalized contact velocity for contact i; $\mathbf{b}_i \equiv \left[\frac{1}{h}\Phi_i(\mathbf{q}^{(l)}), 0, 0\right]^T \in \mathbb{R}^3$ is the unilateral constraint stabilization term; and $\boldsymbol{N} \equiv \mathbf{D}^T\mathbf{M}^{-1}\mathbf{D} \in \mathbb{R}^{3n_c \times 3n_c}$ is the contact associated symmetric positive-semidefinite Schur complement matrix, which is typically very sparse. The new quantities introduced – n_c, $\mathbf{D}$, $\mathbf{D}_i$, $\mathbf{r}_i$, $\mathbf{b}_i$, and $\boldsymbol{N}$ – should be further qualified by a superscript (l) to indicate that they are evaluated in the system configuration corresponding to $t^{(l)}$. For brevity, the superscript was omitted. The CCP then assumes the form

$$\begin{aligned} \text{Find } \boldsymbol{\gamma}_i^{(l+1)}, \text{ for } i = 1, \ldots, n_c \\ \text{such that } \Upsilon_i \ni \boldsymbol{\gamma}_i^{(l+1)} \perp -\left(\boldsymbol{N}\boldsymbol{\gamma}^{(l+1)} + \boldsymbol{r}\right)_i \in \Upsilon_i^\circ \\ \text{where } \Upsilon_i = \{[x,y,z]^T \in \mathbb{R}^3 | \sqrt{y^2+z^2} \leq \mu_i x\} \\ \text{and } \Upsilon_i^\circ = \{[x,y,z]^T \in \mathbb{R}^3 | x \leq -\mu_i \sqrt{y^2+z^2}\} \ . \end{aligned} \tag{10}$$

This CCP represents the first order optimality condition of a quadratic optimization problem with conic constraints whose solution provides the set of normal and friction forces associated with the set of contacts in $\mathcal{A}(\mathbf{q}, \delta)$:

$$\begin{aligned} \min f(\boldsymbol{\gamma}) = \frac{1}{2}\boldsymbol{\gamma}^T \mathbf{N} \boldsymbol{\gamma} + \boldsymbol{r}^T \boldsymbol{\gamma} \\ \text{subject to } \boldsymbol{\gamma}_i \in \Upsilon_i \text{ for } i = 1, 2, \ldots, n_c \ . \end{aligned} \tag{11}$$

The overall approach is summarized as follows: a multibody dynamics frictional contact application is formulated as a DVI problem, which based on decisions $(D1)$ and $(D2)$ morphs into an NCP. The latter is convexified based on $(D3)$ to become a CCP. The CCP is solved by considering an equivalent quadratic optimization problem with conic constraints, whose solution is the desired $\gamma \in \mathbb{R}^{3n_c}$. Equation (5) is then used to expeditiously compute the velocity $\mathbf{v}^{(l+1)}$. The generalized coordinates $\mathbf{q}^{(l+1)}$ are recovered using Eq. (8) and the simulation is advanced at $t^{(l+1)}$.

The solution methodology adopted raises two concerns. First, the CCP was obtained through a relaxation of the nonpenetration complementarity condition, see Eqs. (6) and (9). As $h \to 0$, the solution of the modified time-stepping scheme approaches the solution of the same measure differential inclusion as the original numerical scheme [Anitescu 2006]. However, at large step-sizes h the results might display artifacts that can be traced back to the convexification decision. Second, the CCP-based solution methodology fails to produce a unique solution to the frictional contact problem since the coefficient matrix $\boldsymbol{N}$ is positive semidefinite. This does not come as a surprise. Just like any other solution method for rigid body dynamics that relies on a DVI formulation, see for instance, [Moreau and Jean 1996; Stewart and Trinkle 1996; Anitescu 2006; Glocker and Pfeiffer 2006; Preclik et al. 2009; Shojaaee et al. 2012], the CCP approach lacks the uniqueness attribute in force and velocity distributions [Stewart 2000; Delannay et al. 2007; Preclik and Rüde 2011]. This issue can be traced back to the interplay between Coulomb's dry friction model and the rigid body model. The simplest illustration that does not even bring friction into discussion is the case of a perfectly rigid four-legged stool that is symmetric; there is a lack of uniqueness in relation to the reaction force distribution in the four legs. For frictionless problems the lack of uniqueness is of secondary importance since it can be shown that the change in velocity is unique [Anitescu 2006]. Unfortunately, a similar statement cannot be made in the presence of friction.

2.3 Convexification Artifacts

One drawback of the methodology adopted is that at high sliding velocity the relaxation in Eq. (9) introduces numerical artifacts. Consider for instance a simple benchmark test – a 3D rigid ball sliding fast. The ball is in contact with a plane; i.e., the ground, and has an initial velocity of $-2\,\mathrm{m/s}$ in the x direction. It has a radius of $1\,\mathrm{m}$ and the contact has a friction value of $\mu = .2$. The ball, which is initially sliding, slowly begins to roll due to friction and eventually gets into a steady state rolling motion. The time it takes to get to this state is $t_{rolling} = \frac{2v_0}{7\mu g}$ [Trinkle 2003]. For an initial velocity of $2\,\mathrm{m/s}$ and $g = 9.81 m/s^2$, the ball will be fully rolling at $t_{rolling} = .291s$. A numerical integration step size of $h = 0.0025\,\mathrm{s}$ was used to capture these dynamics. Figure 2 shows the response obtained with an unrelaxed approach and as such displays no artifacts. The "primal approach" is that proposed in [Cadoux 2009; Acary et al. 2011]. The dual formulation is the one described herein if no relaxation is considered. The resulting NCP is solved both with the PATH solver [Dirkse and Ferris 1995] or via a CCP using a fixed point iteration, the latter being called the "dual approach". Both the primal and the dual approaches are solved using Mosek [Andersen and Andersen 2000; Andersen et al. 2003], which relies on an interior point method. Although a fixed point iteration typically fails to converge in the solution of the non-relaxed formulation, which justifies the use of a nonsmooth Newton step in [Cadoux 2009; Acary et al. 2011], for the simple problem at hand the fixed point iteration worked fine. Numerical experiment details are provided in [Mazhar et al. 2014], where the connection between the primal and dual formulations is also highlighted: a relaxation of the primal approach in [Cadoux 2009; Acary et al. 2011] leads to a convex program with conic constraints whose dual is practically the approach adopted herein.

Figure 3, which shows results obtained with the relaxed methodology, displays one artifact in the high sliding regime that can be traced back to changing the expression of the relative normal contact velocity by subtracting the relaxation term $\mu_i \sqrt{(\mathbf{v}^T \mathbf{D}_{i,u})^2 + (\mathbf{v}^T \mathbf{D}_{i,w})^2}$; i.e., $\mu||\mathbf{v}_T||$. Subtracting this term ends up canceling out the last term in the expression of the generalized relative contact velocity $\mathbf{v} + \mu||\mathbf{v}_T||\mathbf{n}$ used in [De Saxcé and Feng 1998] to set up the NCP ($\mathbf{n}$ denotes the unit normal vector at the point of contact). The immediate consequence of this serendipitous cancellation is that the general NCP becomes a much more easily solved quadratic program with conic constraint. The NCP to CCP relaxation affects the response since the complementarity problem will be posed relative to a slightly different relative contact velocity, which in turn slightly changes the direction and magnitude of the resulting frictional contact force. In regimes with large

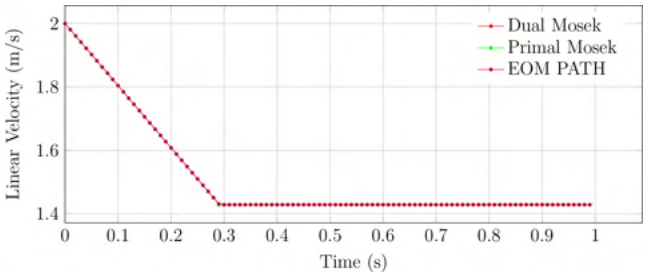

Fig. 2. Translational velocity of 3D ball decreases steadily until the moment it starts rotating with no slip. Solution obtained with non-relaxed approaches: primal, dual, and straight NCP. The first two use Mosek and a fixed point iteration; the latter uses PATH.

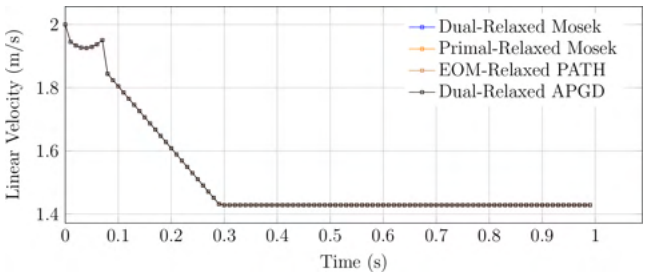

Fig. 3. Solution artifact creeps into the translational velocity of 3D ball at large slip speed.

relative sliding velocities and friction values this artifact results in "jitter" at the contact point as illustrated in Figure 3. Note that such an artifact exists independent of the approach used to solve the optimization problem in Eq. 11. Consequently, the APGD method will not eliminate it. Further details and numerical experiments are provided in [Mazhar et al. 2014].

2.4 Embedding a Simple Cohesion Model

The ability of the DVI formulation to accommodate cohesion was postulated in [Tasora et al. 2013], yet no equations or simulation results were provided therein. Following [Tasora et al. 2013], three observations suffice to explain how a cohesion model can be embedded into the solution framework. First, the condition in Eq. (3), $0 \leq \widehat{\gamma}_{i,n} \perp \Phi_i(\mathbf{q}) \geq 0$, is replaced by $0 \leq \widehat{\gamma}_{i,n} + \hat{c}_i \perp \Phi_i(\mathbf{q}) \geq 0$, where for contact i, $\hat{c}_i > 0$ represents a user prescribed parameter, expressed in Newtons, that controls the amount of cohesion. Effectively, this relaxes the formulation by allowing $\widehat{\gamma}_{i,n}$ to assume negative values; i.e., pulling at the point of contact, with the pulling force capped at the value $\hat{c}_i > 0$. Second, note that the actual normal force at the interface between the two bodies in contact continues to be $\widehat{\gamma}_{i,n}$, so the generalized force associated with this interaction enters the equations of motion unchanged; i.e., as $\widehat{\gamma}_{i,n}\,\mathbf{D}_{i,n}$ in Eq. (3). Lastly, the Coulomb dry friction model is amended to reflect the presence of the cohesion force. To this end, the cone constraint $\sqrt{y^2+z^2} \leq \mu_i \widehat{\gamma}_{i,n}$ is posed as $\sqrt{y^2+z^2} \leq \mu_i(\widehat{\gamma}_{i,n} + \hat{c}_i)$ in the minimization problem of Eq. (3). The discretization proceeds just like in Eqs. (5) through (8), subject to the three observations above.

3. THE ACCELERATED PROJECTED GRADIENT DESCENT METHOD

In introducing Nesterov's method in conjunction with this problem, the discussion concentrates first on the task of minimizing a generic cost function $f(\mathbf{x})$. The use of $\mathbf{x}$ instead of γ is meant to emphasize that Nesterov's method is a general purpose minimization approach. The particular expression of f; i.e., a quadratic function, the projection onto the friction cones, and the large number of variables represent traits specific to the multibody dynamics context in which Nesterov's method is subsequently applied. For a typical granular dynamics system with one million bodies, the quadratic problem with conic constraints in Eq. (11) has approximately 12 million variables and four million constraints.

3.1 Preamble: Nesterov's Method

Nesterov first proposed accelerated gradient schemes in 1983 [Nesterov 1983]. Accelerated gradient descent methods can be seen as simple gradient descent methods with the introduction of 'momentum' in the search direction. Momentum refers to the the concept that the search direction should depend on past iterations in addition to the current iteration. Instead of taking the search direction to be opposite of the gradient direction at the current iteration, the introduction of momentum effectively uses a weighted combination of the current and past gradient directions.

The original gradient descent method, often attributed to Cauchy [Cauchy 1847], is expressed as

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \nabla f(\boldsymbol{x}_k) \tag{12}$$

$$\text{with } \alpha_k = \arg\min_{\alpha} f(\boldsymbol{x}_k - \alpha \nabla f(\boldsymbol{x}_k)) \ , \tag{13}$$

where $f(\boldsymbol{x})$ is a smooth function to be minimized and α_k is a step size for iteration k. Under certain conditions, for example if $f(\boldsymbol{x})$ is convex and Lipschitz continuous, convergence to the global solution can be guaranteed.

The scheme in Eqs. (12)–(13) solves an unconstrained optimization problem. The gradient descent method can be extended to solve constrained optimization problems under certain conditions [Bertsekas 1976]. The resulting projected gradient descent method takes the following form, where $\Pi_{\mathcal{C}}$ represents projection onto the feasible convex set $\mathcal{C}$ and α_k is the step size that should satisfy a sufficient decrease condition:

$$\boldsymbol{x}_{k+1} = \Pi_{\mathcal{C}}(\boldsymbol{x}_k - \alpha_k \nabla f(\boldsymbol{x}_k)) \, . \tag{14}$$

The projected gradient descent method can be shown to have a sub-linear rate of convergence when the objective function $f(\boldsymbol{x})$ is convex and $\mathcal{C}$ is a convex set:

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}^\star) \simeq \mathcal{O}(1/k) \ . \tag{15}$$

Nesterov's method has an improved convergence rate of $\mathcal{O}(1/k^2)$. In fact, it was shown to be an 'optimal' first-order method for smooth problems in terms of its performance among all first-order methods, up to a constant [Nemirovsky and Yudin 1983].

The following set of equations represents one iteration of the accelerated gradient descent (AGD) scheme [Nesterov 2003]. Note that $\boldsymbol{y}_0 = \boldsymbol{x}_0 \in \mathbb{R}^n$, $\theta_0 = 1$, $q \in [0,1]$ is a tuning parameter, and t_k is the step size for the current iteration.

$$\boldsymbol{x}_{k+1} = \boldsymbol{y}_k - t_k \nabla f(\boldsymbol{y}_k) \tag{16}$$
$$\theta_{k+1} \quad \text{solves } \theta_{k+1}^2 = (1-\theta_{k+1})\,\theta_k^2 + q\theta_{k+1} \tag{17}$$
$$\beta_{k+1} = \frac{\theta_k(1-\theta_k)}{\theta_k^2+\theta_{k+1}} \tag{18}$$
$$\boldsymbol{y}_{k+1} = \boldsymbol{x}_{k+1} + \beta_{k+1}(\boldsymbol{x}_{k+1}-\boldsymbol{x}_k)\ . \tag{19}$$

Assume $f(\boldsymbol{x})$ is convex and Lipschitz continuous with constant L; i.e., $||\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})||_2 \leq L||\boldsymbol{x}-\boldsymbol{y}||_2, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. Then, the method described by Eqs. (16)–(19) converges for any $t_k \leq 1/L$. Note that $q = 1$ leads to $\theta_k = 1$, $\beta_k = 0$, and $\boldsymbol{y}_k = \boldsymbol{x}_k$ for all $k \geq 0$, which reduces to the gradient descent method. In general, the parameter q can tune the performance of the method depending on the specifics of the objective function $f(\boldsymbol{x})$. For example, if $f(\boldsymbol{x})$ is also strongly convex, i.e., $\exists \mu > 0 : f(\boldsymbol{x}) \geq f(\boldsymbol{x}^\star) + (\mu/2)||\boldsymbol{x}-\boldsymbol{x}^\star||_2^2, \forall \boldsymbol{x} \in \mathbb{R}^n$, then the optimal value is $q = \mu/L$. If the objective function is not strongly convex, or the strong convexity parameter μ is unknown, then it is often assumed that $q = 0$. Since the original statement of the accelerated method in [Nesterov 1983] had $q = 0$, the convergence rate of $\mathcal{O}(1/k^2)$ is still valid.

The AGD scheme can be extended to constrained optimization in the same way that gradient descent was extended to projected gradient descent. The resulting algorithm, called Accelerated Projected Gradient Descent (APGD) can be expressed by the following sequence of steps performed at each iteration $k \geq 0$. Once again, let $\boldsymbol{y}_0 = \boldsymbol{x}_0 \in \mathbb{R}^n$, and $\theta_0 = 1$.

$$\boldsymbol{x}_{k+1} = \Pi_{\mathcal{C}}(\boldsymbol{y}_k - t_k \nabla f(\boldsymbol{y}_k)) \tag{20}$$
$$\theta_{k+1} \quad \text{solves } \theta_{k+1}^2 = (1-\theta_{k+1})\,\theta_k^2 \tag{21}$$
$$\beta_{k+1} = \frac{\theta_k(1-\theta_k)}{\theta_k^2+\theta_{k+1}} \tag{22}$$
$$\boldsymbol{y}_{k+1} = \boldsymbol{x}_{k+1} + \beta_{k+1}(\boldsymbol{x}_{k+1}-\boldsymbol{x}_k)\ . \tag{23}$$

If $f(\boldsymbol{x})$ is convex and Lipschitz continuous with constant L, the method described by Equations (20)–(23) converges for any $t_k \leq 1/L$. An equivalent algorithm was proved in [Beck and Teboulle 2009] to converge with the same $\mathcal{O}(1/k^2)$ rate as the AGD method.

3.1.1 *Adaptive Step Size.* The global Lipschitz constant may be unknown or too restrictive, implying that choosing a constant $t < 1/L$ may not achieve best performance. The APGD method was therefore adjusted to allow the step size, t_k, to vary at each iteration while still guaranteeing convergence. Two possible adaptive strategies are considered, both adapted from [Becker et al. 2011].

In the first, a local estimate of the Lipschitz parameter, L_k, is computed and used as long as the associated step, $t_k = 1/L_k$, is appropriate to preserve convergence; i.e., the following condition is satisfied:

$$f(\boldsymbol{x}_{k+1}) \leq f(\boldsymbol{y}_k) + \nabla f(\boldsymbol{y}_k)^T(\boldsymbol{x}_{k+1}-\boldsymbol{y}_k) + \frac{L_k}{2}||\boldsymbol{x}_{k+1}-\boldsymbol{y}_k||_2^2\ . \tag{24}$$

Backtracking is performed as long as Eq. (24) is violated. Specifically, set $L_k = 2L_k$ until Eq. (24) is satisfied and proceed with the new estimate. With this approach, the step length t_k will shrink as L_k increases. Once $L_k \geq L$, no more backtracking steps will be needed and t_k will be constant for the remaining iterations. To start off, L_0 can be estimated as $L_0 = ||\nabla f(\boldsymbol{z}_0) - \nabla f(\boldsymbol{z}_1)||_2 / ||\boldsymbol{z}_0 - \boldsymbol{z}_1||_2$, for $\boldsymbol{z}_0 \neq \boldsymbol{z}_1$.

The second strategy allows the step to both grow and shrink throughout the iterative process. The same backtracking is used to increase L_k, yet it may also decrease to lead to larger steps when feasible. Specifically, L_k is decreased at each iteration according to $L_{k+1} = 0.9L_k$, and increased if needed to satisfy Eq. (24). The approach may result in slightly more total backtracking steps, but may allow longer steps and improved overall performance.

3.1.2 *Adaptive Restart.* Depending on the nature of the problem solved, Nesterov's method can yield a numerical solution sequence in which the amount of momentum applied adversely impacts the solution time. In [O'Donoghue and Candes 2012], it is shown that high momentum leads to rippling in the objective function value and decreased performance of the iterative method. In Eqs. (20)–(23), note that $\beta_k \to 1$, is characterized as high momentum. This is addressed through a restarting of the method, that is, a resetting of the momentum [O'Donoghue and Candes 2012]: if $\nabla f(\boldsymbol{y}_{k-1})^T(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) > 0$ at iteration k, then set $\theta_k = 1$ and $\boldsymbol{y}_k = \boldsymbol{x}_k$. In other words, the momentum is reset whenever the projection of the momentum term, $\beta_k(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$, onto the negative gradient, $-\nabla f(\boldsymbol{y}_{k-1})$, is negative. This would imply that the momentum opposes the negative gradient, which is known to be a descent direction and therefore not helping the method.

3.1.3 *Fallback.* If a non-monotone iterative method is terminated prematurely at iteration k, it is possible that the current iterate, $\boldsymbol{x}_k$, is not the best approximation of the solution. A fall-back strategy has been adopted herein to achieve monotone behavior in terms of a selected metric $r(\boldsymbol{x}_i) \in \mathbb{R}$. The fall-back strategy should return $\hat{\boldsymbol{x}}$, the 'best' candidate according to the selected metric: $\hat{\boldsymbol{x}} = \boldsymbol{x}_i$, where $i = \arg\min_{j\in[0,k]} r(\boldsymbol{x}_j)$.

3.2 The Overall Algorithm

The generic multibody dynamics CCP of Eq. (11) is solved with the APGD method augmented with adaptive step size, adaptive restart, and a fall-back strategy, see Algorithm 1. The input for the algorithm is the matrix $\boldsymbol{N}$ and vector $\boldsymbol{r}$, the tolerance τ, the maximum number of iterations N_{max}, and the initial guess for the vector of unknowns, usually taken as $\boldsymbol{\gamma}_0 = \mathbf{0}$.

Of several alternatives discussed in [Heyn 2013], the fall-back metric selected is

$$r = ||\boldsymbol{\psi}||_2, \boldsymbol{\psi} = \frac{1}{3n_c g_d}(\boldsymbol{\gamma} - \Pi_{\mathcal{C}}(\boldsymbol{\gamma} - g_d(\boldsymbol{N}\boldsymbol{\gamma}+\boldsymbol{r}))) \in \mathbb{R}^{3n_c}, \tag{25}$$

where g_d is a small constant parameter, for example $g_d = 1\times10^{-6}$. To justify this selection, first note that if $\boldsymbol{\gamma} = \boldsymbol{\gamma}^*$ is optimal, then $\Pi_{\mathcal{C}}(\boldsymbol{\gamma}^* - g_d(\boldsymbol{N}\boldsymbol{\gamma}^*+\boldsymbol{r})) = \boldsymbol{\gamma}^*$, so $\boldsymbol{\psi} = \mathbf{0}$. Second, consider the case when $\boldsymbol{\gamma}$ is not optimal. Then, it can be verified that

$$\Pi_{\mathcal{C}}(\boldsymbol{\gamma} - g_d(\boldsymbol{N}\boldsymbol{\gamma}+\boldsymbol{r})) = \boldsymbol{\gamma} - g_d\boldsymbol{\psi}\ . \tag{26}$$

The left hand side is equivalent to taking a step of length g_d in the negative gradient direction and projecting back to the feasible region. The right hand side says that the same point can be reached by taking a step of length g_d in the direction opposite of $\boldsymbol{\psi}$. In the limit, as $g_d \to 0$, the direction $\boldsymbol{\psi}$ approaches the plane tangent to the constraint manifold. Note that r could be used to measure convergence for any value of g_d, but a small value was used in practice for the reasons just stated.

Algorithm 1 APGD ($\boldsymbol{N}$, $\boldsymbol{r}$, τ, N_{max}, $\boldsymbol{\gamma}_0$)

1: $\hat{\boldsymbol{\gamma}}_0 = \mathbf{1}_{n_c}$
2: $\boldsymbol{y}_0 = \boldsymbol{\gamma}_0$
3: $\theta_0 = 1$
4: $L_k = \frac{||\boldsymbol{N}(\boldsymbol{\gamma}_0 - \hat{\boldsymbol{\gamma}}_0)||_2}{||\boldsymbol{\gamma}_0 - \hat{\boldsymbol{\gamma}}_0||_2}$
5: $t_k = \frac{1}{L_k}$
6: **for** $k := 0$ toN_{max} **do**
7: $\quad \boldsymbol{g} = \boldsymbol{N}\boldsymbol{y}_k + \boldsymbol{r}$
8: $\quad \boldsymbol{\gamma}_{k+1} = \Pi_{\mathcal{C}}\left(\boldsymbol{y}_k - t_k g\right)$
9: $\quad$ **while** $\frac{1}{2}\boldsymbol{\gamma}_{k+1}^T\boldsymbol{N}\boldsymbol{\gamma}_{k+1} + \boldsymbol{\gamma}_{k+1}^T\boldsymbol{r} \geq \frac{1}{2}\boldsymbol{y}_k^T\boldsymbol{N}\boldsymbol{y}_k + \boldsymbol{y}_k^T\boldsymbol{r} + \boldsymbol{g}^T\left(\boldsymbol{\gamma}_{k+1} - \boldsymbol{y}_k\right) + \frac{1}{2}L_k||\boldsymbol{\gamma}_{k+1} - \boldsymbol{y}_k||_2^2$ **do**
10: $\quad\quad L_k = 2L_k$
11: $\quad\quad t_k = \frac{1}{L_k}$
12: $\quad\quad \boldsymbol{\gamma}_{k+1} = \Pi_{\mathcal{C}}\left(\boldsymbol{y}_k - t_k\mathbf{g}\right)$
13: $\quad$ **end while**
14: $\quad \theta_{k+1} = \frac{-\theta_k^2 + \theta_k\sqrt{\theta_k^2+4}}{2}$
15: $\quad \beta_{k+1} = \theta_k\frac{1-\theta_k}{\theta_k^2+\theta_{k+1}}$
16: $\quad \boldsymbol{y}_{k+1} = \boldsymbol{\gamma}_{k+1} + \beta_{k+1}\left(\boldsymbol{\gamma}_{k+1} - \boldsymbol{\gamma}_k\right)$
17: $\quad r = r\left(\boldsymbol{\gamma}_{k+1}\right)$
18: $\quad$ **if** $r < r_{min}$ **then**
19: $\quad\quad r_{min} = r$
20: $\quad\quad \hat{\boldsymbol{\gamma}} = \boldsymbol{\gamma}_{k+1}$
21: $\quad$ **end if**
22: $\quad$ **if** $r < \tau$ **then**
23: $\quad\quad$ **break**
24: $\quad$ **end if**
25: $\quad$ **if** $\boldsymbol{g}^T\left(\boldsymbol{\gamma}_{k+1} - \boldsymbol{\gamma}_k\right) > 0$ **then**
26: $\quad\quad \boldsymbol{y}_{k+1} = \boldsymbol{\gamma}_{k+1}$
27: $\quad\quad \theta_{k+1} = 1$
28: $\quad$ **end if**
29: $\quad L_k = 0.9L_k$
30: $\quad t_k = \frac{1}{L_k}$
31: **end for**
32: **return** Value at time step $t^{(l+1)}$, $\boldsymbol{\gamma}^{(l+1)} := \hat{\boldsymbol{\gamma}}$.

4. NUMERICAL EXPERIMENTS

The numerical experiments reported herein were carried out using Chrono, a library for rigid and flexible body dynamics [Mazhar et al. 2013]. Chrono is an open source BSD-3 licensed C/C++ physics-based dynamics engine that relies on multi-core and GPU parallelism to handle simulations of rigid bodies interacting through friction and contact. It aims at modeling mechanical systems, e.g., cars, trucks, tracked vehicles, granular flow, etc., that include elements such as joints, driving constraints, force elements, user defined controllers, etc. Chrono comes with several solvers for rigid and flexible body dynamics including projected Gauss-Seidel (GS), Jacobi and Krylov-based. This numerical solution portfolio was augmented herein with a solver based on the APGD method. The numerical experiments carried out to compare APGD against projected GS and Jacobi are thus drawing on the same equation formulation framework. Any efficiency and/or robustness differences can be traced back solely to how the CCP in Eq. (11) is solved by APGD, Jacobi, and GS. The latter is the default sequential solver in Chrono and the benchmark method in this paper. Parallel solutions in Chrono rely on Jacobi.

4.1 APGD Performance Analysis

Projected Gauss-Seidel, or variants thereof, represent the most commonly used algorithms in the DVI-based solution of the frictional contact multibody dynamics problem, see for instance [Moreau and Jean 1996; Glocker and Pfeiffer 2006; Preclik et al. 2009; Shojaaee et al. 2012]. The GS algorithm employed herein is based on work reported in [Anitescu and Tasora 2010]. It has successive over-relaxation and a projection step that enforces feasibility at each iteration. The algorithm relies on block diagonal preconditioning and uses the matrices $\boldsymbol{B}_i = \frac{1}{g_i}\boldsymbol{I}_3 \in \mathbb{R}^{3\times3}$, where for each contact $i \in [1, n_c]$

$$g_i = \frac{tr\left(\boldsymbol{D}_i^T\boldsymbol{M}^{-1}\boldsymbol{D}_i\right)}{3} . \tag{27}$$

The convergence is controlled through r, computed in line 8 of Algorithm 2. A Jacobi algorithm is very similar, essentially Algorithm 2 minus the update at line 6. While slower than GS to converge, parallelizing Jacobi is straightforward.

Algorithm 2 Gauss-Seidel ($\boldsymbol{N}$, $\boldsymbol{r}$, τ, N_{max}, $\boldsymbol{\gamma}_0$)

1: **for** $k := 0$ to N_{max} **do**
2: $\quad \boldsymbol{\gamma} = \boldsymbol{\gamma}_k$
3: $\quad$ **for** $i = 1$ to n_c **do**
4: $\quad\quad \hat{\boldsymbol{\gamma}}_{i,(k+1)} = \Pi_{\mathcal{C}}\left(\boldsymbol{\gamma}_{i,(k)} - \omega\boldsymbol{B}_i\left(\boldsymbol{N}\boldsymbol{\gamma} + \boldsymbol{r}\right)_i\right)$
5: $\quad\quad \boldsymbol{\gamma}_{i,(k+1)} = \lambda\hat{\boldsymbol{\gamma}}_{i,(k+1)} + (1-\lambda)\,\boldsymbol{\gamma}_{i,(k)}$
6: $\quad\quad \boldsymbol{\gamma}_i = \boldsymbol{\gamma}_{i,(k+1)}$
7: $\quad$ **end for**
8: $\quad r = r\left(\boldsymbol{\gamma}_{k+1}\right)$
9: $\quad$ **if** $r < \tau$ **then**
10: $\quad\quad$ **break**
11: $\quad$ **end if**
12: **end for**
13: **return** Value at time step $t^{(l+1)}$, $\boldsymbol{\gamma}^{(l+1)} := \boldsymbol{\gamma}_{(k+1)}$.

4.1.1 *Speed of Convergence Study: APGD vs. Gauss Seidel vs. Jacobi.* In this study, a container with a width and depth of $3\,\mathrm{m}$ was filled with 4000 equally sized spheres of radius $0.15\,\mathrm{m}$ and mass $1\,\mathrm{kg}$. A heavy slab was placed on top of the bed of spheres and allowed to settle using the GS solver to a 'reference' configuration as illustrated in Fig. 4. The friction coefficient between all bodies in the model was $\mu = 0.1$. The reference configuration of the system was subsequently used as the initial state for the APGD, GS, and Jacobi convergence analysis. This type of stacking problem is challenging since there are numerous force chains forming within the granular material that couple the distribution of normal and friction forces well beyond several layers of neighbor bodies. The presence of a heavy slab exacerbates this force chain formation.

Mimicking the process of advancing the simulation by one step, each of the three solvers was cold started with $\gamma_0 = \mathbf{0}$ from the GS-determined reference configuration. As illustrated in Fig. 5, for a slab mass of $1000\,\mathrm{kg}$, the APGD method converges at the fastest rate and has the lowest objective function value after 1000 iterations when advancing the simulation by one step. Within this iteration budget, both Jacobi and GS approach a residual of $r = 1.2 \times 10^{-5}$, while APGD reaches a residual one order of magnitude lower, $r = 1.1 \times 10^{-6}$. Finally, APGD has an objective function value twice as small as GS and almost seven times smaller than Jacobi. The data in Fig. 5 after 1000 iterations is also provided under the $1000\,\mathrm{kg}$ rows in Table I. The table reports similar information and

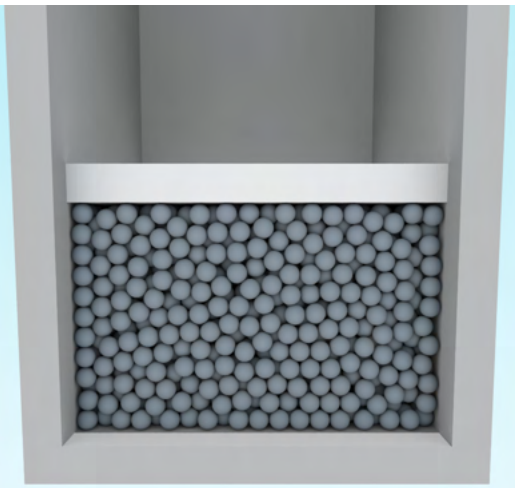

Fig. 4. 'Reference' configuration for convergence test involving APGD, GS and Jacobi.

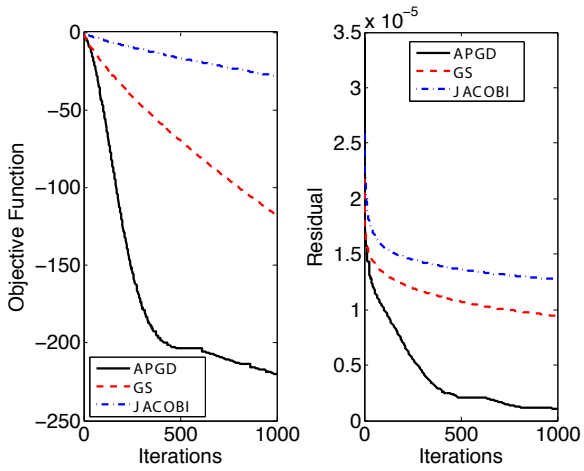

Fig. 5. APGD vs. GS vs. Jacobi: Residual r and Objective Function $f(\gamma)$ after 1000 iterations for the pressure test with a slab of 1000 kg, see Fig 4.

captures a similar trend; i.e., better convergence for APGD, for slab masses up to 1 000 000 kg

For a second numerical experiment using a 1000 kg slab, Table II reports the amount of time it took each solver when cold started from the reference configuration to reach r =7.0 $\times$ 10^{-6}. The results show that for small tolerances, GS had a difficult time converging, while Jacobi failed to converge after 500 000 iterations. For this tolerance, APGD was more than an order of magnitude faster than GS in reaching the set tolerance.

4.1.2 *APGD Scaling Analysis.* APGD has been implemented for parallel execution in Chrono using OpenMP. The collision detection stage is parallelized following an approach described in [Mazhar et al. 2011]. The scaling analysis of the parallel APGD implementation is carried out using a mixer model: a cylindrical container inside of which a four blade mixer rotates counter-clockwise at a rate of $\pi/2$ rad s^{-1}, see Fig. 6. The mixture in the container is made up of spheres, ellipsoids, cylinders and cubes with a density of 1000 kg m^{-3} and an average radius/half-width of 0.05 m. The friction coefficient was $\mu = 0.5$ and the cohesive force parameter, discussed in detail in the next section, was $\hat{c}$ =200 N. Using this

Table I. APGD vs. GS vs. Jacobi: Residual r and Objective Function $f(\gamma)$ after 1000 iterations for different slab masses.

Residual r			
Mass	**Jacobi**	**Gauss Seidel**	**APGD**
1×10^3	1.27×10^{-5}	9.43×10^{-6}	1.10×10^{-6}
1×10^4	1.48×10^{-5}	1.27×10^{-5}	3.18×10^{-6}
1×10^5	1.52×10^{-5}	1.36×10^{-5}	9.09×10^{-6}
1×10^6	1.53×10^{-5}	1.37×10^{-5}	1.18×10^{-5}
Objective Function $f(\gamma)$			
Mass	**Jacobi**	**Gauss Seidel**	**APGD**
1×10^3	-28.29	-117.70	-220.14
1×10^4	-35.63	-162.99	-883.54
1×10^5	-37.02	-176.94	-3199.27
1×10^6	-37.15	-210.23	-4696.48

Table II. APGD vs. GS vs. Jacobi: Number of iterations and time taken to converge to a tolerance of r =7 $\times$ 10^{-6} for a slab of 1000 kg. Jacobi was unable to converge (UtC) within 500 000 iterations.

Solver	Residual	Iterations	Time[s]
Jacobi	7.54×10^{-6} (UtC)	500 000	24 300
Gauss Seidel	6.99×10^{-6}	11 485	494.8
APGD	6.97×10^{-6}	202	10.6

Table III. Parallel APGD scaling with eight cores. Time taken by solver to resolve one time step at blade steady state rotation.

Objects	Contacts	Residual r	Solution Time [s]
101 506	355 551	6.08×10^{-5}	3.97
200 270	804 768	6.42×10^{-5}	8.13
301 790	1.33×10^6	5.36×10^{-5}	12.53
400 555	1.89×10^6	4.58×10^{-5}	17.23

setup, several simulations were run with an increasing number of objects. All simulations used a time step of 5×10^{-4} s. The tests were run on an eight core Intel Xeon E5520 processor. Table III reports the timing results for a single time step when the mixer has reached steady state. The number of objects ranges from approximately 100 000 to 400 000. The results indicate that the simulation time scales linearly with the number of bodies, and therefore contact events, in the model.

4.1.3 *Comparison with Other Solvers.* For large stacking and/or granular dynamics problems with hundreds of thousands of contacts, run times of the APGD method embedded into a dual relaxed frictional contact formulation (i) increase linearly with the number of bodies/contacts; and (ii) are one to two orders of magnitude shorter than those associated with existing Chrono solvers built around the projected Gauss-Seidel or Krylov-subspace approaches [Heyn et al. 2013]. These two observations are backed by results shown in Tables II and III, as well as a range of analyses carried out elsewhere [Mazhar et al. 2013; Mazhar et al. 2014].

If the comparison against projected Gauss-Seidel or Krylov-subspace; i.e., first order approaches, is favorable, what can one say when second order, Hessian-based, methods are brought into the discussion? Second order methods, such as interior point or

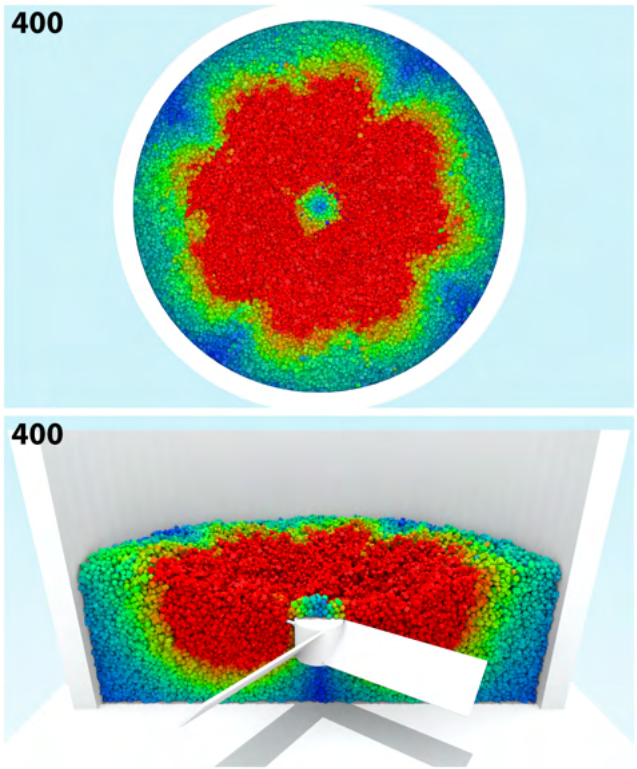

Fig. 6. Mixer simulation with cohesive material made up of several different geometries. The mixer blade rotates counter-clockwise pulling material down.

other Newton method-based, are expected to converge in fewer yet costlier iterations as they are more expensive to setup at each time step and require the solution of a sparse linear system at each iteration. When simulating the dynamics of granular material, for instance, the size of these linear systems is large and the structure of the coefficient matrix changes at each time step owing to the making and braking of contacts. The sparse linear solvers used can be direct or iterative, parallel or sequential. If they are parallel, there are OpenMP, MPI, and GPU-based flavors, see for instance [Schenk and Gärtner 2004; Demmel 2011; Li et al. 2013]. If they are iterative, a preconditioner typically comes into play. Moreover, the APGD approach takes many inexpensive iterations and can be stopped early in the convergence process if visually pleasing animations are the only matter of concern.

It becomes manifest that a full-blown APGD vs. second order methods comparison rapidly escalates and a comprehensive discussion falls outside the scope of this paper. An investigation into this topic is shown in [Mazhar et al. 2014] wherein, a comparison of several reference solvers all used in conjunction with the dual and relaxed approach that led to Eq. (11) was performed. The lessons learned so far are as follows: (a) benchmark tests indicate that the results obtained with Mosek, PATH and APGD are the same – both in the relaxed as well as the unrelaxed dual approaches, albeit there will be differences between the relaxed and non-relaxed results; and (b) the run time comparison is somewhat inconclusive since run times are comparable to the point where implementation details; i.e., software design issues and the underlying sparse linear solver, dictate the overall solution time. An APGD-Mosek efficiency comparison in [Mazhar et al. 2014] indicates that APGD is faster for problems with thousands of contacts but that Mosek it is slowly closing the performance gap. It is not clear at this point whether for very large problems the embedded parallel sparse linear solver will pose a computational bottleneck in Mosek or whether the Mosek solver will eventually become more effective.

As a side note, even for small benchmark problems the APGD and Mosek were significantly yet unsurprisingly faster than PATH. The latter solves the full mathematical program with equilibrium constraints problem; i.e., no reduction to a CCP is undertaken and the dimension of the resulting problem is large and poised to increase much faster with the model size when compared to the CCP dual approach adopted here.

4.2 Other APGD Simulations

4.2.1 *Snow - Clay Simulation.* The purpose of this test was twofold: illustrate how a cohesion model can be embedded in the overall simulation framework, and show how APGD handles the modified CCP. In Fig. 7 several balls of material, each consisting of approximately 75 000 equally sized spheres, are dropped with an initial velocity of $10\,\mathrm{m/s}$ onto the edge of a cube rotated $45°$. The friction value for contact between the spheres was $\mu = 1.0$ and for contacts with the walls it was $\mu = 0.1$. The cohesion was increased from $\hat{c}$ =500 N to $\hat{c}$ =20 000 N to demonstrate different types of materials. For a cohesion value of $\hat{c}$ =500 N the material behaves like heavy snow. Increasing it to $\hat{c}$ =4000 N makes the material behave like clay, while for a cohesion of $\hat{c}$ =20 000 N material fractures rather than deforming under large forces. For each case, the integration time step was h =0.0005 s. Fig. 8 shows two cases, $\hat{c}$ =500 N and $\hat{c}$ =4000 N, after several time frames. The final frames for both cases contain approximately 1.6 million rigid spheres and more than six million contacts, which leads to a CCP in approximately 20 million variables.

4.2.2 *Bulldozer Dynamics.* The purpose of this test was twofold. First, it posed a challenging numerical problem since the mass of the bulldozer is approximately five orders of magnitude larger than the mass of a pebble that the bulldozer operates on. The multi-scale attribute of the model leads to ill conditioned $\boldsymbol{N}$ matrices. Second, the model uses several kinematic joints to connect idlers and sprockets to the chassis of the bulldozer, and track shoes to each other. These joints translate into kinematic geometric constraints of the form $\mathbf{g}^{(G)}(\mathbf{q}) = \mathbf{0}_b$, where $\mathbf{g} \in \mathbb{R}^b$ is a function specific to each joint type [Haug 1989]. For instance, $b = 3$ for a spherical joint, $b = 5$ for a revolute joint, $b = 5$ for a translational joint, etc. There are also kinematic driving constraints, which assume the form $\mathbf{g}^{(D)}(\mathbf{q}, t) = \mathbf{0}_b$. These geometric and driving kinematic constraints result in nonlinear algebraic equations added to the set in Eq. (3). Their presence slightly obfuscates the numerical algorithm by marginally increasing the size and complicating the structure of $\boldsymbol{N}$ and $\boldsymbol{r}$ but otherwise does not change the nature of the optimization problem in Eq. (11) [Anitescu and Hart 2004]. APGD was used to solve this optimization problem, a snapshot of the simulation being shown in Fig. 9.

The tracked vehicle has a curved blade attached to the front. Two sets of tracks are attached to the $5000\,\mathrm{kg}$ chassis. Each track has a $250\,\mathrm{kg}$ sprocket that drives the track as well as five cylindrical rollers, each with a mass of $250\,\mathrm{kg}$. Each of the 50 track shoes has a mass of $20\,\mathrm{kg}$. The total mass of the vehicle is $7300\,\mathrm{kg}$. The track model is created using revolute joints to connect the track shoes. Revolute joints are also used to attach rollers and sprockets to the chassis. Each sprocket is driven at a constant angular velocity of $4\,\mathrm{rad/s}$. The frictional value between the ground and the track

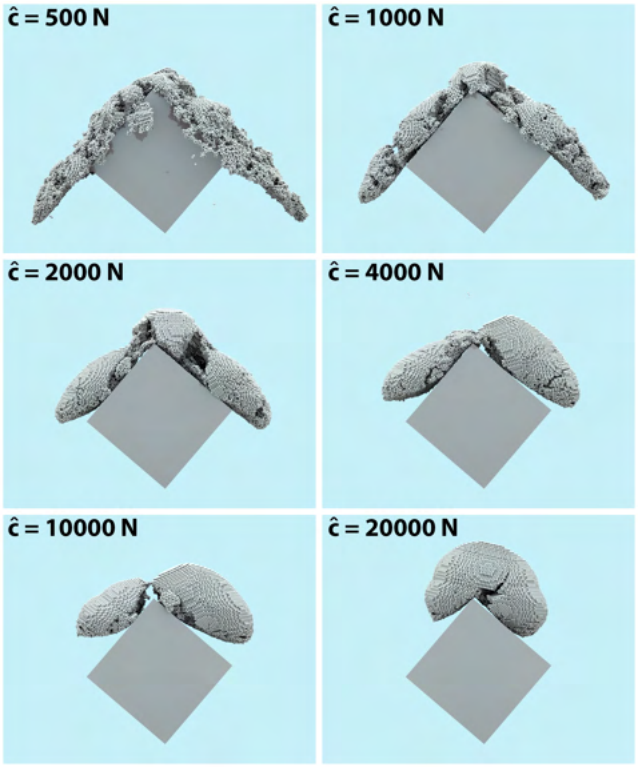

Fig. 7. Comparison between balls with different cohesion values ranging from $\hat{c}$ =500 N to $\hat{c}$ =20 000 N.

shoe is $\mu = 1.0$; the blade is considered frictionless. In total there are 114 rigid objects and 111 constraints joining the vehicle parts together.

The bulldozer drives towards and engages a pile of elliptical granular material with a density of $2000\,\mathrm{kg/m^3}$ and cohesion $\hat{c}$ =500 N. Inter-particle friction is $\mu = 0.2$. There are 160 000 $0.067\,\mathrm{kg}$ ellipsoids representing the soil elements that interact through approximately 0.5 million contacts at each step of the simulation. The integration time step was h =0.0001 s.

4.2.3 *Modeling Deformation and Brittleness Using Cohesion.* The largest Chrono simulation used 10 million elements to model the motion of a tire provided with a detailed tread geometry over deformable terrain. The tire deformation was neglected and the interest was in understanding the deformation of the terrain. The ability of the APGD method to analyze the dynamics of millions of bodies interacting at the interface provides an opportunity to approach old problems from a new perspective. For instance, Fig. 10 provides a sequence of snapshots from the deformation of a teapot that hits a wall. At this point there is no claim of physical behavior in the plastic deformation of the pot. Nonetheless, the use of cohesion in conjunction with the representation of a part as a very large number of discrete elements that, up to a point, stick together thanks to the cohesion force can provide a different perspective on the topic of plasticity and fracture.

This idea is also illustrated in Fig. 11. At a cohesion value of $\hat{c} = 1 \times 10^4$ N, the object behaves in a highly brittle manner and quickly loses structural integrity upon contact with the wall. At $\hat{c} = 1 \times 10^5$ N, the object is strong enough to maintain its shape and only fractures at internal weak points.

4.2.4 *Granular Flow.* The results in Fig. 3 suggest that the convexification process can produce artifacts. However, the solution methodology adopted has merit – a solution always exists and one has a clear idea when the method ought to work and when a different approach must be considered. For instance, in granular dynamics when handling stacks of bodies the relative sliding velocity between bodies is small and the relaxation approach is expected to yield good results. Indeed, the results reported in this and the next subsection confirm the predictive attributes of the APGD algorithm in Chrono.

The first validation setup is similar to that of an hourglass: granular material is placed in a trough and one of the walls slides slightly producing a gap through which the bodies move out. The gap width controls the rate of flow, which is first established experimentally. In the setup discussed, see Fig. 12, the moving wall is on the right. In the experiment, the granular material was composed of approximately 39,000 glass spheres with diameter of $0.5\,\mathrm{mm}$ and mass of $1.6315 \times 10^{-7}\,\mathrm{kg}$ based on a material density of $2500\,\mathrm{kg/m^3}$. The moving wall was angled at $45°$ and its position could be precisely controlled through a linear actuator.

In the experimental test and associated simulation the spheres were first dropped into the trough and allowed to settle. The linear actuator then slid the moving wall to the right producing a different gap for each of three experiments: $2.0\,\mathrm{mm}$, $2.5\,\mathrm{mm}$, and $3.0\,\mathrm{mm}$. The $2.0\,\mathrm{mm}$ gap experiment was used for calibration purposes to evaluate the value of the friction coefficient. The value $\mu = 0.3$ was identified as producing a simulation a flow rate that matched the experimental case. This value was subsequently used for the $2.5\,\mathrm{mm}$ and $3.0\,\mathrm{mm}$ simulations. At no point during the duration of the flow and for neither of the two gap scenarios was the error between experiment and simulation larger than 3% [Heyn 2013].

4.2.5 *Impact Problem.* The discretization process that yields Eqs. (5) through (7) starting from Eq. (3) is one of several ways to produce a numerical solution of the DVI problem associated with frictional contact in multibody dynamics. One can argue that the very form of the equations of motion in Eq. (3) can and should be improved. A compelling case is made in [Smith et al. 2012], where the CCP approached adopted herein is shown to yield incorrect dynamics for certain impact problems. The issue can be traced to the left side of the complementarity condition in Eq. (6), which upon multiplication by h contains the first two terms of a Taylor series expansion of the gap function evaluated at $\mathbf{q}^{(l+1)}$. The consequence of this numerical discretization choice is that impacts are always inelastic, since with no additional physics brought into the model, the very condition $\Phi_i(\mathbf{q}^{(l+1)})$ implies the inelastic attribute of the discretization scheme. How big of a problem this represents is the focus of this subsection. Clearly, the answer is independent of the method to solve the CCP that leads to the optimization problem in Eq. (11). At that point in the solution sequence, any method to solve the optimization problem, APGD or GS, should produce roughly the same solution albeit at different computational costs. We used APGD and a ball drop test, see Fig. 13, to gauge the extent of the error in this benchmark test problem in which balls of three materials – polypropylene, wood, and teflon – are dropped one at a time onto a bed of granular material made up of 500,400 spherical bodies with $1.0\,\mathrm{mm}$ diameter and material density of $2.5\,\mathrm{g/cm^3}$. Figure 13 shows four snapshots from the impact simulation at t =0.0008 s, t =0.0128 s, t =0.0248 s, and t =0.0368 s respectively. The last snapshot represents the instant of deepest penetration. The images show a cut-away through the center of the ball that exposes the profile of the crater.

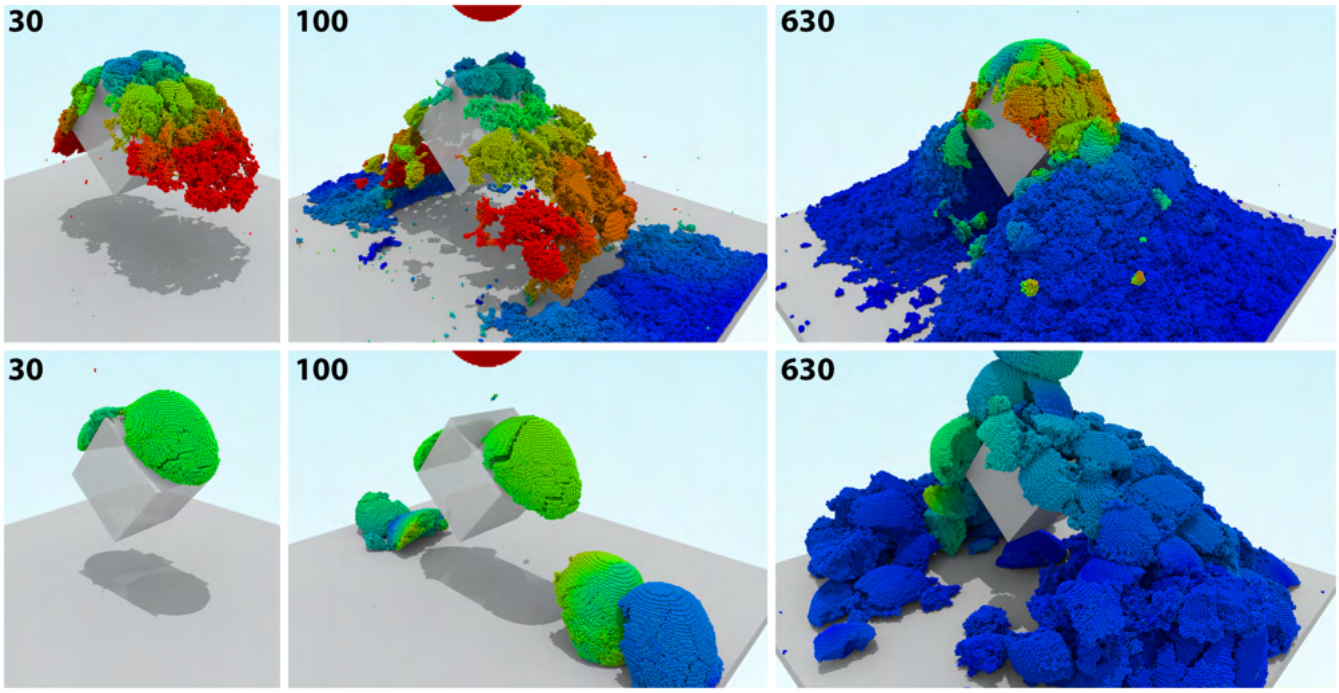

Fig. 8. Balls made up of spheres are dropped onto a cube rotated 45° in an enclosed container have different cohesion parameters: $\hat{c}$ =500 N and $\hat{c}$ =4000 N. The final frame contains approximately 1.6 million rigid spheres and 6-7 million contacts, which leads to a CCP in approximately 20 million variables.

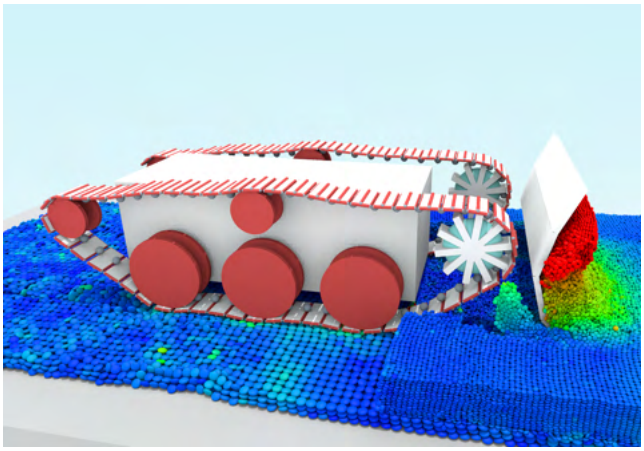

Fig. 9. A 7300 kg bulldozer interacting with a cohesive material for which $\hat{c}$ =500 N.

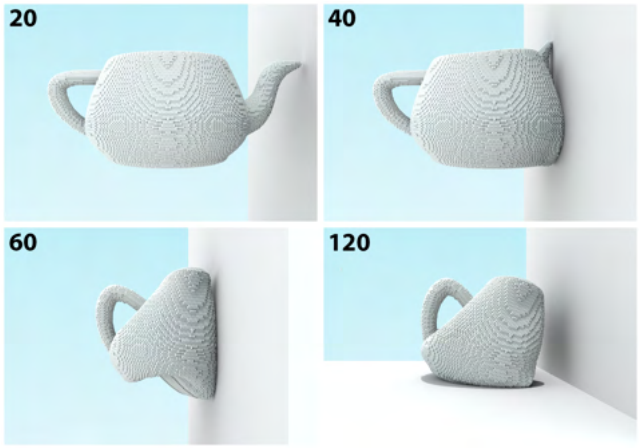

Fig. 10. Walled teapot made up of 80 000 spheres with a cohesion value of $\hat{c}$ =5 × 10^6 N. The integration time step was h =0.001 s.

It was shown in [Uehara et al. 2003] and confirmed in [Ambroso et al. 2005] that the penetration depth d reached by the bottom of the ball had the following empirical formula:

$$d = \frac{0.14}{\mu}\left(\frac{\rho_b}{\rho_g}\right)^{1/2} D_b^{2/3} H^{1/3}\,, \tag{28}$$

where h represents the height from which the ball is dropped (relative to the bottom of the ball and with respect to the granular testbed), $H = h + d$, $\mu = 0.3$ is the friction coefficient, ρ_b is the density of the ball, ρ_g is the density of the material that makes up the granular bodies, and D_b is the diameter of the ball. All experiments performed in [Ambroso et al. 2005] used a granular medium composed of glass spheres with a diameter range of 0.25 mm-0.35 mm and material density of 2.5 g/cm^3. In [Uehara et al. 2003], a variety of granular materials were used, ranging from popcorn kernels of size 4 mm × 6 mm × 7 mm to beach sand of size 0.5 ± 0.4 mm.

The linear fit based on the simulation results suggests that the penetration depth satisfies the following scaling:

Fig. 11. Two objects made up of 154 000 spheres with cohesion values of $\hat{c} = 1 \times 10^4$ N (top) and $\hat{c} = 1 \times 10^5$ N (bottom). The integration time step was $h = 0.001$ s.

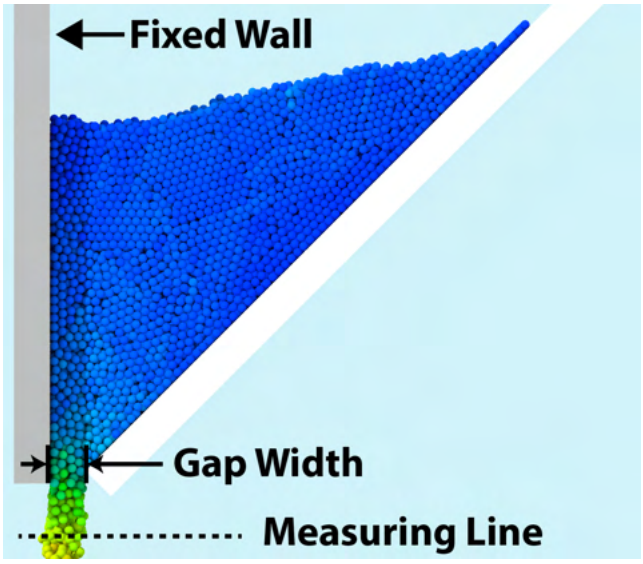

Fig. 12. Schematic of mass flow rate validation experiment. The colors indicate the speed magnitude of each rigid body.

$$d = \frac{0.1221}{\mu} \left(\frac{\rho_b}{\rho_g} \right)^{1/2} D_b^{2/3} H^{1/3} , \tag{29}$$

which suggests that the empirical constants determined from simulation and experiments for these impact tests show a 13.7% difference.

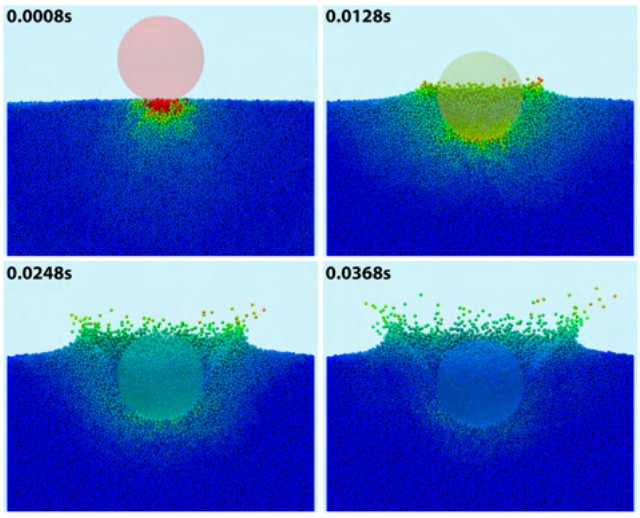

Fig. 13. Snapshots from impact simulation with $h = 20$ cm and $\rho_s = 0.7\,\mathrm{g/cm^3}$ (wood). Simulation time shown in the upper left corner.

5. DISCUSSION

This paper makes two points. For the first time, an approach for handling cohesion outlined in [Tasora et al. 2013] and summarized in Section 2.4 has been demonstrated to work in practice for problems with millions of contacts. To the best of our knowledge, this treatment of cohesion in rigid multibody dynamics is new. While leading to visually pleasing results, it remains to gauge to what extent this cohesion model can reproduce physical behavior in applications such as snow dynamics, deformable terrain modeling, or crack propagation. Second, a new algorithm called APGD has been shown to effectively solve an optimization problem that yields the set of frictional contact forces at each integration time step.

Substituting the projected Jacobi and/or GS steps with APGD can in principle both simplify and accelerate the solution of the velocity-based shock propagation method proposed in [Cottle et al. 2009; Erleben 2007]. Therein, upon partitioning the problem at each time step into blocks, the iterative solution proceeds using an unconventional block-level mixture of Jacobi and GS that draws on ideas proposed in [Bridson et al. 2002; Guendelman et al. 2003]. In [Guendelman et al. 2003], the resolution of the collision and contact events is local and treats each pairwise event independently, first for collisions and subsequently for contacts. The events are visited in random fashion to yield a relaxation that is similar to Moreau's Gauss-Seidel method. The authors indicate that many iterations are needed before the events are resolved and an ad-hoc approach that relies on ordered graphs to correct for body interpenetration is proposed. Nonconvex objects are handled using a mixed representation of the geometry that combines triangulated surfaces with signed distance functions defined on grids. The major difference between the methodology embraced here and the one in [Guendelman et al. 2003] goes back to the global nature of the former approach. If in [Guendelman et al. 2003] the goal is to generate a visually pleasing evolution of stacked bodies fast, herein we strive to maintain a predictive attribute of the overall method by globally resolving at each time step all friction and contact forces present in the system.

The relaxation of the complementarity condition in Eq. (9) has two consequences. First, on the upside, the numerical solution can leverage a wider spectrum of methods to handle the resulting

quadratic program with conic constraints in Eq. (11). We took advantage of this by proposing the APGD algorithm to accelerate the frictional contact force computation. Second, on the downside, the relaxation embedded in the solution introduces artifacts that were discussed in Section 2.3. It should be pointed out that there are approaches that do not fall back on a relaxation, see for instance [Bertails-Descoubes et al. 2011; Daviet et al. 2011; Acary et al. 2011]. In [Daviet et al. 2011], the problem is formulated as an NCP which is solved using a Newton method and in [Bertails-Descoubes et al. 2011] a nonsmooth Newton approach is utilized to solve a fixed point problem.

Ultimately it comes down to either (i) embracing a relaxed approach that leads to a tame numerical problem, e.g., a CCP or LCP, which has a global solution that might displays unphysical artifacts, or (ii) staying faithful to the Coulomb dry friction model and dealing with a numerical problem that in circumstances not known a priori might not have a solution for large and/or complex systems. This paper does not attempt to elucidate this conundrum. Instead, it proposes an approach to solve the CCP that within the bounds of a relaxed DVI formulation: (a) is faster than other first order methods used today, e.g.: projected Jacobi, Gauss-Seidel, Krylov-subspace or variants discussed in [Moreau and Jean 1996; Anitescu and Hart 2004; Glocker and Pfeiffer 2006; Tasora et al. 2008; Preclik et al. 2009; Shojaaee et al. 2012; Tonge et al. 2012; Heyn et al. 2013]; i.e., first order methods that do not rely on Jacobian/Hessian information for convergence; (b) can resolve bilateral constraints; and (c) can handle cohesion by a simple modification of the gap complementarity condition. It should be pointed out that another issue eschewed here that plagues both (i) and (ii) is the lack of uniqueness in the solution.

6. CONCLUSIONS AND DIRECTIONS OF FUTURE WORK

This contribution introduces an algorithm, called APGD, that is used at each time step of a simulation to produce the friction and normal forces acting at the contact points in a system of mutually interacting rigid bodies. The APGD method has been implemented into an open source parallel simulation framework and demonstrated to scale to accommodate the analysis of systems with millions of rigid bodies. The benchmark pressure test results reported in Table I suggest the method is numerically robust. Numerous tests have indicated that almost invariably the APGD solver is one order of magnitude faster than Chrono's default Gauss-Seidel solver. The ability to handle bilateral constraints, cohesion, and rolling and spinning friction suggests that the method is versatile and can be used to model physics that goes beyond the dry Coulomb sliding/sticking friction model. The APGD method can be applied in other complementarity-based approaches, such as [Smith et al. 2012], that address the limited support for impact phenomena in Chrono.

Chrono is a matrix-free simulation framework, which prevents the use of (i) optimized linear algebra kernels, and (ii) Interior Point methods that might prove more apt at solving the discretized DVI problem of interest. In terms of (i), a non-matrix free implementation, which is presently pursued, will allow the use of optimized kernels for matrix-vector multiplication in APGD. Based on results available in the literature [Su and Keutzer 2012; Intel 2013] and APGD profiling, we anticipate an increase in performance by better use of caches, better leverage of parallelism, from vectorization, and use of larger virtual page sizes.

ACKNOWLEDGMENTS

Financial support was provided by Army Research Office award W911NF-12-1-0395 and National Science Foundation award CMMI-GOALI–1362583.

REFERENCES

ACARY, V., CADOUX, F., LEMARECHAL, C., AND MALICK, J. 2011. A formulation of the linear discrete Coulomb friction problem via convex optimization. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik 91,* 2, 155–175.

AMBROSO, M. A., SANTORE, C. R., ABATE, A. R., AND DURIAN, D. J. 2005. Penetration depth for shallow impact cratering. *Physical Review E 71*, 051305.

ANDERSEN, E. AND ANDERSEN, K. 2000. The mosek interior point optimizer for linear programming: An implementation of the homogeneous algorithm. In *High Performance Optimization*, H. Frenk, K. Roos, T. Terlaky, and S. Zhang, Eds. Applied Optimization, vol. 33. Springer US, 197–232.

ANDERSEN, E., ROOS, C., AND TERLAKY, T. 2003. On implementing a primal-dual interior-point method for conic quadratic optimization. *Mathematical Programming 95,* 2, 249–277.

ANITESCU, M. 2006. Optimization-based simulation of nonsmooth rigid multibody dynamics. *Mathematical Programming 105,* 1, 113–143.

ANITESCU, M., CREMER, J. F., AND POTRA, F. A. 1996. Formulating 3D contact dynamics problems. *Mechanics of Structures and Machines 24(4)*, 405–437.

ANITESCU, M. AND HART, G. D. 2004. A constraint-stabilized time-stepping approach for rigid multibody dynamics with joints, contact and friction. *International Journal for Numerical Methods in Engineering 60(14)*, 2335–2371.

ANITESCU, M. AND TASORA, A. 2010. An iterative approach for cone complementarity problems for nonsmooth dynamics. *Computational Optimization and Applications 47,* 2, 207–235.

BECK, A. AND TEBOULLE, M. 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences 2,* 1, 183–202.

BECKER, S. R., CANDÈS, E. J., AND GRANT, M. C. 2011. Templates for convex cone problems with applications to sparse signal recovery. *Mathematical Programming Computation 3,* 3, 165–218.

BERTAILS-DESCOUBES, F., CADOUX, F., DAVIET, G., AND ACARY, V. 2011. A nonsmooth Newton solver for capturing exact Coulomb friction in fiber assemblies. *ACM Transactions on Graphics (TOG) 30,* 1, 6.

BERTSEKAS, D. 1976. On the Goldstein-Levitin-Polyak gradient projection method. *Automatic Control, IEEE Transactions on 21,* 2, 174–184.

BODIN, K., LACOURSIERE, C., AND SERVIN, M. 2012. Constraint fluids. *IEEE Transactions on Visualization and Computer Graphics 18,* 3 (Mar.), 516–526.

BONNEFON, O. AND DAVIET, G. 2011. Quartic formulation of Coulomb 3D frictional contact. Rapport technique RT-0400, INRIA.

BRIDSON, R., FEDKIW, R., AND ANDERSON, J. 2002. Robust treatment of collisions, contact and friction for cloth animation. *ACM Trans. Graph. 21,* 3 (July), 594–603.

BRILLIANTOV, N. V., SPAHN, F., HERTZSCH, J.-M., AND PÖSCHEL, T. 1996. Model for collisions in granular gases. *Physical Review E 53,* 5, 5382.

CADOUX, F. 2009. An optimization-based algorithm for Coulomb's frictional contact. *ESAIM: Proc. 27*, 54–69.

CAUCHY, A. 1847. Méthode générale pour la résolution des systemés d'équations simultanées. *Comp. Rend. Sci. Paris 25,* 1847, 536–538.

COTTLE, R. W., PANG, J.-S., AND STONE, R. E. 2009. *The linear complementarity problem*. Vol. 60. Siam.

CUNDALL, P. 1971. A computer model for simulating progressive large-scale movements in block rock mechanics. In *Proceedings of the International Symposium on Rock Mechanics. Nancy, France*.

CUNDALL, P. 1988. Formulation of a three-dimensional distinct element model–Part I. A scheme to detect and represent contacts in a system composed of many polyhedral blocks. *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts 25,* 3, 107–116.

CUNDALL, P. AND STRACK, O. 1979. A discrete element model for granular assemblies. *Geotechnique 29*, 47–65.

DAVIET, G., BERTAILS-DESCOUBES, F., AND BOISSIEUX, L. 2011. A hybrid iterative solver for robustly capturing Coulomb friction in hair dynamics. In *ACM Transactions on Graphics (TOG)*. Vol. 30. ACM, 139.

DE SAXCÉ, G. AND FENG, Z.-Q. 1998. The bipotential method: a constructive approach to design the complete contact law with friction and improved numerical algorithms. *Mathematical and Computer Modelling 28,* 4, 225–245.

DELANNAY, R., LOUGE, M., RICHARD, P., TABERLET, N., AND VALANCE, A. 2007. Towards a theoretical picture of dense granular flows down inclines. *Nature Materials 6,* 2, 99–108.

DEMMEL, J. W. 2011. Superlu users' guide. *Lawrence Berkeley National Laboratory*.

DIRKSE, S. P. AND FERRIS, M. C. 1995. The PATH solver: a nommonotone stabilization scheme for mixed complementarity problems. *Optimization Methods and Software 5,* 2, 123–156.

ERLEBEN, K. 2007. Velocity-based shock propagation for multibody dynamics animation. *ACM Transactions on Graphics (TOG) 26,* 2, 12.

FILIPPOV, A. 1967. Classical solutions of differential equations with multi-valued right-hand side. *SIAM Journal on Control 5,* 4, 609–621.

FLICKINGER, D. M., WILLIAMS, J., AND TRINKLE, J. 2013. What's wrong with collision detection in multibody dynamic simulation? In *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE.

GLOCKER, C. AND PFEIFFER, F. 2006. An LCP-approach for multibody systems with planar friction. In *Proceedings of the CMIS 92 Contact Mechanics Int. Symposium*. Lausanne, Switzerland, 13–20.

GUENDELMAN, E., BRIDSON, R., AND FEDKIW, R. 2003. Nonconvex rigid bodies with stacking. In *ACM Transactions on Graphics (TOG)*. Vol. 22. ACM, 871–878.

HAUG, E. J. 1989. *Computer-Aided Kinematics and Dynamics of Mechanical Systems Volume-I*. Prentice-Hall, Englewood Cliffs, New Jersey.

HEYN, T. 2013. PhD thesis. Ph.D. thesis, Department of Mechanical Engineering, University of Wisconsin–Madison, http://sbel.wisc.edu/documents/TobyHeynThesis_PhDfinal.pdf.

HEYN, T., ANITESCU, M., TASORA, A., AND NEGRUT, D. 2013. Using Krylov subspace and spectral methods for solving complementarity problems in many-body contact dynamics simulation. *International Journal for Numerical Methods in Engineering 95,* 7, 541–561.

INTEL. 2013. The Intel Math Kernel Library Sparse Matrix Vector Multiply Format Prototype Package. Available online at http://software.intel.com/en-us/articles/the-intel-math-kernel-library-sparse-matrix-vector-multiply-format-prototype-package.

JAEGER, H. M., NAGEL, S. R., AND BEHRINGER, R. P. 1996. Granular solids, liquids, and gases. *Rev. Mod. Phys. 68*, 1259–1273.

JOHNSON, K. L. 1987. *Contact mechanics*. Cambridge University Press.

KANE, C., REPETTO, E., ORTIZ, M., AND MARSDEN, J. 1999. Finite element analysis of nonsmooth contact. *Computer methods in applied mechanics and engineering 180,* 1, 1–26.

KAUFMAN, D. M. AND PAI, D. K. 2012. Geometric Numerical Integration of Inequality Constrained, Nonsmooth Hamiltonian Systems. *SIAM Journal on Scientific Computing 34,* 5, A2670–A2703.

LI, A., SERBAN, R., AND NEGRUT, D. 2013. A SPIKE-based approach for the parallel solution of sparse linear systems on GPU cards. Tech. Rep. TR-2013-05–http://sbel.wisc.edu/documents/TR-2013-05.pdf, SBEL, University of Wisconsin - Madison.

LUDING, S. 2005. *Molecular Dynamics Simulations of Granular Materials*. Wiley-VCH Verlag GmbH, 297–324.

MAZHAR, H., BOLLMANN, J., FORTI, E., PRAEGER, A., OSSWALD, T., AND NEGRUT, D. 2013. Studying the Effect of Powder Geometry on the Selective Laser Sintering Process. Tech. Rep. TR-2013-03: http://sbel.wisc.edu/documents/TR-2013-03.pdf, Simulation-Based Engineering Laboratory, University of Wisconsin-Madison.

MAZHAR, H., HEYN, T., AND NEGRUT, D. 2011. A scalable parallel method for large collision detection problems. *Multibody System Dynamics 26*, 37–55. 10.1007/s11044-011-9246-y.

MAZHAR, H., HEYN, T., PAZOUKI, A., MELANZ, D., SEIDL, A., BARTHOLOMEW, A., TASORA, A., AND NEGRUT, D. 2013. Chrono: a parallel multi-physics library for rigid-body, flexible-body, and fluid dynamics. *Mechanical Sciences 4,* 1, 49–64.

MAZHAR, H., MELANZ, D., FERRIS, M., AND NEGRUT, D. 2014. An Analysis of Several Methods for Handling Hard-Sphere Frictional Contact in Rigid Multibody Dynamics. Tech. Rep. TR-2014-11–http://sbel.wisc.edu/documents/TR-2014-11.pdf, Simulation-Based Engineering Laboratory, University of Wisconsin-Madison.

MAZHAR, H., SCHNEIDER, J., AND NEGRUT, D. 2014. Preliminary results for helical anchoring project. Tech. Rep. TR-2014-10: http://sbel.wisc.edu/documents/TR-2014-10.pdf, Simulation-Based Engineering Laboratory, University of Wisconsin-Madison.

MOREAU, J. J. AND JEAN, M. 1996. Numerical treatment of contact and friction: The contact dynamics method. In *Proceedings of the Third Biennial Joint Conference on Engineering Systems and Analysis*. Montpellier, France, 201–208.

NEMIROVSKY, A. AND YUDIN, D. B. 1983. *Problem complexity and method efficiency in optimization.* John Wiley & Sons.

NESTEROV, Y. 1983. A method of solving a convex programming problem with convergence rate $\mathcal{O}(1/k^2)$. In *Soviet Mathematics Doklady*. Vol. 27(2). 372–376.

NESTEROV, Y. 2003. *Introductory lectures on convex optimization: A basic course*. Vol. 87. Springer.

O'DONOGHUE, B. AND CANDES, E. 2012. Adaptive restart for accelerated gradient schemes. *ArXiv e-prints*.

PÖSCHEL, T. AND SCHWAGER, T. 2005. *Computational granular dynamics: models and algorithms*. Springer.

PRECLIK, T. M., ERGER, K. I., AND RÜDE, U. 2009. Iterative rigid multibody dynamics. In *Proceeding of Multibody Dynamics ECCOMAS Thematic Conference*.

PRECLIK, T. M. AND RÜDE, U. 2011. Solution existence and non-uniqueness of Coulomb friction. Technical Report 4, Friedrich-Alexander-University Erlangen-Nurnberg, Institut Fur Informatik, Nurnberg, Germany.

SCHENK, O. AND GÄRTNER, K. 2004. Solving unsymmetric sparse systems of linear equations with pardiso. *Future Generation Computer Systems 20,* 3, 475–487.

SHOJAAEE, Z., SHAEBANI, M. R., BRENDEL, L., TOEROEK, J., AND WOLF, D. E. 2012. An adaptive hierarchical domain decomposition method for parallel contact dynamics simulations of granular materials. *Journal of Computational Physics 231,* 2 (JAN 20), 612–628.

SMITH, B., KAUFMAN, D. M., VOUGA, E., TAMSTORF, R., AND GRINSPUN, E. 2012. Reflections on Simultaneous Impact. *ACM Transactions on Graphics 31,* 4, 106:1–106:12.

STEWART, D. E. 2000. Rigid-body dynamics with friction and impact. *SIAM Review 42(1)*, 3–39.

STEWART, D. E. AND TRINKLE, J. C. 1996. An implicit time-stepping scheme for rigid-body dynamics with inelastic collisions and Coulomb friction. *International Journal for Numerical Methods in Engineering 39*, 2673–2691.

SU, B.-Y. AND KEUTZER, K. 2012. clSpMV: A cross-platform OpenCL SpMV framework on GPUs. In *Proceedings of the 26th ACM international conference on Supercomputing*. ACM, 353–364.

TASORA, A. AND ANITESCU, M. 2013. A complementarity-based rolling friction model for rigid contacts. *Meccanica 48,* 7, 1643–1659.

TASORA, A., ANITESCU, M., NEGRINI, S., AND NEGRUT, D. 2013. A compliant visco-plastic particle contact model based on differential variational inequalities. *International Journal of Non-Linear Mechanics 53,* SI (Jul), 2–12.

TASORA, A., NEGRUT, D., AND ANITESCU, M. 2008. Large-scale parallel multi-body dynamics with frictional contact on the Graphical Processing Unit. *Journal of Multi-body Dynamics 222,* 4, 315–326.

TONGE, R., BENEVOLENSKI, F., AND VOROSHILOV, A. 2012. Mass splitting for jitter-free parallel rigid body simulation. *ACM Transactions on Graphics (TOG) 31,* 4, 105.

TRINKLE, J. C. 2003. Formulation of multibody dynamics as complementarity problems. In *Volume 5: 19th Biennial Conference on Mechanical Vibration and Noise, Parts A, B, and C*. ASME.

UEHARA, J. S., AMBROSO, M. A., OJHA, R. P., AND DURIAN, D. J. 2003. Low-speed impact craters in loose granular media. *Phys. Rev. Lett. 90*, 194301.

VU-QUOC, L., LESBURG, L., AND ZHANG, X. 2004. An accurate tangential force–displacement model for granular-flow simulations: Contacting spheres with plastic deformation, force-driven formulation. *Journal of Computational Physics 196,* 1, 298–326.

VU-QUOC, L. AND ZHANG, X. 1999. An elastoplastic contact force–displacement model in the normal direction: displacement–driven version. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences 455,* 1991, 4013–4044.

Received Sep 2014; accepted Feb 2015