Available online at www.sciencedirect.com

ScienceDirect

Comput. Methods Appl. Mech. Engrg. 278 (2014) 160–185

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

An impulse-based energy tracking method for collision resolution

XuHai Tang, Adriana Paluszny*, Robert W. Zimmerman

Department of Earth Science and Engineering, Imperial College, London, United Kingdom

Received 22 May 2013; received in revised form 19 March 2014; accepted 14 May 2014
Available online 23 May 2014

Highlights

- DEM can be based on penalties or impulses to resolve collisions.
- The impulse-based energy tracking method (ETM) resolves multiple collisions consistently.
- ETM models multiple collisions iteratively yet simultaneously, while tracking the system energy.
- ETM does not rely on penalties and does not require computation of penetration of bodies.
- The method is validated in the context of energy conservation and meso-scale angles of repose.

Abstract

Discrete element methods can be based on either penalties or impulses to resolve collisions. A generic impulse based method, the energy tracking method (ETM), is described to resolve collisions between multiple non-convex bodies in three dimensions. As opposed to the standard sequential impulse method (SQM) and simultaneous impulse method (SMM), which also apply impulses to avoid penetration, the energy tracking method changes the relative velocity between two colliding bodies iteratively yet simultaneously. Its main novelty is that impulses are applied gradually at multi-point contacts, and energy changes at the contact points are tracked to ensure conservation. Three main steps are involved in the propagation of the impulses during the single- and multi-contact resolution: compression, restitution-related energy loss, and separation. Numerical tests show that the energy tracking method captures the energy conservation property of perfectly elastic single- and multi-point collisions. ETM exhibits improved angular velocity estimation, as compared to SMM and SQM, as demonstrated by two numerical examples that model multi-point contact between box-shaped objects. Angles of repose estimated for multi-object pack repositioning of spheres, cubes, and crosses are in good agreement with the reported experimental values.

Keywords: Discrete element method; Energy conservation; Impulse method; Multiple-body collision

1. Introduction

The numerical simulation of collision-driven interactions between a large number of bodies has many applications in engineering, ranging from the simulation of rock fragmentation in mining engineering processes [1–3], to the simulation of particle flow for tumble mills in agriculture [4]. A robust numerical method to simulate multi-body interaction must satisfy run-time constraints and handle large amounts of fragments of a range of sizes, as well as

* Corresponding author. Tel.: +44 20 7594 7435; fax: +44 20 7594 7444.
E-mail address: apaluszn@imperial.ac.uk (A. Paluszny).

http://dx.doi.org/10.1016/j.cma.2014.05.004
0045-7825/© 2014 Elsevier B.V. All rights reserved.

being stable and physically realistic. Current numerical approaches to model these processes include penalty-based [5], analytical [6], and impulse-based methods [7].

The impulse method, first proposed by Hahn [8], applies the impulse–momentum form of Newton's second law to compute collision response within a multi-body system. The family of impulse methods provides an efficient and accurate alternative to resolve high-frequency collisions of hundreds or thousands of colliding bodies. Impulse methods have been shown to be energy conservative [9], do not rely on penalty parameters, do not require the computation of penetration between bodies, and thrive on their ability of directly using meso-scale material properties. Impulse methods are regularly applied to model haptic interfaces in robotics research [10] and to simulate the motion of tendons, joints, and muscles [11,12]. Impulse methods have been used in engineering for simulated sensorless manipulation of objects [13], and in combination with the finite element method for the simulation of gravity-driven fragmentation [3]. Unlike *a posteriori* methods, the impulse-based method does not rely on arbitrary penalty and damping parameters that define repulsion as a function of penetration. Instead, this method models collisions as a function of predicted impact trajectory and impulse. Thus, impulse methods do not require geometric intersection checks, but instead relies on collision time estimation algorithms, e.g. the efficient Lin–Canny algorithm [14,15]. Contact detection is based on the computation of the relative normal velocity of colliding bodies at each contact point. Newton's impact law, as defined for a single-point collision, is applied either sequentially in sequential impulse methods (SQM) [16] or simultaneously in simultaneous impulse methods (SMM) [17,18] to deal with multiple collisions and multiple contact points. There are at least two versions of simultaneous impulse methods. In 1989, Baraff [17] proposed a simple version of SMM which treated multiple collisions as a linear complementary problem; this method only considered normal impulses and the change of normal velocities, and it did not consider friction. The generic SMM [18] enforces constraints on all collisions by solving a system of linear equations, taking into account impulses in the normal and tangential directions, and considering friction. Due to the simultaneous evaluation of collisions, SMM cannot capture the propagation of contact forces during a collision. Sequential impulse methods (SQM) treat multiple concurrent collisions one-by-one, and non-penetration constraints are enforced to multiple collisions sequentially, iterating over the contact points until no objects collide. In SQM, Newton's impact law is locally applied to each contact, making it particularly well suited for parallelization. A drawback of SQM is that it inverts relative normal velocities of collisions one-by-one, yielding results that depend on the order in which collisions are resolved [16,19].

The penalty method iteratively computes penalty parameters to enforce collisions as a set of non-penetration constraints [20,21]. It is widely used due to its simplicity and flexibility [22–24]. However, this method relies on the definition of an arbitrary, experimentally estimated, normal and tangential stiffness of an enforced spring system that prevents object penetration [25–27]. Relative high collision velocities lead to large overlaps and exaggerated repulsion forces, and introduce instability to the simulation. This results in gearing the development of the method towards increasing stability and disregarding possible emerging artificial energy gains and losses [28,29]. The Lagrange multiplier method is an alternative approach that adds unknowns to solve bilateral constraints, such as joint constraints [29]. The midpoint integration rule combined with the Lagrange multiplier method [30] and penalty regularization of contact rates [31,32] has been proposed to ensure energy conservation during collisions. Penalties are applied to solve collision and contact of objects within the context of the discrete element method (DEM) [33,34], smoothed particle hydrodynamics (SPH) [35,36], and the material point method (MPM) [37,38]. Sphere-conglomerates are used to avoid penetration computation penalties [2]; however, these models are unable to represent smooth surfaces and require large number of spheres that scale with level of detail. Smoothed particle hydrodynamics and the material point method are point-based methods that are best suited to simulate explosions and large deformation problems, and rely on point-to-point contact detection algorithms, resulting in difficult numerical computation of contact force direction [39]. The main geometric hurdles of the penalty method are the need to calibrate micro-to-macro material properties of conglomerates using laboratory experiments, and the arbitrary definition of penalty parameters during the definition of the non-penetration constraints.

The hybrid finite element–discrete element method [40] can model deformation of colliding bodies as a function of impact. Deformation of the bodies is penalized by stark computational costs and is well suited for high-resolution models [24], e.g. impact-driven non-linear degradation of bodies. Models that seek to study emerging behavior from the interaction of hundreds and thousands of objects often assume bodies to be rigid at the moment of impact [41]. One-way coupling of brittle deformation, such as fragmentation, can be used in combination with the rigidity assumption to reduce computational cost [3]. Impulse-based methods pose no restriction in their use within the context of deformability.

In analytical methods, the simulation of non-penetration constraints is treated as a linear complementary problem [42], which does not rely on penalty parameters and does not require use of a damping parameter to remain stable. Its main difficulty is in considering the influence of friction on the collision. For instance, for systems with dynamic or sliding friction, the linear complementary problem matrix becomes non-positive definite and non-symmetric, constraints at the contact point become non-linear, and the problem becomes a non-linear complementarity problem [29].

Energy conservation is critical to ensure stability of a numerical method, especially for contact and collision problems [28,43]. A number of conserving schemes have been developed to ensure energy conservation. These schemes make use of the penalty regulation of normal contact constraint and inherit the conservation property from continuum problems. These conservation schemes can conveniently be combined with the finite element method to simulate frictionless [44] and frictional [43] contact and collision. Hesch and Betsch [45] formulated the node-to-segment contact method and solved large deformation contact problems with the conserving scheme. More recently, an energy and momentum-conserving temporal discretization scheme [46] was developed for adhesive contact problems without considering friction and dissipation. Even though the conserving scheme improves numerical stability, it also inherits from the penalty method the difficulty of having to determine penalty parameters. In order to remove penalty sensitivity, Chawla and Laursen [47] proposed an energy and momentum conserving algorithm, which makes use of Lagrange multipliers instead of penalty parameters.

The generalized impulse method presented here, called the energy tracking method (ETM), addresses the individual shortcomings of the sequential impulse method (SQM) and simultaneous impulse method (SMM). It is aimed at efficiently and accurately modeling high-frequency collisions of non-convex bodies. ETM treats multiple concurrent collisions as a series of single-point collisions, and introduces an additional level of iteration at each collision time step. At the beginning of the collision analysis the relative velocity is negative, indicating that the bodies are on a collision course. An impulse is applied to each body to avoid penetration, in a step-wise manner, incrementally increasing the relative velocity until it reaches zero. At each step, velocities at each collision point are updated. During this process, kinetic energy decreases and elastic energy at the contact points increases. Subsequently, the relative normal velocity continues to increase until all of the elastic energy is released. As opposed to the standard SQM, ETM gradually applies impulses to change relative normal velocities of multiple collisions iteratively yet simultaneously. In contrast to the SMM, ETM yields low angular velocity errors when dealing with multi-contact collisions, due to its ability to simulate the interaction of forces during collision. ETM is advantageous as it processes collisions in a simultaneous, albeit iterative, manner. Energy is shown to be conservative for large multi-body systems of spheres, cubes and non-convex crosses. Numerical experiments evaluating the angle of repose of systems of these objects compare well with macroscopic laboratory experiments reported in the literature [48].

2. Impulse method: contact model for single-point collision

The energy tracking method is based on two levels of iteration: iteration for time steps, $\mathbf{t}$, and iteration for impulse computation at each collision time step, $\mathbf{s}$. As time progresses, $\mathbf{t} = \{t_0, t_1, t_2, \ldots\}$, positions and velocities of bodies are updated. The non-penetration strategy [14,15], based on the fast search of the closest features among convex polyhedra, identifies collision candidates based on the concept of collision time. Accurate estimation of collision time, t_c, is required to apply impulses and avoid geometric overlap due to collision (Fig. 1). Impulses are then computed for each contact point using an iterative method, over a set of iterations $\mathbf{s} = \{s_0, s_1, s_2, \ldots\}$. The latter captures the propagation of impulses at a single instance in time. s_k indicates the step k. At step k, the linear velocity and angular velocity of body j are denoted as $\mathbf{v}^j(s_k)$ and $\omega^j(s_k)$. The impulse applied at contact point c_i is denoted as $\mathbf{p}_i(s_k)$.

In SQM and SMM, the relative normal velocity is directly inverted. The energy tracking method (ETM), on the other hand, gradually applies the impulses to change the relative normal velocity over a set of iterations, applying impulses to multiple collisions at the same time. This procedure is described for a single-point collision. Without loss of generality, the colliding objects are assumed not to deform.

A body b_N collides with another body b_M at contact point c_i (Fig. 2). A set of impulses, $\{\mathbf{p}_i(s_0), \mathbf{p}_i(s_1), \mathbf{p}_i(s_2), \ldots\}$, is applied at the contact points to each colliding body sequentially. The subscript i in $\mathbf{p}_i(s_k)$ denotes that the impulse is applied to the contact point c_i, and s_k denotes this impulse is applied at the iteration s_k. The velocities of colliding body b_j ($j = N$ or M) at the contact point c_i at iteration s_k is denoted as

$$\mathbf{u}_i^j(s_k) = \mathbf{v}^j(s_k) + \boldsymbol{\omega}^j(s_k) \times \mathbf{r}_i^j \tag{1}$$

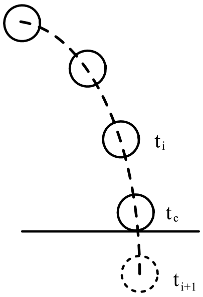


Fig. 1. Estimation of time of contact. Each circle represents a moving object approaching the ground at consecutive time steps. A collision is detected between time steps t_i and t_{i+1}. Collision time is estimated as t_c.

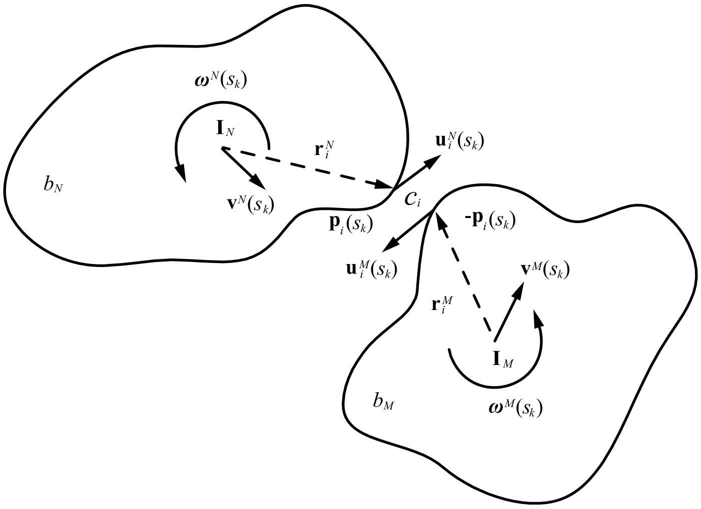


Fig. 2. Collision between bodies b_N and b_M at iteration s_k : $\mathbf{v}^N(s_k)$ and $\mathbf{v}^M(s_k)$ are linear velocities, $\omega^N(s_k)$ and $\omega^M(s_k)$ are angular velocities, $\mathbf{r}_i^N$ and $\mathbf{r}_i^M$ are the vectors from the center of mass of bodies to the contact point c_i, $\mathbf{u}_i^N(s_k)$ and $\mathbf{u}_i^M(s_k)$ are the velocities at the contact point, and $\mathbf{p}_i(s_k)$ is the impulse applied to the two colliding bodies to avoid penetration.

where $\mathbf{r}_i^j = \{r_{i,x}^j, r_{i,y}^j, r_{i,z}^j\}^{\mathrm{T}}$ is the vector from the center of mass of body b_j to the contact point c_i, which is assumed to be constant during the collision. $\mathbf{u}_i^j(s_k)$ is the velocity of body b_j at the contact point c_i. $\mathbf{v}^j(s_k)$ is the linear velocity at the center of mass, and $\omega^j(s_k)$ is the angular velocity. The relative velocity of two colliding bodies at contact point c_i, $\mathbf{u}_i(s_k)$, is defined as

$$\mathbf{u}_i(s_k) = \mathbf{u}_i^N(s_k) - \mathbf{u}_i^M(s_k). \tag{2}$$

The change of relative velocity from iteration s_k to iteration s_{k+1} is defined as

$$\triangle\mathbf{u}_i(s_k) = \mathbf{u}_i(s_{k+1}) - \mathbf{u}_i(s_k). \tag{3}$$

First, an assumption for the change of relative velocity before and after applying an impulse without considering energy loss is described. Energy loss caused by the impulse in the normal direction of the contact surface is handled by Stronge's hypothesis [49]. The consideration of friction is introduced in Section 5 for single-point collisions and Section 6 for multiple collisions. When a pair of impulses of opposite signs $\mathbf{p}_i(s_k)$ and $-\mathbf{p}_i(s_k)$ is applied to colliding bodies, these impulses effectively change their linear and angular velocities:

$$\begin{aligned} \mathbf{p}_i(s_k) &= m^N \triangle\mathbf{v}^N(s_k) \\ -\mathbf{p}_i(s_k) &= m^M \triangle\mathbf{v}^M(s_k) \end{aligned} \tag{4}$$

$$\begin{aligned}&\mathbf{r}_i^N \times \mathbf{p}_i(s_k) = \mathbf{I}^N \triangle\omega^N(s_k)\\&\mathbf{r}_i^M \times -\mathbf{p}_i(s_k) = \mathbf{I}^M \triangle\omega^M(s_k)\end{aligned} \tag{5}$$

where $\mathbf{I}^N$ and $\mathbf{I}^M$ are the inertia tensors, which are constant during the collision and are computed using an explicit exact expression [50], and

$$\begin{aligned}&\triangle\mathbf{v}^N(s_k) = \mathbf{v}^N(s_{k+1}) - \mathbf{v}^N(s_k)\\&\triangle\mathbf{v}^M(s_k) = \mathbf{v}^M(s_{k+1}) - \mathbf{v}^M(s_k)\end{aligned} \tag{6}$$

$$\begin{aligned}&\triangle\omega^N(s_k) = \omega^N(s_{k+1}) - \omega^N(s_k)\\&\triangle\omega^M(s_k) = \omega^M(s_{k+1}) - \omega^M(s_k)\end{aligned} \tag{7}$$

where m^N and m^M are the masses of body b_N and b_M.

Once the change of relative velocity is known, impulses are computed from the collision matrix [7],

$$\mathbf{p}_i(s_k) = \mathbf{K}_i^{-1} \triangle\mathbf{u}_i(s_k) \tag{8}$$

where the collision matrix $\mathbf{K}_i$, which is constant, non-singular, symmetric, and positive definite, is defined as

$$\mathbf{K}_i = \left(\frac{1}{m^N} + \frac{1}{m^M}\right)\mathbf{1} - \left(\tilde{\mathbf{r}}_i^N(\mathbf{I}^N)^{-1}\tilde{\mathbf{r}}_i^N + \tilde{\mathbf{r}}_i^M(\mathbf{I}^M)^{-1}\tilde{\mathbf{r}}_i^M\right) \tag{9}$$

where $\mathbf{1}$ is the identity matrix, and

$$\tilde{\mathbf{r}}_i^N = \begin{pmatrix} 0 & -\tilde{r}_{i,z}^N & \tilde{r}_{i,y}^N \\ \tilde{r}_{i,z}^N & 0 & -r_{i,x}^N \\ -\tilde{r}_{i,y}^N & \tilde{r}_{i,x}^N & 0 \end{pmatrix} \tag{10}$$

$$\tilde{\mathbf{r}}_i^M = \begin{pmatrix} 0 & -\tilde{r}_{i,z}^M & \tilde{r}_{i,y}^M \\ \tilde{r}_{i,z}^M & 0 & -r_{i,x}^M \\ -\tilde{r}_{i,y}^M & \tilde{r}_{i,x}^M & 0 \end{pmatrix} \tag{11}$$

where $\tilde{\mathbf{r}}_i^N$ and $\tilde{\mathbf{r}}_i^M$ are the cross-product matrix of $\mathbf{r}_i^N$ and $\mathbf{r}_i^M$.

ETM applies Stronge's hypothesis [49] to express the energy dissipation in the normal direction. Let W_{release} be the work done by the normal component of the collision impulse during a collision. Then, [7]

$$W_{\text{release}} = -e_n^2 W_{mc} \tag{12}$$

where $e_n \in (0, 1)$ is the normal restitution coefficient which is a measurement of bounce, where $e_n = 1$ implies a perfectly elastic collision and $e_n = 0$ implies that the relative normal velocity is reduced to zero after the collision, and W_{mc} is the work done by the normal impulse at the point of maximum compression, where the relative normal velocity changes sign.

For the following analysis, a local orthogonal coordinate system in $\{\mathbf{t}_i, \mathbf{n}_i, \mathbf{q}_i\}$ is defined at the locality of the contact in terms of the normal and tangential components of the collision (Fig. 3). It follows that

$$\mathbf{t}_i = \frac{\mathbf{n}_i \times \mathbf{p}_i(s_k)}{|\mathbf{u}_i(s_k)|} \tag{13}$$

$$\mathbf{q}_i = \mathbf{t}_i \times \mathbf{n}_i \tag{14}$$

where $\mathbf{n}_i$ is the normal direction of collision surface, and $\mathbf{t}_i$ and $\mathbf{q}_i$ are the two tangential directions at the contact point. It follows that the impulse $\tilde{\mathbf{p}}_i(s_k) = \{\tilde{p}_{i,t}(s_k), \tilde{p}_{i,n}(s_k), \tilde{p}_{i,q}(s_k)\}^{\mathrm{T}}$ and relative velocity $\tilde{\mathbf{u}}_i(s_k)$ can be formulated in terms of this local coordinate system as

$$\begin{cases}\tilde{\mathbf{u}}_i(s_k) = \mathbf{L}_i\mathbf{u}_i(s_k)\\ \tilde{\mathbf{p}}_i(s_k) = \mathbf{L}_i\mathbf{p}_i(s_k)\end{cases} \tag{15}$$

where $\mathbf{L}_i = (\mathbf{t}_i, \mathbf{n}_i, \mathbf{q}_i)^{\mathrm{T}}$ defines the transformation from global to local coordinates.

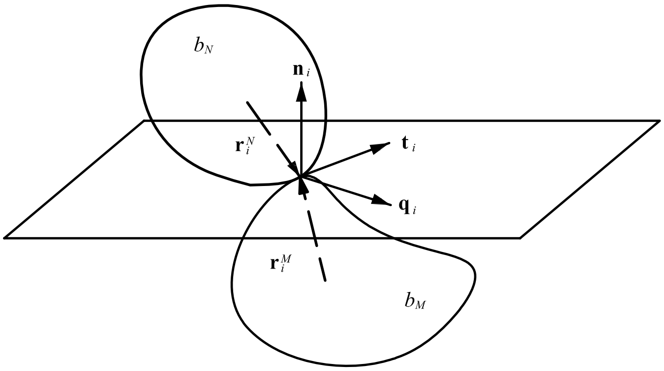


Fig. 3. A local coordinate system $(\mathbf{t}_i, \mathbf{n}_i, \mathbf{q}_i)$ is defined at the contact point between bodies b_M and b_N.

3. Single-point collision: impulse and work

Mirtich [7] proposed that the work done by contact forces, $\triangle W_i(s_k)$, is path independent and is a function of the relative velocity before and after applying the impulse:

$$\triangle W_i(s_k) = \frac{1}{2}\left(\mathbf{u}_i(s_{k+1}) + \mathbf{u}_i(s_k)\right)^{\mathrm{T}} \mathbf{K}_i^{-1} \left(\mathbf{u}_i(s_{k+1}) - \mathbf{u}_i(s_k)\right). \tag{16}$$

The amounts of work done by the normal and tangential impulses are considered separately.

Lemma 3.1. *During a collision, the work done by the impulse can be decomposed into the work done by the impulse in the normal direction and the work done by the impulse in the tangential direction, as follows*

$$\triangle W_{i,t}(s_k) = \frac{1}{2}(2\tilde{u}_{i,t}(s_k) + \triangle\tilde{u}_{i,t}(s_k))\tilde{p}_{i,t}(s_k) \tag{17}$$

$$\triangle W_{i,n}(s_k) = \frac{1}{2}(2\tilde{u}_{i,n}(s_k) + \triangle\tilde{u}_{i,n}(s_k))\tilde{p}_{i,n}(s_k) \tag{18}$$

$$\triangle W_{i,q}(s_k) = \frac{1}{2}(2\tilde{u}_{i,q}(s_k) + \triangle\tilde{u}_{i,q}(s_k))\tilde{p}_{i,q}(s_k). \tag{19}$$

Proof. According to Eq. (16), the work done by the contact force from iteration s_k to s_{k+1} is:

$$\triangle W_i(s_k) = \frac{1}{2}(\mathbf{u}_i(s_{k+1}) + \mathbf{u}_i(s_k))^{\mathrm{T}}\mathbf{K}_i^{-1}(\mathbf{u}_i(s_{k+1}) - \mathbf{u}_i(s_k)). \tag{20}$$

Substituting Eq. (8) into the previous equation yields

$$\triangle W_i(s_k) = \frac{1}{2}(\mathbf{u}_i(s_{k+1}) + \mathbf{u}_i(s_k))^{\mathrm{T}}\mathbf{p}_i(s_k). \tag{21}$$

Considering the collision in the local coordinate system defined by Eqs. (13) and (14), it follows that:

$$\triangle W_i(s_k) = \frac{1}{2}(\mathbf{L}_i^{-1}\tilde{\mathbf{u}}_i(s_{k+1}) + \mathbf{L}_i^{-1}\tilde{\mathbf{u}}_i(s_k))^{\mathrm{T}}\mathbf{L}_i^{-1}\tilde{\mathbf{p}}_i(s_k) \tag{22}$$

$$\triangle W_i(s_k) = \frac{1}{2}(\tilde{\mathbf{u}}_i(s_{k+1}) + \tilde{\mathbf{u}}_i(s_k))^{\mathrm{T}}\mathbf{L}_i^{-\mathrm{T}}\mathbf{L}_i^{-1}\tilde{\mathbf{p}}_i(s_k). \tag{23}$$

Since $\mathbf{L}_i$ is an orthogonal matrix, $\mathbf{L}_i^{-1} = \mathbf{L}_i^{\mathrm{T}}$,

$$\triangle W_i(s_k) = \frac{1}{2}(\tilde{\mathbf{u}}_i(s_{k+1}) + \tilde{\mathbf{u}}_i(s_k))^{\mathrm{T}}\tilde{\mathbf{p}}_i(s_k) \tag{24}$$

and

$$\triangle W_i(s_k) = \frac{1}{2}\begin{bmatrix} 2\tilde{u}_{i,t}(s_k) + \triangle\tilde{u}_{i,t}(s_k) \\ 2\tilde{u}_{i,n}(s_k) + \triangle\tilde{u}_{i,n}(s_k) \\ 2\tilde{u}_{i,q}(s_k) + \triangle\tilde{u}_{i,q}(s_k) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \tilde{p}_{i,t}(s_k) \\ \tilde{p}_{i,n}(s_k) \\ \tilde{p}_{i,q}(s_k) \end{bmatrix}. \tag{25}$$

Thus,

$$\triangle W_i(s_k) = \triangle W_{i,t}(s_k) + \triangle W_{i,n}(s_k) + \triangle W_{i,q}(s_k) \tag{26}$$

$$\triangle W_{i,t}(s_k) = \frac{1}{2}(2\tilde{u}_{i,t}(s_k) + \triangle\tilde{u}_{i,t}(s_k))\tilde{p}_{i,t}(s_k) \tag{27}$$

$$\triangle W_{i,n}(s_k) = \frac{1}{2}(2\tilde{u}_{i,n}(s_k) + \triangle\tilde{u}_{i,n}(s_k))\tilde{p}_{i,n}(s_k) \tag{28}$$

$$\triangle W_{i,q}(s_k) = \frac{1}{2}(2\tilde{u}_{i,q}(s_k) + \triangle\tilde{u}_{i,t}(s_k))\tilde{p}_{i,q}(s_k). \quad \square \tag{29}$$

Lemma 3.2. *During a collision, if the impulse is only applied in the normal direction of the colliding surface, where $\tilde{p}_{i,t}(s_k) = 0$ and $\tilde{p}_{i,r}(s_k) = 0$ in local coordinates, then, the work done by the contact force can be expressed as a function of the change of relative normal velocity:*

$$\triangle W_i(s_k) = \frac{1}{2}(2\tilde{u}_{i,n}(s_k) + \triangle\tilde{u}_{i,n}(s_k))(H_{21}A + H_{22} + H_{23}B)\triangle\tilde{u}_{i,n}(s_k) \tag{30}$$

where,

$$\mathbf{H} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} = \mathbf{L}_i\mathbf{K}_i^{-1}(\mathbf{L}_i^{-1}) \tag{31}$$

$$A = -\frac{H_{23}H_{12} - H_{13}H_{32}}{H_{33}H_{11} - H_{13}H_{31}} \tag{32}$$

$$B = -\frac{H_{31}H_{12} - H_{11}H_{32}}{H_{31}H_{13} - H_{11}H_{33}}. \tag{33}$$

Proof. According to Eq. (18), the work done by the normal contact force is independent of the change of the relative velocity in the tangential direction, and can be expressed as a function of the change of relative velocity in the normal direction:

$$\triangle W_i(s_k) = \frac{1}{2}(2\tilde{u}_{i,n}(s_k) + \triangle\tilde{u}_{i,n}(s_k))\tilde{p}_{i,n}(s_k). \tag{34}$$

To replace $\tilde{p}_{i,n}(s_k)$ by a function of $\triangle\tilde{u}_{i,n}(s_k)$, the relationship between $\tilde{p}_{i,n}(s_k)$ and $\triangle\tilde{u}_{i,n}(s_k)$ is expressed as:

$$\mathbf{p}_i(s_k) = \mathbf{K}_i^{-1}\triangle\mathbf{u}_i(s_k) \tag{35}$$

$$\mathbf{L}^{-1}\tilde{\mathbf{p}}_i(s_k) = \mathbf{K}_i^{-1}\mathbf{L}^{-1}\,\triangle\tilde{u}_i(s_k) \tag{36}$$

$$\tilde{\mathbf{p}}_i(s_k) = \mathbf{L}_i\mathbf{K}^{-1}\mathbf{L}_i^{-1}\,\triangle\tilde{u}_i(s_k) \tag{37}$$

where

$$\tilde{\mathbf{p}}_i(s_k) = \mathbf{H}\,\triangle\tilde{u}_i(s_k) \tag{38}$$

$$\mathbf{H} = \mathbf{L}\mathbf{K}^{-1}(\mathbf{L})^{-1} \tag{39}$$

$$\begin{bmatrix} \tilde{p}_{i,t}(s_k) \\ \tilde{p}_{i,n}(s_k) \\ \tilde{p}_{i,q}(s_k) \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} \triangle\tilde{u}_{i,t}(s_k) \\ \triangle\tilde{u}_{i,n}(s_k) \\ \triangle\tilde{u}_{i,q}(s_k) \end{bmatrix} \tag{40}$$

$$\begin{bmatrix} \tilde{p}_{i,t}(s_k) \\ \tilde{p}_{i,n}(s_k) \\ \tilde{p}_{i,q}(s_k) \end{bmatrix} = \begin{bmatrix} H_{11}\triangle\tilde{u}_{i,t}(s_k) + H_{12}\triangle\tilde{u}_{i,n}(s_k) + H_{13}\triangle\tilde{u}_{i,q}(s_k) \\ H_{21}\triangle\tilde{u}_{i,t}(s_k) + H_{22}\triangle\tilde{u}_{i,n}(s_k) + H_{23}\triangle\tilde{u}_{i,q}(s_k) \\ H_{31}\triangle\tilde{u}_{i,t}(s_k) + H_{32}\triangle\tilde{u}_{i,n}(s_k) + H_{33}\triangle\tilde{u}_{i,q}(s_k) \end{bmatrix} \tag{41}$$

where $\tilde{p}_{i,t}(s_k) = 0$ and $\tilde{p}_{i,q}(s_k) = 0$.

$$\begin{bmatrix} 0 \\ \tilde{p}_{i,n}(s_k) \\ 0 \end{bmatrix} = \begin{bmatrix} H_{11}\triangle\tilde{u}_{i,t}(s_k) + H_{12}\triangle\tilde{u}_{i,n}(s_k) + H_{13}\triangle\tilde{u}_{i,q}(s_k) \\ H_{21}\triangle\tilde{u}_{i,t}(s_k) + H_{22}\triangle\tilde{u}_{i,n}(s_k) + H_{23}\triangle\tilde{u}_{i,q}(s_k) \\ H_{31}\triangle\tilde{u}_{i,t}(s_k) + H_{32}\triangle\tilde{u}_{i,n}(s_k) + H_{33}\triangle\tilde{u}_{i,q}(s_k) \end{bmatrix}. \tag{42}$$

Regarding $\triangle\tilde{u}_{i,n}(s_k)$ as unknown and solving above equation, $\triangle\tilde{u}_{i,t}(s_k)$ and $\triangle\tilde{u}_{i,q}(s_k)$ can be expressed as function of $\triangle\tilde{u}_{i,n}(s_k)$:

$$\triangle\tilde{u}_{i,t}(s_k) = -\frac{H_{33}H_{12} - H_{13}H_{32}}{H_{33}H_{11} - H_{13}H_{31}}\triangle\tilde{u}_{i,n}(s_k) = A\triangle\tilde{u}_{i,n}(s_k) \tag{43}$$

$$\triangle\tilde{u}_{i,q}(s_k) = -\frac{H_{31}H_{12} - H_{11}H_{32}}{H_{31}H_{13} - H_{11}H_{33}}\triangle\tilde{u}_{i,n}(s_k) = B\triangle\tilde{u}_{i,n}(s_k). \tag{44}$$

Then $\tilde{p}_{i,n}(s_k)$ is expressed as a function of $\triangle\tilde{u}_{i,n}(s_k)$,

$$\tilde{p}_{i,n}(s_k) = (H_{21}A + H_{22} + H_{23}B)\triangle\tilde{u}_{i,n}(s_k). \tag{45}$$

Substituting the previous equation into Eq. (18) yields:

$$\triangle W_i(s_k) = \frac{1}{2}(2\tilde{u}_{i,n}(s_k) + \triangle\tilde{u}_{i,n}(s_k))(H_{21}A + H_{22} + H_{23}B)\triangle\tilde{u}_{i,n}(s_k). \quad \Box \tag{46}$$

By manipulating Eq. (46), the change of the relative normal velocity is defined as follows, if the change of energy is known:

$$\triangle\tilde{u}_{i,n}(s_k) = -\tilde{u}_{i,n}(s_k) \pm \sqrt{C + \tilde{u}_{i,n}^2(s_k)} \tag{47}$$

$$C = \frac{2\triangle W_i}{H_{21}A + H_{22} + H_{23}B}. \tag{48}$$

Since the relative normal velocity increases during simulation, the positive solution is chosen:

$$\begin{aligned} \triangle u_{i,n}^{\max}(s_k) &= -\tilde{u}_{i,n}(s_k) + \sqrt{C + \tilde{u}_{i,n}^2(s_k)} \\ C &= \frac{2\triangle W_i}{H_{21}A + H_{22} + H_{23}B}. \end{aligned} \tag{49}$$

4. Impulse with friction

In SQM and SMM, the impulse is defined by Eq. (8), once the change of relative velocity is determined by Newton's impact law. In ETM, the change of relative velocity is first computed in the normal direction, after which friction is considered. A static friction condition is defined, assuming that there is no tangential velocity after collision, such that the change of relative normal velocity is given by

$$\tilde{\mathbf{u}}_i(s_{k+1}) = \begin{bmatrix} 0 & \tilde{u}_{i,n}(s_k) + \triangle\tilde{u}_{i,n}(s_k) & 0 \end{bmatrix}^{\mathrm{T}}. \tag{50}$$

The impulse in this case is expressed as $\mathbf{p}_i(s_k) = (\mathbf{K}_i)^{-1}\triangle\mathbf{u}_i(s_k)$ (Eq. (8)). Decomposing this impulse into its normal and tangential directions,

$$\mathbf{p}_{i,n}(s_k) = \mathbf{n}_i^{\mathrm{T}}\mathbf{p}_i(s_k)\mathbf{n}_i \tag{51}$$

$$\mathbf{p}_{i,q}(s_k) = \mathbf{p}_i(s_k) - \mathbf{p}_{i,n}(s_k) \tag{52}$$

where the static friction condition is defined as

$$|\mathbf{p}_{i,q}(s_k)| \leq \mu |\mathbf{p}_{i,n}(s_k)|. \tag{53}$$

If the static friction condition is satisfied, the above impulse is expressed by Eq. (8). Otherwise, the impulse is recomputed as follows, to ensure that the change of relative normal velocity is $\triangle\tilde{u}_{i,n}(s_k)$ after applying the impulses. The impulse is expressed as

$$\mathbf{p}_i(s_k) = \tilde{p}_{i,n}(s_k)\mathbf{n}_i + \mu\tilde{p}_{i,n}(s_k)\mathbf{q}_i \tag{54}$$

where

$$\tilde{p}_{i,n}(s_k) = \frac{\triangle\tilde{u}_{i,n}(s_k)}{\mathbf{n}_i^{\mathrm{T}}\mathbf{K}_i(\mathbf{n}_i + \mu\mathbf{q}_i)}. \tag{55}$$

Proof. The tangential impulse is derived as follows.

$$\mathbf{p}_{i,q}(s_k) = \mu\tilde{p}_{i,n}(s_k)\mathbf{q}_i. \tag{56}$$

So

$$\mathbf{p}_i(s_k) = \mathbf{p}_{i,n}(s_k) + \mathbf{p}_{i,q}(s_k) = \mathbf{p}_{i,n}(s_k) + \mu\tilde{p}_{i,n}(s_k)\mathbf{q}_i \tag{57}$$

$$\mathbf{K}_i\mathbf{p}_i(s_k) = \triangle\mathbf{u}_i(s_k) \tag{58}$$

$$\mathbf{n}_i^{\mathrm{T}}\mathbf{K}_i\mathbf{p}_i(s_k) = \mathbf{n}_i^{\mathrm{T}}\triangle\mathbf{u}_i(s_k) \tag{59}$$

$$\mathbf{n}_i^{\mathrm{T}}\mathbf{K}_i(\mathbf{p}_{i,n}(s_k) + \mathbf{q}_i\mu\tilde{p}_{i,n}(s_k)) = \triangle\tilde{u}_{i,n}(s_k) \tag{60}$$

$$\tilde{p}_{i,n}(s_k) = \frac{\triangle\tilde{u}_{i,n}(s_k)}{\mathbf{n}_i^{\mathrm{T}}\mathbf{K}_i(\mathbf{n}_i + \mu\mathbf{q}_i)}. \quad \square \tag{61}$$

5. Energy tracking: single collision

Once collisions are identified, a series of impulses are applied to avoid penetration. Unlike SQM and SMM, ETM does not invert the relative normal velocity, $\tilde{u}_{i,n}(s_0)$, directly, $\tilde{u}_{i,n}(s_1) = -e_n\tilde{u}_{i,n}(s_0)$. Instead, it updates the relative normal velocity gradually over a set of iterations, $\{s_0, s_1, s_2, \ldots, s_\lambda\}$, by applying a set of impulses, $\{\mathbf{p}_i(s_0), \mathbf{p}_i(s_1), \mathbf{p}_i(s_2), \ldots\}$, sequentially. This is advantageous, as it allows tracking of the change of relative normal velocity and elastic energy at each contact point, and the fluctuation of kinetic energy of system during a single collision (Fig. 4). The collision response is divided into three stages: compression, in which $\tilde{u}_{i,n}(s_k)$ is negative; restitution, stage in which the estimated energy loss can be accounted for; and separation, in which the relative normal velocities are positive. The initial change of relative normal velocity, $\triangle\dot{u}_n$, is defined as

$$\triangle\dot{u}_n = \frac{\tilde{u}_{i,n}(s_0)}{\lambda} \tag{62}$$

where $\tilde{u}_{i,n}(s_0)$ is the initial relative normal velocity, and λ is a positive integer which controls the rate of change of the relative normal velocity during collision analysis. The parameter λ counts the number of iterations for impulse computation at each collision time step. Without loss of generality, the change of relative normal velocity is initially chosen as $\triangle\dot{u}_n$ for each iteration, and is recomputed at different stages to ensure energy conservation (Algorithm 1).

5.1. First stage

The compression of two colliding bodies is modeled while taking friction into account. The assumption of the change of relative normal velocity from iteration s_k to iteration s_{k+1} is expressed as

$$\triangle\tilde{u}_{i,n}(s_k) = \begin{cases} \triangle\dot{u}_n & \text{if } \tilde{u}_{i,n}(s_k) + \triangle\dot{u}_n < 0 \\ |\tilde{u}_{i,n}(s_k)| & \text{if } \tilde{u}_{i,n}(s_k) + \triangle\dot{u}_n > 0. \end{cases} \tag{63}$$

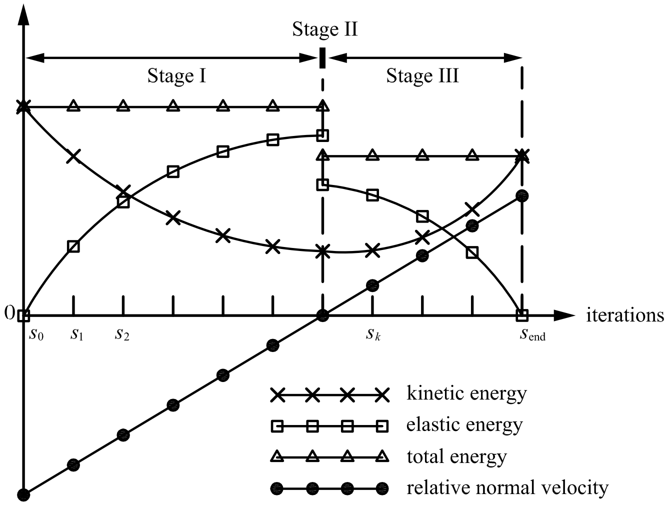


Fig. 4. The change of energy and relative normal velocity during collision resolution for a single collision. During compression (Stage I), the relative normal velocity increases from negative to zero, elastic energy at the contact point increases, and dynamic energy decreases. During separation (Stage III), the relative normal velocity and dynamic energy increase until all elastic energy is released. The total energy is conserved. In this example, the collision is assumed to be perfectly elastic.

Algorithm 1 Collision response for a single collision

```
Require: Body $b_N$ and body $b_M$ collide at contact point $c_i$
initialize the iteration number $k = 0$
initialize $W_{i,n}(s_0) = 0$
calculate the initial change of relative normal velocity, $\triangle\dot{u}_n$ (Eq. (62))
compute relative normal velocity before collision,
while $\tilde{u}_{i,n}(s_k) < 0$ do
    compute relative normal velocity, $\tilde{u}_{i,n}(s_k)$
    compute $\triangle\tilde{u}_{i,n}(s_k)$ (Eq. (63))
    compute impulse, $\mathbf{p}_i(s_k)$ (Eqs. (50)–(55)) considering friction
    calculate work done by impulse (Eq. (16)) and update the energy at the contact point by $W_{i,n}(s_{k+1}) = W_{i,n}(s_k) + \triangle W_{i,n}(s_k)$
    update the linear and angular velocities of $b_N$ and $b_M$ by applying impulses
    k=k+1
end while
compute the energy dissipation using Stronge's hypothesis: $W_{i,n}(s_{k+1}) = e_n^2 W_{i,n}(s_k)$
k=k+1
while $W_i(s_k) > 0$ do
    compute the maximum change of relative normal velocity, $\triangle u_{i,n}^{max}(s_k)$, from the residual energy at the contact point (Eq. (49))
    compute $\triangle\tilde{u}_{i,n}(s_k)$ (Eq. (66))
    compute impulse, $\mathbf{p}_i(s_k)$ (Eqs. (50)–(55)) considering friction
    compute work done by impulse, $W_i$ (Eq. (16)) and update the energy at the contact points: $W_{i,n}(s_{k+1}) = W_{i,n}(s_k) + \triangle W_{i,n}(s_k)$
    update the linear velocity and angular velocity of colliding bodies $b_N$ and $b_M$
    k=k+1
end while
```

The change of relative normal velocity is defined as $\triangle\dot{u}_n$, unless the relative normal velocity becomes positive after applying the impulse. In this case, the change of relative normal velocity is recomputed to ensure that it reaches zero after applying the impulse for a single collision.

Once the assumption for the change of the relative normal velocity is established, the impulse is obtained from Eqs. (50)–(55). Work done by the impulse in the normal direction, $\triangle W_{i,n}(s_k)$, is expressed by Eq. (16). The elastic energy at the contact point before the collision is set to zero: $W_i(s_0) = 0$. At each step, the energy due to the normal contact force at contact point c_i is updated, according to

$$W_{i,n}(s_{k+1}) = W_{i,n}(s_k) + \triangle W_{i,n}(s_k). \tag{64}$$

5.2. Second stage

At the end of the first stage, the work done by the normal impulse reaches the maximum value, and the sign of the relative normal velocity changes to positive. Energy dissipation in the normal direction is expressed by Stronge's hypothesis [49],

$$W_i(s_{k+1}) = e_n^2 W_i(s_k). \tag{65}$$

5.3. Third stage

Separation of two colliding bodies is modeled taking friction into account. The change of relative normal velocity from iteration s_k to iteration s_{k+1} is expressed as

$$\triangle \tilde{u}_{i,n}(s_k) = \begin{cases} \triangle \dot{u}_i & \text{if } \triangle \dot{u}_n \leq \triangle u_{i,n}^{\max}(s_k) \\ \triangle u_{i,n}^{\max}(s_k) & \text{if } \triangle \dot{u}_n > \triangle u_{i,n}^{\max}(s_k). \end{cases} \tag{66}$$

The change of relative normal velocity is defined as $\triangle \dot{u}_n$. However, there is a limitation to the change of relative normal velocity: its maximum change is capped by the maximum work which can be done by the impulse, equal to the elastic energy absorbed at the contact point. Therefore, there is a maximum change of relative normal velocity, $\triangle u_{i,n}^{\max}(s_k)$, which releases the residual energy at the contact point. This maximum change of the relative normal velocity is expressed by Eq. (49). In Eq. (47) there are two solutions for the change of relative normal velocity, the positive sign is always chosen, as the relative velocity is assumed to increase during this stage. Once the assumption of the change of relative normal velocity is established, the impulse is computed using Eqs. (50)–(55). The work done by the impulse, $\triangle W_i(s_k)$, is expressed by Eq. (16). As in Stage I, the energy at each contact point c_i is updated at each step according to Eq. (64). The relative normal velocity and energy are updated gradually over iterations. The iterating stops when $W_i(s_k) \leq 0$ (see Algorithm 1).

6. Energy tracking: multiple collisions

Fig. 5 illustrates multiple collisions, $\mathbf{c} = \{c_0, c_1, c_2, \ldots, c_n\}$, between multiple bodies, $\mathbf{b} = \{b_0, b_1, b_2, \ldots, b_m\}$, detected at a time step t_c, where m is the total colliding bodies and n is the total contact points. At each iteration, multiple collisions are treated as concurrent collisions and impulses are computed and applied to contact points sequentially. The interaction of multiple collisions is simulated by repeating the computation until all relative normal velocities are non-negative (Algorithm 2). The computation for multiple collisions consists of three main stages.

6.1. First stage

Impulses are applied to eliminate all negative relative normal velocities among contact points. At each iteration s_k, the contact point with the smallest relative normal velocity, the minimum contact point, $c_{\min}$, is identified. If the minimum contact point is colliding (its relative normal velocity is negative), a pair of impulses, obtained using Eqs. (50)–(55) considering friction, is applied to $c_{\min}$ to resolve the collision. The change of relative normal velocity is given by Eq. (63). The work done by normal impulses is computed using Eq. (16), and the energy at the contact point is updated. The linear and angular velocities of the colliding bodies, and the relative velocities of all related contact points, are also updated. The procedure is repeated until $c_{\min}$ is not colliding. At the end of this stage, all relative normal velocities will be non-negative, and the total energy among collisions reaches its maximum value.

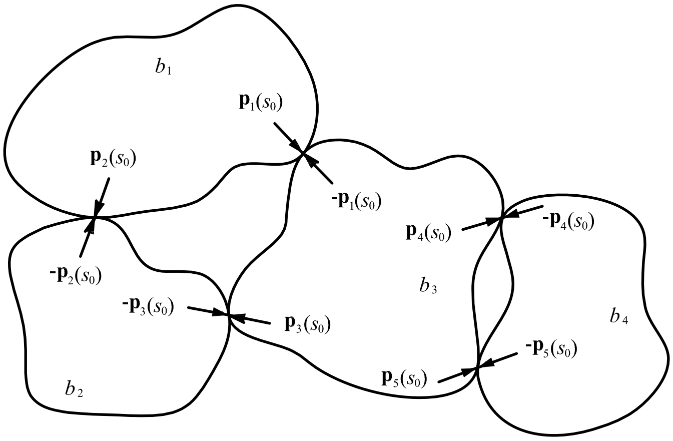


Fig. 5. Collision of multiple bodies: multiple bodies, $\mathbf{b} = \{b_0, b_1, b_2, \ldots, b_m\}$, collide with multiple contact points $\mathbf{c} = \{c_0, c_1, c_2, \ldots, c_n\}$, m is the total number of colliding bodies and n is the total number of contact points. A set of impulses, $(\mathbf{p}_0(s_k), \mathbf{p}_1(s_k), \mathbf{p}_2(s_k), \ldots \mathbf{p}_n(s_k))$, is applied to contact points at iteration s_k.

Algorithm 2 Collision response for multiple collisions

Require: A set of bodies $\{b_0, b_1, ...b_n\}$ collide at contact points $c_i \in \{c_0, c_1, ...c_m\}$

```
initialize $W_i = 0$ for each contact point $i$
identify the contact point with smallest relative normal velocity, $c_{min}$
and its relative normal velocity $\tilde{u}_{min,n}$
while $\tilde{u}_{min,n} < 0$ do
    while $\tilde{u}_{min,n} < 0$ do
        compute $\triangle\tilde{u}_{i,n}(s_k)$ (Eq. (63))
        compute impulses (Eqs. (50)–(55))
        calculate work done by impulse (Eq. (16)) and update the energy at the contact point
        update the linear and angular velocities of the colliding bodies
        recompute the minimum contact point, $c_{min}$ and its relative normal velocity $\tilde{u}_{min,n}$
    end while
    apply Stronge's hypothesis to all the contact points
    search contact point with maximum elastic energy, max$\{W_i\}$
    while max$\{W_i\} > 0$ do
        for all $c_i$ do
            compute $\triangle\tilde{u}_{i,n}(s_k)$ (Eq. (66))
            compute impulses (Eq. (50)–(55))
            calculate work done by impulse (Eq. (16)) and update the energy at the contact point
            update the linear and angular velocities of the colliding bodies
        end for
        recompute contact point with maximum elastic energy, max$\{W_i\}$
    end while
    recompute the minimum contact point, $c_{min}$ and its relative normal velocity $\tilde{u}_{min,n}$
end while
```

6.2. Second stage

Energy dissipation in the normal direction is applied sequentially over all collisions, applying Stronge's hypothesis (Eq. (65)).

6.3. Third stage

Impulses are applied to progressively release energy at all collision points. At each iteration s_k, contact points, **c**, are sequentially evaluated. If the elastic $W_{i,n}(s_k)$ is positive, a pair of impulses, $\mathbf{p}_i(s_k)$ and $-\mathbf{p}_i(s_k)$ is applied to contact point c_i to release energy. This computation is repeated until the energy at all contact points is zero, i.e. $\max\{W_i\} = 0$. For each iteration s_k and each collision c_i, impulses are expressed by Eqs. (50)–(55). The change of relative velocity is given by Eq. (66). The work done by normal impulses is computed using Eq. (16). Subsequently, the energy of the collision, and the velocities of the colliding bodies related to contact point c_i, are updated. The three stages described above are repeated until no negative relative normal velocity remains. As in SQM, ETM cannot ensure termination after a certain number of iterations for perfectly elastic collisions. For oscillating cases, termination is enforced after the three stages have been repeated a finite amount of times, e.g. 300, after which the normal restitution is set to zero. Once the restitution is zero, the computation is ensured to terminate after a few iterations.

7. Numerical tests

ETM is investigated using selected numerical tests. The collision of two tetrahedra validates the performance of ETM for single-point collisions. A series of multi-contact scenarios are selected to exhibit ETM advantages as compared to SQM and SMM. In some cases gravity is disregarded, certain examples are assumed perfectly elastic to evaluate possible artificial energy losses and gains, and in all the cases time step size is assumed to be 1/600 seconds. Energy conservation and repose angles are evaluated for a series of packing and repositioning numerical tests involving spheres, cubes, and non-convex cross objects.

In the following tests, the total system energy, E_t, is expressed as

$$E_t = \sum_i^n \left(E_i^p + E_i^k\right) = \sum_i^N \left(m^i g h_i + \int_{V_i} \frac{\mathbf{u}^2(\mathbf{x})}{2} dv\right) \tag{67}$$

where n is the number of bodies, E_i^p and E_i^k are the potential energy and kinetic energy of body i, m^i is the mass of body i, g is gravity acceleration, h_i is the height of the center of gravity of body i, V_i is the volume, and $\mathbf{u}(\mathbf{x})$ is the linear velocity at integration point $\mathbf{x}$. The relative error of E_t is measured as follows:

$$\xi_{E_t} = \left|\frac{E_t^b - E_t^f}{E_t^b}\right| \tag{68}$$

where E_t^b and E_t^f are the energies before and after collision.

7.1. Single-point collision

The energy tracking method is investigated using a single-point collision of two tetrahedra, a and b (Fig. 6(a)). Initial linear and angular velocities of body a are $\boldsymbol{v}_a = (0, 0, -100)$ m/s and $\omega_a = (5, 10, 3)$ rad/s, respectively. Body b is initially static and only moves after the collision. No external force acts on these bodies. In this test, the collision is assumed to be perfectly elastic and frictionless, i.e. $e_n = 1$ and $\mu = 0$. The experiment constitutes a simple closed system, which is expected to be energy conservative.

The progression of the relative normal velocity, elastic and kinetic energy at the contact point, and the total system energy are plotted in Fig. 6(b) for $\lambda = 200$. During the simulation of collision, the total energy of the system remains constant. During the first stage, the relative normal velocity changes from −211.84 to 0 m/s, the elastic energy at the contact point increases from 0 to 60 124 J, and the kinetic energy decreases from 83 809.2 to 23 685.2 J. During the third stage, the relative normal velocity changes from 0 to 211.84 m/s, the elastic energy at the contact point decreases from 60 124 to 0 J, and the kinetic energy decreases from 23 685.2 to 83 809.2 J. Only a single contact point is taken into account here. In Fig. 6(b), energies are divided by 300 to aid visualization.

The influence of λ is investigated by measuring the relative error of total energy. A variation of the error is measured for increasing λ values of {1, 5, 10, 100, 200, 500}, yielding errors,

$$\xi_{E_t} = \{1.73 \times 10^{-16}, 6.94 \times 10^{-16}, 1.21 \times 10^{-15}, 1.73 \times 10^{-16}, 1.51 \times 10^{-14}, 1.42 \times 10^{-14}\} \tag{69}$$

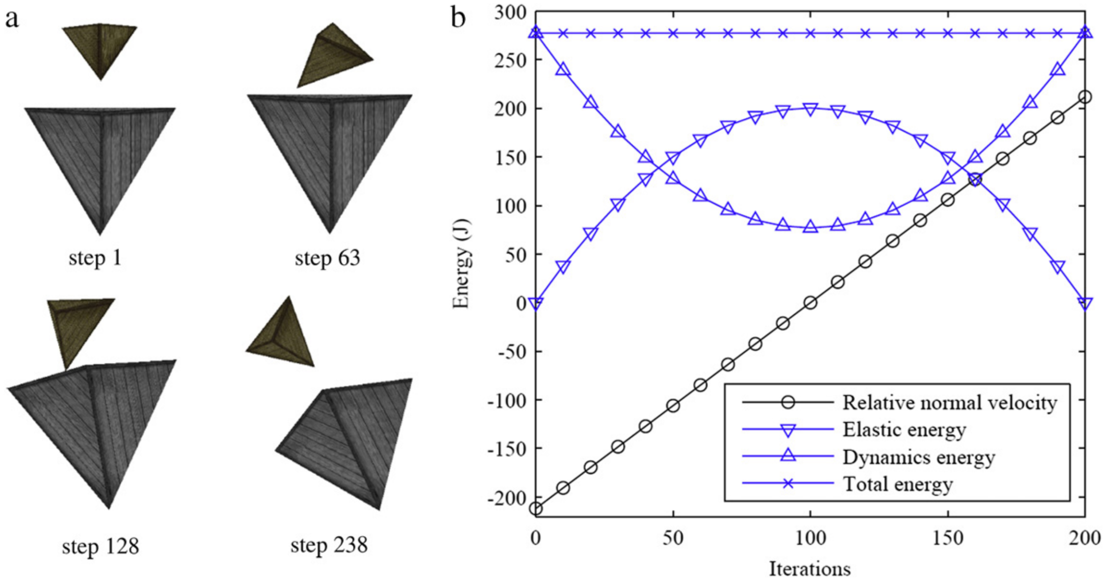


Fig. 6. Single point collision. (a) Collision of two tetrahedra. (b) Progression of relative normal velocity, and the elastic, kinetic and total energy of a system of two colliding tetrahedra. All but the relative normal velocity are scaled by 1/300 to facilitate plotting.

that are initially close to the numerical epsilon, but increase by two orders of magnitude for $\lambda = 500$. The consistently low errors indicate that the single collision can be resolved exactly. The increase of this error, with increasing λ, reflects the accumulation of the numerical error due to the increased number of iterations and operations. This numerical error is also observed to be low.

7.2. Multiple point collision: ETM vs. SMM

The simultaneous impulse method (SMM) inserts all collisions into a system of linear equations, so that constraints on the relative normal velocities, $\tilde{u}_{i,n}(s_1) = -e_n\tilde{u}_{i,n}(s_0)$, are enforced simultaneously. $\tilde{u}_{i,n}(s_0)$ and $\tilde{u}_{i,n}(s_1)$ are the relative normal velocities before and after collision, respectively. In some cases, this empirical velocity constraint does not match real physical behavior. The example investigated in this section is one of them.

Consider the collision of two box-shaped bodies, where one is static and the other dynamic, and gravity is ignored. The initial linear velocity of the cube is (0, 0, −10) m/s and its angular velocity is zero. The moving cube is expected to rotate after the collision, as all of its contact points are located on one half of the contact surface, and are repelled upwards after the collision, generating a positive rotational moment that throws the body out of balance. For SMM, since relative normal velocities at all contact points are the same before the collision, $\tilde{u}_{i,n}(s_0) = -10$ m/s, the relative normal velocities at all contact points are the same after the collision, $\tilde{u}_{i,n}(s_0) = 10$ m/s. Therefore, SMM does not model the rotation which is a function of the propagation of the impulses during collision (Fig. 7(a)). Angular velocities in the x, y and z directions of the SQM and ETM are plotted in Fig. 7(b). SMM fails to predict post-collision non-zero angular velocities. In contrast, ETM predicts angular velocities that agree with the expected rotation.

7.3. Multiple point collision: ETM vs. SQM

The collision of a 2 × 2 × 1 m box-shaped body and the ground is now investigated. Gravity is ignored, and the collision is assumed to be perfectly elastic. The initial linear and angular velocities are (1, 1, −10) m/s and zero, respectively. Since there is no initial angular velocity of the cube before collision, and the ground surface is perfectly horizontal, the linear velocity after the collision should be exactly inverted, and the angular velocity should remain zero.

In this test, the relative error of the velocity is used to track the effect of λ on the quality of the simulation:

$$\xi_{\mathbf{v}} = \frac{|\mathbf{v}_f - \mathbf{v}_o|}{|\mathbf{v}_o|} \tag{70}$$

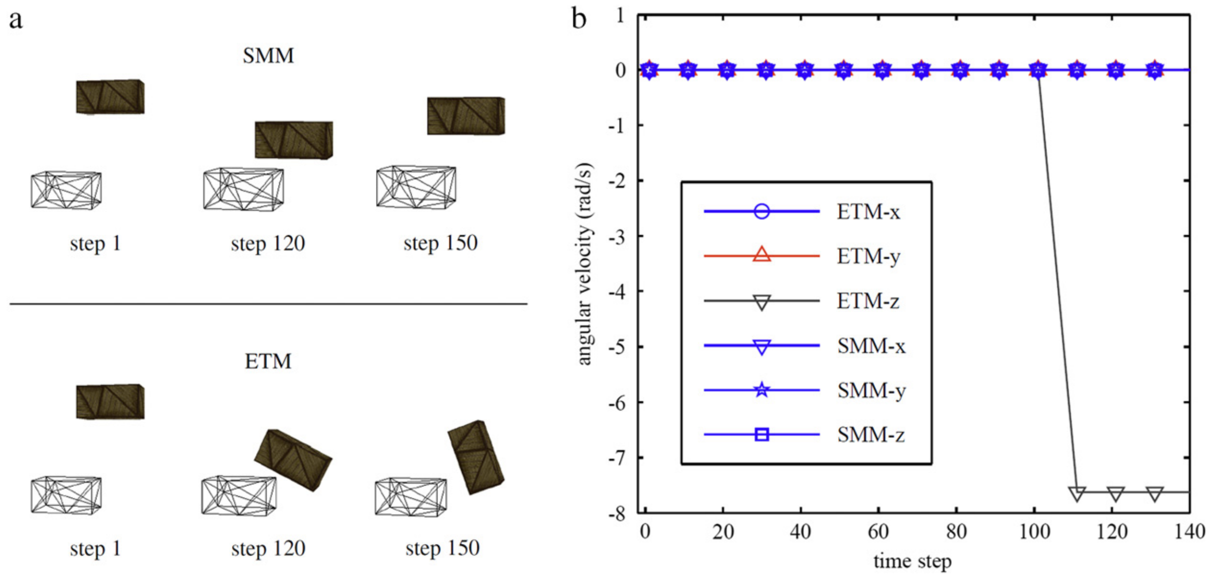


Fig. 7. Multiple-point collision: ETM vs. SMM. (a) Collision of a moving body against a static body with multiple contact points. (b) Angular velocities.

where $\mathbf{v}_o$ is the initial linear or angular velocity of the object at the beginning of simulation, and $\mathbf{v}_f$ are the final linear or angular velocity of the object after collision resolution.

SQM treats multiple collisions as concurrent collisions, and simply inverts the relative normal velocity after applying the impulse. Thus, impulses are applied to collisions one by one, and not simultaneously. In contrast, ETM applies the impulses to multiple collisions simultaneously, but does so by gradually increasing the relative normal velocity during the collision. In this numerical experiment, multiple collisions occur simultaneously between the box-shaped object and the ground, exposing the difficulty of SQM in treating impulse propagation during collisions, as compared to ETM.

The object collides with the ground at time step 166 (Fig. 8(a)). The relative error of the linear velocities remains close to zero, and does not vary with λ for either method. The relative error of the angular velocity is lower for ETM as compared to SQM from the onset, and further decreases as λ increases (Fig. 8(b)). For higher values of λ, i.e. $\lambda > 50$, the accumulated numerical error becomes measurable and the relative velocity error does not further reduce. In the simulations herein, $\lambda = 30$.

7.4. Multiple body collision: energy conservation

The energy conservative property of multiple bodies is now investigated. A necessary condition for the impulse-based method to be energy-conservative has been previously identified [9]. Namely, and, in the context of an SQM formulation, the relative contact velocity must be exactly reversed, as opposed to only inverting it in the normal direction. In contrast, ETM ensures conservation by constraining iterations based on the tracking of energy during collision resolution. This property is illustrated by quantifying artificial energy loss and gain during collision resolution, in terms of the measurement of the relative error in the computation of energy conservation of the collision. To this end, two perfectly elastic cases are examined, in which 540 and 1080 spherical bodies fall into a static box. Relative energy errors for at each collision step, over 250 time steps, are plotted in Fig. 9. Overall, relative errors for both models are consistently below 8×10^{-3} for both SQM and ETM.

7.5. Multiple body collision: packing and repositioning

The experiment models the interaction between multiple bodies of different shapes, including cubes, spheres, and concave units, discretized using a tetrahedral mesh (Fig. 10). Cubes and spheres are typical convex proxies for rock fragments, while specialized three-dimensional concrete crosses are widely used in coastal structure engineering to

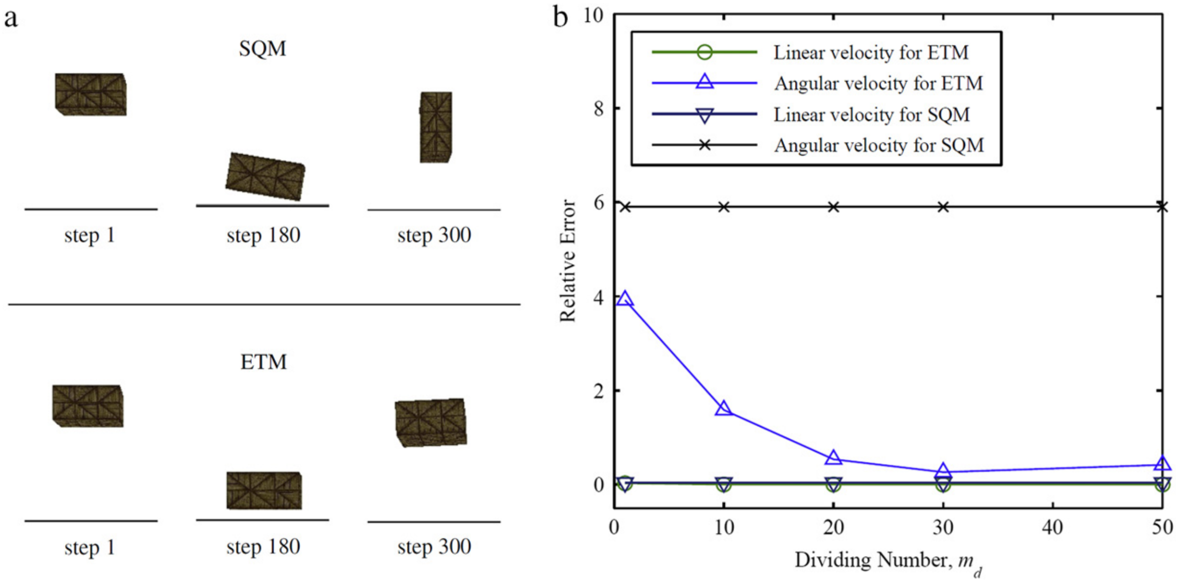


Fig. 8. Multiple-point collision: ETM vs. SQM. (a) A box-shaped object collides with the ground at multiple contact points. (b) The relative error of velocities of the two approaches. The relative error of the linear velocities is low. A higher relative error is measured for the angular velocity of both ETM and SQM. In the case of ETM, this error decreases with the increase of the dividing number, λ.

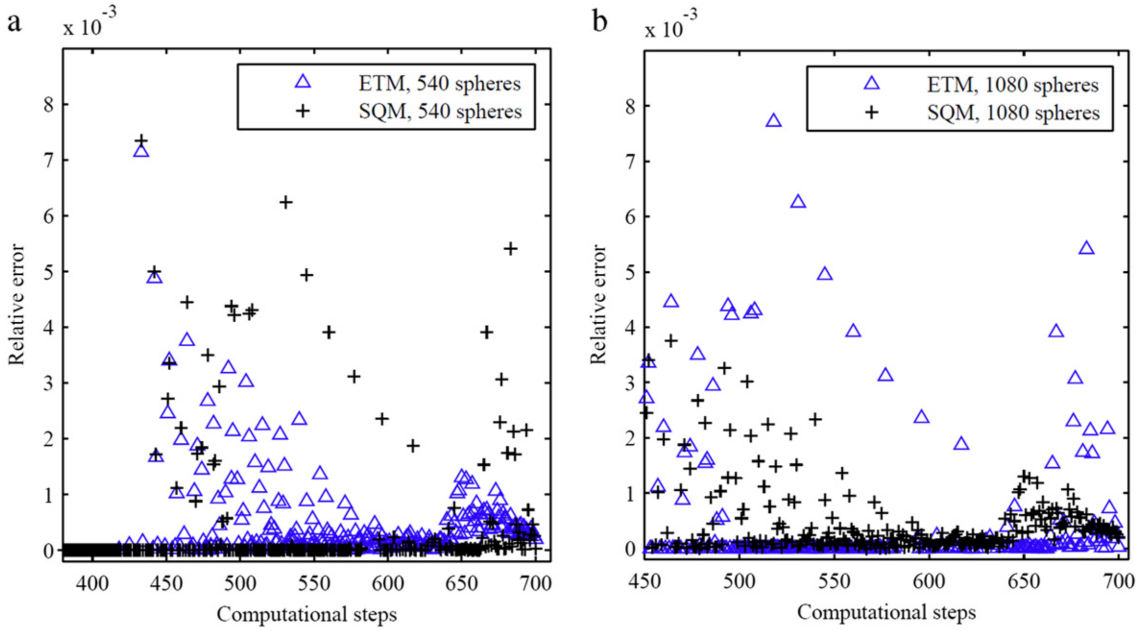


Fig. 9. Relative error of energy during the packing of (a) 540 and (b) 1080 bodies.

protect against wave action for hydraulic stability. Gravitational acceleration is assumed to be $-9.8\ \text{m/s}^2$. Spheres, cubes, and concave units are initially set up in an array randomly oriented about their centers. These bodies undergo gravity-driven fall and settle into a box with static boundary walls (Fig. 11).

The ETM is used to investigate the repose angle of the slope generated by removing the right boundary of the previously generated stacks (Fig. 12). Snapshots of the final repose angles for different object shapes and for different frictions are shown in Fig. 13.

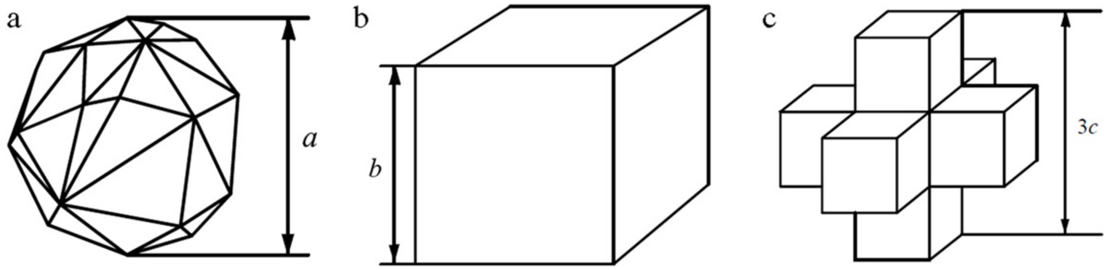
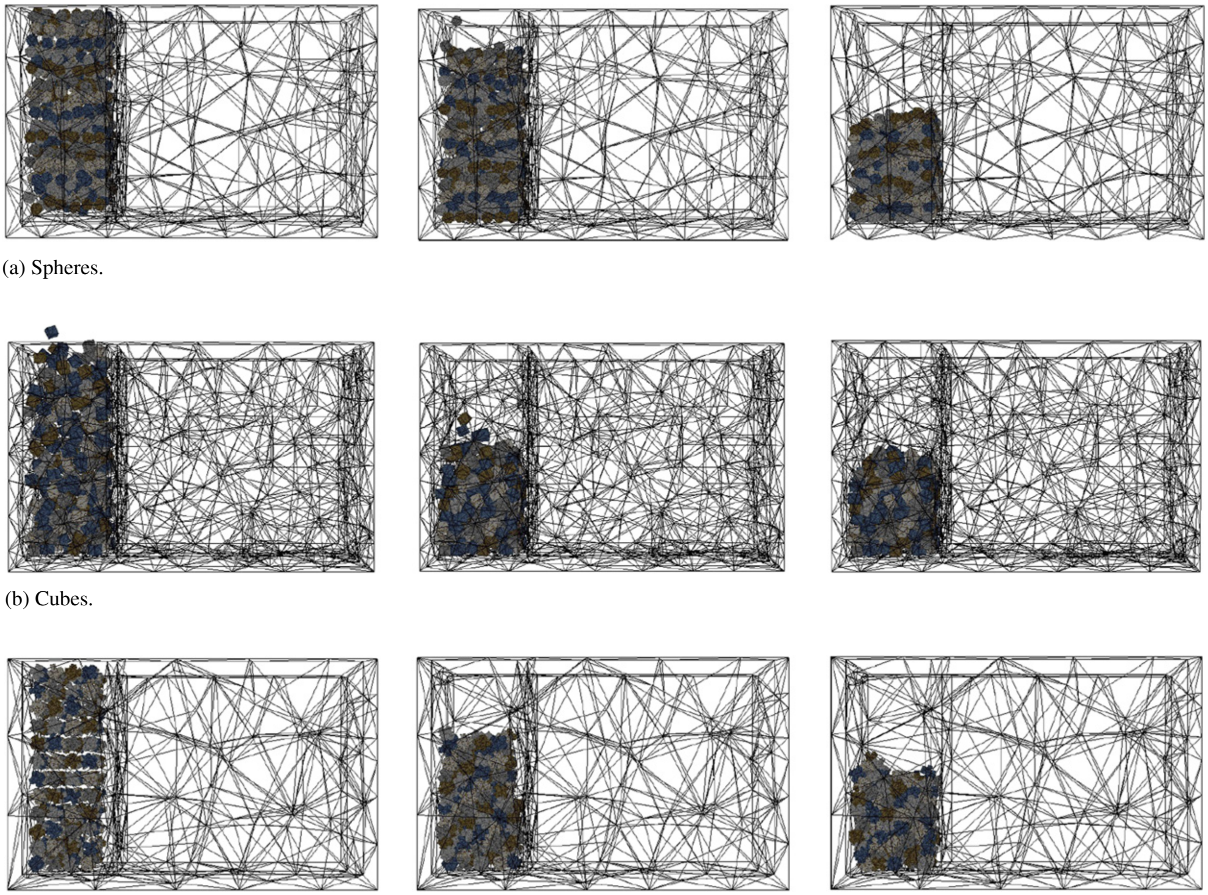


Fig. 10. Geometry of colliding bodies: (a) sphere, where $a = 3$ m, (b) cube, where $b = 2.4$ m, and (c) cross, where $3c = 3.78$ m.



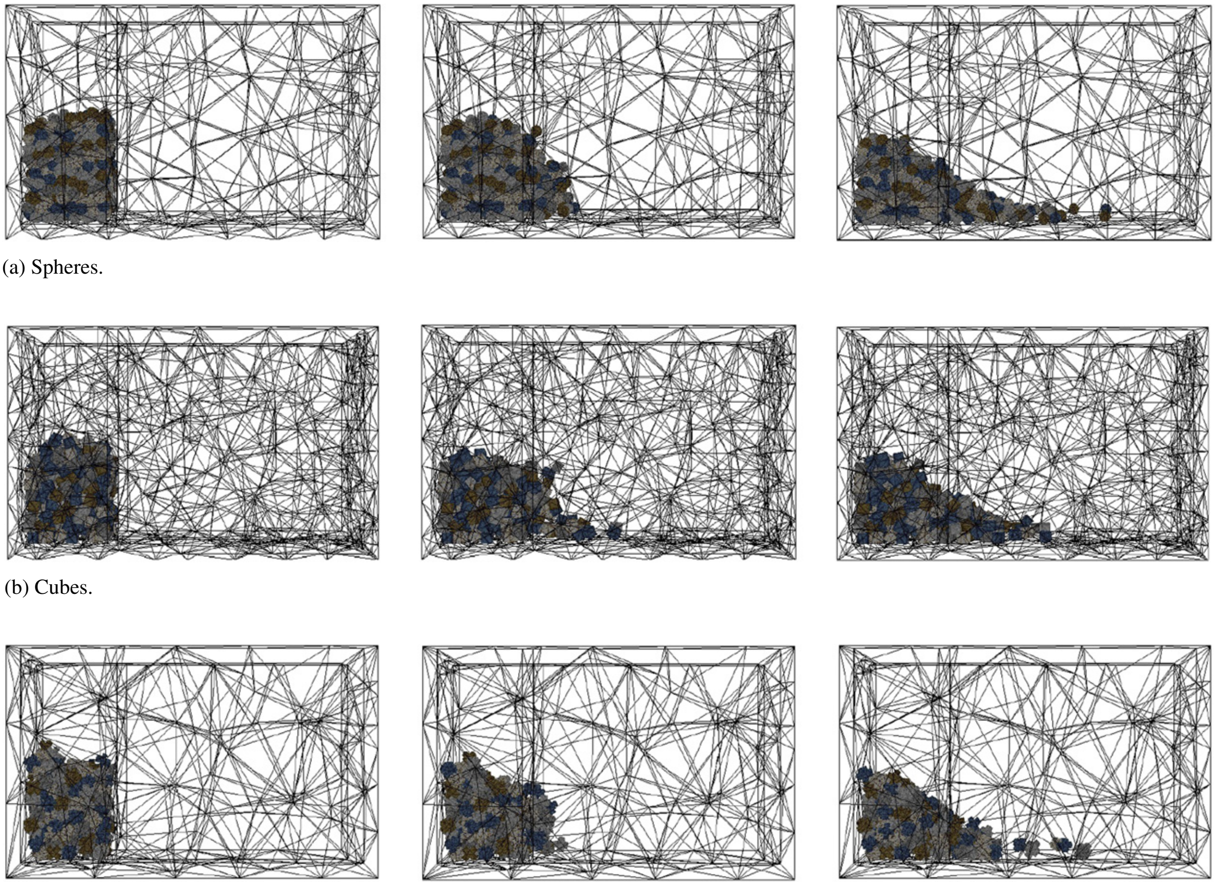
(a) Spheres.

(b) Cubes.

(c) Cross units.

Fig. 11. Packing of multiple bodies with different shapes, $\mu = 0.3$ and $e_n = 0.3$. In (a), 540 spheres are packed. In (b) and (c), 325 cubes and non-convex cross units are packed. The box is subdivided by a wall, which is later removed to model repositioning. Objects are colored randomly, and the container mesh is displayed using a wireframe.

To measure the angle of repose, the floor of the container is subdivided using an 8 × 16 grid, where the longest side of the container is subdivided sixteen times. Eight lines are fitted, using least squares, through the highest nodes identified in each of the sixteen cells. The slopes of these are regarded as the repose angles in these eight zones (Fig. 14). The average repose angle corresponds to the average measured slope. The absolute error is the average difference of the average repose angle as compared to the maximum and minimum measured repose angles. For the spheres, the absolute error ranges from 0.0255° to 0.8°, for cubes from 1.29° to 3.17°, and for crosses from 1.2° up



(a) Spheres.

(b) Cubes.

(c) Cross units.

Fig. 12. Settling of multiple bodies with different shapes, $\mu = 0.64$ and $e_n = 0.64$.

to 12.89°. The angle of repose and respective measurement error is plotted as a function of the micro-scale friction coefficient in Fig. 15.

For comparison, angles of repose obtained experimentally and numerically for spheres [51] are plotted along the numerical results. Results are in good agreement with the laboratory experiments which have shown that the angle of repose increases non-linearly with the friction coefficient [48]. The discrete element method (DEM) has been used to investigate the numerical methodologies to design concrete armor units [52,53]. A numerically generated repose angle for the same cross-shaped particle, computed using 3D DEM [53], is also plotted for comparison. As expected, crosses exhibit more interlocking than spheres and cubes and yield, almost consistently, higher repose angles when the friction coefficient is non-zero. When the colliding bodies are frictionless, the bodies avalanche without generating a slope (Fig. 13). For the frictionless computation, the measured angle of repose mainly depends on the packing proper of bodies. For all shapes, repose angles are close to, but greater than, those measured for dry sand. Results obtained by ETM are comparable to those reported in [53].

8. Concluding remarks

A novel energy tracking method (ETM) has been proposed to solve collision responses in a multi-body system. The ETM belongs to the family of impulse methods, which apply impulses to avoid penetration. In previous versions of the impulse method, *i.e.* SMM and SQM, the relative velocity at the contact point after collision is directly derived from the relative velocity before collision—in a purely simultaneous or sequential manner, and the impulse is a function of the change of the relative normal velocity. In this generalized form of the impulse method, the relative velocity is adjusted gradually over a set of iterations, while within each of these iterations, their effects are considered

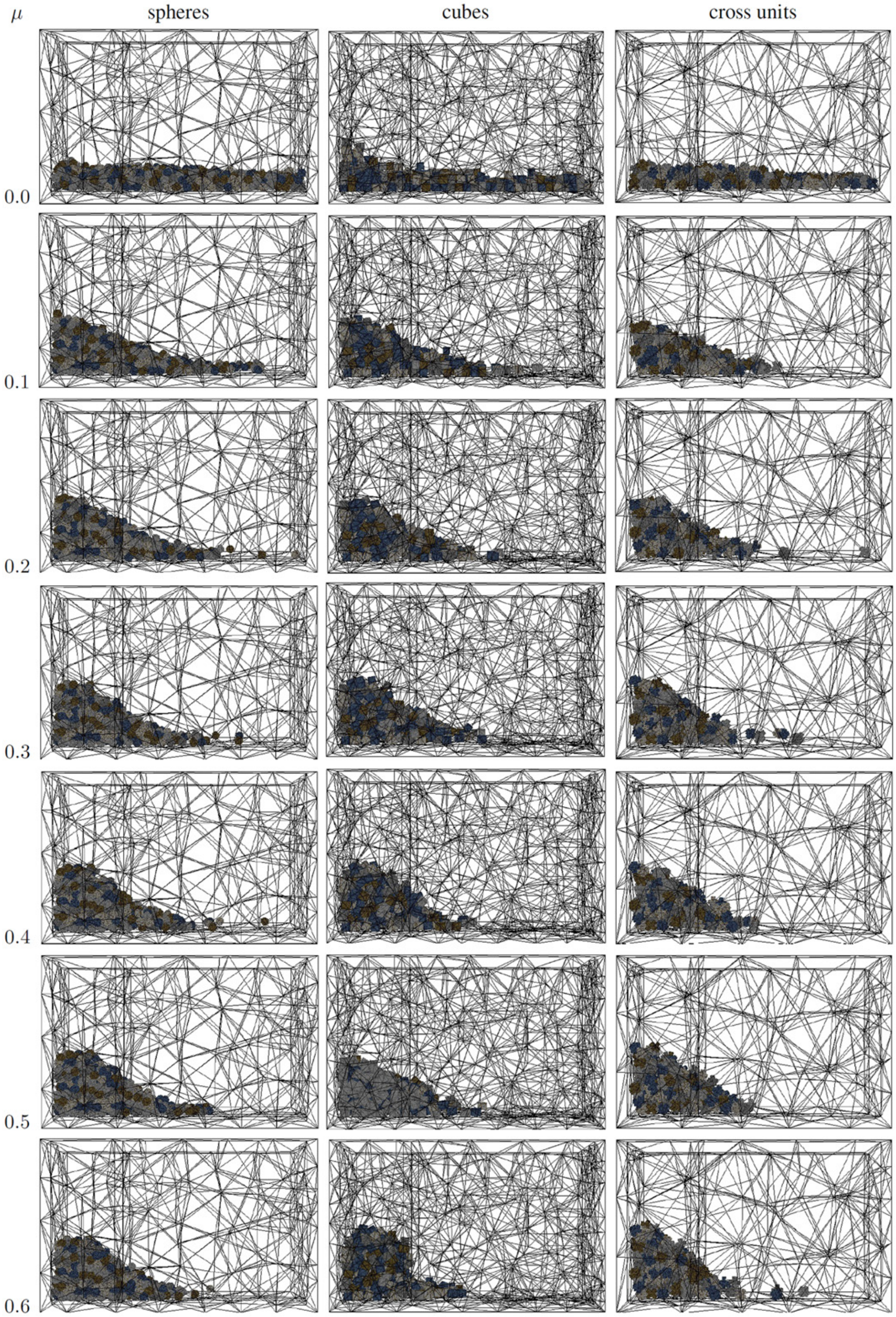


Fig. 13. Repositioning of multiple bodies with frictions ranging from 0.05 up to 0.6. Normal restitution is assumed $e_n = 0.3$.

simultaneously. By applying velocities gradually to the contact points, the impulses are able to influence multiple contact points at the same time, enabling propagation of impulse effects, and thereby improving the overall ability of

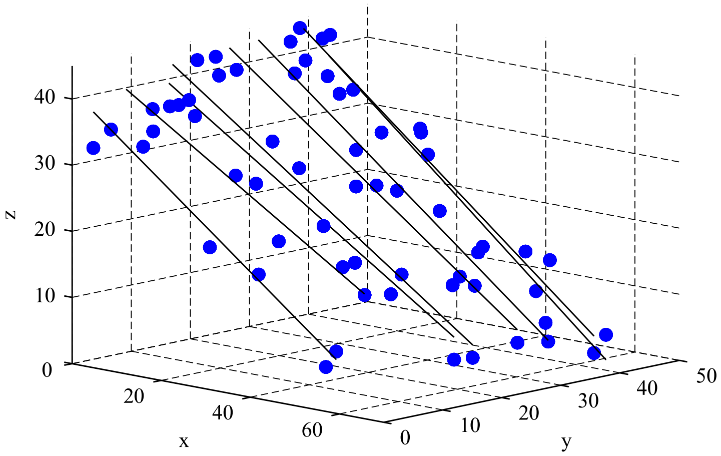


Fig. 14. Eight lines are fitted through the ensuing dataset to estimate the repose angle.

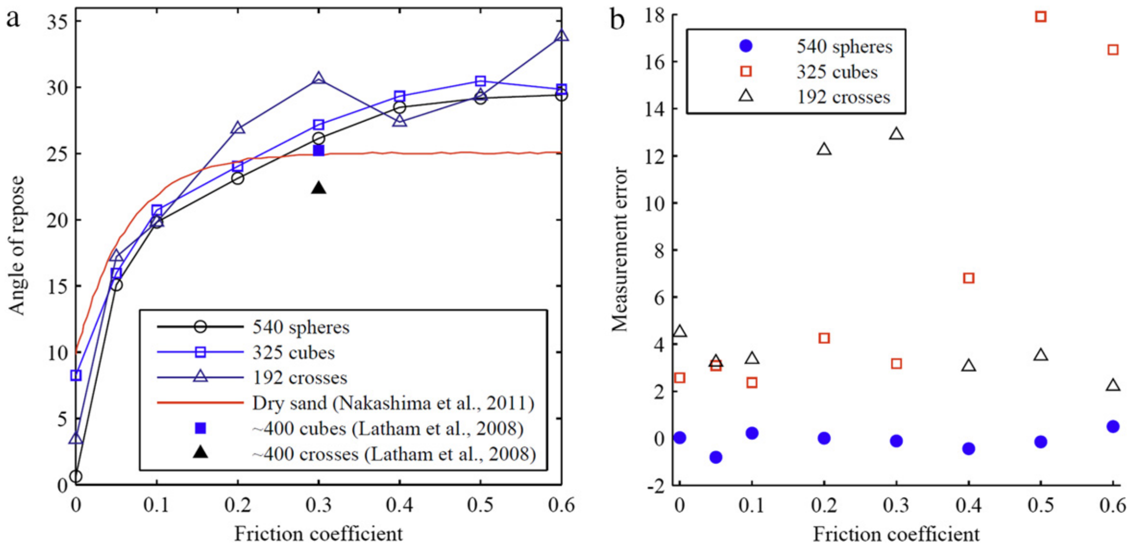


Fig. 15. Friction dependent angle of repose. (b) Variation for each shape as compared to experimental and numerical results reported in the literature, (b) measurement error for each case.

the method to capture propagation of forces during dynamics simulations. Furthermore, ETM ensures that no artificial energy loss or gain results from the resolution of the collisions, setting the stage for a more rigorous analysis of energy lost due to other factors, such as brittle deformation, during impact.

Acknowledgment

The authors thank Rio Tinto for supporting this work, through the Rio Tinto Centre for Advanced Mineral Recovery at Imperial College London (P16954).

Appendix A. Sequential method (SQM)

In the SQM, the relative velocity at single collision is updated directly by impulse in one iteration. $\mathbf{u}_i(s_0)$ and $\mathbf{u}_i(s_1)$ denote the relative velocities at contact point c_i before and after applying impulse respectively. The change of relative velocity can be expressed as

$$\triangle\mathbf{u}_i(s_0) = \mathbf{u}_i(s_1) - \mathbf{u}_i(s_0). \tag{71}$$

To simulate single-point collision with friction, first the static friction condition for the SQM is established. Assuming that the body is static in the tangential directions after collision, the relative velocity after applying the impulse is calculated as

$$\mathbf{u}_i(s_1) = -e_n \mathbf{u}_i(s_0)\mathbf{n}_i \tag{72}$$

where e_n is the coefficient of normal restitution, and $\mathbf{n}_i$ is the normal direction in local coordinates. The above equation can be rewritten in a local coordinate system as

$$\begin{bmatrix} \tilde{u}_{i,t}(s_1) & \tilde{u}_{i,n}(s_1) & \tilde{u}_{i,q}(s_1) \end{bmatrix} = \begin{bmatrix} 0 & -e_n \tilde{u}_{i,n}(s_0) & 0 \end{bmatrix}. \tag{73}$$

This local coordinate system is defined in Section 2. Then the impulse, $\mathbf{p}_i(s_0)$, is calculated by Eq. (8). The friction condition is defined as:

$$|\mathbf{p}_{i,t}(s_0)| < \mu |\mathbf{p}_{i,n}(s_0)| \tag{74}$$

where μ is the coefficient of friction. Once the static friction condition is not satisfied, then, $\mathbf{p}_i(s_0)$ is recomputed as follows [54]:

$$\tilde{p}_{i,n}(s_0) = \frac{\tilde{u}_{i,n}(s_0)(e_n + 1)}{\mathbf{n}_i^{\mathrm{T}} \mathbf{K}(\mathbf{n}_i - \mu \mathbf{q}_i)} \tag{75}$$

then the impulse is recalculated as:

$$\mathbf{p}_i(s_0) = |\tilde{p}_{i,n}(s_0)|\mathbf{n}_i + \mu |\tilde{p}_{i,n}(s_0)|\mathbf{q}_i. \tag{76}$$

The SQM treats multiple concurrent collisions (Fig. 5) as a series of single-point collisions, and the non-penetration constraints are enforced on multiple collisions sequentially. Because the SQM resolves the problem through a series of local solutions, occasionally the solution does not converge to a global solution of system. Even though this problem has not been fully solved, some approaches have been proposed to improve the stability of the SQM. In the following, the approach proposed by Chatterjee [55] is described. During initialization, all collisions are inserted into a heap data structure $\boldsymbol{\psi}$ according to the order of their relative normal velocities. The contact point with the smallest relative normal velocity is obtained from this heap data structure $\boldsymbol{\psi}$. If the smallest relative normal velocity is negative, impulses should be applied to the contact point having the smallest relative normal velocity, to avoid penetration. The impulse is calculated according to the equations from Eqs. (71) to (76). Then, the linear and angular velocities of the related colliding bodies are updated. After all of the dependent collisions are treated, the heap $\boldsymbol{\psi}$ is updated. This computation continues to iterate over the single-point collisions sequentially, until all the negative relative normal velocities are non-negative. For the requirement of numerical stability of SQM, a proper normal restitution coefficient is required [19]. This approach suffers from two main drawbacks: firstly, different orders of resolving collisions yield different results when multiple collisions are considered; secondly, infinite calculations occur in special cases discussed in [16].

Appendix B. Simultaneous method (SMM)

Alternatively, the SMM [18] inserts collisions into a system of linear equations. All of the constraints are resolved simultaneously. For simplification, SMM is described as the special case used in Section 7.2 with friction, $\mu = 0$, and restitution, $e_n = 1.0$. SMM avoids the problem of non-termination of the SQM. Simulation with different order of resolving collisions always produces the same global solution. However, the SMM enforces constraints, $\mathbf{u}_i(s_1) = -e_n \mathbf{u}_i(s_0)\mathbf{n}_i$, on all the collisions, which causes SQM to fail to simulate the interaction among multiple collisions accurately (Section 7.2).

Consider a single-point collision, c_i, as shown in Fig. 2. SMM also simulates the collision in one iteration. Let $\mathbf{v}^N(s_0)$ and $\omega^N(s_0)$ are the linear and angular velocities of body N before collision. $\mathbf{v}^M(s_0)$ and $\omega^M(s_0)$ are the linear and angular velocities of body M before collision. By applying impulse $\mathbf{p}_i(s_0)$ and $-\mathbf{p}_i(s_0)$ to contact point c_i, after collision the linear and angular velocities of body N become $\mathbf{v}^N(s_1)$ and $\omega^N(s_1)$, and the linear and angular velocities of body M become $\mathbf{v}^M(s_1)$ and $\omega^M(s_1)$. In three dimensions, the linear and angular velocities, together with impulses, lead to a total of fifteen unknowns, **x**:

$$\mathbf{x} = \begin{bmatrix}\mathbf{v}^N(s_1) & \omega^N(s_1) & \mathbf{v}^M(s_1) & \omega^M(s_1) & \mathbf{p}_i(s_0)\end{bmatrix}^{\mathrm{T}}. \tag{77}$$

To obtain **x**, a total of fifteen equations are expressed as a system of linear equations, $\mathbf{Ax} = \mathbf{b}$.

Firstly, due to the principle of impulse and linear momentum,

$$m_N\mathbf{v}^N(s_1) - \mathbf{p}_i(s_0) = m_N\mathbf{v}^N(s_0) \tag{78}$$

and

$$m_M\mathbf{v}^M(s_1) + \mathbf{p}_i(s_0) = m_M\mathbf{v}^M(s_0) \tag{79}$$

where m_N and m_M are the masses of bodies N and M.

Secondly, according to the principle of impulse and angular momentum,

$$\mathbf{I}^N\omega^N(s_1) - \mathbf{r}_i^N\mathbf{p}_i(s_0) = \mathbf{I}^N\omega^N(s_0) \tag{80}$$

and

$$\mathbf{I}^M\omega^M(s_1) + \mathbf{r}_i^M\mathbf{p}_i(s_0) = \mathbf{I}^M\omega^M(s_0) \tag{81}$$

where $\mathbf{r}_i^N$ and $\mathbf{r}_i^M$ are the vectors form the center of bodies N and M to the contact point c_i. $\mathbf{I}^N$ and $\mathbf{I}^M$ are inertia tensors for N and M, respectively. By substituting Eqs. (78), (79), (80), and (81) into the linear system of equations in its block-matrix form, which is expressed as

$$\begin{bmatrix} m_N\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{I}_N & \mathbf{0} & \mathbf{0} & -\mathbf{r}_N \\ \mathbf{0} & \mathbf{0} & m_M\mathbf{I} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_M & \mathbf{r}_M \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v}^N(s_1) \\ \omega^N(s_1) \\ \mathbf{v}^M(s_1) \\ \omega^M(s_1) \\ \mathbf{p}_i(s_0) \end{bmatrix} = \begin{bmatrix} m_N\mathbf{v}^N(s_0) \\ \mathbf{I}^N\omega^N(s_0) \\ m_M\mathbf{v}^M(s_0) \\ \mathbf{I}^M\omega^M(s_0) \\ \mathbf{0} \end{bmatrix} \tag{82}$$

where **I** is the 3 × 3 identity matrix.

Thirdly, the relationship between the relative normal velocity before collision, $\mathbf{u}_{i,n}(s_0)$, and after collision, $\mathbf{u}_{i,n}(s_1)$, is expressed, without considering the influence of the normal restitution coefficient, $e_n = 1$, as:

$$\mathbf{u}_{i,n}(s_1) = -\mathbf{u}_{i,n}(s_0) \tag{83}$$

by substituting Eqs. (1) and (2) into the previous equation,

$$\begin{aligned} &[(\mathbf{v}^N(s_1) + \omega^N(s_1) \times \mathbf{r}_i^N) - (\mathbf{v}^M(s_1) + \omega^M(s_1) \times \mathbf{r}_i^M)]\mathbf{n} \\ &\quad = -[(\mathbf{v}^N(s_0) + \omega^N(s_0) \times \mathbf{r}_i^N) - (\mathbf{v}^M(s_0) + \omega^M(s_0) \times \mathbf{r}_i^M)]\mathbf{n} \end{aligned} \tag{84}$$

which can be rewritten as

$$\begin{aligned} &\begin{bmatrix} n_{i,x} \\ n_{i,y} \\ n_{i,z} \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} v_x^N(s_1) \\ v_y^N(s_1) \\ v_z^N(s_1) \end{bmatrix} + \begin{bmatrix} n_{i,z}r_{i,y}^N - n_{i,y}r_{i,z}^N \\ n_{i,x}r_{i,z}^N - n_{i,z}r_{i,x}^N \\ n_{i,y}r_{i,x}^N - n_{i,x}r_{i,y}^N \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \omega_x^N(s_1) \\ \omega_y^N(s_1) \\ \omega_z^N(s_1) \end{bmatrix} \\ &\quad - \begin{bmatrix} n_{i,x} \\ n_{i,y} \\ n_{i,z} \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} v_x^M(s_1) \\ v_y^M(s_1) \\ v_z^M(s_1) \end{bmatrix} - \begin{bmatrix} n_{i,z}r_{i,y}^M - n_{i,y}r_{i,z}^M \\ n_{i,x}r_{i,z}^M - n_{i,z}r_{i,x}^M \\ n_{i,y}r_{i,x}^M - n_{i,x}r_{i,y}^M \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \omega_x^M(s_1) \\ \omega_y^M(s_1) \\ \omega_z^M(s_1) \end{bmatrix} = d_i \end{aligned} \tag{85}$$

$$d_i = -[(\mathbf{v}^N(s_0) + \omega^N(s_0) \times \mathbf{r}_i^N) - (\mathbf{v}^M(s_0) + \omega^M(s_0) \times \mathbf{r}_i^M)]\mathbf{n}. \tag{86}$$

Fourthly, the Coulomb friction is defined as

$$\begin{cases} \mathbf{p}_{i,t} = \mu_t\mathbf{p}_{i,n}(s_0) \\ \mathbf{p}_{i,k} = \mu_k\mathbf{p}_{i,n}(s_0) \end{cases} \tag{87}$$

then by substituting Eqs. (85)–(87) into the linear system of equations, and expressing it in block form, yields

$$\begin{bmatrix} m_N\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{I}_N & \mathbf{0} & \mathbf{0} & -\mathbf{r}_N \\ \mathbf{0} & \mathbf{0} & m_M\mathbf{I} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_M & \mathbf{r}_M \\ \mathbf{A}_{N,M} & \mathbf{B}_{N,M} & -\mathbf{A}_{N,M} & -\mathbf{C}_{N,M} & \mathbf{D}_{N,M} \end{bmatrix} \begin{bmatrix} \mathbf{v}^N(s_1) \\ \omega^N(s_1) \\ \mathbf{v}^M(s_1) \\ \omega^M(s_1) \\ \mathbf{p}_i(s_0) \end{bmatrix} = \begin{bmatrix} m_N\mathbf{v}^N(s_0) \\ \mathbf{I}^N\omega^N(s_0) \\ m_M\mathbf{v}^M(s_0) \\ \mathbf{I}^M\omega^M(s_0) \\ \mathbf{F}_i \end{bmatrix} \tag{88}$$

where the elements in **A** are expressed as

$$\mathbf{A}_{N,M} = \begin{bmatrix} n_{i,x} & n_{i,y} & n_{i,z} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{89}$$

$$\mathbf{B}_{N,M} = \begin{bmatrix} n_{i,z}r^N_{i,y} - n_{i,y}r^N_{i,z} & n_{i,x}r^N_{i,z} - n_{i,z}r^N_{i,x} & n_{i,x}r^N_{i,y} - n_{i,y}r^N_{i,x} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{90}$$

$$\mathbf{C}_{N,M} = \begin{bmatrix} n_{i,z}r^M_{i,y} - n_{i,y}r^M_{i,z} & n_{i,x}r^M_{i,z} - n_{i,z}r^M_{i,x} & n_{i,x}r^M_{i,y} - n_{i,y}r^M_{i,x} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{91}$$

$$\mathbf{D}_{N,M} = \begin{bmatrix} 0 & 0 & 0 \\ t_x - \mu_t n_x & t_y - \mu_t n_y & t_z - \mu_t n_z \\ q_x - \mu_q n_x & q_y - \mu_q n_y & q_z - \mu_q n_z \end{bmatrix} \tag{92}$$

and,

$$\mathbf{F}_i = \begin{bmatrix} d_i \\ 0 \\ 0 \end{bmatrix}. \tag{93}$$

In SMM, multiple collisions are resolved simultaneously, and the assembly of a system equations, $\mathbf{Ax} = \mathbf{b}$, for multiple collisions is introduced. The vector of unknowns **x** for multiple collisions is defined as

$$\mathbf{x} = \begin{bmatrix} \mathbf{v}^0(s_0) \\ \omega^0(s_0) \\ \cdots \\ \mathbf{v}^m(s_0) \\ \omega^m(s_0) \\ \mathbf{p}_0(s_0) \\ \cdots \\ \mathbf{p}_n(s_0) \end{bmatrix} \tag{94}$$

where m is the number of bodies and n is the number of contact points.

Firstly, applying the principle of impulse and linear momentum for body j,

$$m_j\mathbf{v}^j(s_0) - \sum^k \mathbf{I}^j\mathbf{p}_i(s_0) = m_j\mathbf{v}^j(s_1) \tag{95}$$

assemble Eq. (95) to the matrix **A** and vector **b**:

$$\begin{bmatrix} A_{6j-5,6j-5} & A_{6j-5,6j-4} & A_{6j-5,6j-3} \\ A_{6j-4,6j-5} & A_{6j-4,6j-4} & A_{6j-4,6j-3} \\ A_{6j-3,6j-5} & A_{6j-3,6j-4} & A_{6j-3,6j-3} \end{bmatrix} = \begin{bmatrix} m_j & 0 & 0 \\ 0 & m_j & 0 \\ 0 & 0 & m_j \end{bmatrix} \tag{96}$$

and

$$\begin{bmatrix} A_{6j-5,6m+3i} & A_{6j-5,6m+3i-1} & A_{6j-5,6m+3i} \\ A_{6j-4,6m+3i} & A_{6j-4,6m+3i-1} & A_{6j-4,6m+3i} \\ A_{6j-3,6m+3i} & A_{6j-3,6m+3i-1} & A_{6j-3,6m+3i} \end{bmatrix} = -\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{97}$$

and

$$\begin{bmatrix} b_{6j-5} \\ b_{6j-4} \\ b_{6j-3} \end{bmatrix} = m_j \mathbf{v}^j(s_1). \tag{98}$$

Secondly, according to the principle of impulse and angular momentum, the relation among impulse, angular momentum of body j $(j = 1 \dots m)$, and impulses at contact points c_i $(j = 1 \dots n)$, can be expressed as

$$m_j \mathbf{I}^j \omega^j(s_1) - \sum^{k} r_i^j \mathbf{p}_i(s_0) = \mathbf{I}^j \omega(s_0) \tag{99}$$

subsequently, the linear system of equations is assembled via

$$\begin{bmatrix} A_{6j-2,6j-2} & A_{6j-2,6j-1} & A_{6j-2,6j} \\ A_{6j-1,6j-2} & A_{6j-1,6j-1} & A_{6j-1,6j} \\ A_{6j,6j-2} & A_{6j,6j-1} & A_{6j,6j} \end{bmatrix} = \mathbf{I}^j \tag{100}$$

$$\begin{bmatrix} A_{6j-2,6m+3i-2} & A_{6j-2,6m+3i-1} & A_{6j-2,6m+3i} \\ A_{6j-1,6m+3i-2} & A_{6j-1,6m+3i-1} & A_{6j-1,6m+3i} \\ A_{6j,6m+3i-2} & A_{6j,6m+3i-1} & A_{6j,6m+3i} \end{bmatrix} = -\begin{bmatrix} 0 & (r_i^j)_q & -(r_i^j)_t \\ -(r_i^j)_q & 0 & (r_i^j)_n \\ (r_i^j)_t & -(r_i^j)_n & 0 \end{bmatrix} \tag{101}$$

and

$$\begin{bmatrix} b_{6j-2} \\ b_{6j-1} \\ b_{6j} \end{bmatrix} = \mathbf{I}^j \omega(s_0). \tag{102}$$

Thirdly, Newton's impact law is introduced to the collision c_i between bodies b_N and b_M, and matrix $\mathbf{A}$ is manipulated to yield

$$\begin{bmatrix} A_{6m+3i-2,6N-5} & A_{6m+3i-2,6N-4} & A_{6m+3i-2,6N-3} \\ A_{6m+3i-1,6N-5} & A_{6m+3i-1,6N-4} & A_{6m+3i-1,6N-3} \\ A_{6m+3i,6N-5} & A_{6m+3i,6N-4} & A_{6m+3i,6N-3} \end{bmatrix} = \begin{bmatrix} n_{i,x} & n_{i,y} & n_{i,z} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{103}$$

and

$$\begin{bmatrix} A_{6m+3i-2,6N-2} & A_{6m+3i-2,6N-1} & A_{6m+3i-2,6N} \\ A_{6m+3i-1,6N-2} & A_{6m+3i-1,6N-1} & A_{6m+3i-1,6N} \\ A_{6m+3i,6N-2} & A_{6m+3i,6N-1} & A_{6m+3i,6N} \end{bmatrix}$$
$$= \begin{bmatrix} n_{i,z} r_{i,y}^N - n_{i,y} r_{i,z}^N & n_{i,x} r_{i,z}^N - n_{i,z} r_{i,x}^N & n_{i,x} r_{i,y}^N - n_{i,y} r_{i,x}^N \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{104}$$

and

$$\begin{bmatrix} A_{6m+3i-2,6M-5} & A_{6m+3i-2,6M-4} & A_{6m+3i-2,6M-3} \\ A_{6m+3i-1,6M-5} & A_{6m+3i-1,6M-4} & A_{6m+3i-1,6M-3} \\ A_{6m+3i,6M-5} & A_{6m+3i,6M-4} & A_{6m+3i,6M-3} \end{bmatrix} = -\begin{bmatrix} n_{i,x} & n_{i,y} & n_{i,z} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{105}$$

and

$$\begin{bmatrix} A_{6m+3i-2,6M-2} & A_{6m+3i-2,6M-1} & A_{6m+3i-2,6M} \\ A_{6m+3i-1,6M-2} & A_{6m+3i-1,6M-1} & A_{6m+3i-1,6M} \\ A_{6m+3i,6M-2} & A_{6m+3i,6M-1} & A_{6m+3i,6M} \end{bmatrix}$$

$$= -\begin{bmatrix} n_{i,z}r^M_{i,y} - n_{i,y}r^M_{i,z} & n_{i,x}r^M_{i,z} - n_{i,z}r^M_{i,x} & n_{i,x}r^M_{i,y} - n_{i,y}r^M_{i,x} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{106}$$

Fourthly, Coulomb friction is considered, and the linear system is modified as follows:

$$\begin{bmatrix} A_{6n+3i-2,6n+3i-2} & A_{6n+3i-2,6n+3i-1} & A_{6n+3i-2,6n+3i} \\ A_{6n+3i-1,6n+3i-2} & A_{6n+3i-1,6n+3i-1} & A_{6n+3i-1,6n+3i} \\ A_{6n+3i,6n+3i-2} & A_{6n+3i,6n+3i-2} & A_{6n+3i,6n+3i} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ t_x - \mu_t n_{i,x} & t_y - \mu_t n_{i,y} & t_z - \mu_t n_{i,z} \\ q_x - \mu_q n_{i,x} & q_y - \mu_q n_{i,y} & q_z - \mu_q n_{i,z} \end{bmatrix}. \tag{107}$$

Finally, the vector **b** is updated:

$$\begin{bmatrix} b_{6n+3i-2} \\ b_{6n+3i-1} \\ b_{6n+3i} \end{bmatrix} = \begin{bmatrix} d_i \\ 0 \\ 0 \end{bmatrix} \tag{108}$$

where

$$d_i = -((\mathbf{v}^N(s_0) + \omega^N(s_0)\mathbf{r}^N_i) - (\mathbf{v}^M(s_0) + \omega^M(s_0)\mathbf{r}^M_i))\mathbf{n}. \tag{109}$$

The solution to this linear system of equations, $\mathbf{Ax} = \mathbf{b}$, yields the linear and angular velocities of the colliding bodies after collision, along with the impulse applied to avoid penetration.

Appendix C. Supplementary data

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.cma.2014.05.004.

References

[1] A. Munjiza, J. Latham, Comparison of experimental and FEM/DEM results for gravitational deposition of identical cubes, Eng. Comput. 21 (2–3) (2004) 249–264.

[2] A. Vyazmensky, Numerical modelling of surface subsidence associated with block cave mining using a finite element/discrete element approach (Ph.D. thesis), Fraser University, 2008.

[3] A. Paluszny, X. Tang, R. Zimmerman, A fracture- and impulse-based FDEM approach for fragmentation, Comput. Mech. 52 (5) (2013) 1071–1084.

[4] B. Mishra, A review of computer simulation of tumbling mills by the discrete element method: part II: practical applications, Int. J. Miner. Process. 71 (1–4) (2003) 95–112.

[5] R. Guises, J. Xiang, J.-P. Latham, A. Munjiza, Granular packing: numerical simulation and the characterisation of the effect of particle shape, Granular Matter 11 (5) (2009) 281–292.

[6] D. Baraff, Fast contact force computation for nonpenetrating rigid bodies, Comput. Graph. 28 (1994) 23–34.

[7] B. Mirtich, Impulse-based dynamic simulation of rigid body systems (Ph.D. thesis), University of California at Berkeley, 1996.

[8] J.K. Hahn, Realistic animation of rigid bodies, Comput. Graph. 22 (4) (1988) 299–308.

[9] X. Tang, A. Paluszny, R. Zimmerman, Energy conservative property of impulse-based methods for collision resolution, Internat. J. Numer. Methods Engrg. 95 (6) (2013) 529–540.

[10] J.M. Hollerbach, W.B. Thompson, P. Shirley, The convergence of robotics, vision, and computer graphics for user interaction, Int. J. Robot. Res. 18 (1999).

[11] S. Sueda, A. Kaufman, D.K. Pai, Musculotendon simulation for hand animation, ACM Trans. Graph. 27 (3) (2008).

[12] R. Weinstein, E. Guendelman, R. Fedkiw, Impulse-based control of joints and muscles, IEEE Trans. Vis. Comput. Graphics 14 (1) (2008).

[13] D. Reznik, S. Brown, J. Canny, Dynamic simulation as a design tool for a microactuator array, in: IEEE International Conference on Robotics and Automation, vol. 2, 1997, pp. 1675–1680.

[14] M. Lin, Efficient collision detection for animation and robotics (Ph.D. thesis), University of California, Berkeley, 1993.

[15] M. Lin, J. Canny, A fast algorithm for incremental distance calculation, in: Proceeding of the IEEE International Conference on Robotics and Automation, IEEE, Sacramento, CA, USA, 1991.

[16] K. Erleben, J. Sporring, K. Henriksen, H. Dohlmann, Physics Based Animation, Charles River Media, 2005.

[17] D. Baraff, Analytical methods for dynamic simulation of non-penetrating rigid bodies, Comput. Graph. 23 (3) (1989) 223–232.

[18] M. Coutinho, Dynamic Simulations of Multibody Systems, Springer, 2001.

[19] P. Mosterman, On the normal component of centralized frictionless collision sequences, J. Appl. Mech. 75 (5) (2007) 908–915.

[20] I. Babuska, The finite element method with penalty, Math. Comp. 27 (122) (1973) 221–228.

[21] T. Belytschko, The splitting pinball method for contact-impact problems, Comput. Methods Appl. Mech. Engrg. 105 (3) (1993) 375–393.

[22] S. Hasegawa, N. Fujii, Real-time rigid body simulation based on volumetric penalty method, in: 11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2003, pp. 326–332.

[23] N.G. Bourago, V.N. Kukudzhanov, A review of contact algorithms, Izv. RAN, MTT, No. 1, pp. 45–87, 2005 (translation to English).

[24] Y. Wang, F. Tonon, Discrete element modeling of rock fragmentation upon impact in rock fall analysis, Rock Mech. Rock Eng. 44 (2011) 23–35.

[25] Y. Wang, F. Tonon, Calibration of a discrete element model for intact rock up to its peak strength, Int. J. Numer. Anal. Methods Geomech. 34 (5) (2010) 447–469.

[26] S. Hentz, L. Daudeville, F. Donze, Identification and validation of a discrete element model for concrete, J. Eng. Mech. 130 (6) (2004) 709–719.

[27] J. McCarthy, V. Jasti, M. Marinack, C. Higgs, Quantitative validation of the discrete element method using an annular shear cell, Powder Technol. 203 (2010) 70–77.

[28] T. Laursen, V. Chawla, Design of energy conserving algorithms for frictionless dynamic contact problems, Internat. J. Numer. Methods Engrg. 40 (5) (1997) 863–886.

[29] E. Drumwright, A fast and stable penalty method for rigid body simulation, IEEE Trans. Vis. Comput. Graphics 14 (1) (2008) 231–240.

[30] G. Haikal, Hjelmstad, A finite element formulation of non-smooth contact based on oriented volumes for quadrilateral and hexahedral elements, Comput. Methods Appl. Mech. Engrg. 196 (2007) 4690–4711.

[31] L. Noels, L. Stainier, J. Ponthot, An energy momentum conserving algorithm using the variational formulation of visco-plastic updates, Internat. J. Numer. Methods Engrg. 65 (6) (2006) 904–942.

[32] V. Chawla, T. Laursen, Energy consistent algorithms for frictional contact problems, Internat. J. Numer. Methods Engrg. 42 (5) (1998) 799–827.

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

[34] S. Magnier, F. Donzé, Numerical simulations of impacts using a discrete element method, Mech. Cohesive-frictional Mater. 3 (3) (1998) 257–276.

[35] L.B. Lucy, A numerical approach to the testing of the fission thesis, Astron. J. 82 (1977) 1013–1024.

[36] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics theory and application to non-spherical stars, Mon. Not. R. Astron. Soc. 181 (1977) 375–389.

[37] S. Bardenhagen, J. Brackbill, D. Sulsky, The material-point method for granular materials, Comput. Methods Appl. Mech. Engrg. 187 (3–4) (2000) 529–541.

[38] D. Sulsky, A. Kaul, Implicit dynamics in the material-point method, Comput. Methods Appl. Mech. Engrg. 193 (12–14) (2004) 1137–1170.

[39] J. Campbell, R. Vignjevic, L. Libersky, Computer methods in applied mechanics and engineering, Comput. Methods Appl. Mech. Engrg. 184 (1) (2000) 49–65.

[40] A. Munjiza, The Combined Finite-Discrete Element Method, John Wiley and Sons, Ltd, 2004.

[41] M. Pierce, A model for gravity flow of fragmented rock in caving mines, (Ph.D. thesis), The University of Queensland, 2010.

[42] J. Chardonnet, S. Miossec, A. Kheddar, H. Arisumi, H. Hirukawa, F. Pierrot, K. Yokoi, Dynamic simulator for humanoids using constraint-based method with static friction, in: Proceedings of IEEE International Conference on Robotics and Biomimetics, Kunming, China, 2006.

[43] F. Armero, E. Petőcz, A new dissipative time-stepping algorithm for frictional contact problems: formulation and analysis, Comput. Methods Appl. Mech. Engrg. 179 (1–2) (1999) 151–178.

[44] F. Armero, E. Petőcz, Formulation and analysis of conserving algorithms for frictionless dynamic contact/impact problems, Comput. Methods Appl. Mech. Engrg. 158 (3–4) (1998) 269–300.

[45] C. Hesch, P. Betsch, Transient 3D contact problems—NTS method: mixed methods and conserving integration, Comput. Mech. 48 (4) (2011) 437–449.

[46] S. Gautam, R. Sauer, An energy–momentum-conserving temporal discretization scheme for adhesive contact problems, Internat. J. Numer. Methods Engrg. 93 (10) (2013) 1057–1081.

[47] V. Chawla, T.A. Laursen, Energy consistent algorithms for frictional contact problems, Internat. J. Numer. Methods Engrg. 42 (5) (1998) 799–827.

[48] Y. Zhou, B. Xu, A. Yu, P. Zulli, An experimental and numerical study of the angle of repose of coarse spheres, Powder Technol. 125 (1) (2002) 45–54.

[49] W. Stronge, Impact Mechanics, Cambridge University Press, Cambridge, UK, 2000.

[50] F. Tonon, Explicit exact formulas for the 3-D tetrahedron inertia tensor in terms of its vertex coordinates, J. Math. Stat. 1 (1) (2004) 8–11.

[51] H. Nakashima, Y. Shioji, T. Kobayashi, S. Aoki, H. Shimizu, J. Miyasaka, K. Ohdoi, Determining the angle of repose of sand under low-gravity conditions using discrete element method, J. Terramech. 48 (1) (2011) 17–26.

[52] J. Rogers, B. Hamer, A. Brampton, S. Challinor, M. Glennerster, P. Brenton, A. Bradbury, Beach Management Manual (second ed.), Tech. Rep., Robotics Institute, Carnegie Mellon Univ., 1996.

[53] J. Latham, A. Munjiza, J. Mindel, J. Xiang, R. Guises, X. Garcia, C. Pain, G. Gorman, M. Piggott, Modelling of massive particulates for breakwater engineering using coupled FEMDEM and CFD, Particuology 6 (6) (2008) 572–583.

[54] E. Guendelman, R. Bridson, R. Fedkiw, Nonconvex rigid bodies with staking, ACM Trans. Graph. 22 (3) (2003) 871–878.

[55] A. Chatterjee, A. Ruina, A new algebraic rigid body collision law based on impulse space considerations, J. Appl. Mech. 65 (4) (1998) 939–951.