Available online at www.sciencedirect.com

ScienceDirect

Comput. Methods Appl. Mech. Engrg. 321 (2017) 124–144

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

A frictional contact algorithm for implicit material point method[☆]

Zhen-Peng Chen, Xiong Zhang[*], Xin-Ming Qiu, Yan Liu

School of Aerospace Engineering, Tsinghua University, Beijing 100084, PR China

Received 24 September 2016; received in revised form 19 February 2017; accepted 6 April 2017
Available online 13 April 2017

Highlights

- Propose a frictional contact algorithm for the implicit MPM using the augmented Lagrange formulation.
- Employ the Uzawa algorithm to decouple the unknown variables and the Lagrange multipliers.
- The implicit MPM with the proposed contact algorithm is able to efficiently simulate large deformation problems.

Abstract

The explicit material point method (MPM) works successfully in modeling high frequency problems, but it is very computationally expensive in simulating low frequency with small time steps or quasi-static problems. Thus, several groups have developed an implicit MPM for modeling low frequency problems. Recently, a few attempts were undertaken to investigate the contact problems using the implicit MPM but the accuracy was dissatisfactory. In this paper, an augmented Lagrange formulation for the frictional inequality constraints is introduced. A discretization of the Lagrange multiplier field based on the background grid is proposed to establish the implicit MPM framework with the contact algorithm. To reduce the complexity of the solution, the Uzawa algorithm is employed to decouple the unknown variables and the Lagrange multipliers. Finally, the resulting sequent nonlinear equations are solved by the Newton method, in which the tangential matrix is assembled explicitly. By using the compressed sparse row (CSR) technique, the total storage of the matrix can be greatly reduced. Numerical studies show that the computational efficiency and accuracy of the implicit MPM with the proposed contact algorithm are much higher than the explicit MPM.

Keywords: Implicit material point method; Augmented Lagrange method; Frictional contact

1. Introduction

In recent decades, lots of meshfree and particle methods have been proposed to study challenging problems, such as penetration, impact, fluid–structure interaction (FSI) and explosion. The material point method (MPM) is a fully Lagrange particle method developed by Sulsky et al. [1,2]. The material domain is discretized by a set of Lagrange particles, which move through a predefined Eulerian background grid. In each time step, the particles are rigidly

[☆] Supported by the National Natural Science Foundation of China (11272180, 11390363).

[*] Corresponding author.
E-mail address: xzhang@tsinghua.edu.cn (X. Zhang).

http://dx.doi.org/10.1016/j.cma.2017.04.006
0045-7825/© 2017 Elsevier B.V. All rights reserved.

attached to the background grid and move together with the grid. The solutions are mapped from the grid points to the particles to update their positions and velocities after solving the momentum equations on the background grid. Finally, the deformed grid is discarded and a new regular grid is defined in the next time step. Thus, the mesh distortion is avoided, which often arises in FEM with large deformation. Meanwhile, the numerical difficulties associated with Eulerian method are overcome.

For extreme deformation problems, the MPM takes competitive advantages compared with the FEM which often encounters the mesh distortion and the element entanglement. It has been used to simulate large strain problems [3,4], fluid–structure interaction [5–7], dynamic crack propagation [8], hyper-velocity impact [9,10] and explosion [11,12], saturated soil–structure interaction problems [13], just to name a few.

Most works in the MPM employed an explicit time integration which is effective and efficient for modeling high frequency and high velocity problems. However, it is very computationally expensive and leads to significant numerical oscillation in simulating low frequency and low velocity problems such as rolling, upsetting, sloshing [7] and fluid–structure interaction (FSI). When an explicit time integration is used, the time step size is restricted by the Courant stability condition. If only the low-frequency response is interested, employing an implicit method can dramatically reduce the computational cost. In the past decade, several implicit time integration schemes have been developed and implemented with the MPM for low rate dynamic and quasi-static problems. Cummins and Brackbill [14] proposed an implicit MPM for granular materials based on the theta method and matrix-free Newton–Krylov solver with a first-order contact algorithm; however, the time-step sizes cannot be significantly greater than the explicit algorithm. Sulsky and Kaul [15] proposed a similar approach with a slightly different time discretization, but employed the Newton-CG or Newton-GMRES solver. Guilkey and Weiss [16] developed an implicit integration strategy for the MPM using Newmark integration. The tangent stiffness was formed explicitly and the linearized equations of motion was solved either by a CG or a direct solver. Nair and Roy [17] discussed an implicit time-integration scheme for the generalized interpolation material point method (GIMP), which is an extension of the algorithm by Sulsky and Kaul [15]. Recently, Wang et al. [18] presented an implicit MPM using the Newmark time integration.

In many engineering problems, contact phenomena plays an important role and must be taken into account. The original MPM employs a single-valued velocity field to update the positions of material points, so that the no-slip contact between two different bodies is inherent at no additional cost. To enable the frictional or sliding contact, Bardenhagen et al. [19,20] proposed a contact algorithm, which forbids interpenetration, but allows separation and sliding and rolling with friction. Huang et al. [21] further improved the contact algorithm to make the surface normals collinear at the interface so that the momentum conservation is guaranteed. To allow engagements at successive contact points and subsequent separation, a contact/sliding/separation procedure in a multi-mesh environment is proposed by Hu and Chen [22]. Ma et al. [23] proposed a local multi-mesh contact scheme to reduce the computational cost, and an improved contact detection scheme to avoid the contact occurring earlier than the actual time. Nairn provided a better method to find the contact normal between different materials by using volume gradients and an additional criterion by extrapolating positions or displacements to avoid contact occurring before surface are actually touching [24].

The implicit MPM has been used to simulate problems with contact phenomena, such as grain studies [14,16,25], upsetting [15–17], rolling [26], collision of elastic disks [15,18] and slope collapse [18]. In the original implicit MPM, the sticking contact between two different bodies is handled automatically at no additional computational cost with the single-valued velocity field. Cummins and Brackbill [14] firstly reported a first-order contact algorithm for the implicit MPM, in which the contact force is calculated by using the velocity field assuming no contact happens. The contact force keeps unchanged in the iteration process, which may decreases the computational accuracy considerably with a large time step. Nezamabadi et al. [26] proposed a contact algorithm based on the contact dynamics method in which the contact variables (velocity, force, etc.) are computed simultaneously with bulk variables (stresses, strains, etc.).

This paper aims to provide a highly efficient, accurate and robust contact algorithm for the implicit MPM based on an augmented Lagrange formulation for the frictional inequality constraints. A local multi-mesh algorithm is used to described the velocity fields of different bodies [23], and the Lagrange multiplier field is discretized based on the background grid. The contact problem is one of the saddle point problems, which a re usually indefinite and highly ill-conditioned. All these features bring great challenges to the Newton-like methods, which require the Hessen matrix to be positive definite. In numerical mathematics, the Uzawa iteration [27,28] is an effective algorithm for solving these saddle point problems. By employing the Uzawa algorithm, the condition number of the original problem is dramatically improved and the positive definiteness of the effective stiffness matrix can be reserved. In addition, the Uzawa algorithm decouples the unknown variables and the multipliers, which simplifies the solving process. In each of the linearized problems, we explicitly assemble the tangential matrix and solve the equations with the Intel MKL PARDISO solver because a direct solver is usually more robust than an iterative solver. To reduce the total storage of the stiffness matrix, the compressed sparse row (CSR) technique [29] is employed. Numerical examples validate the accuracy, efficiency and robustness of the contact algorithm. Compared with the first-order contact algorithm [14], our algorithm iterates the contact force during the solution so that a much larger time step can be used.

The remaining part of this paper is organized as follows. Section 2 briefly reviews the implicit MPM, while Section 3 presents the contact condition and the augmented Lagrange formulation. The augmented Lagrangian contact algorithm for the implicit MPM is presented in detail in Section 4 and its numerical implementation is described in Section 5. Section 6 validates the accuracy, efficiency and robustness of the proposed contact algorithm by numerical examples. Finally, the conclusions are summarized in Section 7.

2. Implicit material point method

In this section, the equations governing the motion and deformation of materials are first presented in the updated Lagrangian frame, and then the particle discretization is employed to discretize the weak form so that the discrete momentum equations are established. Finally, the Newmark time integration scheme is used to integrate the momentum equations.

2.1. Governing equations

In material domain Ω, the basic governing equations in the updated Lagrangian frame are

$$\rho\left(\boldsymbol{X},t\right) J\left(\boldsymbol{X},t\right) = \rho_0\left(\boldsymbol{X}\right) \tag{1}$$

$$\nabla\cdot\boldsymbol{\sigma} + \boldsymbol{f} = \rho\ddot{\boldsymbol{u}} \tag{2}$$

$$\rho\dot{w}^{\text{int}} = \boldsymbol{D} : \boldsymbol{\sigma} \tag{3}$$

where the superimposed dot signifies the time derivatives, ρ is the current density, $\boldsymbol{X}$ is the Lagrange coordinate, J is the Jacobian determinant, $\boldsymbol{u}$ is the displacement, $\boldsymbol{f}$ is the body force per unit mass, w is the internal energy per unit mass, $\boldsymbol{\sigma}$ is the Cauchy stress and $\boldsymbol{D}$ is the rate-of-deformation. The boundary conditions and initial conditions are

$$\begin{aligned}\left(\boldsymbol{\sigma}\cdot\boldsymbol{n}\right)|_{\Gamma_t} &= \bar{\boldsymbol{t}}\\ \boldsymbol{u}|_{\Gamma_u} &= \bar{\boldsymbol{u}}\end{aligned} \tag{4}$$

and

$$\begin{aligned}\dot{\boldsymbol{u}}\left(\boldsymbol{X},0\right) &= \dot{\boldsymbol{u}}_0\left(\boldsymbol{X}\right)\\ \boldsymbol{u}\left(\boldsymbol{X},0\right) &= \boldsymbol{u}_0\left(\boldsymbol{X}\right)\end{aligned} \tag{5}$$

respectively, where $\boldsymbol{n}$ is the unit outward normal to the boundary. The displacement boundary and prescribed traction boundary of Ω are signified by Γ_u and Γ_t, respectively.

The weak form is equivalent to the momentum equation and the traction boundary condition can be formulated as

$$\delta\dot{w} = \int_\Omega \rho\delta\boldsymbol{v}\cdot\dot{\boldsymbol{v}}d\Omega + \int_\Omega \boldsymbol{\sigma} : \delta\nabla\boldsymbol{v}d\Omega - \int_\Omega \delta\boldsymbol{v}\cdot\boldsymbol{f}d\Omega - \int_{\Gamma_t}\delta\boldsymbol{v}\cdot\bar{\boldsymbol{t}}d\Gamma = 0 \tag{6}$$

with $\delta\boldsymbol{v}$ denoting the virtual velocity and the velocity boundary conditions satisfied a priori.

2.2. Particle discretization

In the MPM, the material domain is discretized by a set of particles. Taking the integration on particles in Eq. (6) and invoking the arbitrariness of $\delta\boldsymbol{v}_I$ lead to

$$m_I\boldsymbol{a}_I = \boldsymbol{f}_I \qquad I = 1, 2, \ldots, n_g \tag{7}$$

where $\boldsymbol{a}_I$ represents the grid nodal acceleration,

$$m_I = \sum_{p=1}^{n_g} N_{Ip} m_p \tag{8}$$

is the lumped grid nodal mass and m_p is the mass of particle p.

$$\boldsymbol{f}_I = \boldsymbol{f}_I^{\text{ext}} + \boldsymbol{f}_I^{\text{int}} \tag{9}$$

is the grid nodal force,

$$\boldsymbol{f}_I^{\text{int}} = -\sum_{p=1}^{n_p} \frac{m_p}{\rho_p} \boldsymbol{\sigma}_p \frac{\partial S_{Ip}}{\partial \boldsymbol{x}_p} \tag{10}$$

is the internal grid nodal force, and N_{Ip} is the value of the generalized shape function of grid point I evaluated at the site of particle p.

$$\boldsymbol{f}_I^{\text{ext}} = \sum_{p=1}^{n_p} m_p N_{Ip} \boldsymbol{b}_p \tag{11}$$

is the external grid nodal force with the traction $\bar{\boldsymbol{t}}$ omitted for simplicity, $\boldsymbol{\sigma}_p = \boldsymbol{\sigma}(\boldsymbol{x}_p)$ is the Cauchy stress, and $\boldsymbol{b}_p = \boldsymbol{b}(\boldsymbol{x}_p)$ is the body force.

2.3. The implicit time integration

In the Newmark method, the velocity and displacement at time t^{n+1} are approximated as

$$\boldsymbol{v}_I^{n+1} = \boldsymbol{v}_I^n + \Delta t[(1-\gamma)\boldsymbol{a}_I^n + \gamma \boldsymbol{a}_I^{n+1}],\ 0 \le \gamma \le 1 \tag{12}$$

$$\boldsymbol{u}_I^{n+1} = \boldsymbol{u}_I^n + \Delta t \boldsymbol{v}_I^n + \frac{\Delta t^2}{2}[(1-2\beta)\boldsymbol{a}_I^n + 2\beta \boldsymbol{a}_I^{n+1}],\ 0 \leqslant 2\beta \leqslant 1 \tag{13}$$

where $\boldsymbol{a}_I^n$ and $\boldsymbol{a}_I^{n+1}$ denote the acceleration of grid point I at time t^n and t^{n+1} respectively. Typically, $\gamma = 0.5$ and $\beta = 0.25$ are used, which yield the constant average acceleration method.

Substituting Eq. (7) into Eq. (12) yields

$$\boldsymbol{R}(\boldsymbol{v}^{n+1}) = \frac{1}{\Delta t}\boldsymbol{M}(\boldsymbol{v}^{n+1} - \widetilde{\boldsymbol{v}}^n) - \gamma \boldsymbol{f}^{\text{int}}(\boldsymbol{v}^{n+1}, t^{n+1}) \tag{14}$$

where

$$\widetilde{\boldsymbol{v}}^n = \boldsymbol{v}^n + \Delta t \boldsymbol{M}^{-1}[(1-\gamma)(\boldsymbol{f}^{\text{ext}}(t^n) + \boldsymbol{f}^{\text{int}}(\boldsymbol{v}^n, t^n)) + \gamma \boldsymbol{f}^{\text{ext}}(t^{n+1})] \tag{15}$$

$$\boldsymbol{M} = \text{diag}\{m_1, m_2, \ldots, m_{n_g}\}. \tag{16}$$

The nonlinear equation (14) can be solved with the Newton's method. Let subscript j denotes the iteration number so that $\boldsymbol{v}_{(j)}$ is the grid nodal velocity in jth iteration at the time step $n+1$, i.e., $\boldsymbol{v}_{(j)} \equiv \boldsymbol{v}_{(j)}^{n+1}$. To begin the iterative procedure, a starting value $\boldsymbol{v}_{(0)}$ is required, which is usually chosen as the solution of the last time step, i.e. $\boldsymbol{v}_{(0)} \equiv \boldsymbol{v}^n$. Expanding the residual $\boldsymbol{R}(\boldsymbol{v}_{(j+1)})$ about the current value of the grid nodal velocity $\boldsymbol{v}_{(j)}$ and setting the resulting residual equal to zero lead to

$$\boldsymbol{R}(\boldsymbol{v}_{(j+1)}) \approx \boldsymbol{R}(\boldsymbol{v}_{(j)}) + \boldsymbol{A}\Delta \boldsymbol{v}_{(j)} = 0 \tag{17}$$

where

$$\boldsymbol{A} = \frac{\partial \boldsymbol{R}(\boldsymbol{v}_{(j)})}{\partial \boldsymbol{v}} = \frac{1}{\Delta t}\boldsymbol{M} + \gamma \frac{\partial \boldsymbol{f}^{\text{int}}}{\partial \boldsymbol{u}} \cdot \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{v}} \tag{18}$$

is the Jacobian matrix,

$$\frac{\partial \boldsymbol{f}^{\text{int}}}{\partial \boldsymbol{u}} = \boldsymbol{K}^{\text{mat}} + \boldsymbol{K}^{\text{geo}}, \tag{19}$$

$\boldsymbol{K}^{\mathrm{mat}}$ and $\boldsymbol{K}^{\mathrm{geo}}$ are the material and geometric tangential stiffness matrices [30], respectively, whose (I, J) blocks, $\boldsymbol{K}_{IJ}^{\mathrm{mat}}$ and $\boldsymbol{K}_{IJ}^{\mathrm{geo}}$, are given by

$$\boldsymbol{K}_{IJ}^{\mathrm{mat}} = \sum_{p} V_p \boldsymbol{B}_{Ip}^{\mathrm{T}} \boldsymbol{D}_p \boldsymbol{B}_{Jp} \tag{20}$$

$$\boldsymbol{K}_{IJ}^{\mathrm{geo}} = H_{IJ} \boldsymbol{I}_{3\times 3} \tag{21}$$

where V_p represents the volume of the material point p, $\boldsymbol{D}_p$ denotes the elastic–plastic tensor in the Voigt notation, $\boldsymbol{B}_{Ip}$ signifies the gradients of the Ith node's shape functions evaluated at particle p. $\boldsymbol{I}_{3\times 3}$ is a 3 ×3 identity matrix, and

$$H_{IJ} = \sum_{p} \left(\frac{\partial N_I}{\partial \boldsymbol{x}} \right)^{\mathrm{T}} \boldsymbol{\sigma}_p \frac{\partial N_J}{\partial \boldsymbol{x}} \tag{22}$$

is a scalar.

According to the Newmark method,

$$\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{v}} = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{a}} \cdot \frac{\partial \boldsymbol{a}}{\partial \boldsymbol{v}} = \beta \Delta t^2 \boldsymbol{I} \cdot \frac{1}{\gamma \Delta t} \boldsymbol{I} = \frac{\beta}{\gamma} \Delta t \boldsymbol{I} \tag{23}$$

where $\boldsymbol{I}$ is an identity matrix. Substituting Eqs. (19) and (23) into Eq. (18) gives

$$\boldsymbol{A} = \frac{1}{\Delta t} \boldsymbol{M} + \beta \Delta t (\boldsymbol{K}^{\mathrm{mat}} + \boldsymbol{K}^{\mathrm{geo}}) \tag{24}$$

Solving the linearized equations (17) for $\Delta \boldsymbol{v}_{(j)}$, the solution is updated by

$$\boldsymbol{v}_{(j+1)} = \boldsymbol{v}_{(j)} + \Delta \boldsymbol{v}_{(j)} \tag{25}$$

The Newton iterations are continued until the following criterion is satisfied:

$$\|\boldsymbol{R}\|_{\infty} \leqslant \varepsilon_{\mathrm{R}} \tag{26}$$

where ε_{R} is a user-defined value for the convergence criterion and $\|\cdot\|_{\infty}$ represents the infinite norm. After convergence is achieved, the solution is accepted as $\boldsymbol{v}^{n+1} = \boldsymbol{v}_{(j+1)}$.

The most popular methods to solve the linear equations are direct methods (such as LU decomposition) and iterative methods (such as CG and GMRES). Nair and Roy [17] attempted an iterative solution of the linear system at each time step, and if it failed, the direct solver was used. Sulsky and Kaul [15] suggested a matrix-free implicit time-stepping strategy for the MPM similar to the Newton–Krylov solution technique. However, the iterative solvers are not as robust as direct solvers and very likely to fail if the matrix is ill-conditioned. In this paper, the Intel MKL PARDISO solver, which is a high-performance, robust, memory efficient, and easy to use software package for solving large sparse linear systems of equations on shared memory multiprocessors [31], is employed with the compressed sparse row (CSR) format. The CSR storage format makes absolutely no assumptions about the sparsity structure of the matrix and stores only nonzero elements, so it greatly reduces the storage demand for the Jacobian matrix [29].

After Eq. (14) is solved, the velocity and position of the particles are updated by

$$\boldsymbol{v}_p^{n+1} = \boldsymbol{v}_p^n + \Delta t \sum_{I=1}^{N_g} N_{Ip} [(1-\gamma) \boldsymbol{a}_I^n + \gamma \boldsymbol{a}_I^{n+1}] \tag{27}$$

$$\boldsymbol{x}_p^{n+1} = \boldsymbol{x}_p^n + \Delta t \sum_{I=1}^{N_g} N_{Ip} \left\{ \boldsymbol{v}_I^n + \frac{\Delta t}{2} [(1-2\beta) \boldsymbol{a}_I^n + 2\beta \boldsymbol{a}_I^{n+1}] \right\}. \tag{28}$$

3. Contact conditions and augmented Lagrange weak form

3.1. *The normal contact condition [32]*

Assume that two bodies come into contact, as shown in Fig. 1. In that case, the non-penetration condition is given by

$$\gamma_{\mathrm{n}} = \boldsymbol{n}^{\mathrm{r}} \cdot \left(\boldsymbol{v}^{\mathrm{r}} - \boldsymbol{v}^{\mathrm{s}} \right) \leqslant 0 \qquad \text{on } \mathrm{A}_c \tag{29}$$

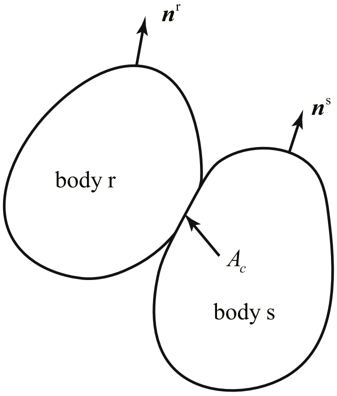


Fig. 1. Two contact bodies.

where γ_n denotes the normal relative velocity between the body r and body s, and $\boldsymbol{n}^{\mathrm{r}} = -\boldsymbol{n}^{\mathrm{s}}$ is the local normal to the contact surface, A_c denotes the contact surface between the bodies r and s. Eq. (29) is necessary, but not sufficient. In MPM models with this criterion it is easy to see contact occurring before surface is actually touching. Additional contact detection algorithms to avoid the contact occurring earlier than the actual time have been proposed by Ma et al. [23] and Nairn [24]. In MPM, the normal vector is calculated by using the mass gradients [20] or the volume gradients on each grid point in the contact detection process. It does not matter which is used if all particle has the same density, but contact is usually between two different materials and they may have different densities. Nairn [24] pointed out that mass gradients must be replaced by volume gradients to handle different materials correctly. In this paper, we use the mass gradient as follows

$$\boldsymbol{n}_I = \sum_{p}^{n_p} m_p \frac{\partial N_{Ip}}{\partial \boldsymbol{x}}. \tag{30}$$

The slip can be described as

$$\boldsymbol{\gamma}_{\mathrm{t}} = (\gamma_\alpha, \gamma_\beta)^{\mathrm{T}} = \left(\boldsymbol{t}_\alpha^{\mathrm{r}} \cdot \left(\boldsymbol{v}^{\mathrm{r}} - \boldsymbol{v}^{\mathrm{s}}\right), \quad \boldsymbol{t}_\beta^{\mathrm{r}} \cdot \left(\boldsymbol{v}^{\mathrm{r}} - \boldsymbol{v}^{\mathrm{s}}\right)\right)^{\mathrm{T}} \tag{31}$$

where $\boldsymbol{t}_\alpha^{\mathrm{r}}$ and $\boldsymbol{t}_\beta^{\mathrm{r}}$ represent any two tangential unit vectors orthogonal with each other.

According to Newton's third law, the contact force $\boldsymbol{f}^{\mathrm{r}}$ and $\boldsymbol{f}^{\mathrm{s}}$ at the surface should satisfy

$$\boldsymbol{f}^{\mathrm{r}} + \boldsymbol{f}^{\mathrm{s}} = 0 \tag{32}$$

which equates with

$$p_{\mathrm{n}}^{\mathrm{r}} - p_{\mathrm{n}}^{\mathrm{s}} = 0 \tag{33}$$

$$\boldsymbol{f}_{\mathrm{t}}^{\mathrm{r}} + \boldsymbol{f}_{\mathrm{t}}^{\mathrm{s}} = 0 \tag{34}$$

where p_n and $\boldsymbol{f}_t$ signify the normal contact pressure and tangential contact force, respectively. The normal contact pressure should be positive, thus

$$p_{\mathrm{n}}^{\mathrm{r}} = -\boldsymbol{f}^{\mathrm{r}} \cdot \boldsymbol{n}^{\mathrm{r}} = p_{\mathrm{n}}^{\mathrm{s}} \geqslant 0. \tag{35}$$

Summarizing Eqs. (29) and (35), we can obtain the non-penetration, non-tension and complementarity conditions:

$$\gamma_{\mathrm{n}} \leqslant 0, \;\; p_{\mathrm{n}} \geqslant 0, \;\; \gamma_{\mathrm{n}} p_{\mathrm{n}} = 0 \qquad \text{on } \mathrm{A_c}. \tag{36}$$

3.2. The frictional contact condition [32,20]

The interfacial behavior related to the frictional response is very important in most engineering problems. Despite the friction has been investigated for a long period, many frictional phenomena have not been fully understood yet. In this article, we restrict ourselves to simple formulations which yield constitutive equations for dry friction. The most widely used constitutive equation is the classical law of Coulomb, which is

$$\|\boldsymbol{f}_{\mathrm{t}}\| \leqslant \mu p_{\mathrm{n}} \tag{37}$$

where μ is the friction coefficient. The complementarity condition states that either the friction stress is situated on the boundary of Coulomb's cone or the slip velocity is zero. Therefore, the slip, stick, and complementarity conditions are summarized as follows

$$\|\boldsymbol{\gamma}_{\mathrm{t}}\| \geqslant 0, \;\; \|\boldsymbol{f}_{\mathrm{t}}\| \leqslant \mu p_{\mathrm{n}}, \;\; \|\boldsymbol{\gamma}_{\mathrm{t}}\|(\|\boldsymbol{f}_{\mathrm{t}}\| - \mu p_{\mathrm{n}}) = 0 \qquad \text{on } \mathrm{A}_{\mathrm{c}}. \tag{38}$$

3.3. *The augmented Lagrange weak form*

The Lagrange multiplier (LM), penalty, and augmented Lagrange multiplier (ALM) methods can be used to solve the problem (6) with constraints (36) and (38), These methods convert a constrained problem to an unconstrained problem by adding constraints to the original variational function. The penalty method is the simplest and do not introduce extra unknown variables, but all constraints will not be exactly satisfied unless the penalty parameter is infinity. However, too large penalty will result in an ill-conditioned problem, which makes the convergence more difficult. Compared with the penalty method, the LM method can exactly satisfy the constraints but introduce extra variables as well. However, this method requires the original function to be strongly convex. Thus, the convergence of LM method may be much harder than the penalty method. The ALM method avoids the disadvantages of the LM and penalty method with the constraints exactly satisfied. The ALM method does not require the strong convexity or the penalty parameter to be infinity (just larger than a certain positive value), and it is much more robust than the LM method and penalty method.

The augmented Lagrange method modifies the original functional $\dot{w}$ by adding terms related to the constraints (36) and (38), namely, the modified functional $\dot{\Pi}$ is given by [33]

$$\dot{\Pi} = \dot{w} + \dot{w}_{\mathrm{n}} + \dot{w}_{\mathrm{t}} \tag{39}$$

where

$$\dot{w}_{\mathrm{n}} = \int_{A_c} l_{\mathrm{n}}(\boldsymbol{v}, \lambda_{\mathrm{n}})\mathrm{dA} \tag{40}$$

$$\dot{w}_{\mathrm{t}} = \int_{A_c} l_{\mathrm{t}}(\boldsymbol{v}, \boldsymbol{\lambda}_{\mathrm{t}})\mathrm{dA} \tag{41}$$

with

$$\begin{aligned} l_{\mathrm{n}} &= \lambda_{\mathrm{n}}\gamma_{\mathrm{n}} + \frac{c}{2}\gamma_{\mathrm{n}}^2 - \frac{1}{2c}[\min(0, \hat{\lambda}_{\mathrm{n}})]^2 \\ &= \begin{cases} \left(\lambda_{\mathrm{n}} + \dfrac{c}{2}\gamma_{\mathrm{n}}\right)\gamma_{\mathrm{n}}, & \hat{\lambda}_{\mathrm{n}} \geqslant 0 \quad \text{contact} \\ -\dfrac{1}{2c}\lambda_{\mathrm{n}}^2, & \hat{\lambda}_{\mathrm{n}} < 0 \quad \text{separation} \end{cases} \end{aligned} \tag{42}$$

and

$$\begin{aligned} l_{\mathrm{t}} &= \boldsymbol{\gamma}_{\mathrm{t}}^{\mathrm{T}}\boldsymbol{\lambda}_{\mathrm{t}} + \frac{c}{2}\boldsymbol{\gamma}_{\mathrm{t}}^{\mathrm{T}}\boldsymbol{\gamma}_{\mathrm{t}} - \frac{1}{2c}[\max(0, \|\hat{\boldsymbol{\lambda}}_{\mathrm{t}}\| - \hat{k})]^2 \\ &= \begin{cases} \boldsymbol{\gamma}_{\mathrm{t}}^{\mathrm{T}}\left(\boldsymbol{\lambda}_{\mathrm{t}} + \dfrac{c}{2}\boldsymbol{\gamma}_{\mathrm{t}}\right), & \hat{\lambda}_{\mathrm{n}} \geqslant 0, \|\hat{\boldsymbol{\lambda}}_{\mathrm{t}}\| \leqslant \hat{k} \quad \text{stick contact} \\ -\dfrac{1}{2c}(\|\boldsymbol{\lambda}_{\mathrm{t}}\|^2 - 2\hat{k}\|\hat{\boldsymbol{\lambda}}_{\mathrm{t}}\| + \hat{k}^2), & \hat{\lambda}_{\mathrm{n}} \geqslant 0, \|\hat{\boldsymbol{\lambda}}_{\mathrm{t}}\| > \hat{k} \quad \text{slip contact} \\ -\dfrac{1}{2c}\|\boldsymbol{\lambda}_{\mathrm{t}}\|^2, & \hat{\lambda}_{\mathrm{n}} < 0 \quad \text{separation} \end{cases} \end{aligned} \tag{43}$$

where c denotes the penalty parameter, $\hat{\lambda}_{\mathrm{n}} = \lambda_{\mathrm{n}} + c\gamma_{\mathrm{n}}$ and $\hat{\boldsymbol{\lambda}}_{\mathrm{t}} = \boldsymbol{\lambda}_{\mathrm{t}} + c\boldsymbol{\gamma}_{\mathrm{t}}$ are the augmented Lagrange multipliers, $\hat{k} = \mu\hat{\lambda}_{\mathrm{n}}$, λ_{n} and $\boldsymbol{\lambda}_{\mathrm{t}} = (\lambda_\alpha, \lambda_\beta)^{\mathrm{T}}$ are the Lagrange multipliers, which in the context of contact mechanics can be interpreted as the nominal contact pressure and the nominal tangential friction force. In Eqs. (42) and (43), the first term and the second term is contributed by the Lagrange method and the penalty method, respectively. The third term represents the contact pressure must be positive and the tangential force must be not greater than the maximum frictional force. The third term makes the modified functional C^1 differentiable at the critical point $\hat{\lambda}_n = 0$ and $\|\hat{\boldsymbol{\lambda}}_t\| = \hat{k}$, which is very important for the Newton-like method.

The stationary condition of the modified functional $\dot{\Pi}$ gives

$$\delta\dot{w} + \int_{A_c} \delta\boldsymbol{v}^{\mathrm{T}} \frac{\partial l_n}{\partial \boldsymbol{v}} \mathrm{dA} + \int_{A_c} \delta\lambda_n \frac{\partial l_n}{\partial \lambda_n} \mathrm{dA} + \int_{A_c} \delta\boldsymbol{v}^{\mathrm{T}} \frac{\partial l_t}{\partial \boldsymbol{v}} \mathrm{dA} + \int_{A_c} \delta\boldsymbol{\lambda}_t^{\mathrm{T}} \frac{\partial l_t}{\partial \boldsymbol{\lambda}_t} \mathrm{dA} = 0 \tag{44}$$

where the partial derivatives of l_n and l_t, depending on the contact and friction status with respect to the target boundary A_c, are as follows:

$$\frac{\partial l_n(\boldsymbol{v}, \lambda_n)}{\partial \boldsymbol{v}} = \begin{cases} \dfrac{\partial \gamma_n}{\partial \boldsymbol{v}} \hat{\lambda}_n, & \hat{\lambda}_n \geqslant 0 \\ 0, & \hat{\lambda}_n < 0 \end{cases} \tag{45}$$

$$\frac{\partial l_n(\boldsymbol{v}, \lambda_n)}{\partial \lambda_n} = \begin{cases} \gamma_n, & \hat{\lambda}_n \geqslant 0 \\ -\dfrac{1}{c} \lambda_n, & \hat{\lambda}_n < 0 \end{cases} \tag{46}$$

$$\frac{\partial l_t(\boldsymbol{v}, \boldsymbol{\lambda}_t)}{\partial \boldsymbol{v}} = \begin{cases} \dfrac{\partial \boldsymbol{\gamma}_t^{\mathrm{T}}}{\partial \boldsymbol{v}} \hat{\boldsymbol{\lambda}}_t, & \hat{\lambda}_n \geqslant 0, \|\hat{\boldsymbol{\lambda}}_t\| \leqslant \hat{k} \\ \dfrac{\hat{k}}{\|\hat{\boldsymbol{\lambda}}_t\|} \dfrac{\partial \boldsymbol{\gamma}_t^{\mathrm{T}}}{\partial \boldsymbol{v}} \hat{\boldsymbol{\lambda}}_t, & \hat{\lambda}_n \geqslant 0, \|\hat{\boldsymbol{\lambda}}_t\| > \hat{k} \\ 0, & \hat{\lambda}_n < 0 \end{cases} \tag{47}$$

$$\frac{\partial l_t(\boldsymbol{v}, \boldsymbol{\lambda}_t)}{\partial \boldsymbol{\lambda}_t} = \begin{cases} \boldsymbol{\gamma}_t, & \hat{\lambda}_n \geqslant 0, \|\hat{\boldsymbol{\lambda}}_t\| \leqslant \hat{k} \\ -\dfrac{1}{c} \left(\boldsymbol{\lambda}_t - \hat{k} \dfrac{\hat{\boldsymbol{\lambda}}_t}{\|\hat{\boldsymbol{\lambda}}_t\|} \right), & \hat{\lambda}_n \geqslant 0, \|\hat{\boldsymbol{\lambda}}_t\| > \hat{k} \\ -\dfrac{1}{c} \boldsymbol{\lambda}_t, & \hat{\lambda}_n < 0. \end{cases} \tag{48}$$

3.4. Equivalence between the Augmented Lagrange weak form and the strong form

The AL weak form Eq. (44) was established from the strong form (2) with the natural boundary condition $(4)_1$ and contact conditions (32), (36) and (38) [34]. To show the equivalence between the AL weak form and the strong form, we will derive the strong form from the AL weak form.

Integrating the first term in Eq. (44) by parts gives

$$\begin{aligned} &\int_{\Omega} \delta\boldsymbol{v}^{\mathrm{T}} (\rho\dot{\boldsymbol{v}} - \nabla \cdot \boldsymbol{\sigma} - \boldsymbol{f}) \mathrm{d}\Omega + \int_{\Gamma_t} \delta\boldsymbol{v}^{\mathrm{T}} (\boldsymbol{\sigma} \cdot \boldsymbol{n} - \bar{\boldsymbol{t}}) \mathrm{d}\Gamma + \int_{A_c} \delta\boldsymbol{v}^{\mathrm{T}} \left(\boldsymbol{\sigma} \cdot \boldsymbol{n} + \frac{\partial l_n}{\partial \boldsymbol{v}} + \frac{\partial l_t}{\partial \boldsymbol{v}} \right) \mathrm{dA} \\ &\quad + \int_{A_c} \delta\lambda_n \frac{\partial l_n}{\partial \lambda_n} \mathrm{dA} + \int_{A_c} \delta\boldsymbol{\lambda}_t^{\mathrm{T}} \frac{\partial l_t}{\partial \boldsymbol{\lambda}_t} \mathrm{dA} = 0. \end{aligned} \tag{49}$$

Invoking the arbitrariness of $\delta\boldsymbol{v}$ and letting the first and second terms of Eq. (49) be zero lead to Eq. (2) and the natural boundary condition $(4)_1$.

The third term in Eq. (49) can be rewritten as

$$\int_{A_c} \delta\boldsymbol{v}^{\mathrm{T}} \left(\boldsymbol{\sigma} \cdot \boldsymbol{n} + \frac{\partial l_n}{\partial \boldsymbol{v}} + \frac{\partial l_t}{\partial \boldsymbol{v}} \right) \mathrm{dA} = \int_{A_c} \delta\boldsymbol{v}^r \cdot (\boldsymbol{f}^r - \boldsymbol{f}^c) \mathrm{dA} + \int_{A_c} \delta\boldsymbol{v}^s \cdot (\boldsymbol{f}^s + \boldsymbol{f}^c) \mathrm{dA} \tag{50}$$

where $\boldsymbol{f}^r = \boldsymbol{\sigma}^r \cdot \boldsymbol{n}^r, \boldsymbol{f}^s = \boldsymbol{\sigma}^s \cdot \boldsymbol{n}^s$ and $\boldsymbol{f}^c = -\left(\frac{\partial l_n}{\partial \boldsymbol{v}} + \frac{\partial l_t}{\partial \boldsymbol{v}} \right)$. Invoking the arbitrariness of $\delta\boldsymbol{v}$ on A_c, we can obtain

$$\boldsymbol{f}^r = -\boldsymbol{f}^s = \boldsymbol{f}^c \tag{51}$$

which is equivalent to Eq. (32).

Next we will examine the fourth term in Eq. (49). In the case of $\hat{\lambda}_{\rm n} = \lambda_{\rm n} + c\gamma_{\rm n} \geqslant 0$, the fourth term can be reduced by invoking the arbitrariness of $\delta\lambda_{\rm n}$ and Eq. (46) to $\int_{A_c} \gamma_{\rm n} {\rm dA} = 0$ so that $\gamma_{\rm n} = 0$. Substituting $\gamma_{\rm n} = 0$ into $\hat{\lambda}_{\rm n} \geqslant 0$ results in $\lambda_{\rm n} = \hat{\lambda}_{\rm n} \geqslant 0$, which means that the contact has occurred. Similarly, in the case of $\hat{\lambda}_{\rm n} = \lambda_{\rm n} + c\gamma_{\rm n} < 0$, the fourth term gives $\lambda_{\rm n} = 0$. Substituting $\lambda_{\rm n} = 0$ into $\hat{\lambda}_{\rm n} < 0$ results in $\gamma_{\rm n} < 0$, which means the separation of the bodies. In both cases, $\gamma_{\rm n}\lambda_{\rm n} = 0$ is valid, which is equivalent to Eq. (36).

Finally, we will examine the fifth term in Eq. (49). In the case of $\hat{\lambda}_{\rm n} \geqslant 0$ and $\|\hat{\boldsymbol{\lambda}}_{\rm t}\| \leqslant \hat{k}$, the fifth term can be reduced by invoking the arbitrariness of $\delta\boldsymbol{\lambda}_{\rm t}$ and Eq. (48) to $\int_{A_c} \boldsymbol{\gamma}_{\rm t} {\rm dA} = 0$, so that $\boldsymbol{\gamma}_{\rm t} = 0$. Substituting $\boldsymbol{\gamma}_{\rm t} = 0$ into $\|\hat{\boldsymbol{\lambda}}_{\rm t}\| \leqslant \hat{k}$ results in $\|\boldsymbol{\lambda}_{\rm t}\| \leqslant \mu\lambda_{\rm n}$, which means that the tangent force is smaller than the maximum frictional force (stick contact). Similarly, in the case of $\hat{\lambda}_{\rm n} \geqslant 0$ and $\|\hat{\boldsymbol{\lambda}}_{\rm t}\| > \hat{k}$, the fifth term can be reduced to $\int_{A_c} \frac{1}{c}(\boldsymbol{\lambda}_{\rm t} - \hat{k}\frac{\hat{\boldsymbol{\lambda}}_{\rm t}}{\|\hat{\boldsymbol{\lambda}}_{\rm t}\|}){\rm dA} = 0$ so that $\|\boldsymbol{\lambda}_{\rm t}\| = \mu\hat{\lambda}_{\rm n} = \mu\lambda_{\rm n}$, which means that the tangent force is equal to the maximum frictional force (slip contact). In the case of $\hat{\lambda}_{\rm n} < 0$, the fifth term gives $\boldsymbol{\lambda}_{\rm t} = 0$, which means that the tangent force is zero (separation). In all cases, $\|\boldsymbol{\gamma}_{\rm t}\|(\|\boldsymbol{\lambda}_{\rm t}\| - \mu\lambda_{\rm n}) = 0$ is valid, which is equivalent to Eq. (38).

Thus, the AL weak form is equivalent to the strong form because the weak form can be derived from the strong form, and vice versa.

4. Augmented Lagrangian contact algorithm in the implicit MPM framework

So far, the weak form of continuum contact mechanics has been established. This section mainly establishes the numerical scheme for solving the contact weak form in the implicit MPM framework. A local multi-mesh algorithm is employed to describe the velocity fields of contact bodies [23].

4.1. The augmented Lagrangian contact MPM framework

The Lagrange multiplier field need not to be continuous because its derivatives of $\boldsymbol{x}$ do not appear in the weak form Eq. (44). Let the Lagrange multiplier trial function $\boldsymbol{\lambda}(\boldsymbol{x}, t) = (\lambda_{\rm n}, \lambda_{\alpha}, \lambda_{\beta})^{\rm T}$ be in the following space:

$$\boldsymbol{\lambda}(\boldsymbol{x}, t) \in j,\ j = \left\{\boldsymbol{\lambda}(\boldsymbol{x}, t) \mid \lambda \in {\rm C}^{-1},\ \text{on } {\rm A_c}\right\}. \tag{52}$$

In the MPM, the momentum equations are solved on the background grid. The contact area $A_{\rm c}$ can be tracked by searching the grid point which is occupied by two different bodies. Thus, the Lagrange multiplier field can be discretized on the background grid as

$$\boldsymbol{\lambda}(\boldsymbol{x}, t) = \sum_{I\in A_c} \boldsymbol{\lambda}_I(t)\, \Lambda_I(\boldsymbol{x}) \tag{53}$$

where $\Lambda_I(\boldsymbol{x})$ denotes the shape function associated with the grid point I, which is chosen as the Dirac delta function, namely,

$$\Lambda_I(\boldsymbol{x}) = \delta(\boldsymbol{x} - \boldsymbol{x}_I). \tag{54}$$

Similarly, test functions are approximated by

$$\delta\boldsymbol{\lambda} = \sum_{I\in A_c} \Lambda_I(\boldsymbol{x})\, \delta\boldsymbol{\lambda}_I,\ \text{on } {\rm A_c} \tag{55}$$

$$\delta\boldsymbol{v} = \begin{cases} \sum \Lambda_I(\boldsymbol{x})\, \delta\boldsymbol{v}_I & \text{on } {\rm A_c} \\ \sum N_I(\boldsymbol{x})\, \delta\boldsymbol{v}_I & \text{in } \Omega. \end{cases} \tag{56}$$

Substituting test functions (55) and (56) into Eqs. (44), and invoking the arbitrariness of $\delta\boldsymbol{v}_I$ and $\delta\boldsymbol{\lambda}_I$ lead to

$$\dot{\boldsymbol{p}}_I^b = \boldsymbol{f}_I^{b,\rm ext} + \boldsymbol{f}_I^{b,\rm int} + \boldsymbol{f}_I^{b,\rm c}, \quad b = {\rm r, s} \tag{57}$$

$$C_I^{\rm nor} = 0 \tag{58}$$

$$\boldsymbol{C}_I^{\rm tan} = 0 \tag{59}$$

where

$$\boldsymbol{f}_I^{b,\rm c} = \boldsymbol{f}_I^{b,\rm nor} + \boldsymbol{f}_I^{b,\rm tan} \tag{60}$$

is the contact force. Note that

$$\frac{\partial \gamma_{\mathrm{n}}}{\partial \boldsymbol{v}^b} = \begin{cases} \boldsymbol{n}^{\mathrm{r}} & b = \mathrm{r} \\ -\boldsymbol{n}^{\mathrm{r}} & b = \mathrm{s} \end{cases} \tag{61}$$

$$\frac{\partial \boldsymbol{\gamma}_{\mathrm{t}}^{\mathrm{T}}}{\partial \boldsymbol{v}^b} = \begin{cases} (\boldsymbol{t}_{\alpha}^{\mathrm{r}}, \quad \boldsymbol{t}_{\beta}^{\mathrm{r}}) & b = \mathrm{r} \\ -(\boldsymbol{t}_{\alpha}^{\mathrm{r}}, \quad \boldsymbol{t}_{\beta}^{\mathrm{r}}) & b = \mathrm{s} \end{cases} \tag{62}$$

thus we have

$$\boldsymbol{f}_I^{b,\mathrm{nor}} = \begin{cases} -\hat{\lambda}_{I\mathrm{n}} \boldsymbol{n}_I^b, & \hat{\lambda}_{I\mathrm{n}} \geqslant 0 \\ 0, & \hat{\lambda}_{I\mathrm{n}} < 0 \end{cases} \tag{63}$$

$$\boldsymbol{f}_I^{b,\mathrm{tan}} = \begin{cases} -(\boldsymbol{t}_{I\alpha}^b \hat{\lambda}_{I\alpha} + \boldsymbol{t}_{I\beta}^b \hat{\lambda}_{I\beta}), & \hat{\lambda}_{I\mathrm{n}} \geqslant 0, \|\hat{\boldsymbol{\lambda}}_{I\mathrm{t}}\| \leqslant \hat{k}_I \\ -\dfrac{\hat{k}_I}{\|\hat{\boldsymbol{\lambda}}_{I\mathrm{t}}\|}(\boldsymbol{t}_{I\alpha}^b \hat{\lambda}_{I\alpha} + \boldsymbol{t}_{I\beta}^b \hat{\lambda}_{I\beta}), & \hat{\lambda}_{I\mathrm{n}} \geqslant 0, \|\hat{\boldsymbol{\lambda}}_{I\mathrm{t}}\| > \hat{k}_I \\ 0, & \hat{\lambda}_{I\mathrm{n}} < 0 \end{cases} \tag{64}$$

represent the normal contact force and the tangential contact force of grid point I, respectively.

$$C_I^{\mathrm{nor}} = \begin{cases} \gamma_{I\mathrm{n}}, & \hat{\lambda}_{I\mathrm{n}} \geqslant 0 \\ -\dfrac{1}{c}\lambda_{I\mathrm{n}}, & \hat{\lambda}_{I\mathrm{n}} < 0 \end{cases} \tag{65}$$

and

$$\boldsymbol{C}_I^{\mathrm{tan}} = \begin{cases} \boldsymbol{\gamma}_{I\mathrm{t}}, & \hat{\lambda}_{I\mathrm{n}} \geqslant 0, \|\hat{\boldsymbol{\lambda}}_{I\mathrm{t}}\| \leqslant \hat{k}_I \\ -\dfrac{1}{c}\left(\boldsymbol{\lambda}_{I\mathrm{t}} - \hat{k}_I \dfrac{\hat{\boldsymbol{\lambda}}_{I\mathrm{t}}}{\|\hat{\boldsymbol{\lambda}}_{I\mathrm{t}}\|}\right), & \hat{\lambda}_{I\mathrm{n}} \geqslant 0, \|\hat{\boldsymbol{\lambda}}_{I\mathrm{t}}\| > \hat{k}_I \\ -\dfrac{1}{c}\boldsymbol{\lambda}_{I\mathrm{t}}, & \hat{\lambda}_{I\mathrm{n}} < 0 \end{cases} \tag{66}$$

denote the constrained equations at grid point I, where

$$\gamma_{I\mathrm{n}} = \boldsymbol{n}_I^r \cdot (\boldsymbol{v}_I^r - \boldsymbol{v}_I^s) \tag{67}$$

$$\boldsymbol{\gamma}_{I\mathrm{t}} = (\boldsymbol{t}_{I\alpha}^{\mathrm{r}} \cdot (\boldsymbol{v}_I^r - \boldsymbol{v}_I^s), \quad \boldsymbol{t}_{I\beta}^{\mathrm{r}} \cdot (\boldsymbol{v}_I^r - \boldsymbol{v}_I^s))^{\mathrm{T}}. \tag{68}$$

4.2. *The implicit time integration in contact simulations*

In contact simulations, the acceleration is contributed by not only the internal and external force, but also the contact force. Because the old background grid is discarded, the internal, external and contact force of the last time step (nth time step) are missed. Thus it is necessary to recover $\boldsymbol{f}_I^{\mathrm{int},n}$ and $\boldsymbol{f}_I^{\mathrm{ext},n}$ by Eqs. (10) and (11), mapping to the new background grid. However, recovering the contact force is difficult because the Lagrange multiplier $\boldsymbol{\lambda}^n$ is never stored in particles. Thus, an approximated formulation is employed as follows:

$$\boldsymbol{a}_I^n = (\boldsymbol{f}_I^{\mathrm{ext},n} + \boldsymbol{f}_I^{\mathrm{int},n} + \boldsymbol{f}_I^{\mathrm{c},n+1})/m_I \tag{69}$$

$$\boldsymbol{a}_I^{n+1} = (\boldsymbol{f}_I^{\mathrm{ext},n+1} + \boldsymbol{f}_I^{\mathrm{int},n+1} + \boldsymbol{f}_I^{\mathrm{c},n+1})/m_I \tag{70}$$

where the third part of $\boldsymbol{a}_I^n$ is $\boldsymbol{f}_I^{\mathrm{c},n+1}$ instead of $\boldsymbol{f}_I^{\mathrm{c},n}$. Substituting Eqs. (69) and (70) into Eq. (12) yields

$$\boldsymbol{R}(\boldsymbol{v}^{n+1}, \boldsymbol{\lambda}^{n+1}) = \frac{1}{\Delta t}\boldsymbol{M}(\boldsymbol{v}^{n+1} - \tilde{\boldsymbol{v}}^n) - \gamma \boldsymbol{f}^{\mathrm{int}}(\boldsymbol{v}^{n+1}, t^{n+1}) - \boldsymbol{f}^{\mathrm{c}}(\boldsymbol{v}^{n+1}, \boldsymbol{\lambda}^{n+1}) \tag{71}$$

After solving Eq. (71) for $\boldsymbol{v}^{n+1}$ and $\boldsymbol{\lambda}^{n+1}$, the velocity and position of the particles are updated by Eqs. (27) and (28).

4.3. Solution of the nonlinear contact equations

4.3.1. The Uzawa algorithm

The contact equations (57)–(59) are similar to the saddle point linear system,

$$\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B}^{\mathrm{T}} \\ \boldsymbol{B} & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \boldsymbol{b} \\ 0 \end{pmatrix} \tag{72}$$

where $\boldsymbol{A}$ is a positive definite matrix, $\boldsymbol{B}$ represents the constraint matrix, $\boldsymbol{x}$ is the unknown variables, $\boldsymbol{\lambda}$ signifies the multipliers. The saddle point problems are usually indefinite and highly ill-conditioned. All these features bring great challenges to the Newton-like methods, which require the Hessen matrix to be positive definite. In numerical mathematics, the Uzawa iteration [27,28] is an effective algorithm for solving these saddle point problems. By implementing the Uzawa algorithm, the ill-condition number of the original problem is dramatically reduced and the positive definiteness of $\boldsymbol{A}$ can be reserved. In addition, the Uzawa algorithm decouples the unknown variables and the multipliers, which simplifies the solving process.

The standard iterative process of the Uzawa algorithm is given by [28]

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + (\boldsymbol{A} + c\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})^{-1}(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{B}^{\mathrm{T}}\boldsymbol{\lambda}^k) \tag{73}$$

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + c\boldsymbol{B}\boldsymbol{x}^{k+1} \tag{74}$$

where c denotes the penalty parameter and k represents the iteration number. The iteration is implemented by two steps to update $\boldsymbol{x}$ and $\boldsymbol{\lambda}$, respectively.

Similarly, the Uzawa algorithm for frictional contact problems is implemented by two steps

$$\boldsymbol{R}(\boldsymbol{v}^{k+1}, \boldsymbol{\lambda}^k) = 0 \tag{75}$$

$$\begin{aligned} \lambda_{\mathrm{n}}^{k+1} &= \max(0, \lambda_{\mathrm{n}}^k + c^k \gamma_{\mathrm{n}}^{k+1}) \\ \boldsymbol{\lambda}_{\mathrm{t}}^{k+1} &= \begin{cases} (\boldsymbol{\lambda}_{\mathrm{t}}^k + c^k \boldsymbol{\gamma}_{\mathrm{t}}^{k+1}) \min\left(1, \dfrac{\hat{k}}{\|\boldsymbol{\lambda}_{\mathrm{t}}^k + c^k \boldsymbol{\gamma}_{\mathrm{t}}^{k+1}\|}\right), & \hat{\lambda}_{\mathrm{n}} \geqslant 0 \\ 0, & \hat{\lambda}_{\mathrm{n}} < 0 \end{cases} \end{aligned} \tag{76}$$

where the superscript $n+1$ is omitted for clarity.

The first step solves Eq. (75) for velocity $\boldsymbol{v}^{k+1}$ by taking the multiplier $\boldsymbol{\lambda}^k$ obtained from the last iteration as fixed. The nonlinear equation (75) can be solved by using the Newton method, which will be presented in Section 4.3.2. After that, the updated velocity $\boldsymbol{v}^{k+1}$ will be used to update the multiplier $\boldsymbol{\lambda}^{k+1}$ in the second step.

The convergence behavior of the iteration does strongly depends upon the penalty parameter. Good choice of the penalty makes the solution converged more quickly. Thus, it makes sense to increase the penalty parameter according to an update scheme, namely

$$c^{k+1} = \begin{cases} \alpha c^k, & \|\boldsymbol{C}(\boldsymbol{v}^{k+1})\|_\infty > 0.25\|\boldsymbol{C}(\boldsymbol{v}^k)\|_\infty \text{ or } \|\boldsymbol{R}(\boldsymbol{v}^{k+1}, \boldsymbol{\lambda}^{k+1})\|_\infty > 0.25\|\boldsymbol{R}(\boldsymbol{v}^k, \boldsymbol{\lambda}^k)\|_\infty \\ c^k, & \text{otherwise} \end{cases} \tag{77}$$

where $\alpha \in [4, 10]$ represents the magnification factor, and

$$\|\boldsymbol{C}\|_\infty = \max_{I \in A_c}(\|C_I^{\mathrm{nor}}\|, \|\boldsymbol{C}_I^{\tan}\|). \tag{78}$$

The iteration will continue until the following convergence criteria are satisfied

$$\|\boldsymbol{C}\|_\infty \leqslant \epsilon_{\mathrm{C}} \text{ and } \|\boldsymbol{R}\|_\infty \leqslant \epsilon_{\mathrm{R}}. \tag{79}$$

4.3.2. Linearization

For a known fixed Lagrange multiplier $\bar{\boldsymbol{\lambda}} = \boldsymbol{\lambda}^{n+1,k}$, the Newton's method is used to solve the nonlinear problem (75) for the velocity $\boldsymbol{v}^{n+1,k+1}$. For clarity, the superscripts $n+1$ and $k+1$ will be omitted. At the beginning of iterations, a first guess $\boldsymbol{v}_{(0)}$ for the root of the function $\boldsymbol{r}(\boldsymbol{v}, \bar{\boldsymbol{\lambda}})$ is required. This results after linearization in the system of equations at the state $(\boldsymbol{v}_{(j)}, \bar{\boldsymbol{\lambda}})$, i.e.

$$\boldsymbol{R}(\boldsymbol{v}_{(j+1)}, \bar{\boldsymbol{\lambda}}) = \boldsymbol{R}(\boldsymbol{v}_{(j)}, \bar{\boldsymbol{\lambda}}) + \boldsymbol{A}\Delta\boldsymbol{v}_{(j)} = 0 \tag{80}$$

where

$$\boldsymbol{A} = \frac{\partial \boldsymbol{R}(\boldsymbol{v}_{(j)}, \bar{\boldsymbol{\lambda}})}{\partial \boldsymbol{v}} = \frac{1}{\Delta t}\boldsymbol{M} + \beta \Delta t(\boldsymbol{K}^{\mathrm{mat}} + \boldsymbol{K}^{\mathrm{geo}}) + (\boldsymbol{A}^{\mathrm{nor}} + \boldsymbol{A}^{\mathrm{tan}}) \tag{81}$$

is the Jacobian matrix of $\boldsymbol{R}(\boldsymbol{v}, \bar{\boldsymbol{\lambda}})$. Denote $\partial \gamma_{\mathrm{n}}/\partial \boldsymbol{v}$ as $\boldsymbol{n}$, $\partial \boldsymbol{\gamma}_{\mathrm{t}}^{\mathrm{T}}/\partial \boldsymbol{v}$ as $\boldsymbol{t}$, thus

$$\boldsymbol{A}^{\mathrm{nor}} = -\frac{\partial \boldsymbol{f}^{\mathrm{nor}}}{\partial \boldsymbol{v}} = \begin{cases} c\boldsymbol{n}\boldsymbol{n}^{\mathrm{T}}, & \hat{\lambda}_{\mathrm{n}} \geqslant 0 \\ 0, & \hat{\lambda}_{\mathrm{n}} < 0 \end{cases} \tag{82}$$

$$\boldsymbol{A}^{\mathrm{tan}} = -\frac{\partial \boldsymbol{f}^{\mathrm{tan}}}{\partial \boldsymbol{v}} = \begin{cases} c\boldsymbol{t}\boldsymbol{t}^{\mathrm{T}}, & \hat{\lambda}_{\mathrm{n}} \geqslant 0, \, \|\hat{\boldsymbol{\lambda}}_{\mathrm{t}}\| \leqslant \hat{k} \\ \dfrac{c}{\|\hat{\boldsymbol{\lambda}}_{\mathrm{t}}\|}\left[\hat{k}\boldsymbol{t}\boldsymbol{t}^{\mathrm{T}} - \dfrac{\hat{k}}{\|\hat{\boldsymbol{\lambda}}_{\mathrm{t}}\|^2}(\boldsymbol{t}\hat{\boldsymbol{\lambda}}_{\mathrm{t}})(\boldsymbol{t}\hat{\boldsymbol{\lambda}}_{\mathrm{t}})^{\mathrm{T}} + \mu(\boldsymbol{t}\hat{\boldsymbol{\lambda}}_{\mathrm{t}})\boldsymbol{n}^{\mathrm{T}}\right], & \hat{\lambda}_{\mathrm{n}} \geqslant 0, \, \|\hat{\boldsymbol{\lambda}}_{\mathrm{t}}\| > \hat{k} \\ 0 & \hat{\lambda}_{\mathrm{n}} < 0 \end{cases} \tag{83}$$

represents the tangential stiffness matrix of the normal pressure and frictional force, respectively [33]. The direct solver is used to solve the linearized problem, in which the Jacobian is factorized by LU decomposition.

5. Numerical implementation

The detailed implementation of the contact algorithm in a time step can be summarized as follows.

1. Form the lumped mass matrix $\boldsymbol{M}^n$ and nodal force $\boldsymbol{f}_{iI}^n$ by Eqs. (8) and (9) at the beginning of the time step n, respectively. And nodal momentum $\boldsymbol{p}_{iI}^n$ by

$$\boldsymbol{p}_I = \sum_{p=1}^{n_p} S_{Ip} m_p \boldsymbol{v}_p. \tag{84}$$

2. Detect the contact nodes which is occupied by two bodies. Adjust the nodal momentum of the contact nodes by

$$\bar{\boldsymbol{v}}_I^{r,n} = \boldsymbol{v}_I^{r,n} - \frac{m_I^s(\boldsymbol{v}_I^{r,n} - \boldsymbol{v}_I^{s,n})}{m_I^r + m_I^s} \cdot \boldsymbol{n}_I^r \boldsymbol{n}_I^r \tag{85}$$

$$\bar{\boldsymbol{v}}_I^{s,n} = \boldsymbol{v}_I^{s,n} + \frac{m_I^r(\boldsymbol{v}_I^{r,n} - \boldsymbol{v}_I^{s,n})}{m_I^r + m_I^s} \cdot \boldsymbol{n}_I^r \boldsymbol{n}_I^r. \tag{86}$$

Although the contact condition has been imposed at the end of last time step, the penetration may still occur at the beginning of the time step n because a new regular grid is employed. Eqs. (85) and (86) are used to eliminate the penetration in the new grid.

3. Choose an initial value for $\boldsymbol{v}_0$ and $\boldsymbol{\lambda}_0$. For example, $\boldsymbol{\lambda}_0$ is simply set to zero and the velocity of last time step can be chosen as $\boldsymbol{v}_0$.
4. Solve the contact equations (57)–(59) using the Uzawa algorithm: Set $k = 0$. Until the criterion (79) is satisfied, do

   (a) Newton iteration: Set $j = 0$. Until the criterion (26) is satisfied, do

      i. Assembly the material stiffness matrix $\boldsymbol{K}^{\mathrm{mat}}$, the geometric stiffness matrix $\boldsymbol{K}^{\mathrm{geo}}$. Form the normal pressure stiffness matrix $\boldsymbol{A}^{\mathrm{nor}}$ and frictional stiffness matrix $\boldsymbol{A}^{\mathrm{tan}}$ by Eqs. (82) and (83) respectively.
      ii. Calculate the nodal contact force $\boldsymbol{f}_{(j)}^{\mathrm{nor}}$ and $\boldsymbol{f}_{(j)}^{\mathrm{tan}}$ by Eqs. (63) and (64), the nodal internal force $\boldsymbol{f}_{(j)}^{\mathrm{int}}$ by Eq. (10).
      iii. Calculate the residual vector $\boldsymbol{R}(\boldsymbol{v}_{(j)}, \bar{\boldsymbol{\lambda}})$ by Eq. (71).
      iv. Solve the linearized equations (80).
      v. Update the nodal velocity: $\boldsymbol{v}_{(j+1)} = \boldsymbol{v}_{(j)} + \Delta \boldsymbol{v}_{(j)}$.
      vi. Set j to $j + 1$.

   (b) Update the Lagrange multipliers by Eq. (76).
   (c) Set k to $k + 1$.

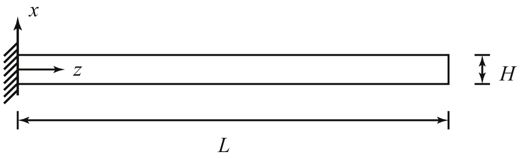


Fig. 2. The cantilever beam.

Table 1
Computational cost of the explicit MPM and implicit MPM.

	Particles	Steps	CPU time (s)
Explicit MPM	10,240	51,577	1521
Implicit MPM	10,240	40	20

5. Update the particle stress $\boldsymbol{\sigma}_p^{n+1}$ by the constitutive equation. Update the particle velocity $\boldsymbol{v}_p^{n+1}$ and position $\boldsymbol{x}_p^{n+1}$ by Eqs. (27) and (28).
6. Discard the deformed background grid.

6. Numerical examples

The corresponding algorithms are implemented in our code MPM3D and extensively tested for both academic and practical examples. To validate the proposed contact algorithm, several numerical examples, including beam vibration, 2D rigid flat punch, cylinder rolling and metal cutting, are investigated in this section. All the simulations in this section are run in a PC with one Intel(R) core(TM) i7-3770 CPU at 3.40 GHz.

6.1. Beam vibration

A cantilever beam is investigated first to study the accuracy and efficiency of the implicit MPM without contact in our codes, compared to the explicit MPM. The cross section of the beam is square with a width of 0.1 m and a height of 0.8 m, and the length of the beam is 16 m, as shown in Fig. 2. The left side of the beam is fixed on the wall and the other side is free. The material of the beam is elastic with a Young's module of 1.0×10^6 Pa, a density of $1.0\ \mathrm{kg/m^3}$ and a Poisson's ratio of 0.3. The gravity is $0.5\ \mathrm{m/s^2}$ in the opposite direction of x axis.

The particle space is 0.05 m and the grid cell size is 0.1 m. The total number of particles is 10,240. In the explicit MPM, the time step size is chosen as $7.8 \times 1.0\mathrm{e}^{-5}$ s which is limited by the CFL condition, while in the implicit MPM, the time step size is chosen as 0.1. The residual error tolerance is chosen as $\epsilon_{\mathrm{R}} = 1.0 \times 10^{-7}$. The Newmark parameter $\gamma = 0.5$ and $\beta = 0.25$ are used.

Figs. 3 and 4 plot the time history of the deflection at the right upper corner of the beam and the total energy, obtained by the implicit MPM and the explicit MUSL MPM, respectively. In MUSL scheme, the updated particle momenta are extrapolated to the grid again before calculating the nodal velocities [2]. The Newmark scheme predicts a consistent result compared with the explicit MUSL MPM with a slight period delay. The energy is conserved very well in the explicit MPM, while the Newmark scheme has a little energy loss as time advances. However, the back-Euler scheme gives severely decayed displacements, and suffers a great energy loss. The cost of the implicit MPM is compared with that of the explicit MPM in Table 1, which shows that the CPU time required by the implicit MPM to finish the simulation is only about 1.3% of that required by the explicit MPM. In this simulation, the average number of Newton iterations in each time step is 2.

6.2. 2D rigid flat punch

A 2D rigid flat punch is pressed by an axial force $P = 1$ N against an elastic half-space, as shown in Fig. 5. For the elastic half-space, the Young's modulus $E_{\mathrm{h}} = 1.0 \times 10^3$ MPa and Poisson's ratio $\upsilon_{\mathrm{h}} = 0.3$, while for the flat punch, the Young's modulus $E_{\mathrm{f}} = 1.0 \times 10^8$ MPa, Poisson's ratio $\upsilon_{\mathrm{f}} = 0.3$, and $a = 1$ mm.

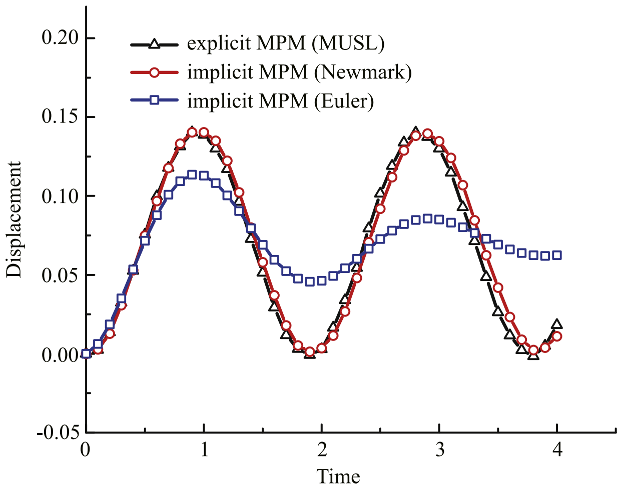


Fig. 3. Time histories of the deflection at the right upper corner of the beam obtained by the explicit MPM and implicit MPM.

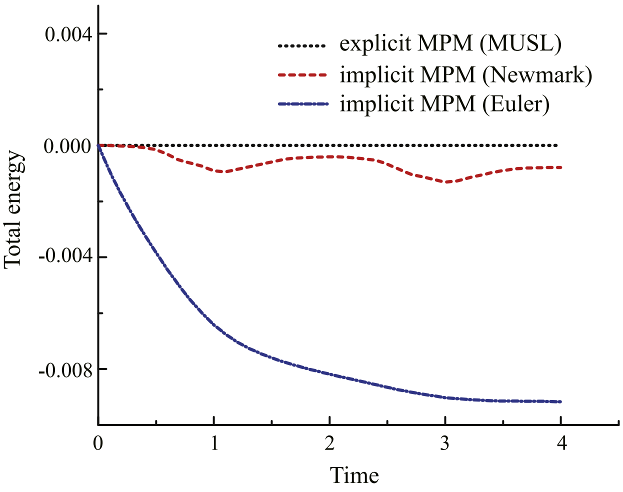


Fig. 4. Time histories of the total energy obtained by the explicit MPM and implicit MPM.

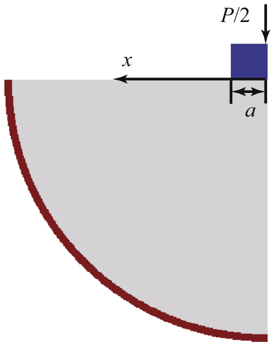
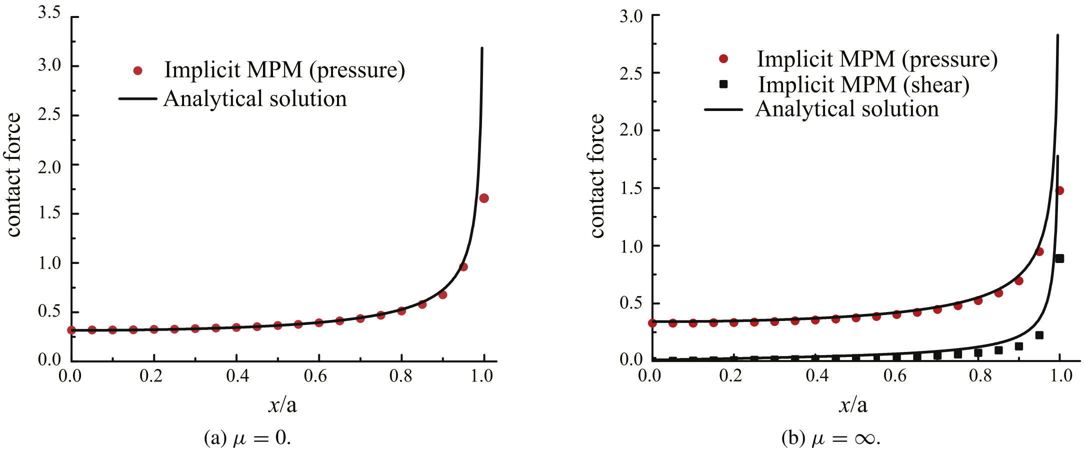


Fig. 5. 2D rigid punch indentation problem.



(a) $\mu = 0$.

(b) $\mu = \infty$.

Fig. 6. Contact pressure distributions: (a) $\mu = 0$; (b) $\mu = \infty$.

The analytical solution [35] of the contact pressure along the x-direction for the frictionless case ($\mu = 0$) is given as

$$p\left(x\right) = \frac{P}{\pi\sqrt{a^2 - x^2}}, \qquad 0 < x < a$$

while the contact pressure and shear for the adhesive case ($\mu = \infty$) is given by

$$p\left(x\right) = \frac{2(1-v)}{\sqrt{3-4v}}\frac{P}{\pi\sqrt{a^2 - x^2}}\cos\left(\eta\ln\left(\frac{a+x}{a-x}\right)\right), \qquad 0 < x < a$$

$$q\left(x\right) = \frac{2(1-v)}{\sqrt{3-4v}}\frac{P}{\pi\sqrt{a^2 - x^2}}\sin\left(\eta\ln\left(\frac{a+x}{a-x}\right)\right), \qquad 0 < x < a$$

where $\eta = \ln\left(3-4v\right)/\left(2\pi\right)$. Note that the pressure and shear approach to infinite as $x \to a$.

In the simulation, the radius of the half space is set to $R = 7a$ with boundary fixed. Due to symmetry, only one half of the model is modeled. The grid cell size is 0.05 mm and the particle space is 0.025 mm. Because this problem is quasi-static, the mass matrix and the inertial term are neglected and γ is set to 1. The convergence parameters are chosen as $\epsilon_R = 1.0 \times 10^{-7}$ and $\epsilon_C = 1.0 \times 10^{-7}$, respectively.

Fig. 6(a) shows the contact pressure distributions along the x direction for the frictionless case, where the solid line represents the analytical solution and the red dots are obtained by the proposed implicit MPM. Fig. 6(b) plots the contact pressure and shear distributions obtained by the implicit MPM for the adhesive case, compared to the analytical solutions. The results obtained by the implicit MPM agree well with analytical solution when $x/a < 0.95$. As x approaches a, the analytical solution predict an infinite elastic result. Both cases verify the computational accuracy of the implicit MPM with the proposed contact algorithm. Fig. 7 shows the distribution of stress σ_{yy} for two cases. The max value occurs at the corner of the rigid punch, which is caused by the stress concentration.

6.3. Cylinder rolling

An elastic cylinder rolling on an inclined elastic plate under gravity, as shown in Fig. 8, is investigated. The radius of cylinder is $r = 2.0$ m, while the size of plate is 10×0.2 m. The gravity $g = 0.141$ m/s^2 is vertically downward. The cylinder has a density of $\rho = 1000$ kg/m^3, Young's modulus of $E = 1.0 \times 10^9$ Pa, and Poisson's ratio of $v = 0.3$. The plate has a density of $\rho = 1.0 \times 10^4$ kg/m^3, Young's modulus of $E = 1.0 \times 10^9$ Pa, and Poisson's ratio of $v = 0.3$. The plate is fixed at the bottom surface.

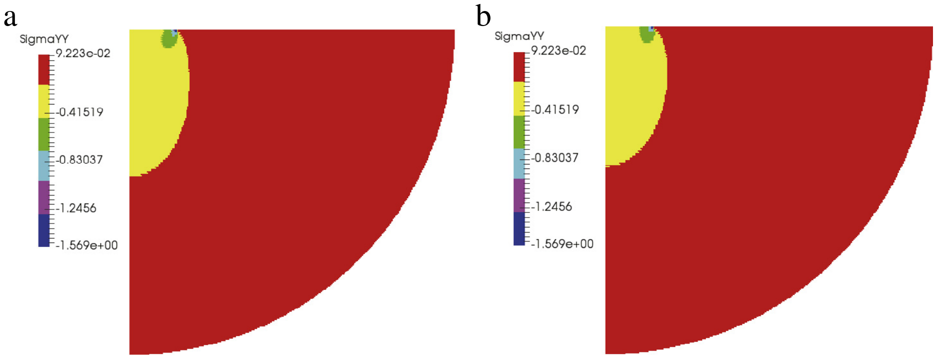


Fig. 7. The distribution of stress σ_{yy}: (a) $\mu = 0$; (b) $\mu = \infty$.

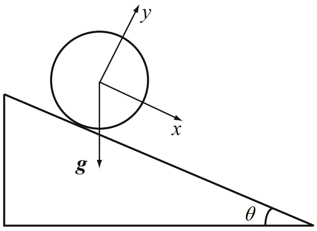
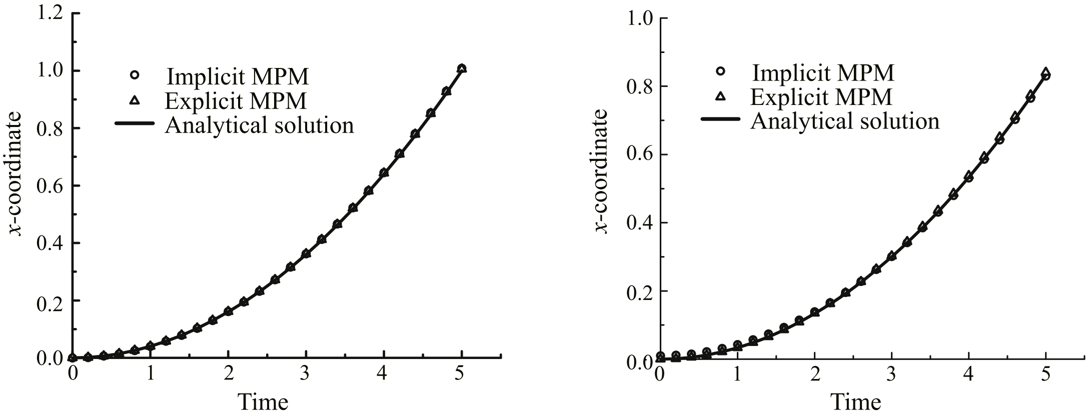


Fig. 8. A sphere rolling on an inclined plate.



(a) Case 1.

(b) Case 2.

Fig. 9. Time history of the center-of-mass position: (a) case 1; (b) case 2.

The analytical x-coordinate of the cylinder's center is given by [19]

$$x\left(t\right)=\begin{cases} x_0+\dfrac{1}{2}gt^2(\sin\theta-\mu\cos\theta) & \tan\theta>3\mu \quad \text{(roll and slip)}\\ x_0+\dfrac{1}{3}gt^2\sin\theta & \tan\theta\leqslant 3\mu \quad \text{(roll without slip)}\end{cases}$$

where $x_0 = 0$ is the x-coordinate of the initial position of the sphere's center.

In the simulation, the cell size is 0.04 m, and particle space is 0.02 m. Two cases are studied. In case 1, the inclined angle $\theta = \pi/4$ with frictional coefficient of $\mu = 0.2$, so that the sphere rolls with slip. In case 2, the inclined angle

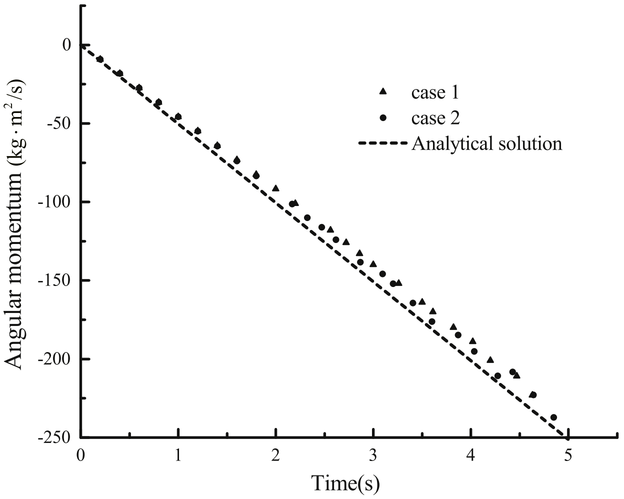


Fig. 10. Time history of the angular momentum.

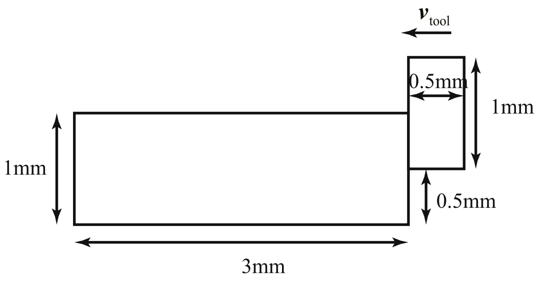


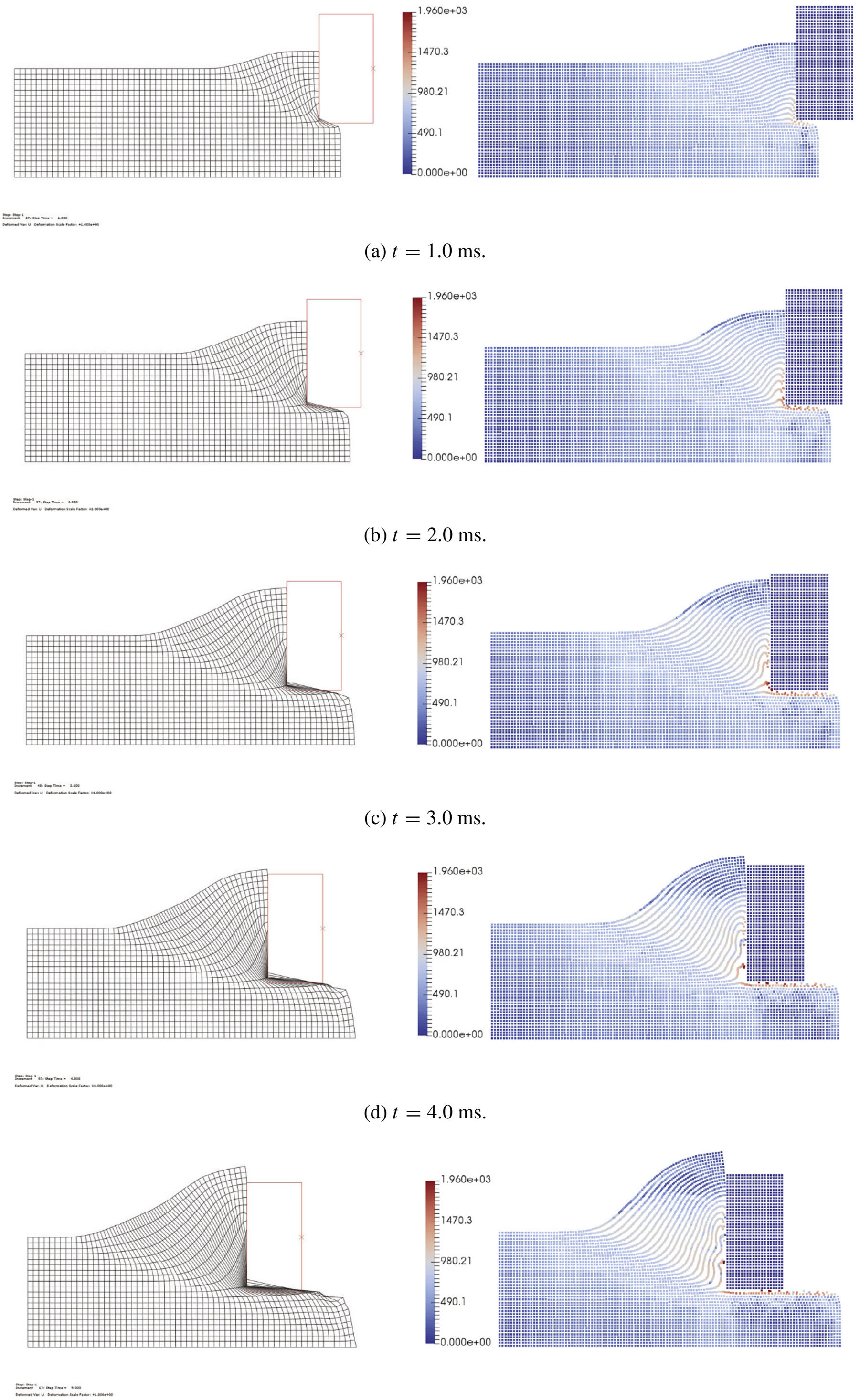
Fig. 11. Sketch of the computational cutting model.

Table 2
Accuracy and efficiency comparison between the implicit MPM and explicit MPM.

	Error		CPU time	
	Implicit MPM	Explicit MPM	Implicit MPM	Explicit MPM
Case 1	0.80%	0.49%	6.1 min	541.2 min
Case 2	0.76%	0.83%	6.0 min	544.5 min

$\theta = \pi/4$ with frictional coefficient of $\mu = 0.4$, so that the sphere rolls without slip. The convergence parameters are taken as $\epsilon_R = 1.0 \times 10^{-7}$ and $\epsilon_C = 1.0 \times 10^{-7}$, respectively. In the implicit MPM, the time step size is set to 0.2, and the Newmark parameters $\gamma = 0.5$ and $\beta = 0.25$ are used. In the explicit MPM, a time scale factor 0.8 is used.

Fig. 9 compares the time history of the center-of-mass position obtained by the implicit MPM and explicit MPM with the analytical solutions for case 1 and case 2. In the rolling simulation, the implicit MPM contact algorithm presents a high computational accuracy for both slip and adhesive case. The time history of the angular momentum of the cylinder with respective to the origin obtained by our scheme for both cases is plotted in Fig. 10, which also agree with the analytical solution reasonably. Table 2 summarizes the displacement error at time $t = 5$s and CPU time cost by the implicit MPM and explicit MPM. The accuracy of the displacement achieved by the implicit MPM is comparable to that achieved by the explicit MPM for both cases, but the cost of the implicit MPM is only about 1.1% of that of the explicit MPM. If the Young's modulus is much bigger, the implicit MPM will be much more efficient than the explicit MPM.


1.960e+03
1470.3
980.21
490.1
0.000e+00
Step: Step-1
Increment 27: Step Time = 1.000
Deformed Var: U Deformation Scale Factor: +1.000e+00
Step: Step-1
Increment 37: Step Time = 2.000
Deformed Var: U Deformation Scale Factor: +1.000e+00
Step: Step-1
Increment 48: Step Time = 3.100
Deformed Var: U Deformation Scale Factor: +1.000e+00
Step: Step-1
Increment 57: Step Time = 4.000
Deformed Var: U Deformation Scale Factor: +1.000e+00
Step: Step-1
Increment 67: Step Time = 5.000
Deformed Var: U Deformation Scale Factor: +1.000e+00

(a) $t = 1.0$ ms.

(b) $t = 2.0$ ms.

(c) $t = 3.0$ ms.

(d) $t = 4.0$ ms.

(e) $t = 5.0$ ms.

Fig. 12. Schematic representations of the deformation for cutting simulations with FEM (left) and the implicit MPM (right).

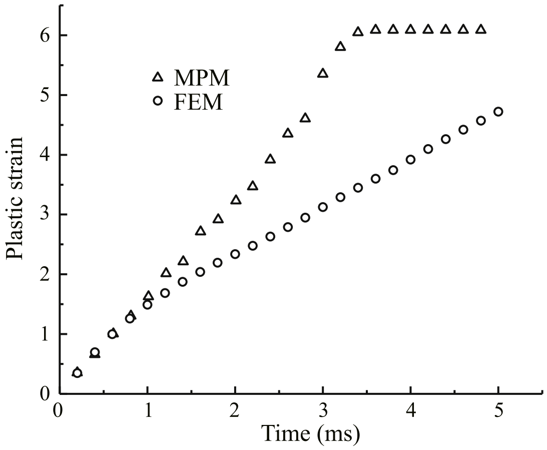


Fig. 13. Time history of the max plastic strain obtained by the implicit MPM and FEM.

6.4. *Cutting*

The finite element method is popular in cutting process simulations. Due to large deformation and element distortions the FEM is confronted with numerical difficulties. The similar numerical modeling has been carried out using the FEM and as well as the explicit MPM [36].

The model set-up is shown in Fig. 11. The fixed boundary condition is applied on the bottom and left sides of the workpiece. The tool moves as a rigid body during the cutting process at the velocity of $\boldsymbol{v}_{\text{tool}} = 0.2$ mm/ms. The workpiece in this simulation is assumed to be under plane strain condition with a width of 3 mm and a height of 1 mm.

In this simulation, we do not focus on the mechanism in the cutting process, but aim to demonstrate the robustness of the proposed contact algorithm and the competitive advantages of the implicit MPM over the FEM in simulating large deformation problems. Thus an elastic–plastic material model is assumed for the workpiece, and no friction is considered for simplicity. Since the cutting velocity is low, the inertial term is not significant. Thus, the cutting process is assumed to be quasi static. The workpiece has a density of $\rho = 7.8 \times 10^{-3}$ g/mm^3, Young's modulus of $E = 200$ GPa, and Poisson's ratio of $\upsilon = 0.3$. The initial yield strength is 700 MPa and the tangential modulus is 300 MPa. The workpiece is meshed by 60 × 20 bilinear elements in FEM. In the implicit MPM simulation, the grid size is 0.05 mm and the particle space is 0.025 mm.

Fig. 12 plots the configurations and equivalent stress field obtained by the ABAQUS/Standard and the implicit MPM at different times. The configurations obtained by the implicit MPM agree well with those obtained by the FEM as a whole except for those near the local cutting point. The severe element distortion in FEM simulation can be seen after $t = 1.0$ ms, especially at $t = 5.0$ ms, which decreases the numerical accuracy. In addition, the contact condition is no more exactly satisfied and nonphysical penetration occurs, which is mainly because there are no adequate intermediate nodes in a single element edge. In the implicit MPM simulations, the background grid is discarded at the end of every time step. Materials are automatically separated if the distance between particles is larger than two grid size. Thus, element distortion is avoided in the cutting simulation. Moreover, the shape of cut surface is consistent with the geometry of the tool. Fig. 13 compares the time histories of the max plastic strain obtained by the implicit MPM and FEM. The numerical results obtained by the two methods agree well to each other before severe element distortion appeared, but diverge significantly afterward. As a whole, the implicit MPM predicts a higher plastic strain than FEM.

7. Conclusion

This paper provides an easy-implementing frictional contact algorithm for the implicit MPM, which is based on an augmented Lagrange formulation. The contact problem is one of the saddle point problems, which are usually indefinite and highly ill-conditioned. All these features bring great challenges to the Newton-like methods, which require the Hessen matrix to be positive definite. By employing the Uzawa algorithm, the condition number of the original problem is dramatically improved and the positive definiteness of the effective stiffness matrix can be reserved. In addition, the Uzawa algorithm decouples the unknown variables and the multipliers, which simplifies the solving process.

The accuracy, efficiency and robustness of the contact algorithm have been validated by the numerical studies. The results obtained by the implicit MPM with the proposed contact algorithm agree well with the analytical solution. The computational accuracy and efficiency are much higher than the explicit MPM. The cutting simulation demonstrates the competitive advantages of the implicit MPM over the FEM and the robustness of the contact algorithm.

References

[1] D. Sulsky, Z. Chen, H.L. Schreyer, A particle method for history-dependent materials, Comput. Methods Appl. Mech. Engrg. 118 (1–2) (1994) 179–196.

[2] D. Sulsky, S.J. Zhou, H.L. Schreyer, Application of a particle-in-cell method to solid mechanics, Comput. Phys. Commun. 87 (1–2) (1995) 236–252.

[3] C. Coetzee, P. Vermeer, A. Basson, The modelling of anchors using the material point method, Int. J. Numer. Anal. Methods Geomech. 29 (9) (2005) 879–895.

[4] Z. Wieckowski, The material point method in large strain engineering problems, Comput. Methods Appl. Mech. Engrg. 193 (39–41) (2004) 4417–4438.

[5] A.R. York, D. Sulsky, H.L. Schreyer, Fluid–membrane interaction based on the material point method, Internat. J. Numer. Methods Engrg. 48 (6) (2000) 901–924.

[6] A. Gilmanov, S. Acharya, A hybrid immersed boundary and material point method for simulating 3D fluid–structure interaction problems, Internat. J. Numer. Methods Fluids 56 (12) (2008) 2151–2177.

[7] J.G. Li, Y. Hamamoto, Y. Liu, X. Zhang, Sloshing impact simulation with material point method and its experimental validations, Comput. & Fluids 103 (2014) 86–99.

[8] Y. Guo, J. Nairn, Three-dimensional dynamic fracture analysis using the material point method, Comput. Model. Eng. Sci. 16 (3) (2006) 141.

[9] X. Zhang, K.Y. Sze, S. Ma, An explicit material point finite element method for hyper-velocity impact, Internat. J. Numer. Methods Engrg. 66 (4) (2006) 689–706.

[10] S. Ma, X. Zhang, X.M. Qiu, Comparison study of MPM and SPH in modeling hypervelocity impact problems, Int. J. Impact Eng. 36 (2009) 272–282.

[11] S. Ma, X. Zhang, Y.P. Lian, X. Zhou, Simulation of high explosive explosion using adaptive material point method, CMES Comput. Model. Eng. Sci. 39 (2) (2009) 101–123.

[12] Y.P. Lian, X. Zhang, X. Zhou, S. Ma, Y.L. Zhao, Numerical simulation of explosively driven metal by material point method, Int. J. Impact Eng. 38 (4) (2011) 238–246.

[13] H. Zhang, K. Wang, Z. Chen, Material point method for dynamic analysis of saturated porous media under external contact/impact of solid bodies, Comput. Methods Appl. Mech. Engrg. 198 (17) (2009) 1456–1472.

[14] S. Cummins, J. Brackbill, An implicit particle-in-cell method for granular materials, J. Comput. Phys. 180 (2) (2002) 506–548.

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

[16] J.E. Guilkey, J. Weiss, Implicit time integration for the material point method: Quantitative and algorithmic comparisons with the finite element method, Internat. J. Numer. Methods Engrg. 57 (9) (2003) 1323–1338.

[17] A. Nair, S. Roy, Implicit time integration in the generalized interpolation material point method for finite deformation hyperelasticity, Mech. Adv. Mater. Struct. 19 (6) (2012) 465–473.

[18] B. Wang, P.J. Vardon, M.A. Hicks, Z. Chen, Development of an implicit material point method for geotechnical applications, Comput. Geotech. 71 (2016) 159–167.

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

[20] S.G. Bardenhagen, J.E. Guilkey, K.M. Roessig, J.U. Brackbill, W.M. Witzel, An improved contact algorithm for the material point method and application to stress propagation in granular material, CMES Comput. Model. Eng. Sci. 2 (2001) 509–522.

[21] P. Huang, X. Zhang, S. Ma, X. Huang, Contact algorithms for the material point method in impact and penetration simulation, Internat. J. Numer. Methods Engrg. 85 (2011) 498–517.

[22] W. Hu, Z. Chen, A multi-mesh MPM for simulating the meshing process of spur gears, Comput. Struct. 81 (2003) 1991–2002.

[23] Z.T. Ma, X. Zhang, P. Huang, An object-oriented MPM framework for simulation of large deformation and contact of numerous grains, CMES Comput. Model. Eng. Sci. 55 (1) (2010) 61–87.

[24] J. Nairn, Modeling imperfect interfaces in the material point method using multimaterial methods, Comput. Model. Eng. Sci. 1 (1) (2013) 1–15.

[25] J. Sanchez, H. Schreyer, D. Sulsky, P. Wallstedt, Solving quasi-static equations with the material-point method, Internat. J. Numer. Methods Engrg. 103 (1) (2015) 60–78.

[26] S. Nezamabadi, F. Radjai, J. Averseng, J.-Y. Delenne, Implicit frictional-contact model for soft particle systems, J. Mech. Phys. Solids 83 (2015) 72–87.

[27] K. Arrow, L. Hurwicz, H. Uzawa, Studies in Nonlinear Programming, Stanford University Press, Stanford, 1958.

[28] D.P. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, Academic Press, 2014.

[29] I.S. Duff, A.M. Erisman, J.K. Reid, Direct Methods for Sparse Matrices, Clarendon Press Oxford, 1986.

[30] T. Belytschko, W.K. Liu, B. Moran, K. Elkhodary, Nonlinear Finite Elements for Continua and Structures, John wiley & Sons, 2013.

[31] https://software.intel.com/en-us/intel-mkl/.

[32] J.J. Moreau, On unilateral constraints, friction and plasticity, in: New Variational Techniques in Mathematical Physics, Springer, 2011, pp. 171–322.

[33] G. Pietrzak, A. Curnier, Large deformation frictional contact mechanics: continuum formulation and augmented Lagrangian treatment, Comput. Methods Appl. Mech. Engrg. 177 (3) (1999) 351–381.

[34] I.C. Dolcetta, U. Mosco, Implicit complementarity problems and quasi-variational inequalities, in: R.W. Cottle, F. Giannessi, J.L. Lions (Eds.), Variational Inequalities and Complementarity Problems Theory and Applications, John Wiley & Sons, New York, London, 1980, pp. 271–283.

[35] K.L. Johnson, K.L. Johnson, Contact Mechanics, Cambridge University Press, 1987.

[36] R. Ambati, X. Pan, H. Yuan, X. Zhang, Application of material point methods for cutting process simulations, Comput. Mater. Sci. 57 (2012) 102–110.