Computational Mechanics
https://doi.org/10.1007/s00466-023-02412-w

ORIGINAL PAPER

Lagrange multiplier imposition of non-conforming essential boundary conditions in implicit material point method

Veronika Singer[1] · Tobias Teschemacher[1] · Antonia Larese[2,3] · Roland Wüchner[4] · Kai-Uwe Bletzinger[1]

Received: 22 June 2023 / Accepted: 16 October 2023

Abstract

The Material Point Method (MPM) is an established and powerful numerical method particularly useful for simulating large-scale, rapid soil deformations. Therefore, it is often used for the numerical investigation of mass movement hazards such as landslides, debris flows, or avalanches. It combines the benefits of both mesh-free and mesh-based continuum-based discretization techniques by discretizing the physical domain with Lagrangian moving particles carrying the history-dependent variables while the governing equations are solved at the Eulerian background grid, which brings many similarities with commonly used finite element methods. However, due to this hybrid nature, the material boundaries do not usually coincide with the nodes of the computational grid, which complicates the imposition of boundary conditions. Furthermore, the position of the boundary may change at each time step and, moreover, may be defined at arbitrary locations within the computational grid that do not necessarily coincide with the body contour, leading to different interactions between the material and the boundary. To cope with these challenges, this paper presents a novel element-wise formulation to weakly impose non-conforming Dirichlet conditions using Lagrange multipliers. The proposed formulation introduces a constant Lagrange multiplier approximation within the constrained elements in combination with a methodology to eliminate superfluous constraints. Therefore, in combination with simple element-wise interpolation functions classically utilized in MPM (and FEM) to approximate the unknown field, a suitable Lagrange multiplier discretization is obtained. In this way, we obtain a robust, efficient, and user-friendly boundary imposition method for immersed methods specified herein for implicit MPM. Furthermore, the extension to frictionless slip conditions is derived. The proposed methodologies are assessed by comparing the numerical results with both analytical and experimental data to demonstrate their accuracy and wide range of applications.

Keywords Material point method · Lagrange multiplier · Weak imposition · Immersed method · Non-conforming boundary conditions

1 Introduction

The Material Point Method (MPM), originally proposed by Sulsky et al. [1], is an established and powerful numerical method particularly well suited to simulating significant and rapid soil deformations. It combines the advantages of both mesh-free and mesh-based methods as the physical domain

Tobias Teschemacher, Antonia Larese, Roland Wüchner and Kai-Uwe Bletzinger have contributed equally to this work.

✉ Veronika Singer
veronika.singer@tum.de

Antonia Larese
antonia.larese@unipd.it

Roland Wüchner
r.wuechner@tu-braunschweig.de

Kai-Uwe Bletzinger
kub@tum.de

[1] School of Engineering and Design, Chair of Structural Analysis, Technical University of Munich, Arcisstraße 21, 80333 Munich, Germany

[2] Department of Mathematics "Tullio Levi Civita", Università degli Studi di Padova, via Trieste 63, 35121 Padua, Italy

[3] Institute for Advanced Study, Technical University of Munich, Lichtenbergstraße 2a, 85748 Garching, Germany

[4] Institute of Structural Analysis, Technische Universität Braunschweig, Beethovenstraße 51, 38106 Braunschweig, Germany

Published online: 18 November 2023

is discretized by Lagrangian moving particles which carry the history-dependent variables while the governing equations are solved at the Eulerian background grid. Thus, it can be interpreted as a modified updated-Lagrangian finite element technique with moving integration points, called material points, while the background grid is reset at the end of each time step. Consequently, in contrast to the standard Finite Element Method (FEM) or any other Lagrangian mesh-based method, the problem of mesh entanglement and the computational expense of re-meshing during the simulation of large strain problems is circumvented in MPM by systematically resetting the computational background grid. In comparison with other continuum-based particle methods, just mentioning the smoothed particle hydrodynamics (SPH) [2], the element-free Galerkin method [3], the reproducing kernel particle method (RKPM) [4], or the Particle Finite Element Method (PFEM) [5–8], MPM has many advantages. The hybrid particle-mesh scheme avoids, among other features [9, 10], expensive neighbor search or re-meshing algorithms while preserving all the advantages of being a FEM-based method.

MPM has been effectively applied to a wide range of challenges during the previous decades, such as hyper-velocity impact [10–12], high fidelity models of diagenesis with advanced contact algorithms [13], simulation of landslides [14, 15] or avalanches [16], multi-phase geomechanical problems [17–19] or within multiphysics, as the partitioned coupling with the discrete element method [20].

In addition, much research has been done to reduce the numerical noise that arises from the transition of material points from one background grid element to another, e.g., the Generalized Interpolation Material Point (GIMP) method [21], the Convected Particle Domain Interpolation (CPDI) [22, 23], the PQMPM [24] or the usage of B-Spline basis functions [25]. Detailed reviews of MPM can be found in [26, 27]. The books published by Zhang et al. [28], Fern et al. [29], and Nguyen et al. [30] provide further examples and a detailed introduction to the theory of MPM.

Since the material points are moving through the Eulerian background grid, boundary conditions can rarely be imposed directly at the nodes of the computational background grid [31, 32]. Instead, similarly to immersed methods such as the Finite Cell Method (FCM) [33, 34], or the Isogeometric B-Rep Analysis (IBRA) [35] essential boundary conditions are usually processed in a weak form. Common approaches in other FEM and immersed FEM methods are the penalty approach ([36] for FEM, e.g., [37] for FCM, e.g., [35, 38] for IBRA, e.g., [39] for MPM), the Nitsche method (e.g., [40] for FCM, e.g., [41] for IGA), Mortar-based methods (e.g., [42] for FEM, e.g., [43] for FCM, [44] for IGA), or the Lagrange multiplier approach ([45] for FEM, e.g., [38, 41] for IGA/IBRA). All methods have their advantages in different numerical scenarios. The penalty approach is typically the simplest avenue, being also comparatively fast. However, the introduction of the penalty factor adds flaws to the numerical system (see also the example from Sect. 4.1). Nitsche methods are very accurate but are mostly more involved in the implementation and their computational complex. Mortar methods condense some degrees of freedom (dofs) by others, which requires a modification of the stiffness matrix. This makes it hard to be handled due to the repeated updates appearing in MPM.

Within the scope of this research, the Lagrange multiplier method is presented, which directly solves the constraint equation and therefore is mature over the penalty approach in terms of accuracy but still less complex than the Nitsche method. However, designing a stable Lagrange multiplier solution is a challenging task and has been studied intensively, especially for FEM and other numerical methods to weakly impose essential boundary conditions [46, 47]. Specifically for the imposition of boundary conditions, Babuška [45] showed that equal order of interpolation of the respective fields demands two different mesh sizes for the displacements and the Lagrange multipliers. Following that approach results in a complicated and computationally expensive construction of the Lagrange multiplier discretization as proposed, for example, by Béchet et al. [47] and extended to 3D by Hautefeuille et al. [48] for the extended finite element method.

These approaches are not suitable for MPM as the set of active background grid elements may vary every time step due to the Lagrangian moving particles within the Eulerian background grid. Moreover, the boundary conditions are not necessarily aligned with the body contour but can instead be positioned arbitrarily within the background grid, leading to different interactions between the material and the boundary at each time step. To cope with these challenges, this paper presents a novel element-wise formulation to weakly impose non-conforming Dirichlet conditions using Lagrange multipliers. The proposed formulation introduces a constant Lagrange multiplier approximation within constrained elements in combination with a methodology to eliminate superfluous constraints. Therefore, in combination with simple element-wise interpolation functions classically utilized in MPM to approximate the displacement field, a suitable Lagrange multiplier discretization is obtained. In this way, we obtain a robust, efficient, and user-friendly boundary imposition method for immersed methods specified herein for implicit MPM.

The paper is structured as follows: the fundamentals of MPM and the used notation are summarized in Sect. 2. The imposition of boundary conditions and the derivation of an appropriate Lagrange multiplier field are presented in Sect. 3. Finally, in Sect. 4, numerical examples in 2D and 3D are discussed, proving the accuracy and demonstrating the broad application range of the proposed methodology.

2 Material point method

This section covers the methodological fundamentals of MPM, starting with the governing equations in Sect. 2.1, before introducing the discretization in time and space in Sect. 2.2 as well as the material point discretization in Sect. 2.3. Finally, the characteristic MPM update scheme in Sect. 2.4 is presented.

2.1 Governing equations

Consider a continuum body $\mathcal{B}$, which occupies a domain Ω in the three-dimensional Euclidean space $\mathcal{E}$ with a regular boundary Γ.

The displacement $\mathbf{u}$ of each point within the body is defined by

$$\mathbf{u} = \mathbf{x} - \mathbf{X} \tag{1}$$

and relates each position $\mathbf{X}$ in the undeformed reference configuration to its position $\mathbf{x}$ in the deformed configuration at time t, which is depicted in Fig. 1.

Mapping an infinitesimal line segment in the reference configuration $\mathrm{d}\mathbf{X}$ into the current configuration $\mathrm{d}\mathbf{x}$ leads to the definition of the deformation gradient $\mathbf{F}$

$$\mathrm{d}\mathbf{x} = \mathbf{F} \cdot \mathrm{d}\mathbf{X}. \tag{2}$$

Cauchy's first equation of motion, defined by

$$\rho \ddot{\mathbf{u}} = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b} \quad \text{in} \quad \Omega \tag{3}$$

is the governing equation for the given problem where $\mathbf{b}$ denotes the volume acceleration. This balance equation holds for every point $\mathbf{x} \in \Omega$ for all times t assuming an isothermal setting. It implies the symmetry condition $\boldsymbol{\sigma} = \boldsymbol{\sigma}^T$ for the Cauchy stress tensor and the definition of the spatial mass density $\rho = J^{-1}\rho_0$ describing the continuity of mass with the Jacobian $J = \det\mathbf{F}$. The first and second material time derivatives of the displacement field $\mathbf{u}$ are the velocity and the acceleration, respectively.

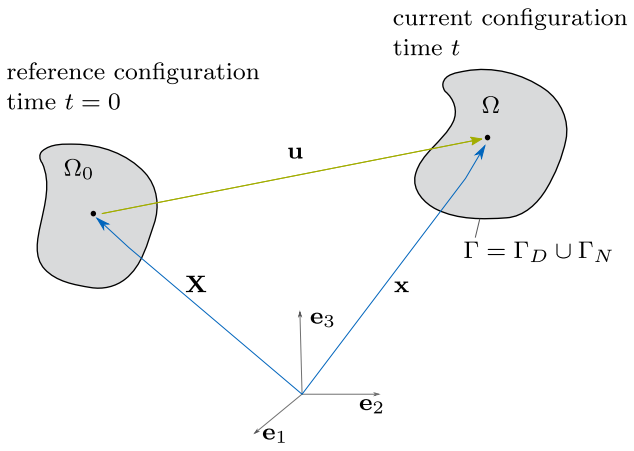

Fig. 1 Current and reference configuration of a continuum body $\mathcal{B}$. The regular boundary Γ is decomposed into Dirichlet and Neumann boundaries, Γ_D and Γ_N, respectively

The problem (Eq. 3) is determined with the boundary conditions

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{on} \quad \Gamma_D \tag{4}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \bar{\mathbf{p}} \quad \text{on} \quad \Gamma_N \tag{5}$$

where $\bar{\mathbf{u}}$ is a prescribed displacement field on the Dirichlet boundary Γ_D and $\bar{\mathbf{p}}$ is a traction vector on the Neumann boundary Γ_N with the outward normal $\mathbf{n}$. Additionally

$$\mathbf{u}(t=0) = \mathbf{u}_0 \quad \text{and} \quad \dot{\mathbf{u}}(t=0) = \dot{\mathbf{u}}_0 \tag{6}$$

are initial conditions while $\mathbf{u}_0, \dot{\mathbf{u}}_0$ are the prescribed displacement and velocity at initial time $t = 0$. The boundary Γ is decomposed into disjoint parts so that the following applies

$$\Gamma = \Gamma_D \cup \Gamma_N \quad \text{and} \quad \Gamma_D \cap \Gamma_N = \emptyset. \tag{7}$$

Since in general a closed-form solution for the given problem cannot be found, a Galerkin method is applied [49, 50]. Therefore the Eq. (3) is multiplied by a weighting function η and integrated over the current volume Ω. By using the variation of the displacements as the weighting function $\eta = \delta\mathbf{u}$, which equals zero on the boundary Γ_D, the weak form of the balance equation is obtained. Applying the product rule and the divergence theorem, the equilibrium can be formulated through the *Principle of Virtual Work* [51]:

$$\begin{aligned} \delta W = &- \int_\Omega \boldsymbol{\sigma} : \delta\mathbf{e}\,\mathrm{d}\Omega - \int_\Omega \rho\ddot{\mathbf{u}} \cdot \delta\mathbf{u}\,\mathrm{d}\Omega \\ &+ \int_\Omega \rho\mathbf{b} \cdot \delta\mathbf{u}\,\mathrm{d}\Omega + \int_{\Gamma_N} \bar{\mathbf{p}} \cdot \delta\mathbf{u}\,\mathrm{d}\Gamma_N = 0 \end{aligned} \tag{8}$$

while the virtual strain $\delta\mathbf{e}$ arises from the gradient of virtual displacement field.

Note that the stress boundary defined by Eq. (5) is part of the weak form (see Eq. 8) and, therefore often referred to as the natural boundary condition, while the Dirichlet condition needs to be prescribed over the boundary Γ_D and is thus often called essential boundary condition. Section 3 discusses this in detail and presents the Lagrange multiplier approach to weakly impose the Dirichlet boundary condition (see Eq. 4).

2.2 Discretization in space and time

To solve the equilibrium Eq. (8), a spatial discretization is necessary to approximate the continuous fields. There-

fore, the computational domain is subdivided into a discrete number of non-overlapping elements, equivalently to conventional finite elements. However, in comparison with classic FEM, the patch of elements in MPM covers the complete computational domain, including empty spaces into which the material is expected to move during the simulation, and is therefore referred to as the computational background grid in the following. Note empty elements are set temporarily inactive during the simulation if they do not contain parts of the body $\mathcal{B}$.

Therefore, the solutions are calculated at the discrete nodes of the elements while the field quantities within the elements are approximated by locally defined basis functions. For example, the displacement field is approximated by

$$\mathbf{u} \approx \mathbf{u}^h = \mathbf{N}_u \hat{\mathbf{u}} \tag{9}$$

where $\mathbf{N}_u$ is the shape function matrix containing the nodal shape functions N_I, while the vector $\hat{\mathbf{u}}$ contains the discrete displacement values at the corresponding nodes I. Also, the other fields, including the virtual displacement and the geometry are approximated with the same principle herein. The superscript h indicates the discretized variables while the subscript u is utilized to distinguish between the shape function matrix of the primal variables and the one resulting from the Lagrange multiplier discretization which is introduced in Sect. 3.1.

Inserting the spatial discretization leads to the semi-discrete equilibrium equation

$$\delta W^h = \frac{\partial W}{\partial \mathbf{u}^h} \delta \mathbf{u}^h = \delta \hat{\mathbf{u}}^T (-\mathbf{R}) = 0 \tag{10}$$

where $\mathbf{R}$ is the residual force vector

$$\mathbf{R} = \int_\Omega \left(\mathbf{B}^T \boldsymbol{\sigma} - \rho \mathbf{N}_u^T \mathbf{b} + \rho \mathbf{N}_u^T \mathbf{N}_u \hat{\ddot{\mathbf{u}}} \right) \mathrm{d}\Omega - \int_{\Gamma_N} \mathbf{N}_u^T \bar{\mathbf{p}} \mathrm{d}\Gamma_N = 0 \tag{11}$$

which equals zero since the virtual displacements are arbitrary. Herein $\mathbf{B}$ is the strain–displacement matrix containing the gradients of the shape functions while $\boldsymbol{\sigma}$ is the Cauchy stress tensor written in Voigt notation.

In addition to the spatial discretization, the continuous time domain is divided into discrete time steps t^n with constant time step size $\Delta t = t^{n+1} - t^n$. Within this paper, we are following the *Newmark-β* [52] implicit time integration scheme specifying $\beta = 0.25$. Therefore, the discretized fields for velocity and acceleration are approximated by the kinematic values already known from the previous time step. Consequently, the residual equation at each time step can be solved for the nodal displacements, which are the remaining unknowns.

However, as the residual force vector is non-linear in general, the Newton-Raphson method, which is based on Taylor's theorem, is used to linearize the equation and approximate the solution iteratively by

$$\mathbf{R}^{n+1} + \frac{\partial \mathbf{R}^{n+1}}{\partial \hat{\mathbf{u}}^{n+1}} \Delta \hat{\mathbf{u}} = \mathbf{R}^{n+1} + \mathbf{K}^{n+1} \Delta \hat{\mathbf{u}} = 0 \,, \tag{12}$$

introducing the tangential stiffness matrix $\mathbf{K}$ as the linearization of the residual force vector. This equation then is subsequently solved for the incremental displacement $\Delta \hat{\mathbf{u}}$ to receive the updated displacement $\hat{\mathbf{u}}^{n+1}$.

Until this point, these steps are identical to a classical updated Lagrangian FEM solution algorithm. In contrast to classical FEM, however, where Gaussian integration is usually applied for the numerical integration of the volume integrals in Eq. (12), in MPM, a particle summation is utilized, which is described in the following.

2.3 Material point discretization

In addition to the computational background grid, which approximates the continuous fields, the body $\mathcal{B}$ is discretized into a finite number n_p of Lagrangian moving particles, commonly called material points, representing a finite volume Ω_p of the body

$$\mathcal{B} \approx \mathcal{B}^h = \sum_{p=1}^{n_p} \Omega_p \,. \tag{13}$$

Applying further the classical MPM approach of a Dirac delta density function [1], the volume integrals within the equilibrium Eq. (12) are transformed into a summation of the particles.

Consequently, the entries of the tangential stiffness matrix related to the nodes I and J of the background grid elements can be calculated by

$$\mathbf{K}_{IJ}^{n+1} = \sum_{p=1}^{n_p} \left((\nabla_x N_I)^T \boldsymbol{\sigma}_p (\nabla_x N_J) + \mathbf{B}_I^T \mathbf{D}_p \mathbf{B}_J + \frac{N_I \rho_p N_J}{\beta \Delta t^2} \mathbf{I} \right) \Omega_p \,, \tag{14}$$

which are then assembled to the global tangential stiffness matrix defined in Eq. (12). Within this equation, the subscript p indicates the material point values, while $\mathbf{D}_p$ is the constitutive matrix. $\mathbf{B}_I$ is the deformation matrix related to each node I, and ∇_x denotes the spatial gradient of the shape function corresponding to node I.

Analogously, the entry of the residual vector related to node I can be calculated by

$$\mathbf{R}_I^{n+1} = \sum_{p=1}^{n_p} \left(\mathbf{B}_I \boldsymbol{\sigma}_p - \rho_p \mathbf{b} N_I + N_I \rho_p \mathbf{N}_u \hat{\ddot{\mathbf{u}}}^{n+1} \right) \Omega_p - \sum_{bp=1}^{n_{bp}} N_I \hat{\bar{\mathbf{p}}} \Gamma_{bp}. \tag{15}$$

The continuous boundary Γ_N herein is discretized into a finite number n_{bp} of mass-less boundary particles such that each boundary particle contains the discrete traction $\hat{\bar{\mathbf{p}}}$ and represents a discrete area Γ_{bp} of the boundary. Further details of this approach can be found in [20, 32].

2.4 MPM update scheme

Due to the dual description of Lagrangian moving material points on the one hand and the computational background grid on the other hand, the MPM workflow includes the classical finite element updated Lagrangian calculation procedure, which is enhanced by repeated inter- and extrapolation of information between background grid nodes (marked with subscript I) and material points (marked with subscript p). Therefore, the MPM procedure per time step can be categorized into the following phases (see Fig. 2):

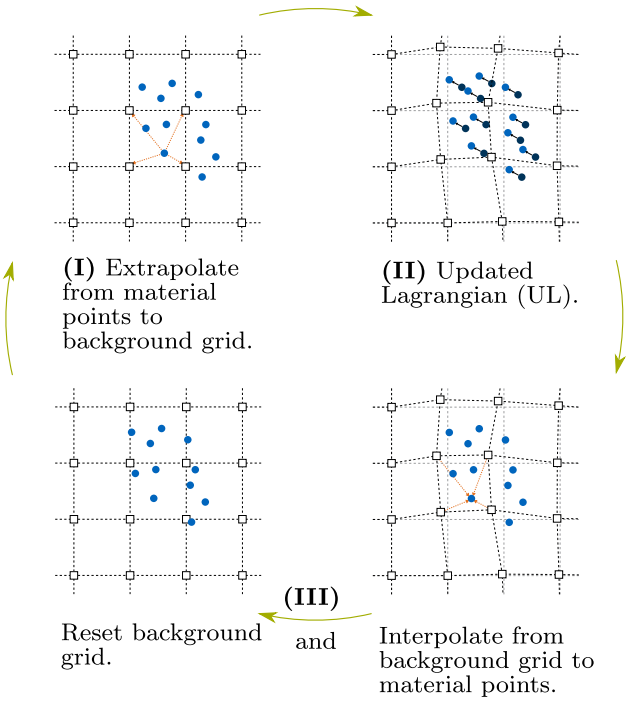

Fig. 2 MPM update scheme

(I) Initialization phase: At the beginning of each time step, the connectivity of each material point is defined, i.e., the background grid element containing the respective material point is searched for, so that the required shape function values N_I can be evaluated at the current position of the material point $\mathbf{x}_p^n$. The kinematic variables $\dot{\mathbf{u}}_p^n$ and $\ddot{\mathbf{u}}_p^n$ are then mapped via mass projection to the corresponding nodes of the background grid as initial conditions:

Calculate nodal velocity:

$$\dot{\mathbf{u}}_I^n = \frac{\sum_{p=1}^{n_p} m_p \dot{\mathbf{u}}_p^n N_I}{m_I^n} \tag{16}$$

Calculate nodal acceleration:

$$\ddot{\mathbf{u}}_I^n = \frac{\sum_{p=1}^{n_p} m_p \ddot{\mathbf{u}}_p^n N_I}{m_I^n} \tag{17}$$

where

$$m_I^n = \sum_{p=1}^{n_p} N_I m_p \tag{18}$$

is the assigned mass at the nodes of the background grid.

(II) Lagrangian phase: This phase is identical to the classical finite element updated Lagrangian calculation procedure [53–56] as it solves the discretized governing equations (Eq. 12) iteratively until a converged solution of the computational background grid is found. At this point, it is important to mark, that solely those nodes of the computational background grid which are assigned a mass m_I are contributing to the final equation system.

(III) Convective phase: The solution obtained at the nodes of the computational background grid is interpolated to the material points. Thus, the update of the position, velocity, and acceleration is defined by:

Material point position:

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \mathbf{N}_u \hat{\mathbf{u}}^{n+1} \tag{19}$$

Material point acceleration:

$$\ddot{\mathbf{u}}_p^{n+1} = \mathbf{N}_u \hat{\ddot{\mathbf{u}}}^{n+1}. \tag{20}$$

Material point velocity:

$$\dot{\mathbf{u}}_p^{n+1} = \dot{\mathbf{u}}_p^n + \frac{1}{2} \Delta t (\ddot{\mathbf{u}}_p^n + \ddot{\mathbf{u}}_p^{n+1}) \tag{21}$$

which is updated via the trapezoidal rule.

Finally, the background grid is reset, an important feature to simulate large strain events without the issues of mesh entanglement and distortion.

Further details of the presented scheme, as well as an extension to mixed formulations, can be found in [57–59].

3 Dirichlet boundary conditions

To solve a problem given by Eq. (8), the definition of proper Dirichlet boundary conditions is a crucial part. For some particular configurations, once the boundaries coincide with the nodal discretization of the computational background grid, the imposition is straightforward as they can be imposed in a FEM fashion, complying naturally with the Kronecker delta property. Those boundaries will be called *grid-conforming* boundaries in the following.

However, this imposition type is not generically applicable in MPM as the material moves independently of the computational background grid, which naturally leads to boundaries that are not aligned with the grid nodes. Therefore, alternative procedures are required, and the boundary has to be enforced in a weak form which will be referred to as *non-conforming* boundary conditions. Within the scope of this research, the Lagrange multiplier method shall be examined to weakly enforce Dirichlet conditions in MPM.

3.1 Lagrange multiplier approach

Enforcing a Dirichlet constraint as given by Eq. (4) with the *Lagrange multiplier method* to a given system leads to the modified principle of virtual work equation:

$$\begin{aligned}\delta W(\mathbf{u},\boldsymbol{\lambda},\delta\mathbf{u},\delta\boldsymbol{\lambda}) =& \delta W(\mathbf{u},\delta\mathbf{u}) + \delta W_{LM}(\mathbf{u},\boldsymbol{\lambda},\delta\mathbf{u}) \\ &+ \delta W_{LM}(\mathbf{u},\boldsymbol{\lambda},\delta\boldsymbol{\lambda}) = \mathbf{0} \end{aligned} \tag{22}$$

where $\boldsymbol{\lambda}$ is the Lagrange multiplier while the displacement $\mathbf{u}$ is the primal variable. In this equation, $\delta W(\mathbf{u},\delta\mathbf{u})$ represents the virtual work given by Eq. (8) while the additional terms arising from the Lagrange multiplier imposition are defined by:

$$\delta W_{LM}(\mathbf{u},\boldsymbol{\lambda},\delta\mathbf{u}) = \int_{\Gamma_D} \boldsymbol{\lambda}^T \delta\mathbf{u} \mathrm{d}\Gamma_D \tag{23}$$

$$\delta W_{LM}(\mathbf{u},\boldsymbol{\lambda},\delta\boldsymbol{\lambda}) = \int_{\Gamma_D} \delta\boldsymbol{\lambda}^T (\mathbf{u}-\overline{\mathbf{u}})\, \mathrm{d}\Gamma_D \,. \tag{24}$$

To solve the modified virtual work equation numerically, the continuous fields $\mathbf{u}$ and $\boldsymbol{\lambda}$ need to be approximated by locally defined basis functions and discrete values at the nodes. This discretization, however, has to be chosen carefully to satisfy the inf-sup condition of Ladyzhenskaya, Babuška [45, 60] and Brezzi [61, 62] which is discussed in Sect. 3.3. At this point, the notation is kept general utilizing the shape function matrix $\mathbf{N}_u$ for the approximation of the displacement field (see Eq. 9) and introducing the shape function matrix $\mathbf{N}_\lambda$ to approximate the Lagrange multiplier field by

$$\boldsymbol{\lambda} \approx \boldsymbol{\lambda}^h = \mathbf{N}_\lambda \hat{\boldsymbol{\lambda}} \tag{25}$$

while $\hat{\boldsymbol{\lambda}}$ are the values at the Lagrange multiplier dofs.

Following, Eqs. (23–24) are rewritten in the discretized version:

$$\delta W^h_{LM}(\mathbf{u}^h,\boldsymbol{\lambda}^h,\delta\mathbf{u}^h) = \delta\hat{\mathbf{u}}^T \int_{\Gamma_D} \mathbf{N}_u^T \mathbf{N}_\lambda \mathrm{d}\Gamma_D \hat{\boldsymbol{\lambda}} \,, \tag{26}$$

$$\begin{aligned}\delta W^h_{LM}(\mathbf{u}^h,\boldsymbol{\lambda}^h,\delta\boldsymbol{\lambda}^h) = \delta\hat{\boldsymbol{\lambda}}^T \Bigg(&\int_{\Gamma_D} \mathbf{N}_\lambda^T \mathbf{N}_u \mathrm{d}\Gamma_D \hat{\mathbf{u}} \\ &- \int_{\Gamma_D} \mathbf{N}_\lambda \mathrm{d}\Gamma_D \hat{\overline{\mathbf{u}}} \Bigg) \,. \end{aligned} \tag{27}$$

Herein $\hat{\overline{\mathbf{u}}}$ are the discrete imposed displacements at the discretized boundary, introduced in Sect. 3.3. Applying the Newton Raphson procedure to the modified virtual work Eq. (22) and considering the discretization leads to the saddle point problem:

$$\begin{bmatrix} \mathbf{K}\,(+\mathbf{K}_{LM}) & \boldsymbol{\Lambda}^T \\ \boldsymbol{\Lambda} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\hat{\mathbf{u}} \\ \Delta\hat{\boldsymbol{\lambda}} \end{bmatrix} = - \begin{bmatrix} \mathbf{R} + \boldsymbol{\Lambda}^T \hat{\boldsymbol{\lambda}} \\ \boldsymbol{\Lambda}\hat{\mathbf{u}} - \overline{\boldsymbol{\Lambda}}\hat{\overline{\mathbf{u}}} \end{bmatrix} \tag{28}$$

with

$$\boldsymbol{\Lambda} = \int_{\Gamma_D} \mathbf{N}_\lambda^T \mathbf{N}_u \mathrm{d}\Gamma_D \quad \text{and} \quad \overline{\boldsymbol{\Lambda}} = \int_{\Gamma_D} \mathbf{N}_\lambda \mathrm{d}\Gamma_D \,. \tag{29}$$

$\mathbf{K}$ and $\mathbf{R}$ are defined by Eq. (12) and $\mathbf{K}_{LM}$ is the linearization of Eq. (26) with respect to the displacements. As the constraints are linear in $\mathbf{u}^h$, this term vanishes and is therefore posed in brackets. From Eq. (29) it can be observed that the contribution of the Lagrange multiplier imposition depends on the shape functions of the two discretized fields, while the evaluation of the integral is discussed in Sect. 3.3.

3.2 Suitable Lagrange multiplier field

The given saddle point problem, which results in a minimum of the primal variables and a maximum for the Lagrange multipliers, can be solved only if the inf-sup condition of Ladyzhenskaya, Babuška [45, 60] and Brezzi [61, 62] is satisfied. This mainly depends on the discretization of the involved fields. While in MPM, classically simple element-wise basis functions are chosen for the displacement field, the Lagrange multiplier discretization has to be selected according to that field approximation which is introduced in Sect. 3.2.1. Additionally, in Sect. 3.2.2, the chosen approach is

enhanced to obtain a suitable Lagrange multiplier discretization for different background grid element types excluding superfluous constraints.

3.2.1 Discretization of the Lagrange multiplier field

As stated by Zienkiewicz et al. [49], a necessary but not sufficient condition for a stable solution is the requirement that the rank of the tangential stiffness matrix $\mathbf{K}$ must be larger or equal to the rank of Lagrange multiplier contribution $\mathbf{\Lambda}$. Thus, for the imposition of non-conforming boundary conditions, this requirement is met by using a different order of interpolation for the involved fields. Accordingly, in combination with simple element-wise basis functions for the displacement field, which are linear functions for triangular and tetrahedral background grid elements and bi- and trilinear functions for quadrilateral and hexahedral elements, respectively, a constant element-wise approximation of the Lagrange multipliers is selected. In particular, from a computational point of view, this is an attractive and straightforward choice since each background grid element is assigned an additional dof $\boldsymbol{\lambda}$ for the spatial directions. It contributes to the final system of equations when the element is intersected by the boundary. Consequently, the boundary conditions are defined locally at the element level, which is an essential feature for general and robust imposition within MPM. Especially for the numerical investigation of flow events with large strains as well as a changing topology of the boundaries, this locality is an essential requirement as the selection of active background grid elements may change in every time step.

3.2.2 Excluding superfluous constraints

The approach of simple element-wise basis functions for the primal variables in combination with the constant approximation of the dual variables does not soundly satisfy the inf-sub condition. It is known from the formulation of mixed elements that this approach fails, as it leads to over-constraint systems [63, 64]. Additionally, for the imposition of boundary conditions, this approach may result in a Lagrange multiplier discretization that is locally over-constrained, leading to spurious oscillations of the resulting Lagrange multipliers.

This effect is also referred to as boundary locking (see e.g., Hughes [65]) with an adverse influence on the convergence behavior of the primal variable. Therefore, the Lagrange multiplier discretization has to be appointed consciously in dependence on the respective background grid, ensuring that conditions are not imposed twice. In this section, a procedure to eliminate superfluous constraints to receive an appropriate Lagrange multiplier field is proposed.

For illustration purposes, the consequences of the straightforward utilization of a constant Lagrange multiplier approximation within all elements which are intersected by the boundary should be exemplified for a linear elastic cantilever beam under dead load. The clamped support is imposed by the non-conforming Lagrange multiplier condition. Details of the example, as well as a sketch of the system, can be found in Sects. 4.2 and 4.3, where this example is investigated further. At this point, we are focusing on the solution for the discrete Lagrange multipliers representing the forces at the boundary for different background grid element types. In Fig. 3a–d, the results for four different background grid types are depicted.

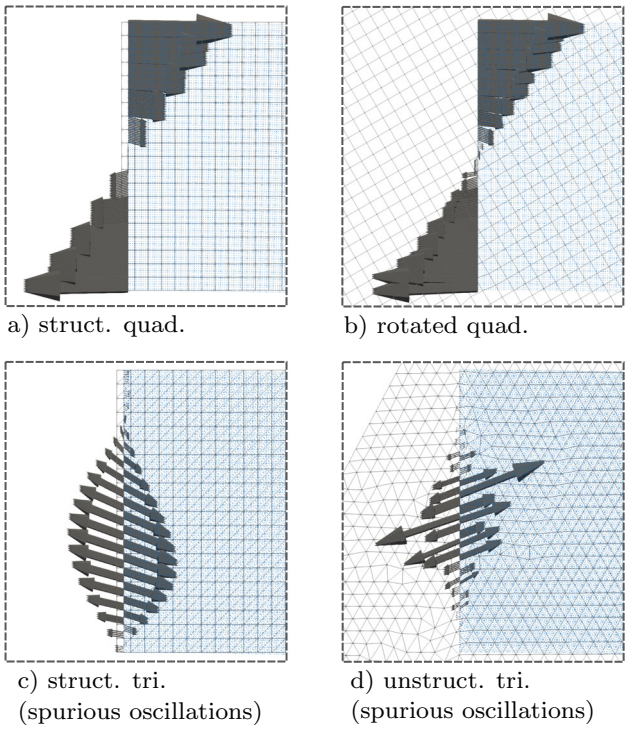

Fig. 3 Solution of the discrete Lagrange multipliers at the boundary for different background grid element types. Sub-figures **a**, **b** illustrate the force distribution in the case of quadrilaterals (structured and rotated) while sub-figure **c**, **d** show the results for triangular background grid elements (structured and unstructured)

It can be observed that for quadrilateral background grid elements, independently whether the boundary intersects the element equally, as illustrated in Fig. 3a or arbitrarily (see Fig. 3b) the resulting forces are representing the expected stress distribution at the boundary. Due to the chosen Lagrange multiplier discretization (see Sect. 3.2.1), the forces are constant within one element.

However, spurious oscillations of the resulting forces can be observed for triangular background grid elements. The resulting force distributions (being the respective Lagrange multipliers) are depicted in Fig. 3c for a structured and Fig. 3d for an unstructured triangular background grid, indicating the breach of the local mesh dependent inf-sub condition.

This phenomenon can be explained mechanically by the possible deformations of the background grid element edges

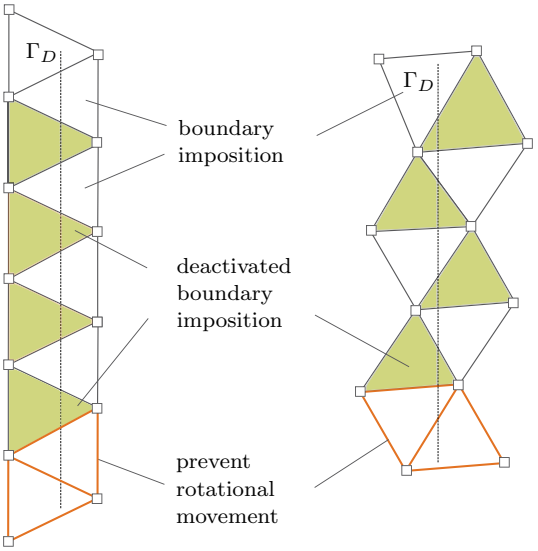

Fig. 4 The boundary condition should be imposed within the white elements, whereas the Lagrange multiplier dofs should be deactivated in the green highlighted elements. (Color figure online)

in combination with the element geometry itself. Due to the chosen basis functions for the primal variables, the displacement along the edges is approximated linearly between the corresponding nodes. This is still the case for edges intersected by a Dirichlet boundary. However, fulfilling the constraints leads to a dependency on the adjacent nodal deformations of that specific edge.

Consequently, the nodal deformations of a triangular element are already determined if it has common edges with two neighboring constrained elements. In Fig. 4 the elements highlighted in green are such particular elements whose nodal deformations and, consequently the resulting strains are already determined by the neighboring constrained elements. Therefore, the constraint imposition within those elements is superfluous and needs to be eliminated to avoid spurious oscillations in the Lagrange multiplier solution. This can be achieved easily within the computation by temporarily deactivating the dofs of respective Lagrange multipliers to eliminate those superfluous constraints.

To prevent the rotational movement of the constrained elements at the boundary, two adjacent elements being intersected by the Dirichlet condition need to be set active once within the boundary geometry definition due to the constant distribution of the Lagrange multiplier. In Fig. 4, those elements are graphically highlighted with an orange contour and are defined at the starting point of the line, defining the boundary geometry.

Applying the proposed scheme to the illustrative example of the cantilever beam under dead load leads to an appropriate Lagrange multiplier field for triangular elements as well. It is successfully eliminating the spurious oscillations of the resulting forces. In Fig. 5 a) and b) the solution of the

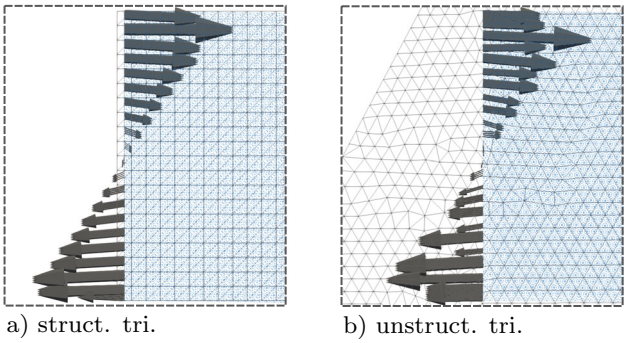

Fig. 5 Solution of the discrete Lagrange multipliers at the boundary for triangular elements considering the proposed elimination procedure excluding the superfluous constraints. Sub-figure **a** illustrates the force distribution for a structured triangular background grid, while sub-figure **b** shows the result for the unstructured background grid

Lagrange multipliers is visualized, showing a distribution of the forces in accordance with the expected stress distribution at the boundary. Due to the elimination procedure and in combination with a non-uniform cut of the boundary with the background grid elements, the distribution of the forces in Fig. 5b is obviously not as smooth as the expected stress distribution. Therefore, a weighting procedure is applied to distribute the element-wise constant forces to the discrete boundary particles. This procedure is explained in Sect. 3.5. In Sect. 4.3, the accordance of the resulting forces with the global equilibrium is demonstrated, while Fig. 14 shows the smooth distribution of the resulting forces at the boundary particles.

This argumentation holds as well for quadrilateral elements. However, superfluous constraints may solely occur if three neighboring elements sharing each an edge with the considered element are also constrained. For the imposition of boundary conditions, this configuration rarely appears, and therefore no spurious oscillations of the resulting forces can be observed in Fig. 3a, b.

For the 3D space, this theory can be extended accordingly, distinguishing between neighboring elements sharing a surface or an edge. Therefore, three constrained neighboring edge elements are sufficient to describe the nodal deformations of a tetrahedral element as long as the edges pass through every node of the geometry. All other constraints within the remaining neighboring elements as well as the constraint within the considered element itself, are superfluous in this case. For hexahedral elements, seven constrained edge elements passing through all nodes would be sufficient to describe the nodal displacements. In general, however, the constrained elements in 3D are not distributed such that the elimination procedure may consider the edge-aligned neighbors only. Instead, it is of utmost importance that the deformation of all nodes of a constraint element is determined, and only under this condition superfluous Lagrange multipliers can be deactivated. It is, therefore, often

inevitable to avoid some local superfluous constraints by maintaining a sufficient boundary imposition for all nodes involved. Consequently, in 3D, some oscillations in the solution of the Lagrange multipliers may be present, which, however, are reduced to a minimum due to the elimination procedure. In Sect. 4.4 it is demonstrated that the proposed elimination still results in a suitable Lagrange multiplier distribution to impose non-conforming arbitrarily shaped Dirichlet conditions in 3D space in an MPM framework with considerable effort.

Although there are often several possibilities to select the elements for the boundary imposition, the solution of the primal variable is not affected by this decision as long as the boundary imposition determines all nodal deformations of intersected elements. The resulting values of the Lagrange multipliers, on the other hand, may change slightly to satisfy the equilibrium.

From a computational point of view, the proposed elimination procedure of superfluous Lagrange multiplier dofs to receive a suitable Lagrange multiplier distribution is attractive as well: since the computational background grid does not change its topology, the neighboring elements sharing a common edge or surface need to be found only once for each element being intersected by the boundary. Consequently, within each time-step superfluous constraints are detected efficiently, and therefore the corresponding Lagrange multiplier dofs are deactivated temporally.

3.3 Particle representation of the boundary

In MPM, the continuous boundary Γ_D shall be partitioned into n_{bp} non-overlapping sub-domains, while each sub-domain is represented by a boundary particle with a respective current area Γ_{bp}:

$$\Gamma_D \approx \sum_{\text{bp}=1}^{n_{\text{bp}}} \Gamma_{\text{bp}}. \tag{30}$$

Therefore, in conformance with the volume integration of the body by material points, a quadrature-based technique is chosen for the boundary as well. Those boundary particles can be expounded as mass-less particles carrying the necessary information for the boundary imposition. This procedure was also proposed in [39] to weakly impose essential boundary conditions by penalty augmentation. This methodology is mature compared to other boundary tracking methods, such as the level set method [63], as it omits the computationally expensive calculation for the intersections of the boundary with the elements of the computational background grid. Instead, the search algorithms inherent in any MPM scheme can also be used for the boundary particles, and consequently, moving boundaries can be handled efficiently.

The boundary particles are introduced at the beginning of the simulation within the respective primitives, e.g., polylines in 2D or meshes in 3D. A user-defined number of boundary particles are placed either equally distributed or at Gauss point positions of the respective primitive and get assigned the corresponding weights representing the respective area Γ_{bp}. In a two-dimensional case, the individual weight has to be multiplied by the corresponding thickness of the model to obtain the resulting sub-domain of each boundary particle. Thus, the local surface integral in Eq. (28) evaluated for one element being intersected by the boundary segment Γ_D can be approximated by:

$$\int_{\Gamma_D} (...)\, d\Gamma_D \approx \sum_{\text{bp}=1}^{n_{\text{bp}}} (...)\, \Gamma_{\text{bp}}\,. \tag{31}$$

In this case, the boundary segment within an intersected background grid element is approximated by summing the respective boundary particle areas located within the respective background grid element. Consequently, a fine discretization of the boundary by mass-less particles leads to a good approximation of the boundary segments.

In addition to the geometrical information, as the current position and the assigned area, the boundary particles carry the kinematic variables needed for prescribed displacements and the unit normal vector, which is essential for the inclined slip condition (see Sect. 3.4). As suggested by [39], the unit normal vectors $\hat{\mathbf{n}}_{\text{bp}}$, defined in the outward direction of the imposed boundary, are initialized at the beginning of the simulation on each boundary particle, which is illustrated in Fig. 6. The interpolated unit normal vector $\hat{\mathbf{n}}_I$ at the node I of the computational background grid can then be approximated within each time step by:

$$\hat{\mathbf{n}}_I = \frac{\sum_{\text{bp}=1}^{n_{\text{bp}}} \hat{\mathbf{n}}_{\text{bp}} \Gamma_{\text{bp}} N_I}{\left\| \sum_{\text{bp}=1}^{n_{\text{bp}}} \hat{\mathbf{n}}_{\text{bp}} \Gamma_{\text{bp}} N_I \right\|}\,. \tag{32}$$

The normal is required to construct the rotation matrix $\mathbf{Q}$ for the inclined roller support described in the following section. The unit vectors at the boundary particles are updated during the calculation procedure according to the boundary particle movement during the calculation procedure.

3.4 Slip boundary condition

One further novel feature of this research is the imposition of slip Dirichlet boundary conditions, which can be oriented arbitrarily in space, named slip conditions. In this case, the movement in the normal direction to the support is restricted, whereas, at the same time, the body is able to move freely in tangential directions. In Fig. 6 the slip condition in 2D is illustrated.

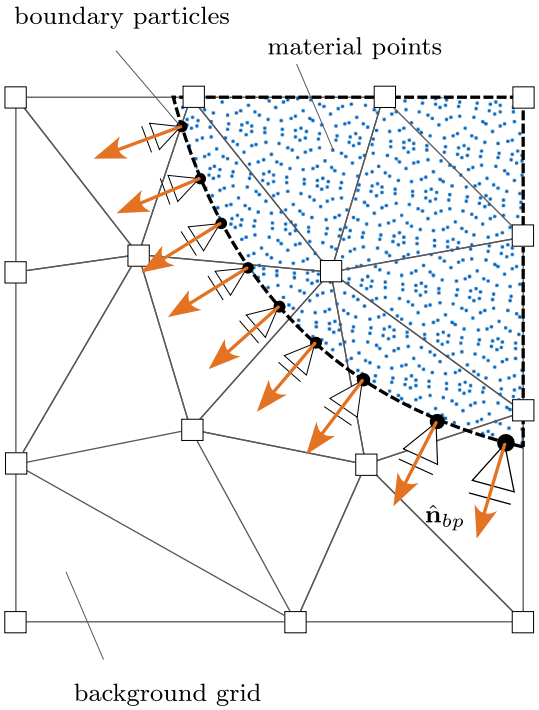

Fig. 6 Slip conditions and definition of $\hat{\mathbf{n}}_{\text{bp}}$ at the boundary particles

Therefore, all globally oriented matrices and vectors are rotated with an orthogonal rotation matrix:

$$\mathbf{Q} = \begin{bmatrix} \hat{n}_x & \hat{n}_y & \hat{n}_z \\ \hat{t}_x & \hat{t}_y & \hat{t}_z \\ \hat{q}_x & \hat{q}_y & \hat{q}_z \end{bmatrix} \tag{33}$$

where $\hat{\mathbf{n}}_I$ is the resulting normal unit vector perpendicular to the inclined surface at the node I and $\hat{\mathbf{t}}_I$, and $\hat{\mathbf{q}}_I$ are the normalized tangent vectors parallel to the surface. For a global imposition, it is necessary that the rotation is applied locally to those nodes affected by the rotation. Therefore, the rotation matrix results in a block matrix

$$\hat{\mathbf{Q}} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q} \end{bmatrix} \tag{34}$$

where $\mathbf{I}$ represents the identity matrix for those nodes which are not affected by the rotation. Applying the modified rotation matrix leads to the partially rotated system:

$$\left(\hat{\mathbf{Q}} \begin{bmatrix} \mathbf{K} & \boldsymbol{\Lambda}^T \\ \boldsymbol{\Lambda} & \mathbf{0} \end{bmatrix} \hat{\mathbf{Q}}^T\right) \left(\hat{\mathbf{Q}} \begin{bmatrix} \Delta\hat{\mathbf{u}} \\ \Delta\hat{\boldsymbol{\lambda}} \end{bmatrix}\right) = -\left(\hat{\mathbf{Q}} \begin{bmatrix} \mathbf{R} + \boldsymbol{\Lambda}^T\hat{\boldsymbol{\lambda}} \\ \boldsymbol{\Lambda}\hat{\mathbf{u}} - \overline{\boldsymbol{\Lambda}}\hat{\overline{\mathbf{u}}} \end{bmatrix}\right). \tag{35}$$

To impose the roller condition, the Lagrange multipliers in tangential directions to the support are set to zero. This can be achieved by setting the respective entries of the rotated stiffness matrix to zero, except for the entry on the main diagonal, which is set to one. As the entries in the residual vector are also set to zero, the resulting Lagrange multipliers in tangential directions, which can be interpreted as reaction forces, are set to zero. Following this, the system can move in tangential directions without restrictions.

3.5 Calculation of reaction forces

Another advantage of the Lagrange multiplier method is that the resulting reaction forces which enforce the Dirichlet condition can be calculated directly. From Eq. (28), one can derive the additional force acting on the nodes of the computational background grid due to the boundary imposition:

$$\mathbf{R}_D = \boldsymbol{\Lambda}^T \hat{\boldsymbol{\lambda}} \,. \tag{36}$$

Therefore, the resulting reaction forces $\mathbf{R}_{I,D}$ at node I of the computational background grid element in spatial directions is defined by:

$$\mathbf{R}_{I,D} = \hat{\boldsymbol{\lambda}} \sum_{\text{bp}=1}^{n_{\text{bp}}} \mathbf{N}_I \Gamma_{\text{bp}} \,, \tag{37}$$

assuming a constant Lagrange multiplier distribution within the element. The reaction forces at the boundary particles are obtained by interpolating the discrete nodal reaction forces with the respective shape functions:

$$\mathbf{R}_{bp} = \sum_{I=1}^{n_n} \gamma_{\text{I,bp}} \mathbf{N}_I \mathbf{R}_{I,D} \tag{38}$$

where $\gamma_{\text{I,bp}}$ is the interpolation weighting factor to maintain the sum of forces. It is defined by the ratio of the boundary particle area Γ_{bp} and the nodal area $\mathbf{A}_I$:

$$\gamma_{\text{I,bp}} = \frac{\Gamma_{\text{bp}}}{\mathbf{A}_I}, \qquad \text{where} \quad \mathbf{A}_I = \sum_{\text{bp}=1}^{n_{\text{bp}}} \mathbf{N}_I \Gamma_{\text{bp}} \,. \tag{39}$$

This applied weighting procedure, a classical MPM approach, provides the resulting reaction forces at the discretized boundary.

The accuracy of the proposed calculation of the reaction forces at the boundary particles is presented explicitly in Sect. 4.3.

3.6 Material and boundary interaction

The Dirichlet boundary conditions in an MPM model can be defined arbitrarily within the computational background grid and, therefore, independently of the body outline. Therefore, in contrast to classical FEM, where the conditions are

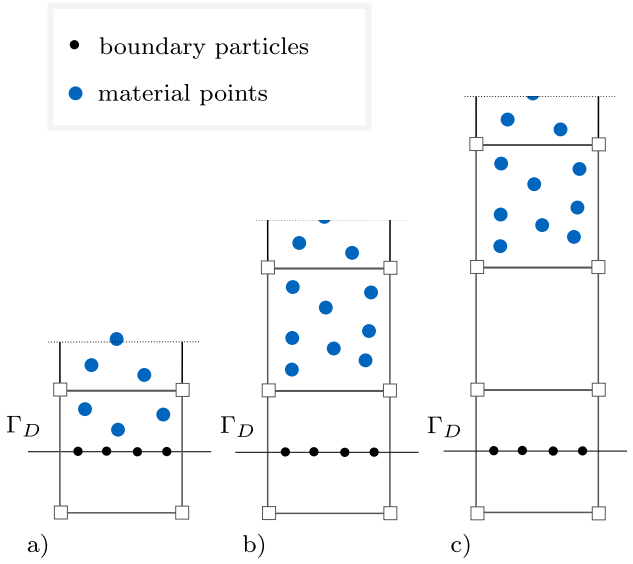

Fig. 7 Configurations of the boundary and the material

classically defined along the body outline, in MPM it is not necessarily the case that both boundary particles and material points lie within one background grid element. Instead, it is possible for the body to move without the influence of the boundary condition if there is at least one empty background grid element between them, preventing mutual interaction via the nodes of the computational background grid. Thus, for the imposition of weak boundary conditions, different configurations of material points interacting with the boundary particles have to be considered and are visualized in Fig. 7. The first configuration visualized in Fig. 7a where material points and boundary particles are within one background grid element, the imposition of the non-conforming boundary is straightforward, as described before. The configurations b and c, however, have to be considered carefully, as the boundary particles are located within a background grid element that contains no material points, and therefore the stiffness of the particular element is zero. For configuration c as well as setups where the material is even further apart, the respective dofs of elements containing material points and those containing the boundary particles are decoupled. Therefore, the material movement is not influenced by the respective boundary imposition which is why the corresponding element containing the boundary particles need not be considered for the final equation system.

However, in configurations depicted in Fig. 7b where material points are located in neighboring elements of those being intersected by the boundary, the boundary condition already influences the material movement. One possibility might be to neglect the boundary in this case as well by deactivating the boundary imposition of all boundary particles that are within a background grid element without material points. However, this may easily lead to material points penetrating through the boundary. This is especially true for configurations where the boundary is defined near the element border of the adjacent element containing the material points. Since the boundary condition is not active for the solution of the governing equations, the nodes of the background grid element are updated within one time step without considering the respective Dirichlet condition, and after updating the material point positions, it may be the case that the material points have already passed through the boundary before it was activated. Therefore, another method is proposed at this point. For those elements that contain only boundary particles but still interact with material points, an artificial stiffness is added to the diagonal of the respective elemental stiffness matrix.

As a consequence, elements that contain solely boundary particles are assigned an artificial stiffness, ensuring that material points from neighboring elements are already influenced by the boundary condition. Due to this influence, the movement of the material points should be decelerated such that they are not penetrating through the boundary but still permitting the movement towards the boundary itself. From the authors' experience, Young's modulus of the considered material multiplied by the volume of the respective element can be set as the artificial stiffness.

3.7 Small cut instability

In immersed methods, the small cut instabilities are well-known problems and have been addressed by several authors just mentioning [40]. In case a boundary cuts an element nearby, the shape function values of the corresponding nodes may be very small, consequently leading to ill-conditioned stiffness matrices. Therefore, the most trivial approach to overcome this issue is to modify the shape function values N_I evaluated at the location of the respective boundary particle such that the minimum value is larger than a user-given threshold ε, leading to the following equation:

$$\bar{N}_I = \frac{N_I^*}{\sum_I^{n_n} N_I^*} \quad \text{where} \quad N_I^* = \begin{cases} \varepsilon, & \text{if } N_I \leq \varepsilon \\ N_I & \text{otherwise} \end{cases} \tag{40}$$

where $\bar{N}_I$ is the modified shape function value and n_n the number of nodes of an element. Yet the partition of unity is ensured by the weighting procedure. This approach was applied in the penalty approach [39] as well to handle the small-cut instabilities where the stabilization tolerance was set to $\varepsilon = 0.01$. This assumption holds as well for the present work.

However, it's not sufficient to check the shape function values of the boundary particles only. As illustrated in Fig. 8, there are several configurations of boundary particles that may have small shape function values. Those particles are

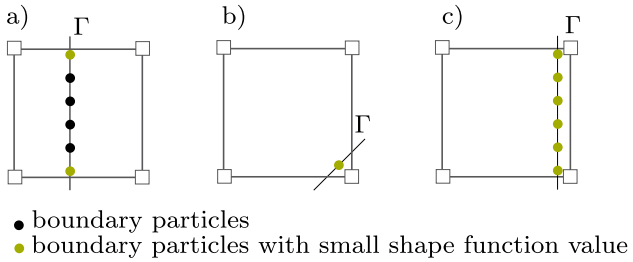

Fig. 8 Possible configurations of boundary particles with small shape function value

highlighted in green. Especially in the configuration 8a, the small values of the shape function are not affecting the condition of the complete stiffness matrix. Therefore the proposed modification of the shape function values is not applied in this case.

Numerical instabilities, however, will appear in the configurations 8b and c where the element is cut nearby a vertex or parallel to one edge or surface in three-dimensional space. As a consequence, the corresponding entries in the global stiffness matrix will be very small compared to the other values leading to ill-conditioned systems. To overcome this issue, the shape function values are modified as proposed by Chandra et al. [39] such that the minimum value is larger than the user-given threshold.

4 Numerical examples

Several examples are presented in this chapter to show the quantitative performance and accuracy of the proposed boundary imposition method. First of all, the proposed method's advantages compared to penalty augmentation are evaluated. Then, the convergence rate of the method is analyzed for different background grid types. Subsequently, the boundary particle distribution is studied, evaluating the reaction forces at the support. The slip condition is validated for a sliding cylinder on an inclined slope in 2D and 3D, comparing the numerical results to the analytic solution. Finally, the application of the proposed method for granular flow is demonstrated by comparing the numerical solution to experimental results and for a possible industrial example.

4.1 Pulled bar

The first example, visualized in Fig. 9 considers a linear elastic cantilever beam under tension loading. The beam has a length of $L = 8\text{m}$, a cross-section of $A = 1 \times 1\text{m}^2$ and a density of $\rho = 1000\text{kg/m}^3$. The numerical models are created in 2D assuming a St. Venant Kirchhoff material with Young's modulus of $E = 90\text{MPa}$ and a Poisson ratio of $\nu = 0.0$ for the constitutive equation. To apply a pure tension load, a horizontal gravitational force of 9.81m/s^2 is considered acting in the x-direction. For a reference solution, the analytic beam solution of the horizontal displacement at point A at the tip of the cantilever is considered.

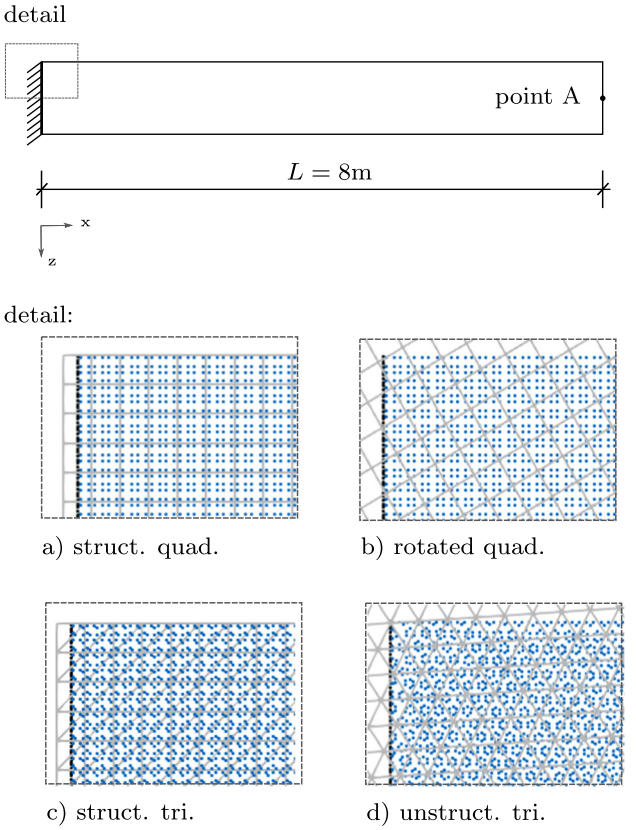

Fig. 9 Cantilever under pure tension loading and a detailed view of the non-conforming boundary condition and the particle discretization within four different background grid types (**a**–**d**)

The clamped support at the left side of the beam is modeled as a non-conforming boundary, imposing the Dirichlet condition weakly either by penalty augmentation [39] or Lagrange multiplier imposition. For the former technique, selecting an appropriate penalty factor is necessary, which is not a trivial task. For each numerical simulation, this factor needs to be calibrated carefully as otherwise, either penetration of the material through the boundary occurs if the factor is chosen too low, or numerical instabilities may appear due to a factor that is too large. Furthermore, the penalty factor introduces a modeling effect in the solution field. Therefore, the obtained solution hardly depends on the selected penalty factor, which is a drawback of the penalty imposition. To show this spring-type behavior of the penalty imposition, the horizontal displacement of point A is calculated with a penalty factor β varying in the range from $\beta = 10^{10}$ to $\beta = 10^{30}$. The results are compared to the solution obtained by the Lagrange multiplier augmentation.

Table 1 Background grid types for the numerical models

(a)	Structured quadrilateral background grid (struct. quad.)
(b)	Rotated quadrilateral background grid (rotated quad.)
(c)	Structured triangular background grid (struct. tri.)
(d)	Unstructured triangular background grid (unstruct. tri.)

Four different background grid types are considered for the following study, summarized in Table 1.

The element size of the background grid is set to 0.04m for all types. For the initialization of the material points, an additional mesh, called body mesh in the following, is created with an element size of 0.02m introducing the material points at the respective Gauss point positions. In the case of a quadrilateral background grid also a quadrilateral body mesh placing 4 particles per element was selected. Whereas, once a triangular body mesh was considered 12 material points per element were initialized. A detail of the final discretized models can be found in Fig. 9a–d. Please notice that for the structured background grids, the grid size was selected such that the imposed boundary is located in the middle of the first row of background grid elements. In case 9b, the initial geometry of the structured grid was rotated by 30° to ensure an arbitrary intersection pattern of the boundary with the computational grid.

As mentioned before, the clamped support is modeled as a non-conforming boundary condition. Therefore, the boundary is divided into line segments with a length of 0.02m placing four equally distributed boundary particles onto each segment which are imposing the Dirichlet condition either by penalty augmentation or Lagrange multiplier imposition.

In Fig. 10a–d, the relative error of the horizontal displacement at point A compared to the analytical solution is plotted against the varying penalty factor for the background grid types displayed in Fig. 9a–d. Additionally, the solution obtained by the proposed Lagrange multiplier imposition is shown in the plots.

As expected, the relative error of the horizontal displacement at point A decreases with an increasing penalty factor until it approaches the solution we obtained with the Lagrange multiplier method. This behavior can be observed for all types of background grids. A further increase in the penalty factor, however, shows a different pattern for the relative error depending on the background grid type. For the structured background grids (a and c), a broad range of penalty factors can be applied to receive similar results compared to the Lagrange multiplier imposition. This, however, is not valid for the unstructured background grid types (b and d). Here, the range of penalty factors to receive results similar to the Lagrange multiplier imposition is rather small. Hence, the difficulty of selecting an appropriate penalty factor can be demonstrated even for this academic example.

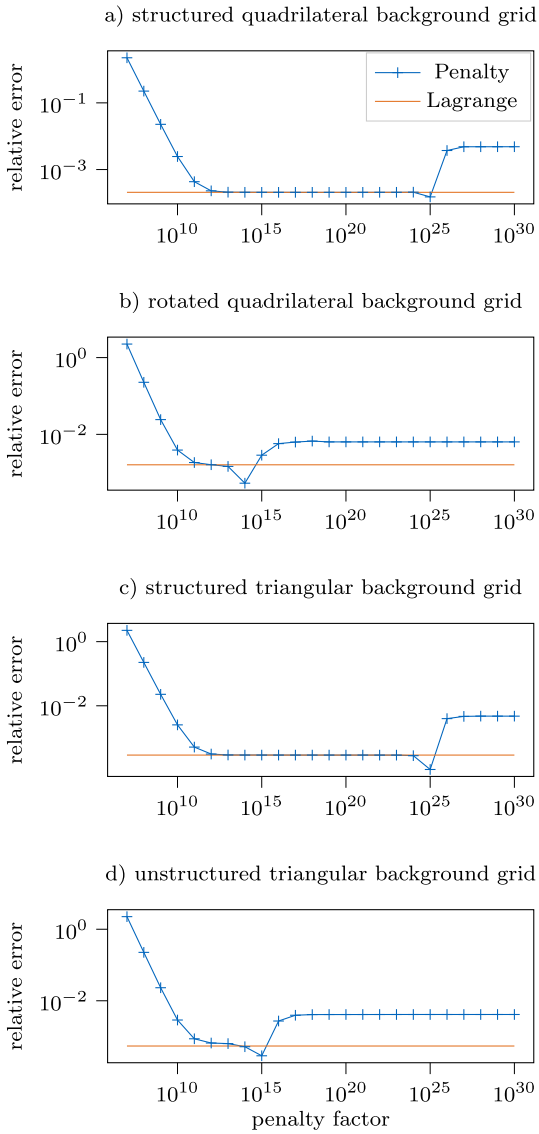

Fig. 10 Comparison of penalty imposition and Lagrange multiplier method

It can be observed that after a further increase of the penalty factor, certain values lead to improved results compared to the solution obtained with the Lagrange multiplier method. However, those values result from plotting the absolute value of the relative error. From here, a further increase in the penalty factor leads to a stiffer behavior compared to the analytical solution and consequently reduces the deformation of the total system. Finally, the relative error converges toward the solution, which would be obtained if all nodes of the background grid elements containing boundary particles would be fixed in space. The increase of the penalty

factor, however, is limited by the conditioning of the stiffness matrix.

In summary, this simple example demonstrates that the appropriate choice of penalty factor is a challenging task. Often, the calibration process leads to trial-and-error estimations where some prior knowledge of the solution is required. Furthermore, it should be noted that for this academic example, there is quite a wide range of feasible penalty factors to obtain reasonable results. However, for more complex simulations, the suitable range of penalty factors may be severely limited to prevent material penetration on the one hand and preserve the numerical accuracy and stability on the other. Therefore, the Lagrange multiplier augmentation is an attractive alternative for the weak imposition of Dirichlet boundary conditions since the time-consuming and challenging task of calibrating the penalty factors is eliminated, and a user-friendly imposition of the essential boundary conditions is enabled.

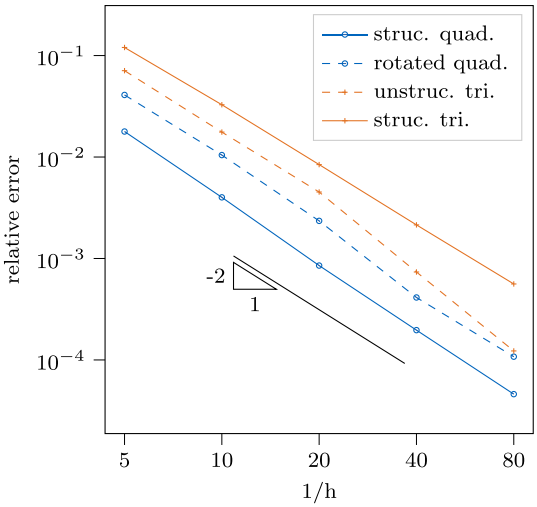

Fig. 11 Mesh convergence study for a cantilever beam. It shows a quadratic convergence rate for both quadrilateral and triangular background meshes for structured and unstructured meshes

4.2 Mesh convergence study

To study the convergence rate of the proposed Lagrange multiplier method, a mesh refinement study is performed. We are considering the same cantilever as the previous example but with the dead load in the global z-direction. The setup of the system, as well as the studied background grid types, can be found in Fig. 9 of the previous example. For the convergence study, again, the mesh types, summarized in Table 1 are considered while a detail of each discretized model can be found in Fig. 9a–d.

For the structured quadrilateral background mesh, the initialization of the mesh is such that the non-conforming boundary condition is imposed at the center-line of the respective background element to ensure a comparable problem setup for different mesh sizes. For the initialization of the material points, a structured quadrilateral body mesh is introduced, placing 16 particles within each element. The size of the body mesh is considered to be half of the background mesh size.

For the unstructured quadrilateral background mesh study, the background grid is rotated by 30° to ensure an arbitrary distribution of the boundary particles within the background grid elements. The body mesh for the initialization of the material points is created analogously to the first study.

To investigate the triangular background mesh, a structured background mesh, and an unstructured background mesh is used. For those studies, the material points are initialized by a triangular body mesh which again has half the size of the body mesh and introduces 12 particles per element at the Gauss-Point positions.

For all configurations, the tip displacement of the cantilever beam at point A is measured, and the relative error compared to the analytical beam solution is calculated. The results are plotted in Fig. 11.

The relative error of the vertical displacement decreases quadratic with a mesh refinement for all considered background grid types.

4.3 Boundary particle distributions and reaction forces

In addition to the previous example, the distribution of the boundary particles within the background grid and the influence on the numerical solution should be investigated. Therefore, again the cantilever beam under dead load (see Fig. 9) with different background grid element types a–d from the previous example is considered. The background grid mesh size of 0.05 m from the previous example is picked, and the distribution of the boundary particles is varied from 40 to 2560. To receive an equal distribution of the boundary particles, the boundary is divided into a varying number of line segments while initializing one boundary particle at each line center.

In Fig. 12 the relative error of the vertical displacement at point A is plotted against the number of boundary particles. One can observe that for all types of background grid elements, the results are independent of the number of boundary particles, assuming a sufficient amount of boundary particles to enforce the boundary condition. Obviously, if the boundary were discretized even more coarsely by boundary particles, various background grid elements would not contain any boundary particles even though they are intersected by the boundary, and thus the constraint imposition would be insufficient.

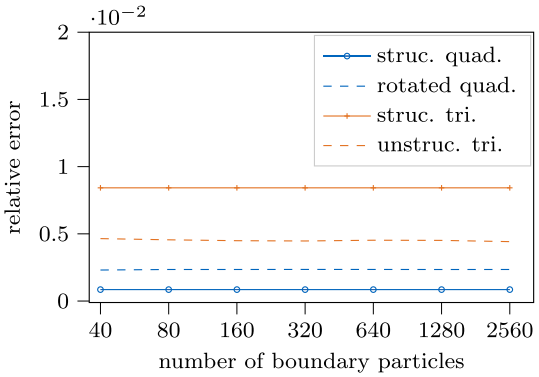

Fig. 12 Convergence study for boundary particle refinement considering the four types of background grids with a constant element size of 0.05 m. The relative error of the vertical displacement at point A of the cantilever is measured

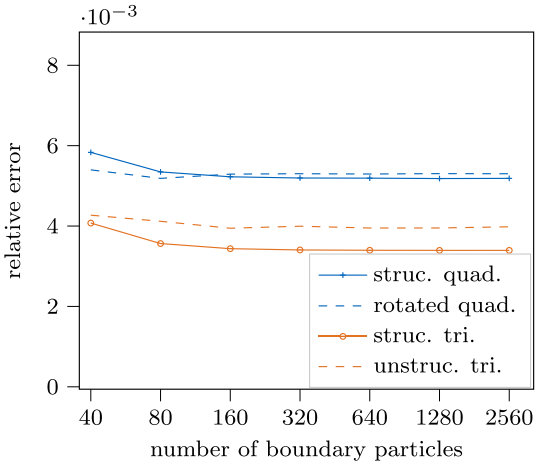

Fig. 13 Convergence study for boundary particle refinement considering the four types of background grids with a constant size of 0.05 m. The relative error of the resulting bending moment M_b at the clamped support of the cantilever is measured

A similar study is performed for the resulting bending moment, which can be calculated from the reaction forces. For the considered linear elastic beam under dead load $g = 9.81\text{m/s}^2$, the resulting analytical bending moment M_b at the boundary can be calculated as follows:

$$M_b = \frac{1}{2} g \cdot \rho \cdot A \cdot L^2 = 313.92[\text{kNm}]. \tag{41}$$

The resulting numerical solution is obtained by calculating the reaction forces at the boundary particles (see the concept described in Sect. 3.5) and multiplying each force with the respective distance of the boundary particle to the centerline of the beam. The relative error compared to the analytic results are plotted in Fig. 13 against the number of boundary particles. It can be observed that by boundary particle refinement the solution converges as expected to a constant value.

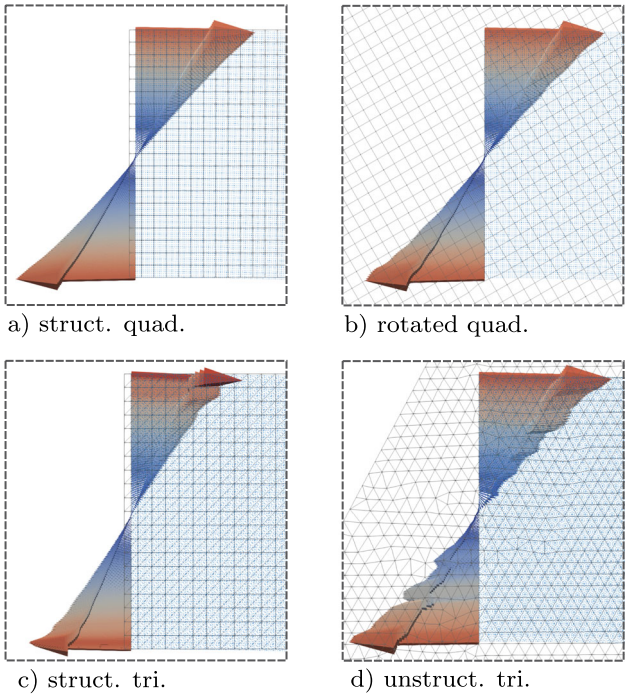

Fig. 14 Reaction forces for the cantilever under dead load for the four different background grid types with a size of 0.05 m

Additionally, the reaction forces at the boundary particles are visualized in Fig. 14 for the different background grid element types a–d.

The results show an excellent agreement with the theoretical results. The forces' distribution at the boundary matches well with the expected solutions. Especially, the weighting procedure, presented in Sect. 3.5 is important so that the resulting Lagrange multiplier forces, which are visualized in Figs. 3a, b and 5a, b are distributed to the boundary particles.

4.4 Cylinder on inclined slope

For the validation of the friction-less sliding condition imposed by the Lagrange multiplier method, a rigid cylinder with a radius r = 1m and a width of b = 1m is chosen. It slips down an inclined slope of 60°, enforced by gravity. The material model for the cylinder is considered to be linear elastic with a high stiffness E = 2*e*9Pa being a numerically rigid body, and a density of $\rho = 7800 \frac{\text{kg}}{\text{m}^3}$. For pure sliding conditions, the displacement of the cylinder can be determined analytically. Therefore, this example is used frequently for validation purposes in various literature (e.g. [39, 66]). Within this example, we are studying the displacement of the cylinder for different background grid element types in two- and three-dimensional space, while the non-conforming slip boundary condition is imposed by the proposed Lagrange multiplier method. In Fig. 15 the 2D system of the cylinder is visualized, including a detailed view of four different

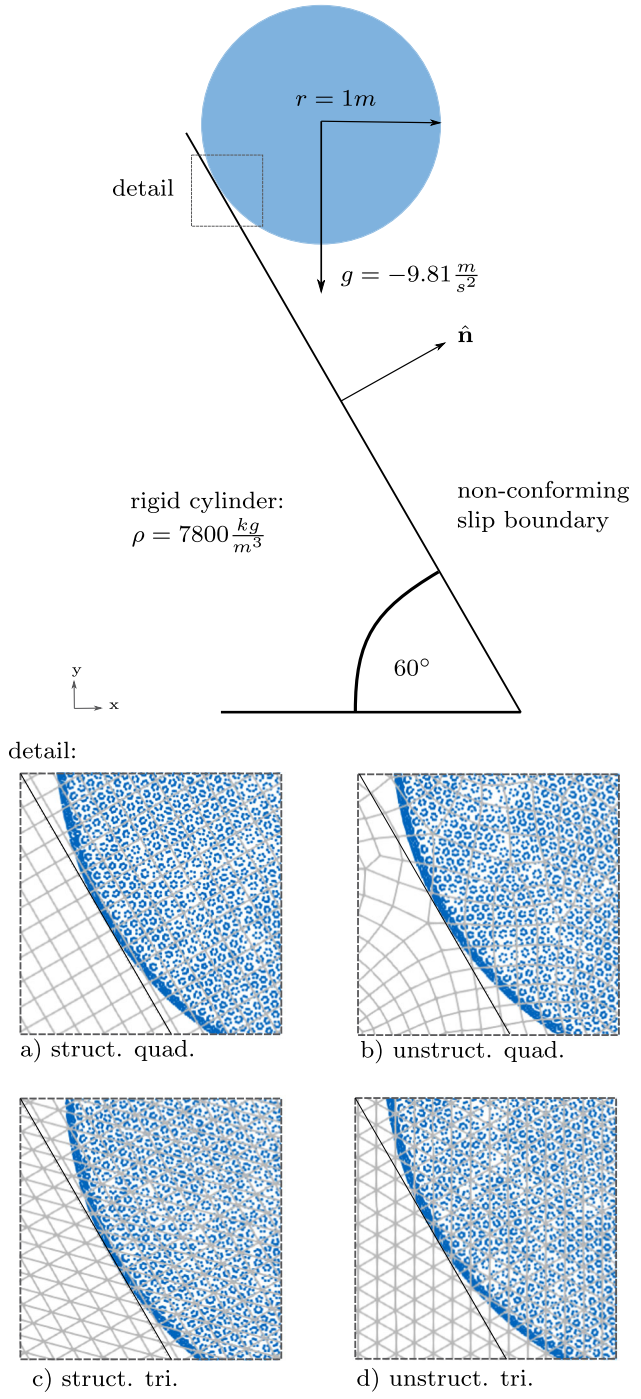

Fig. 15 Cylinder sliding down inclined slope in 2D and a detailed view of the non-conforming boundary condition within four different background grid types

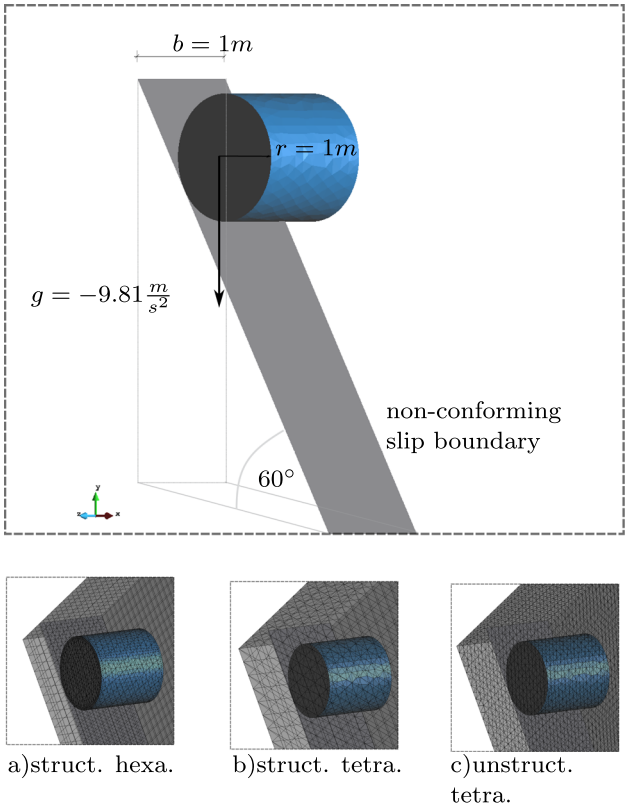

Fig. 16 Cylinder sliding down inclined slope in 3D and a detailed view of the non-conforming boundary condition within three different background grid types

background grid types, visualizing the definition of the non-conforming boundary condition. In Fig. 16 is sketched the three-dimensional model with three zoomed regions visualizing the considered background grid types and the definition of the non-conforming boundary condition.

Similar to the previous validation examples, a structured (Fig. 15a) and unstructured quadrilateral (Fig. 15b), as well as a structured (Fig. 15c) and unstructured triangular (Fig. 15d) background grid, are chosen for the two-dimensional space, resulting in four case studies where the slip condition is imposed as a non-conforming boundary. A mesh size of 0.12 m is considered for the background grids, whereas for the initialization of the material points an unstructured trian-

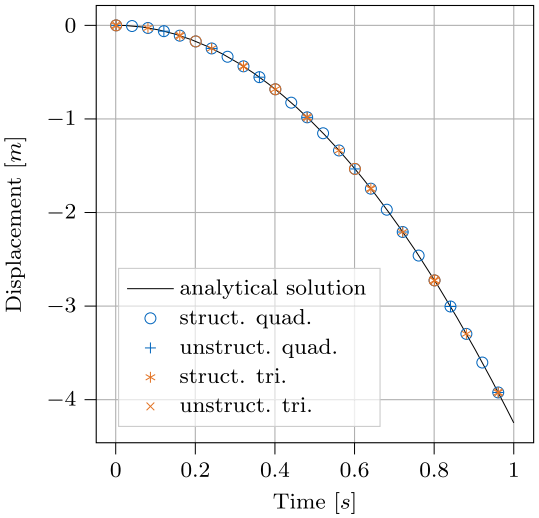

Fig. 17 Cylinder on an inclined plane: simulation results for frictionless slip condition in comparison to the analytical solution for different mesh types in 2D

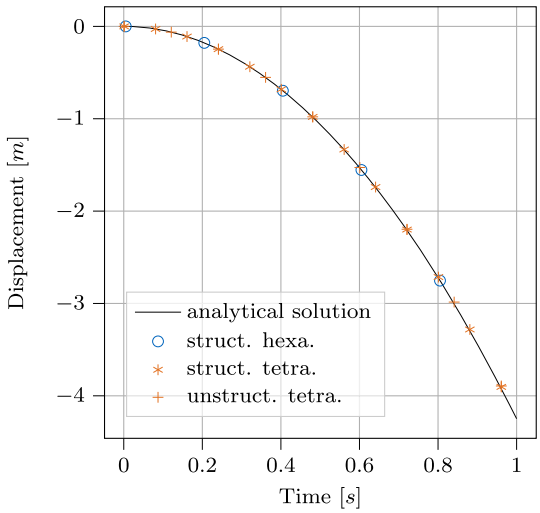

Fig. 18 Cylinder on an inclined plane: simulation results for frictionless slip condition in comparison to the analytical solution for different mesh types in 3D

gular body mesh with the size of 0.06 m is selected placing three particles per element. For the numerical simulation, a time step of $\Delta t = 0.001$s is selected and the displacement of the cylindrical center is plotted for 1s. The results in comparison with the analytical solutions are plotted in Fig. 17 and show a great agreement with the analytical solution, independently of the considered element type.

Within 3D, the numerical results are studied for three different background grid element types. The first case (Fig. 16a) considers a structured hexahedral background mesh with a size of 0.12 m, where the non-conforming boundary is imposed centered within the respective elements. For the tetrahedral elements the size was chosen to be 0.12 m. A structured (Fig. 16b) and an unstructured tetrahedral (Fig. 16c) background grid is studied. The final results are plotted in Fig. 18 in comparison with the analytical displacements, proofing the accuracy of the proposed method also for 3D simulations.

4.5 Granular column collapse

In this validation case, a granular column collapse of non-cohesive soil is simulated according to the two-dimensional experiments conducted by [67]. The setup of the experiment is visualized in Fig. 19.

It shows the granular material at its initial position, confined to a region of $20 \times 10 \times 5\text{cm}^3$, and after releasing flows down due to gravity. During the experiment, the configuration of the material was captured by a high-speed camera at different times until the final run-out of the material. Based on the experimental data, as well as the numerical SPH simulations done by Bui et al. [68], a non-associated elasto-plastic material model is chosen, assuming the Mohr-Coulomb yield criterion with finite strain assumption. The material parameters of the granular material are summarized in Table 2.

The left and bottom side walls of the considered experimental setup are modeled with a non-slip boundary condition. To show the performance of the newly proposed Lagrange multiplier method for granular material with large topological changes, those boundaries are imposed by non-conforming boundary conditions. Therefore, boundary particles are introduced along the left and bottom side walls. In Fig. 19 the zoomed region indicates the boundary particle distribution within the background grid.

For the computational background grid, structured quadrilateral elements with a size of 0.002 m are chosen, whereas for the initialization of the material points a structured quadrilateral body mesh with the size of $5e{-}4$ m is introduced, placing 4 material points at the respective Gauss point positions. The boundary is divided into line segments of $5e - 4$ m, initializing three boundary particles each. The simulation runs until the final run out with a time step of $\Delta t = 5e{-}5$s.

In Fig. 20, the obtained numerical results are compared to the experimental results obtained by [67] at different times. It can be observed that the deformation pattern, as well as the runout, matches very well. In accordance with the SPH simulations conducted by [67] the material is still moving at time $t = 0.35$ s and reaches a final run out of 0.55 m at the end, which as well fits the final material configuration of the experiment. In addition to the material configuration at different times, the failure line of the numerical simulations is compared to the experimental results. A very good agreement can be observed. However, the failure line at the bottom cannot be reproduced accurately by considering a

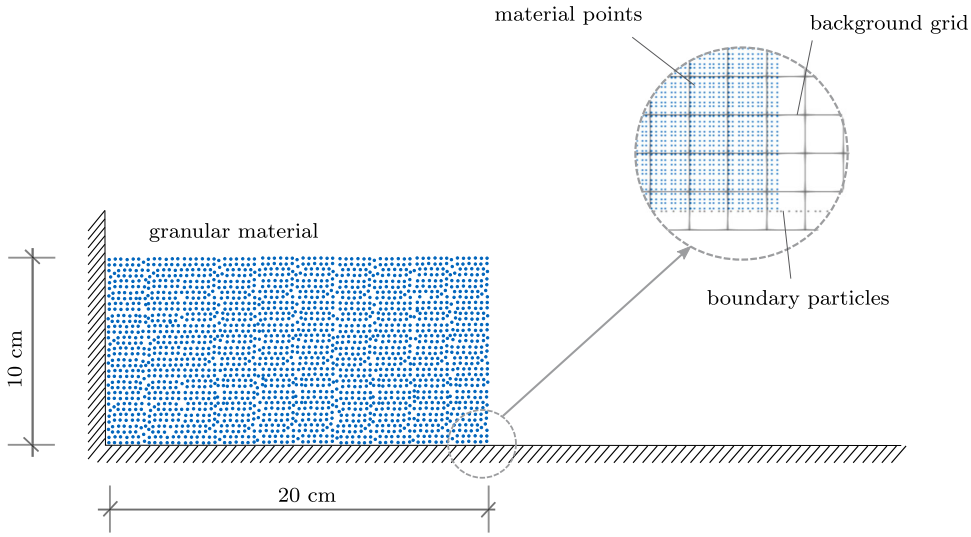

Fig. 19 Simulation model of the granular column collapse experiment with a detailed view of the discretized model and the non-conforming boundary condition

Table 2 Parameters for the granular material

Young's modulus	Poisson's ratio	density
$E = 5.84\text{MPa}$	$\nu = 0.3$	$\rho = 20.4\frac{\text{kN}}{\text{m}^3}$
internal friction angle	cohesion	dilatation angle
$\Phi' = 21.9°$	$c' = 0.0\text{kPa}$	$\Psi = 0.0°$

fixed boundary imposition within the numerical model. To capture this phenomenon in the numerical model, the friction at the boundary needs to be considered, which is not part of the scope of the presented research.

4.6 Cone filled with granular material and moving boundary

The final example in this paper demonstrates the advantages of the proposed method for industry-scale applications. This study considers a cone filled with granular material, while at the bottom, the gravity-driven material can flow out. During the simulation, the cone is moving upwards with a constant velocity of $v_0 = 0.2\frac{\text{m}}{\text{s}}$. The example setup is visualized in Fig. 21 with a zoomed region at the corner to show the discretization of the system. A quadrilateral structured background grid with a size of 0.03 m is used for the computation. An unstructured triangular body mesh with size 0.02 m is created to initialize the material points, placing 16 particles into each element. At the non-conforming boundary, which is imposed as a slip condition, 4 boundary particles are initialized within each line segment with an initial length of 0.02 m. A time step of $\Delta t = 1e{-}3\text{s}$ is considered for the computation.

To represent the elastoplastic behavior of dry soil, the granular material properties were selected to be the same as in the preceding case (see Table 2).

The solution of the numerical simulation is visualized in Fig. 22 showing the material flow for six specific times until the cone is emptied and the complete material rests on the ground. The bottom is modeled as a fixed conforming boundary condition. The boundary particles, imposing the non-conforming slip condition at the outline of the cone, are moving in total 1.2 m in the vertical direction, corresponding to the imposed constant velocity for 6s in total.

This example demonstrates the broad application range of the newly proposed boundary imposition method by the Lagrange multiplier method, as the boundary can be defined independently of the computational background grid. In addition, arbitrarily directed moving boundary conditions may be imposed robustly for both slip and fixed conditions. This feature is highly advantageous for MPM, where the computational background grid is usually kept stationary while the material moves through the grid. Additionally, unlike the penalty approach, which frequently requires user calibration, the Lagrange multiplier imposition does not require the user to choose the problem-dependent penalty factor [39]. This makes the Lagrange multiplier method superior for a straightforward imposition of the boundary condition.

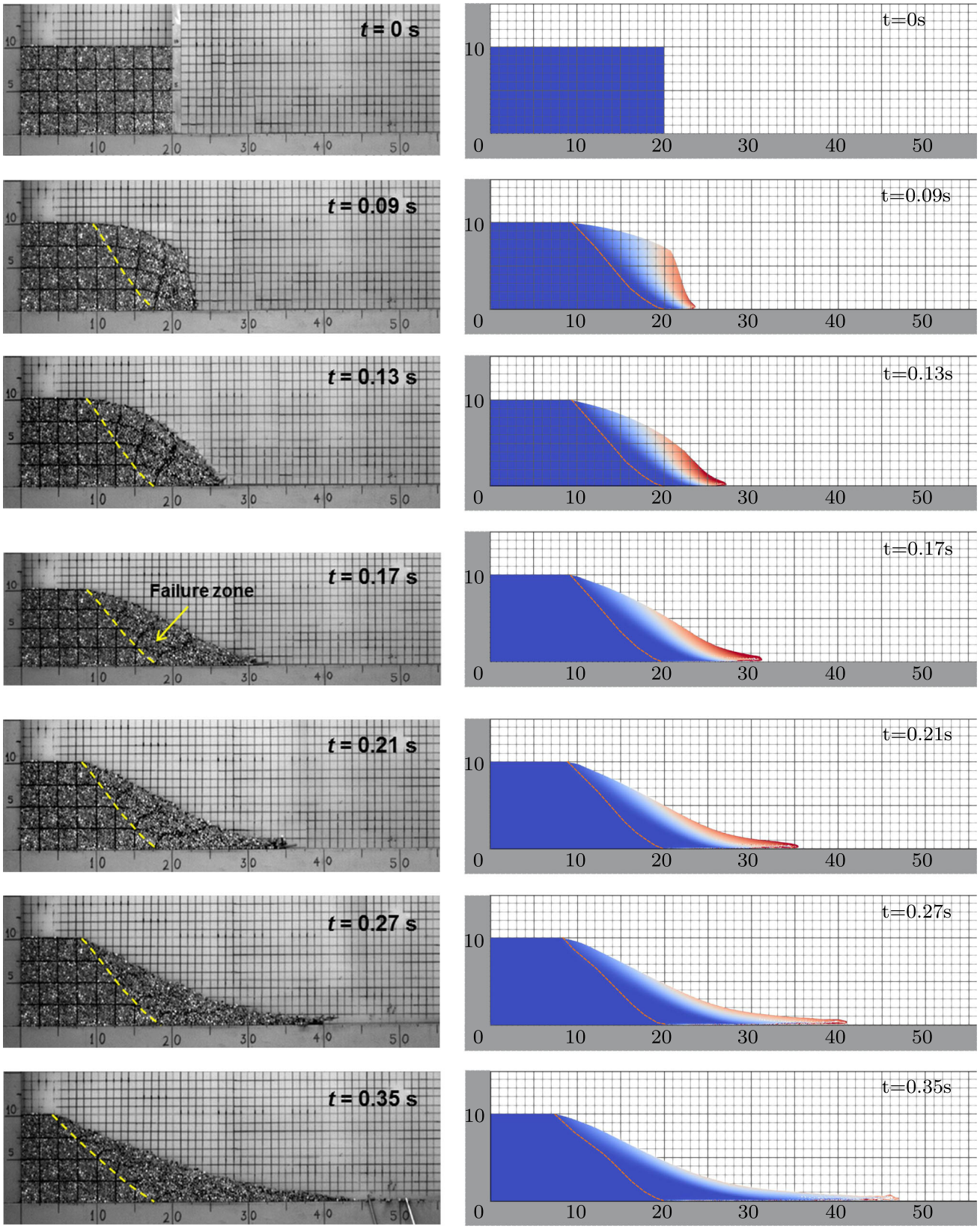

Fig. 20 Numerical results in comparison with the experimental results published by Nguyen et al. [67] for specific times

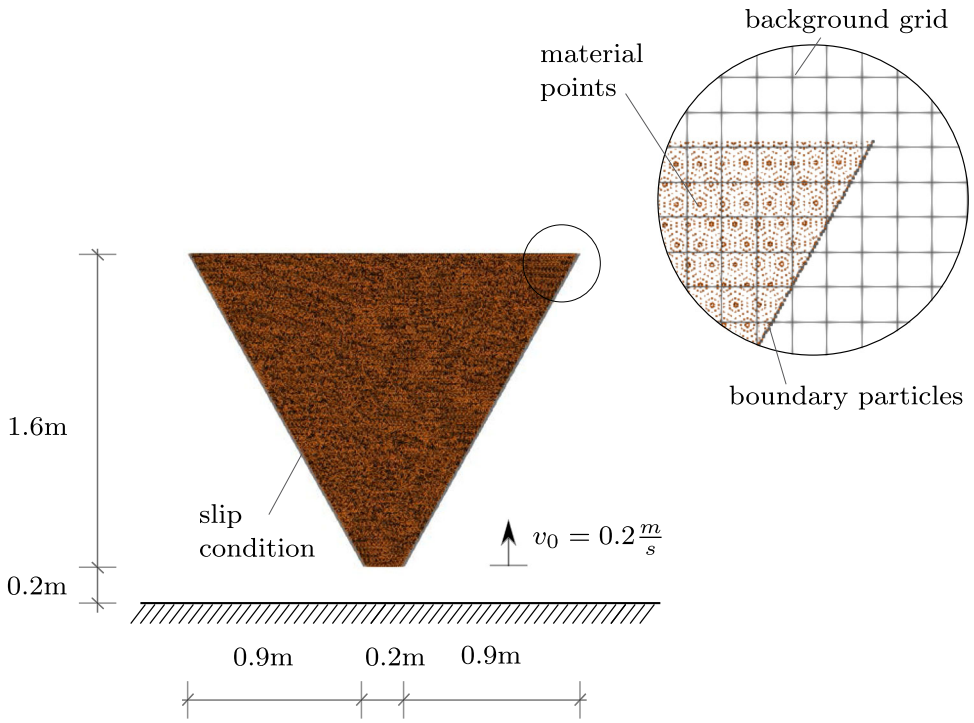

Fig. 21 Cone filled with granular material. The cone outline is modeled as a non-conforming slip condition with prescribed velocity

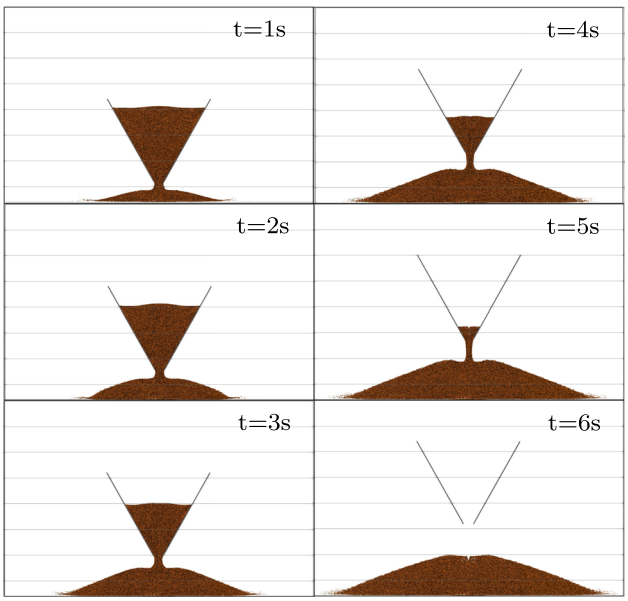

Fig. 22 Simulation results for the specific times 1s to 6s. After 6s the complete granular material is released from the cone

5 Conclusion

This paper presents the Lagrange multiplier method for implicit MPM to weakly impose non-conforming essential boundary conditions in 2D and 3D. With a novel discretization scheme, the method has been employed robustly and efficiently. The boundary geometry is discretized by massless boundary particles that move in accordance with the boundary deformation through the Eulerian computational background grid tracing the shape of the boundary geometry throughout the simulation. However, in combination with the discretization of the physical domain by material points which are as well moving through the computational background grid, the definition of an inf-sub stable Lagrange multiplier field becomes a challenging task as both the set of active background grid cells as well as the interaction between the boundaries and the material points may change at every time step. A local element-wise formulation of the constraints is proposed to robustly impose non-conforming boundary conditions with the Lagrange multiplier method in MPM to cope with that.

As typically applied in MPM, the displacement field is discretized by element-wise simple interpolation functions, whereas a constant approximation of the Lagrange multiplier field within the intersected elements is chosen (see Sect. 3.2.1). From an implementation-driven point, this approach is very attractive, as each affected background grid element is assigned an additional dof for the Lagrange multipliers. Consequently, the contributions of the boundary particles are considered automatically in the assembly procedure.

However, it is demonstrated that this approach may result in spurious oscillations in the Lagrange multiplier solution, as the mesh-dependent inf-sub condition is not necessarily fulfilled. Therefore, a procedure is developed to eliminate superfluous constraints which leads to a suitable Lagrange multiplier field (see Sect. 3.2.2). Consequently,

the resulting forces (being the respective Lagrange multipliers) are correctly representing the global stress distribution at the boundary. Due to the constant approximation of the Lagrange multiplier field, those forces are constant within the respective background grid elements. Therefore, a weighting procedure is presented to calculate the respective reaction forces at the boundary particles (see Sect. 3.5).

The proposed methodologies are also extended for frictionless slip conditions, which may be oriented arbitrarily within space (see Sect. 3.4). On top of that, prescribed kinematic values may also be imposed, leading to moving boundaries within space.

Several numerical examples with increasing complexity have been simulated to assess the quality of the proposed work. The first example demonstrates the advantages of the proposed method compared to the penalty augmentation [39] since the time-consuming and difficult task of calibrating the problem-dependent penalty factors is eliminated. Instead, a more user-friendly imposition of non-conforming essential boundary conditions is enabled.

Additional examples prove the accuracy, as well as the convergence rate of the newly proposed method, by comparing the numerical results to analytical reference solutions. Furthermore, the correctness of the frictionless slip condition is demonstrated within 2D and 3D problems. The comparison with experimental results from the literature confirms the validity of the proposed method, as well for the elastoplastic regime. Finally, the advantages of the proposed method for industrial applications are demonstrated.

It should be noted that for many simulations, the cell-crossing error induced by the use of simple element-wise shape functions of the primal variable can be reduced by particle refinement, as applied to the examples herein, and is thus reduced at low computational cost. However, for examples where this approximation reaches its limit, higher-order shape functions are required to approximate the displacement field. Therefore, for future research, the proposed Lagrange multiplier imposition of non-conforming essential boundary conditions in implicit MPM should be extended to variants of MPM that use higher-order shape functions for the primal variable instead. A crucial task for this extension is to adjust the discretization and interpolation order of the Lagrange multiplier field accordingly in order to satisfy the inf-sub condition while maintaining computational efficiency. Moreover, the increased influence domain of the boundary needs to be considered and studied carefully for this extension, especially for configurations of the material being apart from the boundary and approaching during simulation time.

To conclude, the proposed enhancements are a powerful methodology to impose non-conforming boundary conditions in immersed methods by the Lagrange multiplier method, making MPM more comprehensive for the simulation of large and rapid deformation scenarios. This is of utmost importance for the predictive simulation and assessment of mass movement hazards, including landslides, debris flows, and avalanches.

Acknowledgements This work was supported by the Hans Fischer Fellowship program of the Institute of Advanced Studies of the Technical University of Munich, Germany. This study was carried out within the RETURN Extended Partnership and received funding from the European Union Next-GenerationEU (National Recovery and Resilience Plan – NRRP, Mission 4, Component 2, Investment 1.3 – D.D. 1243 2/8/2022, PE0000005). The authors would like to acknowledge the support of Indam (Istituto Nazionale di Alta Matematica "Francesco Severi").

Funding Open Access funding enabled and organized by Projekt DEAL.

Code Availability For this work, the open-source multiphysics software *KRATOS* [69–71] has been used, which is written in C++ and offers a Python interface. The current version can be obtained from [72].

Declarations

Conflict of interest The authors declare that they have no competing interests.

References

1. Sulsky D, Chen Z, Schreyer H (1994) A particle method for history-dependent materials. Comput Methods Appl Mech Eng 118:179–196
2. Lucy LB (1977) A numerical approach to the testing of the fission hypothesis. Astron J 82:1013–1024
3. Belytschko T, Lu YY, Gu L (1994) Element-free galerkin methods. Int J Numer Meth Eng 37:229–256
4. Liu WK, Jun S, Zhang YF (1995) Reproducing kernel particle methods. Int J Numer Meth Fluids 20:1081–1106
5. Oñate E, Idelsohn SR, Del Pin F, Aubry R (2004) The particle finite element method. An overview. Int J Comput Methods 1:267–307
6. Cremonesi M, Franci A, Idelsohn S, Oñate E (2020) A state of the art review of the particle finite element method (pfem). Arch Comput Methods Eng 27:1709–1735
7. Cremonesi M, Meduri S, Perego U (2020) Lagrangian–Eulerian enforcement of non-homogeneous boundary conditions in the particle finite element method. Comput Part Mech 7:41–56
8. Meduri S, Cremonesi M, Frangi A, Perego U (2022) A Lagrangian fluid-structure interaction approach for the simulation of airbag deployment. Finite Elem Anal Des 198:103659

9. Chen J-S, Hillman M, Chi S-W (2017) Meshfree methods: progress made after 20 years. J Eng Mech 143:04017001
10. Ma S, Zhang X, Qiu X (2009) Comparison study of MPM and SPH in modeling hypervelocity impact problems. Int J Impact Eng 36:272–282
11. Zhang X, Sze K, Ma S (2006) An explicit material point finite element method for hyper-velocity impact. Int J Numer Meth Eng 66:689–706
12. Wilson P (2022) A computational impact analysis approach leveraging non-conforming spatial, temporal and methodological discretisations
13. Guilkey J, Lander R, Bonnell L (2021) A hybrid penalty and grid based contact method for the material point method. Comput Methods Appl Mech Eng 379:113739
14. Andersen S, Andersen L (2010) Modelling of landslides with the material-point method. Comput Geosci 14:137–147
15. Nguyen TS et al (2022) Post-failure process and kinematic behavior of two landslides: case study and material point analyses. Comput Geotech 148:104797
16. Gaume J, van Herwijnen A, Gast T, Teran J, Jiang C (2019) Investigating the release and flow of snow avalanches at the slope-scale using a unified model based on the material point method. Cold Reg Sci Technol 168:102847
17. Zhang H, Wang K, Chen Z (2009) Material point method for dynamic analysis of saturated porous media under external contact/impact of solid bodies. Comput Methods Appl Mech Eng 198:1456–1472
18. Bandara SS (2013) Material point method to simulate large deformation problems in fluid-saturated granular medium. Ph.D. thesis, University of Cambridge Cambridge, UK
19. Yerro Colom A, Alonso Pérez de Agreda E, Pinyol Puigmartí N M (2015) The material point method for unsaturated soils. Géotechnique 65:201–217
20. Singer V, Sautter KB, Larese A, Wüchner R, Bletzinger K-U (2022) A partitioned material point method and discrete element method coupling scheme. Adv Model Simul Eng Sci 9:16. https://doi.org/10.1186/s40323-022-00229-5
21. Bardenhagen SG, Kober EM (2004) The generalized interpolation material point method. Comput Model Eng Sci 5:477–496
22. Sadeghirad A, Brannon RM, Burghardt J (2011) A convected particle domain interpolation technique to extend applicability of the material point method for problems involving massive deformations. Int J Numer Meth Eng 86:1435–1456
23. Sadeghirad A, Brannon RM, Guilkey J (2013) Second-order convected particle domain interpolation (CPDI2) with enrichment for weak discontinuities at material interfaces. Int J Numer Meth Eng 95:928–952
24. Wilson P, Wüchner R, Fernando D (2021) Distillation of the material point method cell crossing error leading to a novel quadrature-based c0 remedy. Int J Numer Meth Eng 122:1513–1537
25. Gan Y, Sun Z, Chen Z, Zhang X, Liu Y (2018) Enhancement of the material point method using b-spline basis functions. Int J Numer Meth Eng 113:411–431
26. Sołowski WT et al (2021) In: Bordas SP, Balint DS (eds) Chapter two - material point method: overview and challenges ahead. Vol. 54 of advances in applied mechanics, pp 113–204, Elsevier. URL https://www.sciencedirect.com/science/article/pii/S0065215620300120
27. de Vaucorbeil A, Nguyen VP, Sinaie S, Wu JY (2020) In: Bordas SP, Balint DS (eds) Chapter two - material point method after 25 years: theory, implementation, and applications. Vol. 53 of advances in applied, mechanics, pp 185-398. Elsevier URL https://www.sciencedirect.com/science/article/pii/S0065215619300146
28. Zhang X, Chen Z, Liu Y (2016) The material point method: a continuum-based particle method for extreme loading cases. Academic Press, Oxford
29. Fern J, Rohe A, Soga K, Alonso E (2019) The material point method for geotechnical engineering. A practical guide. CRC Press, Boca Raton
30. Nguyen VP, de Vaucorbeil A, Bordas S (2023) The material point method. Springer International Publishing, Cham
31. Cortis M et al (2018) Imposition of essential boundary conditions in the material point method. Int J Numer Meth Eng 113:130–152
32. Kafaji IKJ (2013) Formulation of a dynamic material point method (MPM) for geomechanical problems. Ph.D. thesis, University of Stuttgart
33. Parvizian J, Düster A, Rank E (2007) Finite cell method: h-and p-extension for embedded domain problems in solid mechanics. Comput Mech 41(1):121–133
34. Düster A, Parvizian J, Yang Z, Rank E (2008) The finite cell method for three-dimensional problems of solid mechanics. Comput Methods Appl Mech Eng 197:3768–3782
35. Breitenberger M, Apostolatos A, Philipp B, Wüchner R, Bletzinger K-U (2015) Analysis in computer aided design: nonlinear isogeometric b-rep analysis of shell structures. Comput Methods Appl Mech Eng 284:401–457 (**Isogeometric Analysis Special Issue**)
36. Babuska I (1973) The finite element method with penalty. Math Comput 27:221–228
37. Kollmannsberger S et al (2015) Parameter-free, weak imposition of dirichlet boundary conditions and coupling of trimmed and non-conforming patches. Int J Numer Meth Eng 101:670–699
38. Teschemacher T et al (2018) Realization of cad-integrated shell simulation based on isogeometric b-rep analysis. Adv Model Simul Eng Sci 5:1–54
39. Chandra B, Singer V, Teschemacher T, Wüchner R, Larese A (2021) Nonconforming dirichlet boundary conditions in implicit material point method by means of penalty augmentation. Acta Geotech 16(8):2315–2335
40. Larsson K, Kollmannsberger S, Rank E, Larson MG (2022) The finite cell method with least squares stabilized nitsche boundary conditions. Comput Methods Appl Mech Eng 393:114792
41. Apostolatos A, Schmidt R, Wüchner R, Bletzinger K-U (2014) A nitsche-type formulation and comparison of the most common domain decomposition methods in isogeometric analysis. Int J Numer Meth Eng 97:473–504
42. Wilson P, Teschemacher T, Bucher P, Wüchner R (2021) Non-conforming fem-fem coupling approaches and their application to dynamic structural analysis. Eng Struct 241:112342
43. Horger T, Kollmannsberger S, Frischmann F, Rank E, Wohlmuth B (2014) A new mortar formulation for modeling elastomer bedded structures with modal-analysis in 3d. Adv Model Simul Eng Sci 1:1–19
44. Brivadis E, Buffa A, Wohlmuth B, Wunderlich L (2015) Isogeometric mortar methods. Comput Methods Appl Mech Eng 284:292–319
45. Babuška I (1973) The finite element method with lagrangian multipliers. Numerische Mathematik 20:179–192
46. Barbosa H, Hughes T (1991) The finite element method with lagrange multipliers on the boundary: circumventing the babuška-brezzi condition. Comput Methods Appl Mech Eng 85:109–128
47. Béchet E, Moës N, Wohlmuth BI (2009) A stable lagrange multiplier space for stiff interface conditions within the extended finite element method. Int J Numer Methods Eng 78(8):931–954
48. Hautefeuille M, Annavarapu C, Dolbow JE (2012) Robust imposition of dirichlet boundary conditions on embedded surfaces. Int J Numer Meth Eng 90:40–64. https://doi.org/10.1002/nme.3306
49. Zienkiewicz OC, Taylor RL (1977) The finite element method, vol 36. McGraw-hill, London

50. Finlayson BA (2013) The method of weighted residuals and variational principles (Society for Industrial and Applied Mathematics, Philadelphia, PA https://doi.org/10.1137/1.9781611973242
51. Walter W, Walter DP (2002) Mechanics of structures: variational and computational methods. CRC Press https://www.taylorfrancis.com/books/mono/10.1201/9781420041835/mechanics-structures-walter-wunderlich-walter-pilkey
52. Newmark NM (1959) A method of computation for structural dynamics. J Eng Mech Div 85:67–94. https://doi.org/10.1061/JMCEA3.0000098
53. Iaconeta I, Larese A, Rossi R, Guo Z (2017) Comparison of a material point method and a galerkin meshfree method for the simulation of cohesive-frictional materials. Materials 10(10):1150
54. Bonet J, Wood RD (1997) Nonlinear continuum mechanics for finite element analysis. Cambridge University Press, Cambridge
55. Belytschko T, Liu WK, Moran B, Elkhodary K (2013) Nonlinear finite elements for continua and structures. John wiley & sons, Chichester
56. Wriggers P (2008) Nonlinear finite element methods. Springer Science & Business Media, Berlin, Heidelberg
57. Iaconeta I, Larese A, Rossi R, Guo Z (2017) Comparison of a material point method and a Galerkin Meshfree method for the simulation of cohesive-frictional materials. Materials 10:1150
58. Iaconeta I, Larese A, Rossi R, Oñate E (2018) A stabilized mixed implicit material point method for non-linear incompressible solid mechanics. Comput Mech 63:1243–1260
59. Iaconeta I (2019) A Discrete-continuum hybrid modelling of flowing and static regimes. Ph.D. thesis
60. Babuška I (1971) Error-bounds for finite element method. Numer Math 16:322–333. https://doi.org/10.1007/BF02165003
61. Brezzi F (1974) On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. R.A.I.R.O. Analyse Numérique 8:129–151
62. Brezzi F, Fortin M (1991) Mixed and hybrid finite element methods. Springer, New York
63. Dervieux A, Thomasset F (1980) In: Rautmann RA (ed) finite element method for the simulation of a rayleigh-taylor instability. *Approximation Methods for Navier-Stokes Problems*, vol 771 of *Lecture Notes in Mathematics*, pp 145–158. Springer Berlin Heidelberg, Berlin, Heidelberg,
64. Wriggers P (2009) Mixed finite element methods - theory and discretization. Springer Vienna, Vienna. https://doi.org/10.1007/978-3-211-99094-0_5
65. Hughes T (1987) The finite element method: linear static and dynamic finite element analysis. Prentice-Hail, Englewood Cliffs
66. Bardenhagen S, Brackbill J, Sulsky D (2000) The material-point method for granular materials. Comput Methods Appl Mech Eng 187:529–541
67. Nguyen CT, Bui HH, Fukagawa R (2015) Failure mechanism of true 2d granular flows. J Chem Eng Jpn 48:395–402
68. Nguyen CT, Nguyen CT, Bui HH, Nguyen GD, Fukagawa R (2017) A new sph-based approach to simulation of granular flows using viscous damping and stress regularisation. Landslides 14:69–81
69. Dadvand P, Rossi R, Oñate E (2010) An object-oriented environment for developing finite element codes for multi-disciplinary applications. Arch Comput Methods Eng 17:253–297
70. Dadvand P et al (2013) Migration of a generic multi-physics framework to HPC environments. Comput Fluids 80:301–309
71. Ferrándiz VM et al (2020) KratosMultiphysics (Version 8.0). Zenodo
72. Kratos multiphysics. https://github.com/KratosMultiphysics/Kratos