A continuum consistent discrete particle method for continuum–discontinuum transitions and complex fracture problems

S.O. Sperling\textsuperscript{a}, J.P.M. Hoefnagels\textsuperscript{a}, K. van den Broek\textsuperscript{b}, M.G.D. Geers\textsuperscript{a,}\textsuperscript{*}

\textsuperscript{a} Mechanics of Materials, Department of Mechanical Engineering, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands
\textsuperscript{b} VDL ETG Technology and Development, De Schakel 22, 5651 GH Eindhoven, The Netherlands

Received 14 July 2021; received in revised form 2 December 2021; accepted 6 December 2021

Available online 1 January 2022

Abstract

In numerical simulations where complex fracture behavior plays a prominent role in the material’s mechanical behavior, particle methods are an attractive computational tool since they adequately accommodate arbitrary discontinuities. However, existing particle methods are either limited in their constitutive flexibility, like the Discrete Element Method (DEM), or prone to instabilities, like Smoothed Particle Hydrodynamics (SPH) and Peridynamics. In this paper we present an alternative particle formulation, referred to as the Continuum Bond Method (CBM). The method has the same constitutive flexibility as conventional continuum methods like the Finite Element Method (FEM), while still being able to incorporate arbitrary discontinuities as in particle methods like DEM, SPH and Peridynamics. In CBM, the continuum body is divided into a series of material points where each material point carries a fraction of the body’s mass. A triangulation procedure establishes the bonds between the particles that interact with each other. The deformation gradient tensor is determined via a volume weighted averaging procedure over the volumes spanned by pairs of nearest neighboring particles. The obtained approximation of the continuum deformation field on the particles allows for a straightforward implementation of continuum constitutive laws. To assess this property in CBM, simulation outcomes for an elastic nonlinear plastic tensile bar are compared to FEM and SPH results. While the stress–strain curves obtained by FEM, CBM and SPH coincide quite accurately, it is found that the local plastic strains obtained by CBM are much closer to the FEM reference solution than the SPH results. The ability of CBM to account for arbitrary discontinuities is demonstrated via a series of dynamic fracture simulations. It is shown that, without the need of additional crack tracking routines, CBM can account for fracture instability phenomena like branches. In conclusion, CBM is suitable for the implementation of continuum constitutive behavior while maintaining the advantageous discontinuous fracture properties of particle methods.

© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Keywords: DEM; SPH; FEM; Particle method; Fracture

1. Introduction

Material models involving damage and fracture are widely used to perform lifetime predictions for structural components. In the past decades, a lot of research has been dedicated to developing computational schemes that
include fracture in a natural manner. It has been shown that two-dimensional crack propagation problems can be simulated quite accurately. However, capturing more complex fracture events like crack bifurcation, coalescence, dynamic effects and three-dimensional influences is more challenging. Many options for fracture simulations exist, although, a method’s flexibility to incorporate arbitrary discontinuities often interferes with the adopted continuum description. The resulting implications will be reviewed in more detail for some of the most common methods in the literature.

The Finite Element Method (FEM) is the most widely employed numerical method for structural mechanics problems in research and industry. Several adaptions, e.g. element erosion and interface discontinuity methods, have been proposed to enhance FEM to include fracture phenomena. Whereas implementation of these techniques is relatively straightforward, it is known that element erosion routines fail to capture extended fracture phenomena such as branching [1] while the interface discontinuity approach may introduce errors in the material’s bulk properties [2]. A promising alternative was proposed by Belytschko and Black [3] which is known as the eXtended Finite Element Method (XFEM) where fractured surfaces are represented via discontinuous functions in the displacement field. XFEM performs well for crack propagation problems, but fracture behavior involving branching and coalescence require dedicated algorithms that translate the coinciding crack patterns to tailored enrichments of the interpolated displacement field [1,4,5]. This introduces a significant degree of computational complexity, especially when pursuing three-dimensional fracture problems [6,7]. In the past, the Element-Free Galerkin Method (EFGM) has been proposed as an alternative because its meshless nature relieves some of the stringent requirements on enrichment functions for crack junctions [8]. Apart from this, Rabczuk et al. [9] argued that XFEM and EFGM meet the same challenges for intricate fracture simulations since both methods model the crack as a continuous surface and additional criteria for extended fracture behavior are required.

An attractive alternative to these continuum methods discussed above, are discrete particle methods. While in continuum methods the geometry of the body is captured via a continuous interpolated field, discrete particle methods simply represent a body as a set of material points each containing a fraction of the body’s mass. This geometrical simplification yields an advantage over continuum methods regarding the representation of fractured surfaces, which, as discussed in the previous paragraph, can be cumbersome. A prominent example of a particle method is the Discrete Element Method (DEM) [10]. While initially developed for simulating granular media, DEM has been extended to model solids by introducing spring or beam elements between particles [11], which is referred to as the lattice-particle method by some authors. The constitutive behavior of the equivalent continuum material hinges on one-dimensional force–displacement relations, which alleviates the method from evaluating multi-dimensional spatial derivatives [12]. This means that constitutive laws formulated by stress–strain measures need to be converted into force–displacement relations. In the case of a simple linear elastic material, the stiffness terms corresponding to the discrete formulation can be approximated from the continuum equivalent ones by equating the second derivative of the strain energy to the individual components of the fourth-order elasticity tensor, e.g. [13]. Alternatively, one can perform a fitting procedure relating the macroscopic Young’s modulus and Poisson’s ratio to the pairwise interaction parameters, e.g. [11]. More extended continuum constitutive formulations involving e.g. geometric non-linearities and plasticity require more simplifying assumptions and elaborate translation procedures to approximate the discrete equivalent parameters [13–15]. Uchimali et al. [16] introduced a possible solution for the straightforward implementation of continuum constitutive laws in DEM. They have proposed a three-particle interaction potential where the distances and angles within a three-fold set constitute a linearized deformation gradient tensor that can be readily implemented in a continuum constitutive law to extract the interaction forces. However, the classical property of DEM where fracture is modeled by deleting the interaction between two-fold particle pairs is not adopted. Hence, fracture studies involving DEM are mostly performed within the scope of elastic–brittle materials such as concrete [17] and glass [18].

Alternatively, a particle method that employs continuum kinematics to approximate the continuum constitutive behavior is the Smoothed Particle Hydrodynamics (SPH) method. In SPH, the approximate deformation gradient tensor corresponding to a particle is computed via a weighted averaging procedure on the deformed positions within a pre-defined particle cloud. The stresses, found via a constitutive law, are then translated to interaction forces between the particles in these particle clouds. For solid mechanics purposes, Belytschko et al. [19] suggested that the total Lagrangian formulation is the most suitable numerical description for SPH since it minimizes instability related problems. Consequently, the total Lagrangian variant is the most widely employed version of SPH for problems related to material fracture in current literature [20,21]. SPH has been employed in studies involving
complex fracture like machining [22] and scratching [23,24] because it can handle arbitrary discontinuities and large local deformations due to its meshless particle nature. Moreover, the dynamic brittle fracture behavior of a combined SPH pseudo-spring formulation was found to be in adequate agreement with XFEM and DEM results. In the pseudo-spring formulation, the bond weight is scaled according to the elongation between particles, reaching zero when the critical elongation criterion is met [25]. However, even in a total Lagrangian setting, SPH simulations involving large strains exhibit instabilities in the form of particle clumping [26]. The particle clumping phenomena can be countered via the introduction of penalty or viscous forces, but the effect of these corrections on the simulated material behavior is not thoroughly addressed in the literature.

A more recent, and arguably the most popular, particle method for simulating solid materials is the Peridynamics method. Since Silling [27] introduced the method as an alternative formulation of elasticity, the attention of the Peridynamics method has been growing exponentially [28]. Peridynamics is based on an integral formulation of the balance equation which naturally allows for the presence of discontinuities within the domain. Because of this property, it is an attractive method for fracture simulations. The Peridynamics spectrum is quite broad and essentially three versions exist in the literature: Bond-Based (BB), Ordinary State-Based (OSB) and Non-Ordinary State Based (NOSB). The latter employs continuum kinematics to approximate continuum constitutive behavior. Ganzenmüller et al. [29] noted the similarities in the discrete formulation and computational procedure of NOSB Peridynamics and SPH. However, the point of departure in deriving the formulations for SPH and NOSB Peridynamics differ significantly. Similarly to SPH, instability issues are observed in NOSB Peridynamics. A comprehensive overview of the occurring instability phenomena is given by Tupek and Radovitzky [30] and a series of suggestions on penalty stabilization is provided by Breitenfeld et al. [31]. Interestingly, the Peridynamic community seems to attribute the instability related issues to the Peridynamic mechanics description, rather than to the discretization and collocating nature of the particle method as done by the SPH community. Consequently, reformulations accounting for the non-local nature of the averaged deformation gradient, which is a local kinematic quantity, yield more complex, but promising alternatives for NOSB Peridynamics descriptions e.g. [32,33]. Even though NOSB Peridynamics simulations have been conducted using continuum plasticity models [34,35], Tupek and Radovitzky [30] mentioned that instability phenomena still play a significant role in simulation outcomes because of the deformation gradient description. A solution that has been proposed is the translation of continuum constitutive laws to OSB Peridynamic material models, but this is not a straightforward procedure, see e.g. [36]. Alternatively, one could resort to a novel Peridynamic paradigm formulated in [37], where the employed kinematic quantities can be related to the classical continuum ones.

This paper introduces an alternative discrete particle-based framework, called the Continuum Bond Method (CBM), which enables a seamless implementation of continuum constitutive behavior without relying on stabilization routines. The evolution of the discretized body is governed by force interactions between pre-determined particle pairs referred to as bonds, whereby discontinuities are introduced by simply deleting these bonds. The bond configuration is established via a triangulation procedure over the complete body. The deformation kinematics are determined through a volume-weighted averaging procedure over the particle’s adjacent triangles, spanned by neighboring particle pairs. Note that this is different from the weighted least-squares averaging procedure employed by SPH or NOSB Peridynamics and more analogous to nodal averaging in FEM [38,39]. The formulation for the bond interaction forces is derived from an energy variational principle, so no partial differential equations with continuity requirements are considered. To calculate the evolution of each particle in the body, an explicit time integration scheme is employed, which is a common feature in discrete particle routines. A comprehensive description of the complete computational strategy is provided, followed by a discussion on the bond deletion routine. To investigate the constitutive capabilities of CBM, a benchmark example is presented with a finite strain elastic nonlinear plastic constitutive law. As mentioned earlier, the reported effects of stabilization routines on the constitutive response in particle methods are scarce. To assess this, CBM and SPH results for an elastic plastic tensile bar are compared to a finely discretized reference FEM solution. To illustrate the potential of CBM on also handling the transition to more extensive fracture phenomena, a range of dynamic fracture simulations are included, exhibiting different degrees of crack branching.

Throughout this paper, the following notation conventions will be used (using Einstein summation convention and Cartesian tensors):
2. Continuum bond method

The point of departure is the discretization of the continuum body into $N$ particles, each carrying a fraction of the body’s mass. A schematic representation is provided in Fig. 1. Neighboring particles are connected via bonds, which are defined according to a triangulation procedure. For this paper, the discretization and triangulation routines implemented in Gmsh [40] are used. In Fig. 2a a schematic representation of the neighboring triangles $T_i = \{I, II, III, IV, V\}$ connecting to particle $i$ is given. The relation between the reference position $\vec{X}_i$ and the deformed position $\vec{x}_i$ is given by the mapping function $\phi(\vec{X}, t)$. The volume represented by a single particle is defined as

$$\Omega_i = \sum_{\Lambda \in T_i} \frac{\Omega_\Lambda}{3},$$

where $\Omega_i$ and $\Omega_\Lambda$ are the volumes of particle $i$ and triangle $\Lambda$ in the undeformed reference configuration. In the next subsections, the mathematical formulation of the CBM framework is presented. First, the description of the kinematics is given where a particle deformation gradient is related to the relative positions of all points in a particle cloud. Subsequently, the interaction forces within a particle cloud are derived.

2.1. Kinematics

The particle deformation gradient tensor corresponding to $i$ is obtained by volume-weighted averaging of the linearized deformation gradient tensors spanned by the triangles in $T_i$, i.e.

$$\bar{\mathbf{F}}_i = \frac{1}{\Omega_i} \sum_{\Lambda \in T_i} \frac{\Omega_\Lambda}{3} \mathbf{F}_\Lambda.$$
The linearized deformation gradient tensor $\bar{\mathbf{F}}_A$ represents the mapping of a single triangle attached to $i$ as illustrated in Fig. 2b, following

$$\bar{\mathbf{F}}_A \cdot \bar{\mathbf{x}}_{ia} = \bar{\mathbf{x}}_{ia},$$  
$$\bar{\mathbf{F}}_A \cdot \bar{\mathbf{x}}_{ib} = \bar{\mathbf{x}}_{ib},$$  

(3)

where $\bar{\mathbf{x}}_{ia} = \bar{\mathbf{X}}_a - \bar{\mathbf{X}}_i$ and $\bar{\mathbf{x}}_{ib} = \bar{\mathbf{X}}_b - \bar{\mathbf{X}}_i$ denote the difference in position vectors related to the reference and deformed configuration. Note that $\alpha$ and $\beta$ indicate the two particles spanning $A$ together with particle $i$. Some straightforward manipulations yield

$$\bar{\mathbf{F}}_A = \bar{\mathbf{x}}_{ia} \left( \mathbf{X}^{-1}_{A,i} \cdot \bar{\mathbf{X}}_{ia} \right) + \bar{\mathbf{x}}_{ib} \left( \mathbf{X}^{-1}_{A,i} \cdot \bar{\mathbf{X}}_{ib} \right),$$  

(4)

where

$$\mathbf{X}_{A,i} = \bar{\mathbf{X}}_a \bar{\mathbf{x}}_{ia} + \bar{\mathbf{X}}_b \bar{\mathbf{x}}_{ib}.$$  

(5)

Then substituting Eq. (4) into Eq. (2) gives

$$\bar{\mathbf{F}}_i = \frac{1}{\Omega_i} \sum_{\Lambda \in T_i} \bar{\mathbf{x}}_{ia} \bar{\mathbf{\psi}}_{A,ia} + \bar{\mathbf{x}}_{ib} \bar{\mathbf{\psi}}_{A,ib},$$  

(6)

with

$$\bar{\mathbf{\psi}}_{A,ia} = \left( \frac{\Omega_A}{3} \right) \mathbf{X}^{-1}_{A,i} \cdot \bar{\mathbf{X}}_{ia}.$$  

(7)

Eq. (6) reveals that the formulation of the particle deformation gradient tensor consists of a double summation: over the triangles $A$ and over the particles $\alpha$ and $\beta$. Exploiting this property allows to substitute the summation over $T_i$ by a summation over the cloud of nearest neighboring particles $P_i = \{ j, k, l, m, n \}$ (see Fig. 2a) such that

$$\bar{\mathbf{F}}_i = \frac{1}{\Omega_i} \sum_{p \in P_i} \bar{\mathbf{x}}_{ip} \bar{\mathbf{z}}_{ip},$$  

(8)

where $\bar{\mathbf{z}}_{ip}$ is referred to as the bond vector connecting particle $i$ to $p$. The bond vector is calculated with

$$\bar{\mathbf{z}}_{ip} = \bar{\mathbf{\psi}}_{A,ip} + \bar{\mathbf{\psi}}_{A,p,ip},$$  

(9)

considering $A_A$ and $A_B$ to be two adjacent triangles sharing the side spanned by particles $i$ and $p$.

### 2.2. Discrete balance equation

In the derivation of the discrete dynamic balance equation, d’Alembert’s principle is taken as a point of departure since it conveniently allows to exploit variational principles to find the governing equations. Here, it is postulated...
that the variation of the work done by the inertia forces equals the difference in variational internal work $W^{\text{int}}$ and external work $W^{\text{ext}}$ as
\[ \sum_{i=1}^{N} m_i \ddot{a}_i \cdot \delta \vec{x}_i = \delta W^{\text{ext}} - \delta W^{\text{int}}, \] (10)
where the variational operator is denoted by $\delta$. Additionally, $m_i$ and $\ddot{a}_i$ represent the mass and acceleration of particle $i$. The variation of the externally applied work experienced by the particles in the body is described such that
\[ \delta W^{\text{ext}} = \sum_{i=1}^{N} \vec{f}^\text{ext}_i \cdot \delta \vec{x}_i, \] (11)
where $\vec{f}^\text{ext}_i$ represent the gravitational and externally applied forces. To establish a particle framework in which conventional continuum constitutive laws can be implemented in a straightforward manner, $W^{\text{int}}$ is formulated in terms of the internal mechanical work due to stress. For continuum bodies, this yields
\[ \delta W^{\text{int},c} = \int_{\Omega} \mathbf{P} : \delta \mathbf{F}^T d\Omega, \] (12)
where $\mathbf{P}$ denotes the first Piola–Kirchhoff stress tensor. The discrete particle equivalent of this statement, using the definition of the particle deformation gradient tensor, is written as
\[ \delta W^{\text{int},c} \approx \delta W^{\text{int}} = \sum_{i=1}^{N} \bar{\mathbf{P}}_i : \delta \bar{\mathbf{F}}^T_i \Omega_i, \] (13)
where the particle first Piola–Kirchhoff stress tensor corresponding to the particle deformation gradient tensor is defined as $\bar{\mathbf{P}} = \mathcal{C}(\bar{\mathbf{F}})$. The function $\mathcal{C}$ represents an arbitrary continuum constitutive law which can be derived from the conservative part of the free energy density. Substituting Eq. (8) in Eq. (13) yields
\[ \delta W^{\text{int}} = \sum_{i=1}^{N} \bar{\mathbf{P}}_i : \sum_{p \in \mathbf{P}_i} (\bar{\mathbf{P}}_p \cdot \bar{\zeta}_{ip}) \cdot \delta \vec{x}_p - (\bar{\mathbf{P}}_i \cdot \bar{\zeta}_{ip}) \cdot \delta \vec{x}_i. \] (14)
Then, by shifting the summation term, expanding the double dot product and substituting $\delta \vec{x}_p = \delta \vec{x}_p - \delta \vec{x}_i$ one obtains
\[ \delta W^{\text{int}} = \sum_{i=1}^{N} \sum_{p \in \mathbf{P}_i} (\bar{\mathbf{P}}_i \cdot \bar{\zeta}_{ip}) \cdot \delta \vec{x}_p - (\bar{\mathbf{P}}_p \cdot \bar{\zeta}_{pi}) \cdot \delta \vec{x}_i. \] (15)
By changing the order of summation for the first term on the right hand side such that
\[ \sum_{i=1}^{N} \sum_{p \in \mathbf{P}_i} (\bar{\mathbf{P}}_i \cdot \bar{\zeta}_{ip}) \cdot \delta \vec{x}_p = \sum_{i=1}^{N} \sum_{p \in \mathbf{P}_i} (\bar{\mathbf{P}}_p \cdot \bar{\zeta}_{pi}) \cdot \delta \vec{x}_i, \] (16)
the internal force acting on a particle can be extracted from the variation of the internal mechanical work as
\[ \delta W^{\text{int}} = \sum_{i=1}^{N} \left[ \sum_{p \in \mathbf{P}_i} \vec{f}_{ip} \right] \cdot \delta \vec{x}_i, \] (17)
where the interaction force reads
\[ \vec{f}_{ip} = \bar{\mathbf{P}}_p \cdot \bar{\zeta}_{pi} - \bar{\mathbf{P}}_i \cdot \bar{\zeta}_{ip}. \] (18)
Finally, substituting Eqs. (11) and (17) into Eq. (10), assuming that the expression holds for all kinematically admissible variations of the particle positions, results in the discrete balance equation written as
\[ m_i \ddot{a}_i = \vec{f}^\text{ext}_i - \sum_{p \in \mathbf{P}_i} \vec{f}_{ip}. \] (19)
Note that the interaction force given in Eq. (18) has a striking similarity with the SPH and NOSB Peridynamics definitions for the interaction force between a particle pair (see Eqs. (21) and (35) in [29]). The interaction force formulation of CBM differs from that of SPH or NOSB Peridynamics mainly with respect to volume integration. In CBM, the volume fractions related to the adjacent triangles of bond $i_p$ are included in the interaction force $\vec{f}_{i_p}$ through the bond vectors $\vec{\zeta}_{i_p}$ and $\vec{\zeta}_{p_i}$. In SPH and NOSB Peridynamics, the integration from stress to force between a particle pair involves the multiplication of the complete particle volumes, rather than fractions of volumes related to particle pairs.

3. Implementation

In this section the computational implementation aspects of the CBM are discussed. First, the computational routine in an explicit dynamic context is presented. The individual steps of the routine are discussed and computational choices are motivated. Next, a brief overview of the elasto-plastic constitutive model for the upcoming simulation comparisons is specified. Then, some mathematical considerations regarding bond erosion are emphasized. Finally, a discussion on the different dissipation sources in CBM is given and an artificial damping term is introduced reflecting the dissipative mechanisms related to fracture. Note, that the proposed numerical viscosity is incorporated to dampen the inter-particle vibrations that occur due to sudden bond removal, maintaining a stable fracture simulation. Therefore, it should not be confused with ad hoc stabilization forces employed in SPH and NOSB Peridynamic formulations to correct for instability phenomena originating from the kinematic description.

3.1. Time integration scheme

The kinematics of the particles within a body are calculated using an explicit time integration scheme. The Velocity-Verlet routine is an extensively used scheme in particle methods, most prominently in Molecular Dynamics (MD) and DEM. A global overview of the CBM computational routine is given in Algorithm 1 listed in the Appendix. The structure of the CBM routine is slightly different than standard routines employed in MD, DEM or explicit FEM. Here, the routine for a single time step is structured in 5 separate loops, where loop number 1 is the first velocity half-step and position update. The second loop checks whether a bond is broken according to the inter-particle linear strain criterion $\epsilon_{i_p} \geq \epsilon_{\text{crit}}$ with

$$\epsilon_{i_p} = \frac{\|\vec{x}_{i_p}\| - \|\vec{X}_{i_p}\|}{\|\vec{X}_{i_p}\|}.$$  (20)

If so, the bond is deleted and the interaction vectors of the surrounding particles must be updated. The equations related to this update are discussed in Section 3.3. The third loop evaluates for each particle in the body the nearest neighbors belonging to its pre-defined particle cloud and calculates the particle deformation gradient tensor according to Eq. (8). With the particle deformation gradient, the stress tensor acting on a particle can be computed according to a given continuum constitutive law. Note that, even though the body’s dynamics are included in this scheme, no deformation rates are presently employed in the determination of the constitutive response, i.e. no strain rate dependent phenomena, like viscosity, are included in the constitutive law. However, the extension to include deformation rates is rather straightforward, simply replace $\vec{x}_{i_p}$ for $\vec{v}_{i_p}$ to obtain the time derivative of the particle deformation gradient tensor. The fourth loop sums the interaction forces, resulting from stresses with Eq. (18), for each particle to obtain the net force. Since $\vec{f}_{i_p} = -\vec{f}_{p_i}$, it is computationally more efficient to loop over all bonds rather than to perform a routine similar to loop number 3. Finally, the net particle forces are used to compute the accelerations and the second velocity half-step. The boundary conditions are imposed in the form of prescribed accelerations and are exerted such that the impact speed experienced by the total body remains minimal. The exact formulation of the boundary evolution in terms of a prescribed acceleration is given for each discussed numerical example. Note that the prescribed velocities and prescribed displacements on particles are a consequence of the prescribed acceleration through the time integration routine of the Velocity-Verlet algorithm. For the unprescribed particles, the velocities and displacements are found equivalently, but the acceleration is determined by the calculated net force acting on the particle.
3.2. Elasto-plastic continuum constitutive model

An elasto-plastic constitutive model suitable for large deformations is adopted, which was shown to be consistent with the classical return mapping used in infinitesimal strain theory. The scheme originates from [41] and a summary of the implementation in the CBM computational routine is given in Algorithm 2 listed in the Appendix. Note that, Algorithm 2 serves solely as the constitutive model and is therefore an extension of line 18 in Algorithm 1. Hence, the current implementation of CBM allows for other constitutive models as well and is not limited to the one exploited here. The deformation gradient tensor is assumed to consist of an elastic and plastic part, i.e. \( F = F_e \cdot F_p \), also known as the multiplicative split. The plasticity routine is based on \( J_2 \)-flow theory exploiting the Kirchhoff stress and the logarithmic strain. More details about the properties and characteristics of this finite strain elasto-plastic constitutive model are given in [41,42].

3.3. Bond omission routine

In particle methods, fracture is generally modeled by deleting the link between two particles. This can be realized by deleting an explicit bond, like in DEM, or by removing a particle from an interactive cloud, like in SPH and Peridynamics. As a consequence, a constitutive relationship between the two particles vanishes and their relative motion is no longer restricted by this bond. An equivalent procedure is possible in CBM, where one breaks the bond between two particles and thereby nullifies the interaction between them. However, it is important that the bond vectors of the surrounding bonds are updated properly since the kinematic description is based on pairs of vector sets. Additionally, the averaging volume related to the particle deformation gradient tensor must also be updated. Note that, after fracture occurs, there exists a distinction between the volume fraction represented by the particle, and the averaging volume related to the adjacent triangles. The volume fraction represented by the particle, from which the particle mass is calculated, is determined in the reference configuration and is conserved throughout the simulation (i.e. mass remains conserved). The averaging volume, however, changes when triangles are omitted due to broken bonds. A bond constitutes an edge of either one or two triangles. When a bond is deleted, the triangle(s) attached to this bond are removed as well. This implies that for all particles spanning the deleted triangle(s), the averaging volume must be updated with Eq. (1) and the bond vectors corresponding to the deleted triangle(s) are updated with Eq. (9).

3.4. Artificial damping for sudden fracture events

There are, in general, three sources of dissipation that are accounted for in CBM: (i) constitutive dissipation (e.g. in plasticity): the intrinsic dissipation behavior included in the constitutive model, which is automatically incorporated by the established constitutive consistency; (ii) dissipation through instantaneous fracture by elastic release: the discrete removal of elastic energy related to the instantaneous bond deletion during fracture; (iii) local viscous dissipation: resulting from small-scale viscous effects (e.g. atomic friction) that dampen vibrations caused by dynamic fracture. In the first numerical example, the elasto-plastic bar, only dissipation source (i) is present because no fracture occurs. In the second numerical example, dynamic fracture of an elastic–brittle plate, dissipation sources (ii) and (iii) are present since fracture does occur and the adopted constitutive model is elastic. The instantaneous removal of bonds, reflecting brittle fracture, also releases energy in the neighborhood of the failed bond, through which elastic waves propagate across the sample. In reality, such elastic waves always get damped and hence dissipated. In an explicit dynamic integration scheme, these elastic waves may also destabilize the simulation. While any kind of source (iii) term can be introduced in CBM, for the current case an artificial viscosity term is adopted that actively dampens vibrations at the inter-particle scale to recover a stable fracture simulation. The employed artificial damping is inspired by Monaghan and Gingold [43], where a tailored artificial viscosity was derived that acts on the relative motion between particles from a general bulk viscosity formulation. The adopted derivation exploits this concept. The viscous pressure resulting from the relative motion between particles \( i \) and \( p \) is described as

\[
\pi_{ip} = -\beta l_0 \rho c_0 \frac{\vec{v}_{ip} \cdot \vec{x}_{ip}}{\|\vec{x}_{ip}\|^2 + 0.01 l_0^2},
\]

(21)
where \( \rho \) denotes the density, \( l_c \) the characteristic spacing between particles, \( c_0 \) the material’s speed of sound and \( \beta \) the control parameter. Note that
\[
\frac{\partial \vec{v}}{\partial x} \approx \frac{\vec{v}_{ip} \cdot \vec{x}_{ip}}{\| \vec{x}_{ip} \|^2 + 0.01l_c^2}, \tag{22}
\]
is the discrete equivalent of the spatial variation of the velocity field and is calculated on the deformed configuration. Since the framework is defined in a total Lagrangian form, a pull-back to the reference configuration is required. If the artificial damping forces simply add to the interaction forces, the structure of Eq. (18) will be maintained, ensuring momentum balance. As a result, the numerical damping force acting between particles \( i \) and \( p \) is described as
\[
\vec{f}_{AV}^{ip} = \left( J_{ip} \pi_{pi} \vec{F}_i^{-T} \right) \cdot \vec{\xi}_{pi} - \left( J_{pi} \pi_{ip} \vec{F}_i^{-T} \right) \cdot \vec{\xi}_{ip}, \tag{23}
\]
where \( J_I = \text{det}(\vec{F}_I) \). Note that \( \vec{f}_{AV}^{ip} = -\vec{f}_{AV}^{pi} \) since \( \pi_{pi} = \pi_{ip} \).

Note, that it is also possible to include a bond damage formulation that gradually decreases the interaction between particle pairs rather than abruptly eliminating bonds. This would, intuitively, partly eliminate the mentioned inter-particle dynamic phenomena and render the artificial damping obsolete for a sufficiently fine temporal discretization. However, the elimination of bonds, whether gradual or instantaneous, may still affect the approximated deformation gradient tensor. This results in changes in the stress and interaction forces, which would still induce dynamic phenomena in an explicit dynamic time integration scheme. This is inherent for all particle methods that exploit a particle averaging scheme, hence this phenomena is also present in SPH and NOSB Peridynamics. Yet, this is different for DEM or BB Peridynamics because the constitutive interaction is there solely defined at the bond level.

4. Examples

To illustrate CBM’s capabilities to accurately incorporate continuum constitutive models, an elasto-plastic tensile bar benchmark problem is investigated. An adequately refined FEM simulation is used as reference to validate the global and local material behavior found by the CBM simulations. Also, the SPH routine and the influence of stabilization on the simulation outcome is compared against CBM results for equivalent particle discretizations. Afterwards, the fracture capabilities of CBM are assessed by evaluating a series of dynamic fracture simulations. On a final note, all particle images in the upcoming examples are generated using the visualization program OVITO [44].

4.1. Elasto-plastic tensile bar

The geometry of the tensile bar considered in this example is schematically depicted in Fig. 3a. The specimen’s bottom row of particles are fixed in both directions, the upper row of particles exhibit a prescribed displacement in the vertical direction of \( u^* \) and are fixed in the horizontal direction. A plane strain situation is assumed and a nonlinear hardening law (inserted in line 5 of Algorithm 2) is used given by:
\[
h(\varepsilon_p) = h_0 + (\tau_{y\infty} - \tau_0)(1 - e^{-\alpha \varepsilon_p}), \tag{24}
\]
where \( \varepsilon_p \) denotes the equivalent plastic strain. The material and geometrical parameters for the tensile bar are listed in Table 1. Since the goal of the current investigation is to assess the ability of CBM to accurately account for plasticity in a continuum sense, damage phenomena involving localization are not studied in this first example. It was found that the prescribed displacement up till the point of necking is \( u^* = 2 \text{ mm} \) for the considered geometry specified in Table 1.

4.1.1. FEM comparison

For the FEM simulations triangular 6-node elements are used in combination with a 3-point Gauss integration scheme. A converged solution was achieved at a discretization of \( \sim 200 \) degrees of freedom judging from the global stress–strain response. Since the FEM discretization and the particle discretization of CBM are fundamentally different, a heavily refined FEM solution (\( \sim 7000 \) degrees of freedom) is interpolated at the spatial positions corresponding to the particle locations of the CBM discretization. This enables an intuitive comparison in the local constitutive response at the exact same spatial locations for FEM and CBM while minimizing the interpolation error.
Table 1

<table>
<thead>
<tr>
<th>Description</th>
<th>Symbol</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shear modulus</td>
<td>$G$</td>
<td>80.19</td>
<td>GPa</td>
</tr>
<tr>
<td>Bulk modulus</td>
<td>$\kappa$</td>
<td>164.21</td>
<td>GPa</td>
</tr>
<tr>
<td>Initial flow stress</td>
<td>$\tau_y^0$</td>
<td>450</td>
<td>MPa</td>
</tr>
<tr>
<td>Residual flow stress</td>
<td>$\tau_y^\infty$</td>
<td>715</td>
<td>MPa</td>
</tr>
<tr>
<td>Linear hardening coefficient</td>
<td>$h_0$</td>
<td>129</td>
<td>MPa</td>
</tr>
<tr>
<td>Saturation exponent</td>
<td>$\alpha$</td>
<td>16.93</td>
<td>–</td>
</tr>
<tr>
<td>Height of tensile bar</td>
<td>$H$</td>
<td>40</td>
<td>mm</td>
</tr>
<tr>
<td>Height of neck</td>
<td>$H_n$</td>
<td>18</td>
<td>mm</td>
</tr>
<tr>
<td>Width of neck</td>
<td>$W_n$</td>
<td>6</td>
<td>mm</td>
</tr>
</tbody>
</table>

The FEM simulations are executed using an implicit incremental-iterative solver while the CBM simulations were performed with an explicit dynamic routine presented in Algorithm 1. To recover a quasi-static case, the excitation velocity of the prescribed boundary must remain a factor of 1000 below the material’s sound of speed. Because physical inertia effects are neglected in a quasi-static situation, an arbitrary value for the density can be adopted. Note, however, that the critical time step for which the simulation remains stable is restricted by

\[ \Delta t_{\text{crit}} < \frac{l_c}{c_0}. \]  

This implies that decreasing $\rho$, and thereby increasing $c_0$ to allow for a shorter simulation time, will not provide any computational gain since the critical time step will scale accordingly. So, in the CBM simulations the numerical density is set to $\rho = 0.01 \text{ [g/mm}^3\text{]}$ and the total simulation time to $t_{\text{sim}} = 1 \text{ ms}$. The prescribed acceleration applied
Fig. 4. The global first Piola–Kirchhoff stress, found by CBM, in the vertical (tensile) direction plotted against the global linear strain for (a) discretizations of 100, 200 and 350 particles and (b) compared to a static, implicit FEM solution where both simulations (CBM and FEM) contain approximately 200 degrees of freedom.

to the top particles is formulated as

\[
\vec{a}^*(t) = \begin{cases} 
\vec{\tilde{u}}^* \left( \frac{120}{\tau^5} t^3 - \frac{180}{\tau^4} t^2 + \frac{60}{\tau^3} t \right) & \text{if } t \leq \tau, \\
0 & \text{if } t > \tau,
\end{cases}
\]

(26)

where \( \tau \) is the time constant related to the duration of the step and \( \vec{\tilde{u}}^* \) is the prescribed displacement vector. This formulation of the prescribed acceleration results in a S-shaped evolution of the boundary displacement moving from 0 to \( u^* \) over time. By choosing the prescribed acceleration in this form, impact phenomena on the specimen due to the imposed boundary conditions remain minimal. For the current situation, the time constant is taken as \( \tau = \frac{4}{5} t_{\text{sim}} \) and the time step as \( \Delta t = 0.2 l_c / c_0 \).

In Fig. 4a the global first Piola–Kirchhoff stress in the vertical direction is plotted against the global linear strain for three particle discretizations (see Fig. 3b). Two conclusions can be drawn here: First, the length-scale that is introduced by the particle discretization has little influence on the global response, i.e. proper classical continuum behavior is obtained. Secondly, a converged solution is achieved for a relatively coarse discretization of 100 particles.

In Fig. 4b the CBM and FEM responses are plotted together. Both simulations contain approximately 200 degrees of freedom, which for CBM means 100 particles and for FEM 100 nodes. It can be concluded that the mechanical response obtained from CBM coincides quite accurately with the response found by FEM.

Next, the local material response is investigated. Since particle methods create a simplified geometrical presentation of the concerned body, a fine discretization is required to reproduce an apparent tensile bar. Note that this is solely done for visualization purposes, since from Fig. 4a it was already established that a converged solution is achieved for 100 particles. For this, the CBM discretization is refined to approximately 1800 particles which coincides with a characteristic particle spacing of \( l_c = 0.55 \) mm. Fig. 5a gives a qualitative comparison of the equivalent plastic strain map over the deformed tensile bar for FEM (left bar) and CBM (right bar). Remember, that the particle locations corresponding to the CBM discretization are probed as material points on the finely discretized FEM domain to provide an objective comparison while minimizing interpolation error. It is obvious that the plastic response in the neck of the tensile bars are quite similar for FEM and CBM. A comparison of the equivalent Kirchhoff stress, defined as \( \tau_{eq} = \sqrt{\frac{1}{2} \tau_{dev} : \tau_{dev}} \) with \( \tau_{dev} \) as the deviatoric part of the Kirchhoff stress tensor, is given in Fig. 5b. Again, the results for FEM and CBM match very well.

4.1.2. SPH comparison

SPH is among the current standards in particle methods since it has been implemented in commercial packages and a significant body of literature is devoted to it. The goal of this subsection is to scrutinize the newly developed
CBM routine relative to a more standard particle method, SPH in this case, to evaluate the difference in simulation results. Note, that the ability of a particle method to account for more extensive continuum constitutive descriptions is the central theme in this comparative evaluation. A thorough explanation of SPH is beyond the scope of this work and more information about the fundamentals can be found in [45–47], a clear overview of the computational scheme is given by Leroch et al. [24].

In SPH, field variables, like the deformation gradient, are approximated via a weighted averaging scheme over some pre-defined particle cloud. The so-called kernel radius $r$ determines the size of these particle clouds in which material points interact with each other, which is analogous to the cut-off radius in MD. The kernel radius $r$ is a numerical parameter inherent to the SPH framework, hence it has no constitutive or micro-mechanical meaning. However, the kernel radius does influence the constitutive response in the discrete body as will be shown hereafter. For the present SPH simulations, the typically used cubic spline weighing function is employed which is formulated as:

$$w(q_{ij}) = \begin{cases} 
10 \frac{1}{7\pi r^2} & \text{if } 0 \leq q_{ij} \leq 1, \\
\frac{1}{4} \left(2 - q_{ij}\right)^3 & \text{if } 1 < q_{ij} \leq 2, \\
0 & \text{if } q_{ij} > 2, 
\end{cases}$$

(27)

where $q_{ij} = \|\vec{X}_j - \vec{X}_i\|/r$.

Since the invention of SPH, a significant amount of attention has been devoted to instability problems affecting the solution. The suggestion to resort to a total Lagrangian formulation when dealing with solid mechanics problems is widely accepted because it enables fracture simulations. Nonetheless, nonphysical bundling of particles is still observed in the deformed configurations of the SPH routine. In order to suppress this phenomenon, several solutions are proposed, among which the introduction of viscous forces, penalty pressures or additional integration points.
Fig. 6. The global first Piola–Kirchhoff stress, found by SPH, in the vertical (tensile) direction plotted against the global linear strain at varying penalty factors $\psi$ for (a) $r = 2l_c$ and (b) $r = 4l_c$.

called stress points. The stress point enrichment solution does not maintain the true particle nature of the SPH method, since node and integration point no longer coincide, hence, simulation outcomes are usually dependent on some degree of stabilization. Here, the stabilization routine proposed by Ganzenmüller [26] is employed to suppress the particle clumping. So, a penalty force defined as

$$f_{i}^{\text{pen}} = \psi \sum_{j \in P_i} \frac{\Omega_i \Omega_j w(q_{ij}) E}{\| \vec{x}_j - \vec{x}_i \|^2} \left( \delta_{ij} + \delta_{ji} \right) \frac{\vec{x}_j - \vec{x}_i}{\| \vec{x}_j - \vec{x}_i \|},$$

(28)

is added to the internal forces that act on particle $i$. The Young’s modulus $E$ is set as the penalty stiffness. The symbol $\delta_{ij}$ denotes the error of the given relative position between particles $i$ and $j$ compared to the relative position according to the approximate deformation gradient tensor of particle $i$. The variable $\psi$ is called the penalty factor and is thus one of the numerical parameters in the SPH method. Since this dependency of $r$ and $\psi$ will be present for all SPH results, a comparison with CBM is less straightforward as it was with FEM. Consequently, to assess the difference in constitutive response between CBM and SPH, the $[r, \psi]$ dependence has to be taken into account. Specifically, $\psi$ values from 0.01 to 1 are considered. Note that for $\psi = 1$ the stabilization forces are approximately of the same order as the internal forces since the Young’s modulus is taken as the penalty stiffness. For the kernel radius $r$ three values are examined: $2l_c$, $3l_c$ and $4l_c$. All simulation settings from the previous subsection are preserved and the particle discretizations ($l_c = 0.55$ mm, $\sim 1800$ particles) for SPH and CBM have been chosen exactly the same. The global stress–strain response for varying penalty factors is given in Fig. 6a for $r = 2l_c$ and in Fig. 6b for $r = 4l_c$. Note that rather extreme cases in the considered range of $[r, \psi]$ are shown, but that the effect on the stress–strain curves is minimal and similar to the CBM curve. The local behavior, though, is greatly influenced by the settings of $[r, \psi]$, as shown in Fig. 7. The rows from left to right show the plastic strain map obtained by SPH for a kernel radius of $r = 2l_c$, $r = 3l_c$ and $r = 4l_c$ and the columns from top to bottom the results for penalty factors $\psi = 0.01$, $\psi = 0.1$ and $\psi = 1$ over the central necked section of the deformed tensile bar. The color scale is refined in order to emphasize the differences between the plastic maps. Clearly, a low stabilization factor and a relatively small interaction radius results in spurious plastic strains. Additionally, increasing either $r$ or $\psi$ induces a smoothing. On the other hand, increasing both $r$ and $\psi$ influences the distribution of plastic strains and alters the global stiffness (see Fig. 6b).

In order to quantitatively compare the overall difference in field quantities obtained by the SPH and CBM method, a global error definition is used where the FEM simulation serves as the reference:

$$\text{Error} = 100 \sqrt{\frac{\sum_{i=1}^{N} (\bullet_i^{\text{FEM}} - \bullet_i^{\text{SPH}})^2}{\sum_{i=1}^{N} (\bullet_i^{\text{FEM}})^2}} \%.$$  

(29)
Fig. 7. Equivalent plastic strain $\varepsilon_p$ plotted over the neck of the deformed tensile bar obtained by SPH. The columns from left to right denote the kernel radii for $r = 2l_c$, $r = 3l_c$ and $r = 4l_c$. The rows from top to bottom denote the penalty factors for $\psi = 0.01$, 0.1 and 1.

Here, the $L^2$ error norm for a field quantity $\bullet$ is considered over all particles in the body between FEM and $\star = \text{SPH}$ or $\star = \text{CBM}$, except the particles that are less than $4l_c$ away from the prescribed boundaries at the top and bottom of the tensile bar. SPH introduces significant errors in the stress field close to the boundaries where Dirichlet conditions are imposed directly on the particles. Fig. 8 shows the difference in equivalent Kirchhoff stress for CBM and SPH plotted over the bottom half of the tensile bar. Note the significant difference at the prescribed boundary of the tensile bar for the SPH method. Clearly, CBM does not exhibit this deficiency. Issues regarding boundary deficiencies in the SPH method have been discussed in the literature, e.g. by Chen et al. [48], but this is not the focus for the current comparison. Hence, the regions where great SPH errors are expected a priori, i.e. close to the prescribed boundaries, are further omitted from the error analysis. In Fig. 9a the global error in the plastic strain field, as defined in Eq. (29), is plotted against the penalty parameter on a logarithmic scale for the three considered kernel radii. Also, the CBM error is included to illustrate the accuracy in local response compared to SPH, which is independent of the SPH penalty factor $\psi$. One can clearly observe that an optimum can be identified in the range $[r, \psi]$ such that the global error in plastic strains between SPH and FEM is minimal. From the given graph this is
approximately at $r = 3l_c$ and $\psi = 0.07$, where the error is 3.3%. Note that the SPH error in this case is still more than twice as large as the CBM error, which is 1.5%. Fig. 9b gives a similar comparison for the equivalent Kirchhoff stress $\tau_{eq}$. Here, the effect of stabilization is largest for the smallest kernel radius considered ($r = 2l_c$). Again, an optimum can be distinguished for $r = 3l_c$ and $\psi = 0.2$ where the error is 2.7%. Note that, the error graphs for $\bullet = \varepsilon_p$ and $\bullet = \tau_{eq}$ are quite different. The difference is due to the fact that Fig. 9b is governed by elastic response of the bar, while Fig. 9a only reflects the plastic behavior. Lastly, a qualitative comparison of the difference in local plastic strain between the different particle methods and FEM is given Fig. 10. Here, the SPH results for the three considered kernel radii are compared at a $\psi$ value where the error with FEM is lowest according to Eq. (29), i.e. the minima identified in Fig. 9a. For the SPH results, clearly, the difference map becomes smoother for an increasing kernel radius whereas at $r = 4l_c$ systematic differences become prominent. Also, the absolute difference in plastic strain between FEM and CBM is included. Some minor differences can be observed at the top and bottom part of the neck. Compared to the SPH results, the differences between CBM and FEM are reasonably small.

### 4.2. Dynamic fracture

In this subsection, the quasi-brittle fracture characteristics of CBM are discussed and demonstrated via a numerical example. The goal of this example is to highlight the method’s capability to naturally account for complex fracture phenomena resulting from fast propagating cracks, and thus prove that CBM can serve as a versatile tool in the simulation of material failure. The fracture behavior in brittle materials is often investigated in particle methods like DEM [14,49], SPH [25] and Peridynamics [50,51]. The ability to include crack branching resulting from dynamic crack propagation in a natural manner is one of the merits of particle methods.

In this example, simple material models are used and the adopted fracture criteria is phenomenological. Fig. 11 provides a sketch of the considered pre-cracked plate including its dimensions. Here, the bottom is fixed and the
Fig. 10. Difference in equivalent plastic strain $\varepsilon_p$ between FEM and the particle methods: SPH and CBM, plotted over the neck of the deformed tensile bar.

Fig. 11. Schematic representation of a pre-cracked plate including geometrical denotations and imposed boundary conditions.

top is displaced by $u^*$. In the simulations, a prescribed velocity $v^*$ is imposed via the prescribed acceleration as

$$\vec{a}^*(t) = \begin{cases} 
\vec{v}^* 
& \text{if } t \leq \tau, \\
0 
& \text{if } t > \tau,
\end{cases}$$

(30)

where $\vec{v}^*$ denotes the applied velocity vector. A linear elastic material model is assumed and the constitutive model reads

$$P = F \cdot (4C : E),$$

(31)

where $4C$ denotes the stiffness tensor and $E$ the Green–Lagrange strain tensor. This constitutive model is an extension of the classical infinitesimal linear elastic model since it can account for geometrical non-linearities and rotations. The situation presented here resembles a thin PMMA plate, hence a plane stress situation is assumed and the material parameters presented in Table 2 are employed. In the following, a bond strain criterion is used to determine the fracture behavior. To relate the material’s fracture toughness to a bond strain criterion, a simple
Table 3
Simulation settings for mode I fracture.

<table>
<thead>
<tr>
<th>Description</th>
<th>Symbol</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Height of plate</td>
<td>H</td>
<td>20 mm</td>
<td></td>
</tr>
<tr>
<td>Width of plate</td>
<td>W</td>
<td>20 mm</td>
<td></td>
</tr>
<tr>
<td>Length of pre-crack</td>
<td>w</td>
<td>10 mm</td>
<td></td>
</tr>
<tr>
<td>Refined strip height</td>
<td>$\Delta H_{ref}$</td>
<td>4 mm</td>
<td></td>
</tr>
<tr>
<td>Particle spacing at edges</td>
<td>$l_{edge}$</td>
<td>1 mm</td>
<td></td>
</tr>
<tr>
<td>Time step</td>
<td>$\Delta t$</td>
<td>0.1 $l_{ref}/c_0$</td>
<td>ms</td>
</tr>
<tr>
<td>Damping control parameter</td>
<td>$\beta$</td>
<td>0.5</td>
<td>–</td>
</tr>
<tr>
<td>Fracture energy release rate</td>
<td>$G_f$</td>
<td>100 J/m²</td>
<td></td>
</tr>
<tr>
<td>Loading velocity</td>
<td>$v^*$</td>
<td>0.75 m/s</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 12. The evolution of the dissipated energy normalized by the total energy at the end of the simulation for different discretizations.

estimate is used relating the elastic energy in a volume to the energy required to create a fractured surface:

$$\epsilon_{crit} = \sqrt{\frac{2G_f}{EI_c}}$$  \hspace{1cm} (32)

with $G_f$ as the material’s fracture energy release rate. First, a discussion regarding the dissipation sources is presented for a simple mode I fracture problem. Subsequently, the ability of CBM to incorporate crack branching events is demonstrated.

4.2.1. Dissipation evaluation

As mentioned in Section 3.4, the dissipation in the current fracture model originates from the instantaneously released elastic energy stored in the deleted bonds and the artificial viscosity that dampens inter-particle vibrations. Evaluating Eqs. (21) and (32), one can conclude that the viscous pressure and bond strain criterion are related to the characteristic distance between particles $l_c$, which is a discretization parameter and not a material property. The discretization dependency in Eqs. (21) and (32) ensures an appropriate scaling of the local dissipation such that the global dissipation is nearly independent of the number of particles. To demonstrate this, several mode I fracture simulations are executed where the discretization over the refined strip is varied by $l_{ref} = 1/4, 1/5, 1/6, 1/7$ and $1/8$ mm, which corresponds to 3487, 4895, 6537, 8293 and 10256 particles. The geometrical specifications and simulation settings are given in Table 3. Fig. 12 shows the evolution of the dissipated energy ($\psi_{diss}$) normalized by the total (sum of elastic, kinetic and dissipated) energy at the end of the simulation ($\psi_{tot,end}$) for the given particle discretizations. After fracture, $\psi_{tot,end}$ does not change anymore, hence the value $1 - \psi_{diss}/\psi_{tot,end}$ indicates the residual kinetic and elastic energy. Logically, in the quasi-static limit the ratio $\psi_{diss}/\psi_{tot,end}$ should be 1 at the end of the simulation when the bodies are separated, but in this example the material dynamics are included, so
the value will stay below 1 until all the remaining dynamics are damped. One can observe that the individual curves are slightly different for each particle discretization, but the observed overall dissipative trend is similar. The discrepancies between the curves are the result of marginally varying crack paths due to the different particle configurations. This can be confirmed from Figs. 13a, 13b and 13c where the fracture trajectories for \( l_{ref}^c = \frac{1}{4}, \frac{1}{6} \text{ and } \frac{1}{8} \text{ mm} \).

### 4.2.2. Crack branching

The aim here is not to capture the fracture behavior of PMMA exactly. Indeed, a lot of parameters influence the fracture results and tailored material descriptions accounting for the proper fracture behavior of PMMA are not available. However, as stated previously, the ability of a particle method to include more extensive fracture events, e.g. crack branches, without any additional routines to account for the fractured surfaces, is what makes particle methods an attractive computational tool to model material failure. Here, the fracture results for the case described above are investigated and the phenomenological trends from these simulations will be assessed through experimental observations on crack propagation in brittle materials provided by Ravi-Chandar and Knauss [52,53,54] and Fineberg and Marder [55]. This will illuminate CBM’s physical and numerical parameters governing crack propagation, while demonstrating its capability to account for more complex fracture events. The simulated plate is discretized into 14 431 particles, the geometrical specifications and simulation settings are given in Table 4.

First, the effect of the fracture energy release rate on the fracture behavior is investigated. Figs. 14a, 14b and 14c contain a set of fractured plates where the fracture energy release rates are equal to \( G_f = 100 \text{ J/m}^2 \), \( G_f = 250 \text{ J/m}^2 \) and \( G_f = 500 \text{ J/m}^2 \). The time for propagating cracks to reach the end of the specimens is \( t = 102 \mu s \), \( t = 135 \mu s \) and \( t = 142 \mu s \), respectively. Increasing \( G_f \) has a decreasing effect on the degree of crack branching in the fractured plates. Note that, \( G_f \) reflects the energy that is dissipated by fracture. The velocity of the propagating crack has an upper limit which relates to the material’s wave speed and the mechanisms involved in the crack formation. Consequently, imposing an excitation speed that forces the crack velocity towards this limit, causes the

---

**Table 4**

<table>
<thead>
<tr>
<th>Description</th>
<th>Symbol</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Height of plate</td>
<td>( H )</td>
<td>20</td>
<td>mm</td>
</tr>
<tr>
<td>Width of plate</td>
<td>( W )</td>
<td>30</td>
<td>mm</td>
</tr>
<tr>
<td>Length of pre-crack</td>
<td>( w )</td>
<td>3</td>
<td>mm</td>
</tr>
<tr>
<td>Refrmed strip height</td>
<td>( \Delta H_{ref} )</td>
<td>8</td>
<td>mm</td>
</tr>
<tr>
<td>Particle spacing at edges</td>
<td>( l^c_{edge} )</td>
<td>1</td>
<td>mm</td>
</tr>
<tr>
<td>Refrmed particle spacing</td>
<td>( l^c_{ref} )</td>
<td>1/6</td>
<td>mm</td>
</tr>
<tr>
<td>Time step</td>
<td>( \Delta t )</td>
<td>0.1 ( l^c_{ref} / c_0 )</td>
<td>ms</td>
</tr>
<tr>
<td>Damping control parameter</td>
<td>( \beta )</td>
<td>0.5</td>
<td>–</td>
</tr>
</tbody>
</table>
plate to compensate for the local excess energy by creating more fractured surfaces. Note that for different values of $G_f$ the crack tip reached the end of the plate at different instances in time. This implies an apparent decrease in crack tip velocity with $G_f$. Eq. (32) clearly indicates that each plate has a different fracture onset and this also influences crack tip time of arrival at the back of the plate. In Figs. 15a and 15b the fracture energy release rate is fixed at $G_f = 100$ J/m$^2$ and the loading speeds are set to $v^* = 0.75$ m/s and $v^* = 0.15$ m/s. Here we can observe
that the degree of branching is reducing for a decreasing excitation speed, which is consistent with the local excess energy argument. However, for \( v^* = 0.75 \text{ m/s} \), the crack tip reaches the back of the plate at \( t = 135 \mu s \), which is later than for Fig. 14a \( (t = 102 \mu s) \), indicating that there is also a reduction in global crack tip velocity present for Fig. 15a, even though branching occurs. This indicates that the degree of branching scales with the apparent crack tip velocity. This is consistent with the experimental observations, e.g. presented in [55] where for a PMMA plate an increase in fractured surface area is observed for an increasing crack tip velocity.

When modeling fracture in particle methods, one must consider also the numerical settings that influence the modeled crack path. Obviously, the precise crack path is influenced by the individual discrete bonds of the particle configuration. To assess this influence, Fig. 16a uses the same simulation settings as Fig. 14a but the discrete configuration has been reconfigured resulting in a minor \((\sim 0.8\%)\) increase of particles (14,542 in total). The resulting crack path is different, but the degree of branching and the time instant at which the crack tip reaches the back of the plate are similar. Also, the artificial damping that is incorporated to prevent instabilities due to impact phenomena from the bond deletion routine influences the fracture behavior. To investigate this effect, in Fig. 16b the damping control parameter is increased from \( \beta = 0.5 \) to a value of \( \beta = 1 \). Comparing to Fig. 14a, it is clear that the global crack tip speed as well as the degree of branching have significantly decreased. Hence, the artificial damping still has a great effect on the fracture physics involved in the CBM routine. It is postulated that the crack tip dynamics at rapidly rising stresses is dominated by nonlinear rate dependent phenomena. This includes inherent viscous effects or time dependent interaction between micro-crack stress fields. As stated previously, the current fracture model does not include any small-scale crack tip mechanics and is, apparently, mistaking the artificial damping for constitutive information on the crack propagation behavior. Hence, if one is interested in modeling the fracture behavior of a specific material, it is important to include the corresponding small-scale rate dependent phenomena characterizing the dynamic fracture behavior of that material, rather than to include an arbitrary viscous law designed to suppress destabilizing vibrations.

5. Conclusions

It has been demonstrated that CBM, in contrast to other particle methods, seamlessly captures the complex constitutive behavior in a classical continuum mechanics sense, similarly to continuum methods. At the same time, CBM naturally exhibits complex fracture characteristics like other particle methods, a feature missing in conventional continuum methods.

In this paper, a novel particle method is proposed based on a volume weighted averaging of adjacent volumes spanned by the nearest neighboring particle pairs. The kinematic description, linking the particle deformation gradient to the relative displacements within a cloud, allows one to relate a continuum mechanics perspective to a particle mechanics formulation. The obtained discrete dynamic balance equation is convenient for implementation in explicit dynamic time integration schemes, such as the Velocity-Verlet scheme. Generally, particle methods are
known for their flexibility in introducing arbitrary discontinuities. In CBM, this is realized by simply deleting bonds between particles and thereby omitting the interaction forces between separated sections. The constitutive flexibility and fracture characteristics are demonstrated by evaluating two numerical examples: (i) Elasto-plastic deformation of a steel tensile bar, compared to a reference FEM simulation as well as SPH simulations for different values of the interaction radius $r$ and penalty parameter $\psi$ and (ii) brittle fracture of a thin PMMA plate for varying physical and numerical quantities. The following conclusions have been drawn:

**Algorithm 1:** Velocity-Verlet time integration procedure for CBM

```plaintext
1 while $t < t_{\text{sim}}$ do
  2 $t = t + \Delta t$ // Time update
  3 $n = n + 1$ // Time step update

  // Loop #1
  4 for $i \in \Omega$ do
    5 $\vec{v}(t + \Delta t) = \vec{v}(t) + \vec{a}(t) \Delta t$ // 1st Velocity half-step
    6 $\vec{x}(t + \Delta t) = \vec{x}(t) + \vec{v}(t + \frac{\Delta t}{2}) \Delta t$ // Position update
  7 end

  // Loop #2
  8 for $(i, p) \in \Omega$ do
    9 if $\vec{e}_{ip} \geq \epsilon_{\text{crit}}$ then
      10 Delete $(i, p)$ // Delete bonds that have reached break criterion
    11 end

  // Loop #3
  12 for $i \in \Omega$ do
    13 $\vec{P}_i = 0$, $\vec{f}_i = \vec{0}$ // Set stress and force to 0
    14 for $p \in \mathcal{P}_i$ do
      15 $\vec{F}^{+\Delta t}_i = \vec{F}^{+\Delta t}_i + \frac{1}{\eta} \vec{v}_p \vec{v}_p$ // Calculate deformation gradient tensor
    16 end
    17 $\vec{P}_i = C(\vec{F}^{+\Delta t}_i)$ // Calculate stress from constitutive law
  18 end

  // Loop #4
  19 for $(i, p) \in \Omega$ do
    20 $\vec{f}_i = \vec{f}_i + \vec{f}_p + \vec{f}^{NN}_p$ // Sum bond forces and optional damping on particle $i$
    21 $\vec{f}_p = \vec{f}_p - \vec{f}_i + \vec{f}^{NN}_i$ // Sum bond forces and optional damping on particle $p$
  22 end

  // Loop #5
  23 for $i \in \Omega$ do
    24 if $i \in \Omega_{\text{pres}}$ then
      25 $\vec{a}(t + \Delta t) = \vec{a}(t + \Delta t)$ // Set prescribed acceleration
    26 else
      27 $\vec{a}(t + \Delta t) = \frac{1}{\eta} \vec{f}_i$ // Determine acceleration from internal forces
    28 end
    29 $\vec{v}(t + \Delta t) = \vec{v}(t + \frac{\Delta t}{2}) + \vec{a}(t + \Delta t) \frac{\Delta t}{2}$ // 2nd Velocity half-step
  30 end
end
```

- The global stress–strain response acquired by CBM and FEM are similar, even with a modest number of degrees of freedom.
- The local material response found by CBM and FEM are in adequate agreement, whereas the local material response of SPH is influenced severely by $r$ and $\psi$. While an optimum for the numerical SPH settings can be found such that global error on the plastic strain field is minimal (3.3%), the CBM error remains lower (1.5%).
- Relative to FEM-based methods, CBM can account for more extensive fracture events, like branching, more easily since additional crack tracking routines are unnecessary.
- The overall characteristics of dynamic fracture in relation to the fracture energy and loading velocity are well captured by CBM. However, to model the exact dynamic fracture response of a specific material, one
must incorporate the proper rate dependent dissipative behavior reflecting the underlying crack propagation mechanisms.

In conclusion, CBM is suitable for the implementation of continuum constitutive behavior while maintaining the advantageous fracture properties of particle methods. On a final note, it is important to mention that CBM is compatible with existing particle codes, allowing one to employ well developed computational tools like e.g. LAMMPS for solid mechanics simulations.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research is co-financed by Holland High Tech through a public–private partnership in research and development within the Dutch top sector of High-Tech Systems and Materials.

Appendix

See Algorithm 1 and Algorithm 2.

Algorithm 2: Elasto-plastic constitutive routine

<table>
<thead>
<tr>
<th>Data: $F^t$, $B^t$, $\varepsilon^t_p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f = F^{t+\Delta t} \cdot (F^T)^{-1}$</td>
</tr>
<tr>
<td>$\tau = \frac{1}{2}C : \ln(B^{t+\Delta t})$</td>
</tr>
<tr>
<td>$\tau_{eq} = \sqrt{\frac{2}{3}} \tau_{dev} : \tau_{dev}$</td>
</tr>
<tr>
<td>$\phi = \tau_{eq} - \tau_0 - h(\varepsilon^t_p)$</td>
</tr>
<tr>
<td>if $\tau &gt; 0$ then</td>
</tr>
<tr>
<td>$N = \frac{3}{2} \tau_{eq}$</td>
</tr>
<tr>
<td>$ \Delta \gamma = \Delta \gamma + \frac{h(\varepsilon^t_p + \Delta \gamma (\varepsilon^t_p + \Delta \gamma (\varepsilon^t_p + \Delta \gamma)))}{2 G \Delta \gamma}$</td>
</tr>
<tr>
<td>$\tau = \tau - 2 \Delta \gamma G N$</td>
</tr>
<tr>
<td>else</td>
</tr>
<tr>
<td>$\Delta \gamma = 0$, $r = \tau$</td>
</tr>
<tr>
<td>$\Delta \gamma = \Delta \gamma + \frac{h(\varepsilon^t_p + \Delta \gamma (\varepsilon^t_p + \Delta \gamma (\varepsilon^t_p + \Delta \gamma)))}{2 G \Delta \gamma}$</td>
</tr>
<tr>
<td>$\tau = \tau - 2 \Delta \gamma G N$</td>
</tr>
<tr>
<td>$P = \tau \cdot (F^{t+\Delta t})^{-T}$</td>
</tr>
</tbody>
</table>

References


