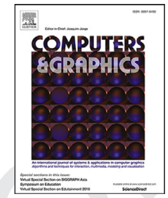
Computers & Graphics xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag



Technical Section

Animation of crack propagation by means of an extended multi-body solver for the material point method[☆]

Joel Wretborn[a,*], Rickard Armiento[b], Ken Museth[a]

[a] DreamWorks Animation, Glendale, CA 91201, USA
[b] Department of Physics, Chemistry and Biology (IFM), Linköping University, SE-581 83 Linköping, Sweden

ARTICLE INFO

Article history:
Received 1 May 2017
Revised 16 October 2017
Accepted 16 October 2017
Available online xxx

Keywords:
Material point method
Cracks
Contact

ABSTRACT

We propose a multi-body solver that extends the Material Point Method (MPM) to simulate cracks in computer animation. We define cracks as the intersection between pieces of bodies created by a pre-fracture process and held together by massless particle constraints (glue particles). These pieces are simulated using a MPM multi-body solver extended by us to efficiently handle *N*-body collisions. Benefits of the present work include (1) Low computational overhead compared to a normal MPM algorithm; (2) good scaling in three dimensions due to our use of sparse grids for background computations; (3) allowing for an easy and controllable setup phase to simulate a desired material failure mode, which is especially useful for computer animation.

1. Introduction

Some of the most interesting natural phenomena involve material fracture, and it is a vital ingredient in simulations where realism is desired. Hence, algorithms for object breakage using various simulation techniques are a topic of high level of interest, both for engineering applications and for computer graphics and animation. Specifically for simulations using the material point method (MPM) by [1], simulation of fracture via crack propagation appear to have been mostly discussed in the engineering literature with a focus on numerical accuracy. The aim of these works is different from what is needed for animation applications, where simulation speed and art directability are prioritized. In the present paper we present an algorithm for fracture that provides attractive features for use in computer graphics while only adding a small overhead over regular MPM simulation.

The MPM method is increasingly relevant for simulations of materials due to improvements in hardware and algorithms. It has proven useful in simulations involving large deformations, where the approach of combining meshless particles with a fixed computational background grid provides a robust framework. Both viscoelastic and viscoplastic materials have been simulated with impressive results. The MPM has also been used for other materials like rubber and sponges, which can undergo large elastic deformations. Inherent to the method is that these materials will break naturally if the stress is too high at any particular location. Normally, a simulated material is homogeneous and isotropic. This is often not the case for their real-world counterpart, as small weaknesses and local inconsistencies are important features for how a crack propagates through a medium. Such irregularities could be introduced in the simulation by modifying the parameters that govern the constitutive model on a per-particle basis or by jittering the particles in their initial configuration, but doing so in a way that both conserves the original collective behavior of the material while achieving the desired break point is difficult.

In the present work, we extend MPM by defining a crack via pre-fracturing of a specimen into different bodies, which are bound together by particle constraints scattered on the crack surface. We call these particles *glue particles*, and their role is to hold the object fragments connected until they break and a crack is formed. The focus of this paper is on simulation of bodies with a single crack, and where the crack propagation is dominated by an opening mode. The pieces from a fractured body are allowed to interact freely in the simulation, and we also present an extension to the contact algorithm by [2] to allow for arbitrarily many colliding bodies in the same solver.

The rest of the paper is organized as follows. A review of related works is discussed in 2. In Section 3 we present the extended contact algorithm, which is utilized for the crack algorithm in Section 4. Simulations based on the two algorithms will be shown in Section 5, followed by a discussion in Section 6 that

[☆] This article was recommended for publication by Prof L Barthe.
[*] Corresponding author.
E-mail address: joel@wbn.se (J. Wretborn).

https://doi.org/10.1016/j.cag.2017.10.005
0097-8493/

Please cite this article as: J. Wretborn et al., Animation of crack propagation by means of an extended multi-body solver for the material point method, Computers & Graphics (2017), https://doi.org/10.1016/j.cag.2017.10.005

points out current artefacts and limitations. Final conclusions then follow in Section 7.

2. Related work

Early works on the simulation of deformable plasticity and fracture in computer graphics were undertaken by [3]. Such approaches to dynamic fracture propagation often involved mesh-based finite element methods due to the ease of calculating stress coefficients along connected points. [4] introduced an element splitting approach to increase numerical accuracy and avoid visible artefacts for brittle fracture, which was later extended by [5] to include ductile fracture. However, mesh based methods can have problems handling large deformations, which may easily occur in fracture scenarios due to high internal stresses needed for a crack to surface. [6] suggested a meshfree method where the surface of a material is modeled using unconnected points. A crack is explicitly represented by a crack front of surface particles, which are added continuously to the crack front during the simulation. [7] proposed a method for fracture of thin sheets that requires a pre-defined crack. Explicitly declared cracks are flexible and give great control over how the material breaks. However, visible artefacts will arise if stresses are not properly aligned to the fracture surface, as the resulting crack will look unrealistic. [8] shows great results combining a pre-computed compound mesh dynamically applied based on the impact location of a projectile, where resulting pieces are simulated by a rigid-body solver.

The MPM was created by [1], and has since then proven to be useful for a range of different phenomena. It was later introduced to computer graphics by [9] with their work on snow. [10] and [11] modified their method to simulate viscoplastic materials like foam. [12] added a heat solver to simulate phase change of materials. [13] and [14] used MPM together with a Drucker–Prager plasticity model to simulate sand and other granular materials. [11] complemented MPM with a particle re-sampling scheme to handle potential non-uniform particle distributions due to high shearing strain. [15] proposed to track a locally affine transformation on each particle that would enable conservation of angular momentum; an improvement over the normally used PIC/FLIP [16] update scheme.

In MPM, all particles discretized to the same grid will share the same description of internal stresses on the Eularian grid. This will yield non-physical contact forces when two distinct bodies collide, but this can be avoided by complementing MPM with a contact algorithm. Due to the particles in MPM being meshless, contact is often resolved on the grid, and any Eularian contact method can be used. [17] resolves contact of overlapping Eularian grids by formulating the problem by the principle of least constraints. A similar approach was employed by [18], who uses this Eularian formulation to simulate contact for a Lagrangian mesh. [2] proposes a method targeted at MPM. They discretize each body to separate background grids, and an impenetrability condition is imposed with respect to the relative grid velocity to resolve collisions. [19] uses this contact scheme on a purely grid based level set method. However, due to the grid being a smeared representation of the current particle configuration, all purely Eularian methods will observe that collisions are detected prematurely. This artefact is discussed in the context of our solver in Section 6.

One existing method to handle the lack of an inherent way to represent cracks in MPM is CRAMP (CRAcks with Material Points) [20,21]. CRAMP introduces cracks on the Eulerian grid by allowing grid nodes to have multiple velocity fields. Particles from opposite sides of a crack are rasterized to different grids, which is determined by a line-crossing algorithm from the particle to the grid. The crack is explicitly represented as a Lagrangian mesh of massless particles, which in 2D constitutes of connected line segments and in 3D a polygon mesh. CRAMP is primarily used for engineering applications to investigate the stress response of a specimen with a non-propagating crack, but was recently extended by [22] to allow for dynamically propagating cracks. The authors, however, also say that their dynamic crack propagation algorithm "... *[requires] substantial computational effort even for two-dimensional calculations*." The importance of speed and art directability makes CRAMP difficult to use in graphics applications. Our goal is a crack algorithm capable of simulating realistic looking crack propagation, with lower computational requirements more suitable for computer graphics. [23] have also developed a scheme for crack growth in generalized interpolation MPM (GIMP) which targets engineering applications. They simulate a crack along a pre-defined cohesive zone in a 2D specimen, where particles from opposite sides of the crack interface interact. The glue particles presented in this paper resemble their use of a cohesive zone.

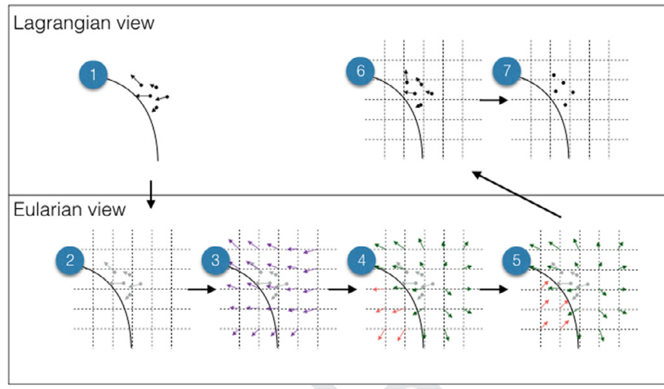


Fig. 1. *Overview of the algorithm.* The steps are classified as *Lagrangian* and *Eulerian* to signify what entity is being manipulated, particles or grid nodes. Explanation of the steps: (1) Initialized particles are used as input; (2) a background grid is created; (3) mass and velocity are rasterized onto the grid, and internal forces are calculated using the constitutive model; (4) new velocities are calculated using external and internal forces (red values are inside a boundary); (5) boundary collisions are resolved; (6) velocity is transferred back to the particles; (7) particle position and deformation are updated. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.) Q2

3. Multi-body solver for MPM

MPM is a hybrid method in that it combines an Eulerian mesh with Lagrangian particles. First, a continuous material is discretized into material points. The particles store all information that will be carried on through the simulation such as, position, velocity, deformation, and other potential properties related to the constitutive model. The Eulerian grid is used in the background to perform certain types of calculations. A particle is rasterized onto the grid by means of a weighting function, which transfers its attributes to the grid nodes. The internal and external forces are solved on the grid, and the attributes are transferred back to the particles and their positions are updated. Afterwards, the grid is discarded and a new simulation step is initialized. An overview of the algorithm can be seen in Fig. 1. We follow closely the implementation by [9], with the exception that we use an explicit time step integration scheme to simplify the grid update.

Our approach to crack propagation is to pre-fracture a specimen into separate bodies that are held together by glue particles. When these particle constraints break, each body must interact with every other body in the simulation. Section 3.1 first describes a two-body collision scheme founded upon the work of [2]. This two-body algorithm is a reformulation of the same method they present in their paper, but we introduce a different notation which we use for our extension to N-body collisions. The difference stems

Please cite this article as: J. Wretborn et al., Animation of crack propagation by means of an extended multi-body solver for the material point method, Computers & Graphics (2017), https://doi.org/10.1016/j.cag.2017.10.005