BSC. THESIS REPORT: APPLYING THE MPM TO SOLID MECHANICS

DAMIAN BIRCHLER (BDAMIAN@STUDENT.ETHZ.CH)
SUPERVISORS:
G. TAURELLIO AND S. KHATRI
EXAMINER:
PROF. P. KOUMOUTSAKOS

1. Scope

For his bachelor thesis the author wrote an MPM code in C++ and performed various simulations with it. The scope of this report is to summarize the work performed and to document any difficulties encountered.

In a first step a one-dimensional problem was studied: the stretching of a "one-dimensional" bar. Once this test case performed satisfactorily the author went on to try a simulation in two dimensions. An infinitely long copper cylinder was prescribed an initial radial velocity and its vibrations were studied. In the final, three-dimensional, simulation copper bars were twisted and torn.

The following three sections will cover the three problems in more depth.

2. One-Dimensional Stretched Bar

In [6] it is shown that the material point positions in MPM can be considered quadrature nodes of a midpoint-rule-like integration in the MPM algorithm. Integrating non-smooth functions — the derivative of the often-used 1^{st} order linear hat function is not smooth — via the midpoint-rule can lead to significant quadrature errors. Therefore, the authors propose using higher order basis functions.

In order to study the effect of higher order kernels on simulation accuracy they perform convergence tests for 1^{st}, 2^{nd} and 3^{rd} order kernels. The kernels used including their derivatives are shown in figure 1 and their analytical formulas can be found in [6]. While for the 1^{st} order kernel the error actually increases as the time step and grid spacing are refined Steffen et al. see second-order convergence for the 2^{nd} and 3^{rd} order kernel.

2.1. Problem description.

Their test problem is a one-dimensional bar that is fixed on one end and prescribed a sinusoidal traction force $q(X,t) = \delta(X-L)\sin(\frac{\pi}{L}t)$ (L is the bar's initial length) on its other end. The computational domain has extent $[0, 1.15]$ and the bar has an initial length of $L = 1$. The number of grid nodes n_g was varied and $n_p = 3n_g$ particles were used to discretize the bar. Since the bar has an initial length of 1 and the computational domain has an extent of 1.15 this configuration results in slightly less than three particles per cell. The discretization was set up in such a way as to ensure that there were not equally many particles in each cell in order to study the effects that unequally distributed particles can have on simulation accuracy.

2.2. Algorithm.

Steffen et al. use the following variant of the MPM algorithm. The superscript G denotes quantities on the grid and the superscript P denotes particle quantities:

The algorithm begins by assigning a mass to each grid point from the particle positions. The particle mass naturally is fixed throughout the computation and it is assumed that at this point in the algorithm the particle positions are known — either from a previous iteration or from the initial values. Because the basis functions by construction observe the "partition-of-unity"

Date: May 29, 2012.