Production ready MPM simulations

Gergely Klár
DreamWorks Animation
greg.klar@dreamworks.com

Jeff Budsberg
DreamWorks Animation
jeff.budsberg@dreamworks.com

Stephen Jones
DreamWorks Animation, NVIDIA
stjones@nvidia.com

Ken Museth
DreamWorks Animation
ken.museth@dreamworks.com

Figure 1: Samples from left to right: sand pillars collapsed by boulder, dragon thrashing in the sand, sand castle collapsing.

ABSTRACT

We present two complementary techniques for Material Point Method (MPM) based simulations to improve their performance and to allow for fine-grained artistic control. Our entirely GPU-based solver is able perform up to five times faster than its multithreaded CPU counterpart as a result of our novel particle and grid transfer algorithms. On top of this, we introduce Adaptive Particle Activation, that both makes it possible to simulate only a reduced number of particles, and to give artists means for fine direction over the simulation.

CCS CONCEPTS

•Computing methodologies →Massively parallel and high-performance simulations; Physical simulation;

KEYWORDS

MPM, simulation, GPU, art-directed, granular materials

ACM Reference format:
Gergely Klár, Jeff Budsberg, Stephen Jones, and Ken Museth. 2017. Production ready MPM simulations. In *Proceedings of SIGGRAPH '17 Talks, Los Angeles, CA, USA, July 30 - August 03, 2017,* 2 pages.
DOI: http://dx.doi.org/10.1145/3084363.3085066

SIGGRAPH '17 Talks, Los Angeles, CA, USA

DOI: http://dx.doi.org/10.1145/3084363.3085066

1 INTRODUCTION

The Material Point Method has proven itself to be a versatile tool for animation of natural phenomena, like sand [Klár et al. 2016], snow [Stomakhin et al. 2013], and even lava [Stomakhin et al. 2014]. The main benefits of MPM in production relate to its ability to preserve overall dynamics over different resolutions, and its robustness to particle overlap. However, these advantages come at a cost that renders MPM impractical for all but hero shots.

We present our work done at DreamWorks Animation to dramatically increase the performance of MPM, and to establish a production-friendly workflow for the VFX artists.

Our core multithreaded MPM solver for sand outperforms the published results six times. We replace this solver with one that runs entirely on the GPU, and that is up to five times faster than its CPU counterpart, even on mid-range GPUs.

Independent of the underlying solver implementation, we further reduce simulation time by introducing our Adaptive Particle Activation (APA) scheme, that dynamically suspends and reinstates particles, significantly reducing the number of points used by the solver while maintaining the desired look. An added benefit of APA is the option to manually activate and deactivate particles, allowing for artistic control.

Finally, we demonstrate how these components fit together to create a workflow that allows for both fast turnaround times and fine tuning.

2 PARTICLE AND GRID TRANSFERS

The steps of a standard MPM simulation are as follows: *transfer from particles to grids, grid solve, transfer from grids to particles,* and *particle update.*

Of these steps, the particle and grid transfers have proven to be the most difficult to parallelize efficiently. This is especially true

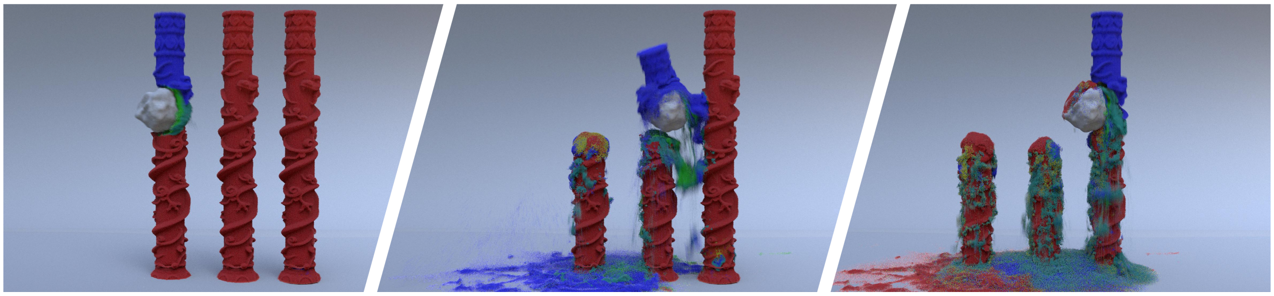

Figure 2: Different stages of particle activation: red - deactivated, green - activated by collision object, teal - activated by neighbors, blue - activated by island detection, yellow - inactive candidate .

for GPU implementations that have to adhere to the hardware's peculiarities to achieve maximal performance.

Our novel transfer algorithms take full advantage of the architecture by means of two level binning of the particles. Particles are assigned to voxels, then the voxels to tiles. The binning to voxels is based on each particle's influence on the grid, resulting in a partitioning where particles of a voxel affect, and are affected by the same set of grid nodes. The tiles, in turn, form a sparse regular lattice over the simulation domain, and voxels are assigned to tiles based on their position.

To facilitate the binning, each voxel is assigned a 32 bit voxel-and-tile key. The binning is done by sorting, run-length encoding, and computing the scan of these keys. All these steps are run entirely on the GPU, and scale linearly with the number of particles. The resulting arrays act together as a hashmap of varying length vectors, but at the simplicity and efficiency of plain arrays. These arrays provide immediate access for the GPU threads to the particle ranges they need to process.

With the use of this binning strategy, we map tiles to *computation blocks,* and voxels to *warps.* This allows us to take full advantage of the shared memory architecture, and reduce the number of required atomic operations to one per voxel from a typical 8-27 per voxel.

3 ADAPTIVE PARTICLE ACTIVATION

In many situations, especially for sand, a large portion of the particles are stationary, and much of the simulation efforts are spent on maintaining their equilibrium. We avoid these unnecessary computations by deactivating particles, and later on reactivating them as necessary.

Particles are eligible for deactivation when the local speed is under a user-defined threshold. If a particle remains eligible for a period of time, it is deactivated, thus ignored by the simulation. To keep the particles coupled to the simulation, we convert the deactivated particles to a narrow-band level set represented as a compact VDB [Museth 2013], then merge that to the collision level set and update the velocity field appropriately.

Similarly, deactivated particles can re-enter the simulation via velocity conditions in the local neighborhood exceeding a threshold, proximity to moving collision objects, or even manual activation regions as per artistic demands. Further, we use a fast level set segmentation operation [Museth et al. 2015] to detect and activate any potential floating islands of material.

This approach not only profoundly improves simulation times, but also provides means for artistic control. With artist controlled activation, both the timing and the propagation of dynamics can be directed.

4 WORKFLOW INTEGRATION

While the internal workings of an MPM solver are fairly complex, its input and output are quite tangible. The solver operates on points with attributes describing their state and physical properties, and it ultimately returns the same points with updated attributes. Between simulation steps point attributes can be freely adjusted, as long as the positions are not altered. This gives great freedom for dynamic control of material properties and in applying custom forces, which are critical in responding to artistic feedback in animation and VFX. Furthermore, points can be freely added or removed from the simulation without risking its stability.

We implemented our MPM solver as a plug-in for Houdini, matching the workflow of our artists. Here all the conventional tools are available to setup, control, and modify the simulation in order to hit the desired art direction.

5 CONCLUSION

The talk describes our latest results to make the adaption of MPM simulation appealing for feature VFX.

Our work makes it possible to run simulations magnitudes faster by moving the solver to the GPU and by the use of Adaptive Particle Activation. This reduced iteration time makes the use of MPM feasible in non-hero shots as well, and promotes its wider use in the industry.

REFERENCES

G. Klár, T. Gast, A. Pradhana, C. Fu, Craig S., C. Jiang, and J. Teran. 2016. Drucker-prager Elastoplasticity for Sand Animation. *ACM Trans. Graph.* 35, 4, Article 103 (July 2016), 12 pages. DOI: http://dx.doi.org/10.1145/2897824.2925906

K. Museth. 2013. VDB: High-resolution Sparse Volumes with Dynamic Topology. *ACM Trans. Graph.* 32, 3, Article 27 (July 2013), 22 pages. DOI: http://dx.doi.org/10.1145/2487228.2487235

K. Museth, D. Bailey, J. Budsberg, J. Lynch, and A. Pearce. 2015. OpenVDB. In *ACM SIGGRAPH 2015 Courses (SIGGRAPH '15).* ACM, New York, NY, USA, Article 20. DOI: http://dx.doi.org/10.1145/2776880.2792701

A. Stomakhin, C. Schroeder, L. Chai, J. Teran, and A. Selle. 2013. A Material Point Method for Snow Simulation. *ACM Trans. Graph.* 32, 4, Article 102 (July 2013), 10 pages. DOI: http://dx.doi.org/10.1145/2461912.2461948

A. Stomakhin, C. Schroeder, C. Jiang, L. Chai, J. Teran, and A. Selle. 2014. Augmented MPM for Phase-change and Varied Materials. *ACM Trans. Graph.* 33, 4, Article 138 (July 2014), 11 pages. DOI: http://dx.doi.org/10.1145/2601097.2601176