Kali: High Quality FEM Destruction in Zack Snyder's *Sucker Punch*

Ben Cole*
MPC

Abstract

MPC introduces Kali, a finite element based simulation toolkit for large scale and detailed destruction, developed in collaboration with Pixelux. The toolkit is described with examples from Zack Snyder's *Sucker Punch*.

1 Introduction

Zack Snyder's *Sucker Punch* required us to deliver 50 large scale fully CG destruction shots, featuring close-up, slow-motion destruction of brick, stone, wood and metal. Conventional VFX toolkits use Rigid Bodies for modelling and simulation of destruction. These tools make it hard to simulate flexing, tearing and breaking in a realistic and controllable way. With Kali, we have replaced the Rigid Body solver with a Finite Element solver. We worked with the development team from Pixelux to take their DMM technology, created for realtime video game applications [Parker and O'Brien 2009], and reapply it to visual effects. Modifications were required to the software and a new pipeline was established to deal with VFX production demands.

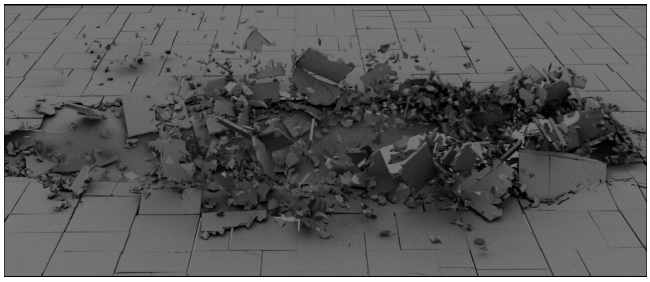

Figure 1: *Ground destruction in* Sucker Punch.

2 Usage

Kali uses a tetrahedral finite element representation of geometry. We have chosen to use a cage based approach, separating the rendered geometry from geometry used for simulation. This allows artists to define simulations as they wish, adding fracture detail and altering the shapes of collision objects independently of the rendered geometry. Simulation detail can be changed shot to shot, with more tetrahedra in areas where attention will be focused. A single tetrahedral mesh can map to many pieces of render geometry, or vice versa.

Render geometry is chopped at render-time against the last frame of simulation, so changes to the model simply require a re-chop, not a re-sim or re-cache. Likewise, changes to the simulation do not require remodelling. This leads to efficiency gains as there is far less cyclic inter-dependency between our modelling, lighting and simulation teams. This approach is general and will work with simulations and animations created with other tools, too.

Kali has an event system that allows TDs to add features to the system as needed. We've created a library of standard events, e.g., for pinning, driving, and controlling material toughness. Every tetmesh has one or more materials assigned to it, controlling flexiblity, brittleness, resilience, etc. A wide range of material types can be defined. Material properties can be changed during simulation using the event system.

In general, simulations are run on the farm in batches with varied parameter ranges. More than one cache can be used with a single renderable asset, allowing different parts to be simulated separately, which is useful for larger environments, or for shot continuity.

On *Sucker Punch* we simulated the destruction of the interior and exterior of a Japanese style pagoda. The pagoda exterior had approximately 10.7 million faces. The interior was approximately 11 million. We simulated destruction of the roof, floor, walls, pillars, doors and furnishings. Individual pillars were simulated with up to a million tetrahedra, which were then used to chop the renderable geometry. Chopping took between 1 minute and 1 hour. Simulation times were 1 or 2 hours for average setups to about 20 hours for the most complex simulations.

There is a simple system for editing caches after simulation, that allows vertices or regions to be moved around to tidy up simulations or add specific behaviour. Caches can also be used as sources for particle emission. The system knows how and when specific regions broke apart, which is useful for calculating realistic emission for secondary simulations.

3 Modelling and Rendering

The modeling department does not need to know how the model will be used in simulation, as long as they follow a set of basic rules. Objects must be closed and faces must have area. These restrictions do not dramatically limit geometry that can be used with the system. For rendering, Kali simulation assets can literally be dropped-in in place of standard geometry assets and rendered using the same set ups [Meeres-Young et al. 2010]. All that is needed in addition is a shader for newly created interior faces.

4 Summary

The work we undertook on *Sucker Punch* using our Kali system demonstrates that a finite element approach to large scale and detailed destruction works effectively in a production pipeline. Performance is good and results are visually interesting and high-quality. The system has also been used on several other VFX projects, including *Harry Potter and the Deathly Hallows (Part Two)* and *X-Men: First Class*.

References

MEERES-YOUNG, G., RICKLEFS, H., AND TOVELL, R. 2010. Managing thousands of assets for the prince of persia city of alamut. In *ACM SIGGRAPH 2010 Talks*, ACM, New York, NY, USA, SIGGRAPH '10, 30:1–30:1.

PARKER, E. G., AND O'BRIEN, J. F. 2009. Real-time deformation and fracture in a game environment. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ACM, New York, NY, USA, SCA '09, 165–175.

*e-mail: ben-c@moving-picture.com

SIGGRAPH 2011, Vancouver, British Columbia, Canada, August 7 – 11, 2011.
ISBN 978-1-4503-0921-9/11/0008