SIGGRAPH Course: Real-time Physics

Multi-Sensory Physics and User Interaction

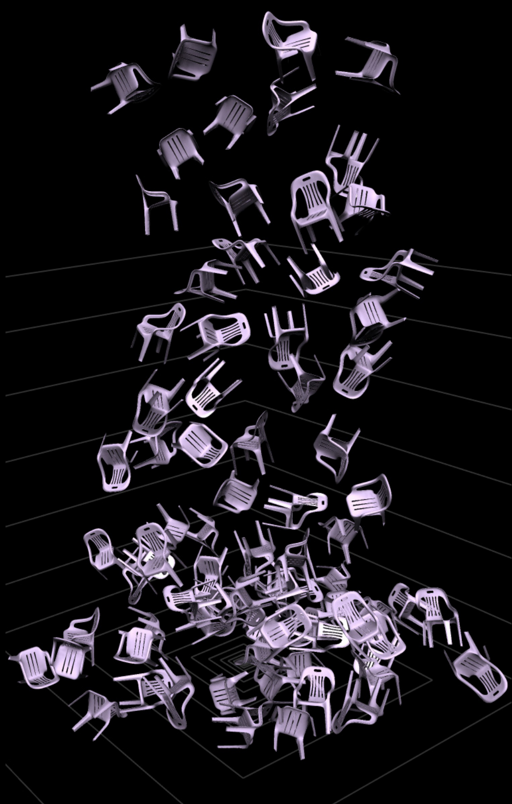
Doug L. James

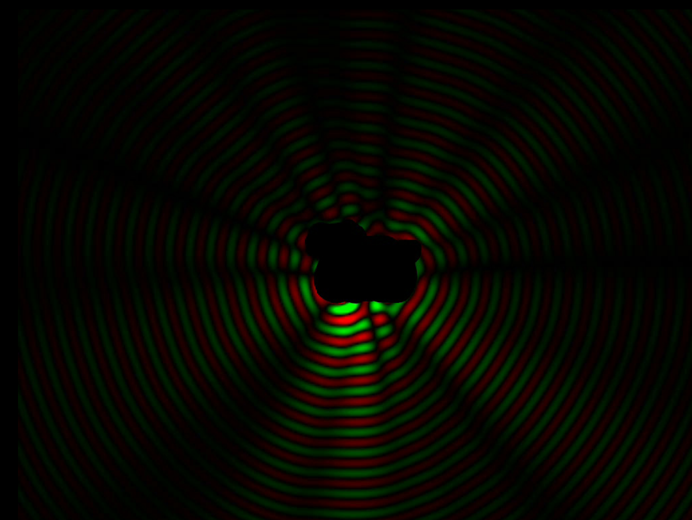
Cornell University

Acknowledgements

- Jernej Barbic
- Christopher Twigg
- Dinesh Pai

Multi-Sensory Simulation





Graphics **Haptics** **Acoustics**

New algorithms (scalable, real-time, multi-rate, …)

Offline vs Real-Time Physics

Offline physics

- *Given an approximation of some accuracy, compute its solution as fast as possible*

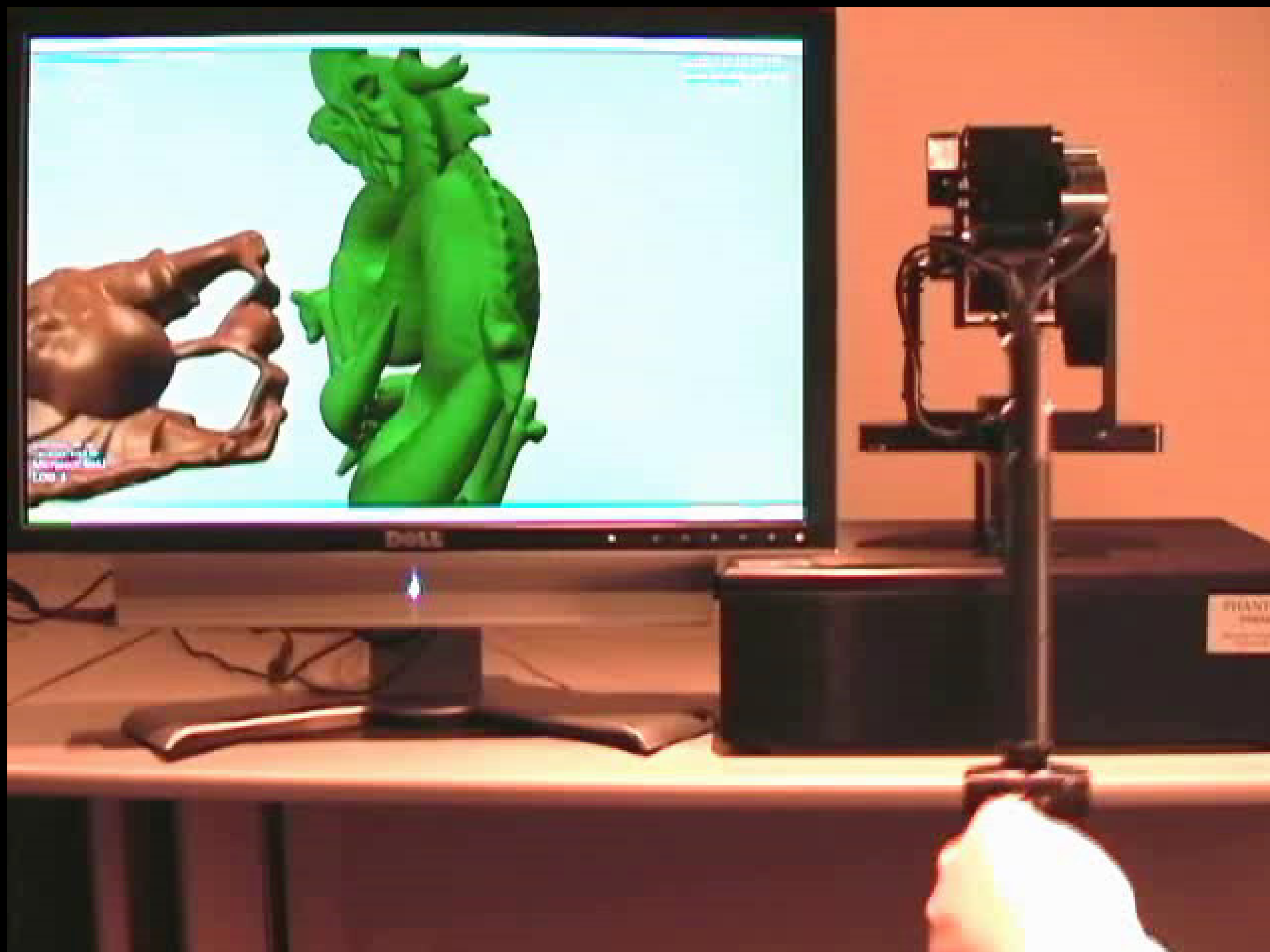
Real-time physics

- *Given a fixed time budget, compute a solution as accurately as possible*
- *Rates often dictated by human sensory system*

Different algorithm design issues

- *Precomputation useful for real-time physics*
- *"Drag race" analogy*

Haptics Preview



[Barbic & James 2007]

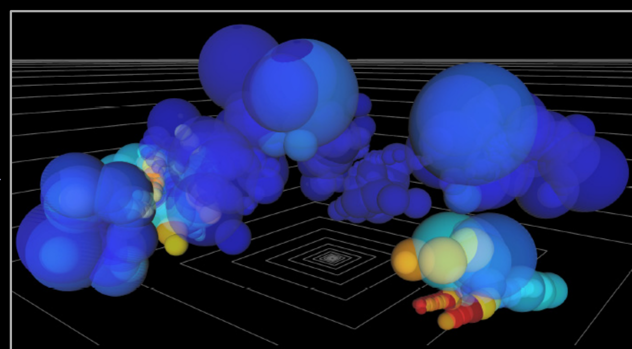
Class Overview

1. Reduced-order deformable models
2. Haptic force-feedback rendering
3. Sound synthesis
4. Interactive motion design
    - Many-Worlds Browsing

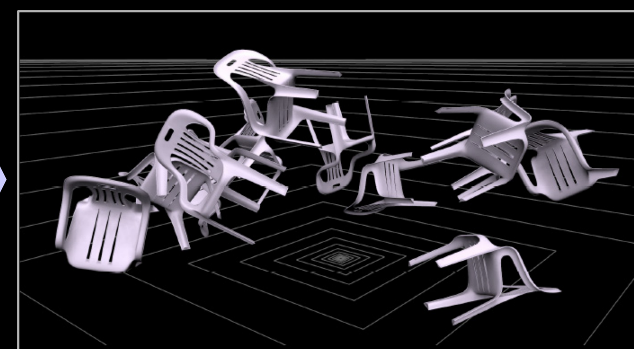
Big Picture

Multi-sensory Feedback

Parameterize



Output-sensitive Processing and Communication



Synthesis

Precomputation | Runtime

Reduced-order Deformable Models

References

Real-Time Subspace Integration for St.Venant-Kirchhoff Deformable Models

Jernej Barbič Doug James *

Carnegie Mellon University

Abstract

In this paper, we present an approach for fast subspace integration of reduced-coordinate nonlinear deformable models that is suitable for interactive applications in computer graphics and haptics. Our approach exploits dimensional model reduction to build reduced-coordinate deformable models for objects with complex geometry. We exploit the fact that model reduction on large deformation models with *linear* materials (as commonly used in graphics) result in internal force models that are simply cubic polynomials in reduced coordinates. Coefficients of these polynomials can be precomputed, for efficient runtime evaluation. This allows simulation of nonlinear dynamics using fast implicit Newmark subspace integrators, with subspace integration costs independent of geometric complexity. We present two useful approaches for generating low-dimensional subspace bases: modal derivatives and an interactive sketching technique. Mass-scaled principal component analysis (mass-PCA) is suggested for dimensionality reduction. Finally, several examples are given from computer animation to illustrate high performance, including force-feedback haptic rendering of a complicated object undergoing large deformations.

CR Categories: I.6.8 [Simulation and Modeling]: Types of Simulation—Animation, I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling, I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: simulation, animation, deformation, precomputation, model reduction, finite element method, interactive, haptics

1 Introduction

Objects undergoing physically based *large* deformations play an important part of computer graphics and animation where shape changes must be visible, and their simulation is notorious for being computationally demanding. For example, the high update rates of force-feedback haptic rendering make it difficult to accurately simulate large deformations, especially with complex geometry and distributed contact interactions (see Figure 1). Many interactive and offline simulations, such as those used in the computer animation industry, would also benefit from having highly interactive large deformation models.

In this paper, we show that dimensional model reduction on deformable models with geometric nonlinearities but *linear materials*, as commonly used in graphics (the so-called St. Venant-Kirchhoff model, or StVK), can lead to extremely fast and precomputable approximations for real-time applications. We exploit the fact that

*e-mail: barbic|djames@cs.cmu.edu

Figure 1: **Large-deformation dynamics at kilohertz rates:** *Force-feedback haptic rendering of distributed contact interactions between a user-controlled ball and a flexible bridge model (59630 triangles, $r = 15$). Subspace dynamics and contact handling are simulated at a hard real-time (1000 Hz) update rate. Reduced coordinates are exploited for real-time Bounded Deformation Tree collision processing [James and Pai 2004].*

dimensional model reduction in this case results in internal force models that are simply cubic polynomials in reduced coordinates. Coefficients of these reduced force polynomials can be precomputed for efficient runtime evaluation of exact internal forces and stiffness matrices. All the integration costs are independent of geometric complexity. Consequently, large deformation physics can be integrated at extremely fast rates using trusted subspace integrators, e.g., implicit Newmark, while graphical rendering is done at slower rates. For example, the large bridge example shown in Figure 1 can only be dynamically rendered at about 40 Hz, but its dynamics can be integrated at more than a kilohertz, thus enabling haptic simulations of complex large-deformation models. In general, the integration speed is proportional to the number of subspace dimensions employed, e.g., with 4 dimensions the bridge dynamics can be integrated at over 200 kHz.

Our proposed approach is most closely related to linear modal vibration models, first introduced to graphics by Pentland and Williams [July 1989]. These linear dynamics models are simple and fast, and have seen extensive use [Shinya and Fournier 1992; Stam 1997; Basdogan 2001; James and Pai 2002; Hauser et al. 2003; Choi and Ko 2005]. Unfortunately, geometric linearity leads to distortions for large deformations, which is a significant limitation for computer graphics. Our proposed approach preserves several nice properties of linear modal analysis, but overcomes the serious limitation of linear Cauchy strain by employing full quadratic Green strain in all computations. Namely, we can still capture the large-scale motion of a model with very few modes. We also preserve the property that progressively more modes can be used to increase simulation accuracy.

Given the deformation basis, our approach can automatically generate a fast reduced-coordinate model. A key challenge there-

Real-Time subspace integration for St. Venant-Kirchhoff deformable models
Jernej Barbič and Doug James
ACM Transactions on Graphics, 24(3),
August **2005**, pp. 982-990.

DyRT: Dynamic Response Textures for Real Time Deformation Simulation with Graphics Hardware

Doug L. James and Dinesh K. Pai
Department of Computer Science, University of British Columbia

Abstract

In this paper we describe how to simulate geometrically complex, interactive, physically-based, volumetric, dynamic deformation models with *negligible main CPU costs*. This is achieved using a ***Dynamic Response Texture***, or **DyRT**, that can be mapped onto any conventional animation as an optional rendering stage using commodity graphics hardware. The DyRT simulation process employs precomputed modal vibration models excited by rigid body motions. We present several examples, with an emphasis on bone-based character animation for interactive applications.

1 Introduction

In this paper we present an efficient rendering technique for simulating real time dynamic deformations for applications such as character animation. This is achieved using a ***Dynamic Response Texture***, or **DyRT**, that can be mapped onto any conventional animation (motion capture or keyframe or rigid body dynamics simulation) as an optional rendering stage. This is because the complexity of rendering deformations using DyRT is comparable to lighting the object. Therefore, *every* deformable object, large or small, can be rendered with realistic dynamic deformation responses, in real time, on commodity graphics hardware.

The physical realism of DyRT is due to the use of precomputed modal analyses [22] of dynamic elastic models computed using, e.g., the Finite Element Method (FEM) [29]. These systems typically have a few clearly dominant dynamic deformation modes that enable us to produce convincing realizations on commodity graphics cards. This is achieved using vertex programs [14] that perform the per-vertex linear superpositions necessary to compute displacement and normal vectors.

A second key component of a DyRT is the use of *rigid motion transfer functions* for rigid (bone) motion input dependence, so that DyRTs respond physically and are not just "canned vibrations." This is in contrast to, e.g., several nVidia vertex program demos [14], such as "warp," which, although extremely useful in context, have limited physical foundations.

In the remainder of this paper we describe the foundations for and the process of applying DyRT to objects, with a particular emphasis on bone-based character animation.

1.1 Related Work

Significant work has been done on simulating dynamic deformable objects, in areas such as human body modeling and interactive simulation. Despite the large amount of pioneering work on deformation [25, 28, 15, 1], there continue to be exciting new applications [18, 17] and improvements in simulation efficiency [2, 6].

Numerous examples of human body modeling exist in the literature with particular areas of interest being deformations of skin and muscles [27, 9], faces [12], and layered models [5]. Support exists in commercial animation packages, such as Maya, for simulating tissue dynamics. There have also been significant recent developments for interactive dynamic tissue simulation, especially for force feedback applications such as surgical simulation [6, 20]. Despite these advances, the simulation of transient vibration responses for secondary animation remains largely absent from the traditional character animation pipeline, and especially so in video games.

Of particular interest for graphics hardware are data-driven deformation models based on linear superposition of precomputable *global deformation bases* [3], which include space warping methods such as FFD. While such models can provide fast simulation and constraint handling for physically-based dynamic [19, 28] and also static [11, 10] deformable models, we are primarily interested in their amenability to graphics hardware simulation [14].

For simulating *free vibrations* of elastic models with modest amplitudes, global deformation bases based on Karhunen-Loeve expansions from modal analysis provide the optimal description [22, 8]. First introduced to the graphics community by the pioneering work of Pentland and Williams [19, 8], more recently they have been used for interactive applications involving precomputed or measured modal data: stochastic simulation of tree-like structures [23], force feedback [4], and contact sound simulation [26].

Our contribution: This is the first paper to show how to simulate geometrically complex, interactive, physically-based, volumetric, dynamic deformation models in real time with *negligible main CPU costs*. We do so with precomputed modal vibration models stored in graphics hardware memory and driven by a handful of inputs defined by rigid body motion.

2 Background on Modal Vibration Models

We briefly summarize the necessary background on modal vibration analysis here, and refer the reader to a suitable text [22]. The linear elastodynamic equation for a finite element model [29],

$$\mathsf{M}\ddot{\mathsf{u}} + \mathsf{C}\dot{\mathsf{u}} + \mathsf{K}\mathsf{u} = \mathsf{F}, \quad (1)$$

DyRT: Dynamic Response Textures for Real Time Deformation Simulation With Graphics Hardware
Doug L. James and Dinesh K. Pai
ACM Transactions on Graphics, 21(3),
July **2002**, pp. 582-585.

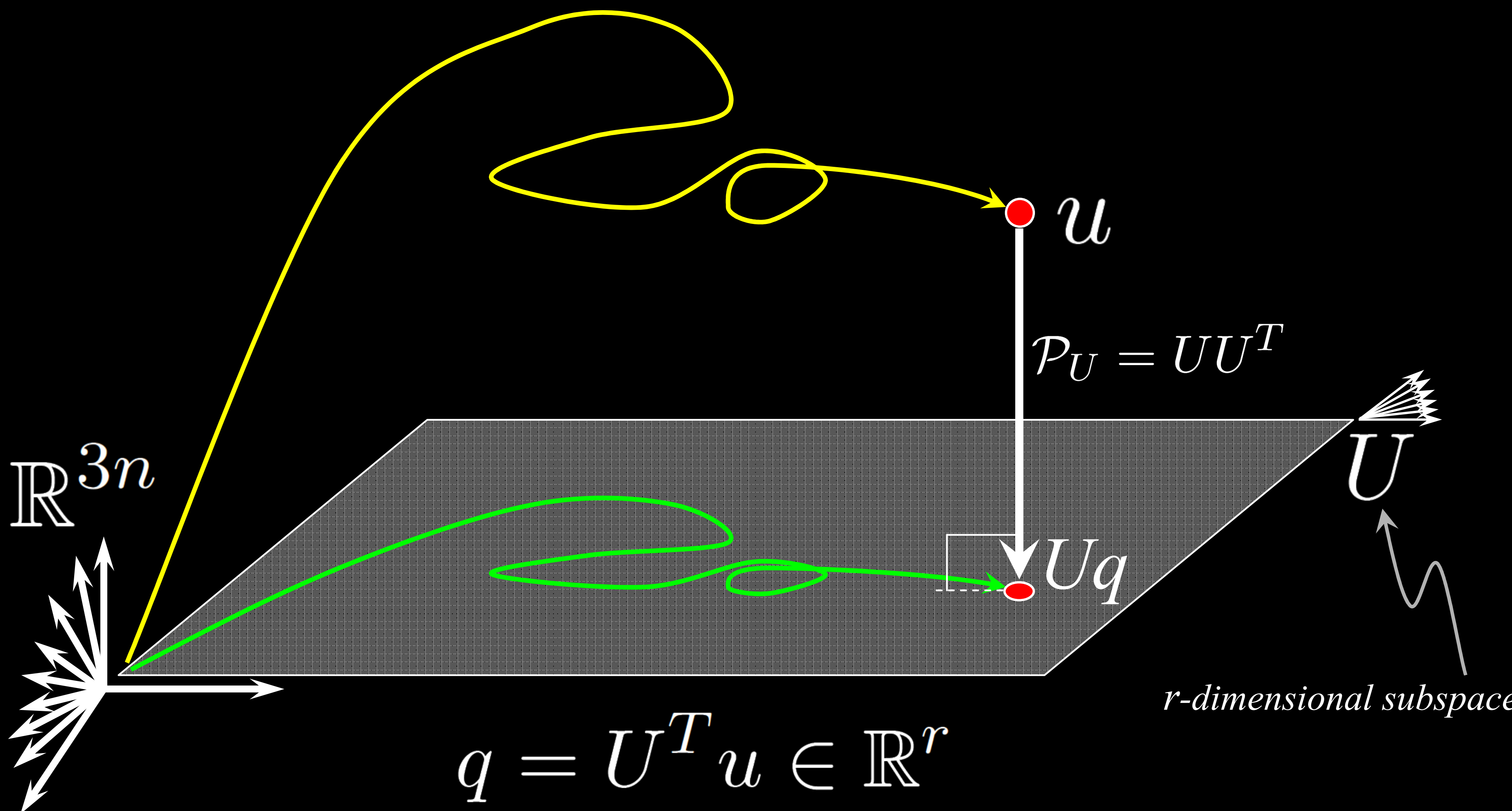
Warm-up: Modal Deformations

$$\begin{pmatrix} u \end{pmatrix} = \begin{pmatrix} U_1 \end{pmatrix} q_1(t)$$

$$\begin{pmatrix} u \end{pmatrix} = \begin{pmatrix} U_1 \end{pmatrix} q_1(t) + \begin{pmatrix} U_2 \end{pmatrix} q_2(t)$$

$$u = U\ q \quad \text{where} \quad q \in \mathbb{R}^r$$

Geometric View of Subspace Deformations



$$q = U^T u \in \mathbb{R}^r$$

"reduced coordinates"

Dimensional Model Reduction

- Systematic reduction in model DoFs
  - Projection onto lower-dimensional basis
- Classical example:
  - *Linear modal analysis* (see texts by Shabana)
  - U are generalized eigenvectors of M and K:

$$K\mathbf{u}_m = \omega_m^2 M \mathbf{u}_m$$

eigenvalue *eigenvector*

Linear Vibration Modes

Dimensional Model Reduction

$$M\ddot{u} + R(u) = f$$

Substitute $u = Uq$

Project U^T

Identity

$$\overbrace{U^T M U}^{\text{Identity}} \ddot{q} + U^T R(Uq) = U^T f$$

Dimensional Model Reduction

$$M\ddot{u} + R(u) = f$$

Substitute $u = Uq$

Project U^T

$$\overbrace{U^T M U}^{\text{Identity}} \ddot{q} + U^T R(Uq) = U^T f$$

$$\boxed{\ddot{q}(t) + \tilde{R}(q(t)) = \tilde{f}(t)}$$

Reduced Equations of Motion

[Krysl, Lall, Marsden 2001]

Reduced Internal Forces

slow in general → R

[Krysl, Lall, Marsden 2001]

$$\tilde{R}(q) = U^T R(Uq)$$

Reduced Internal Forces: Linear Modal Analysis

$$\tilde{R}(q) = U^T R(Uq)$$

$$= \begin{pmatrix} \omega_1^2 \, q_1 \\ \omega_2^2 \, q_2 \\ \vdots \\ \omega_r^2 \, q_r \end{pmatrix}$$

Linear force terms.

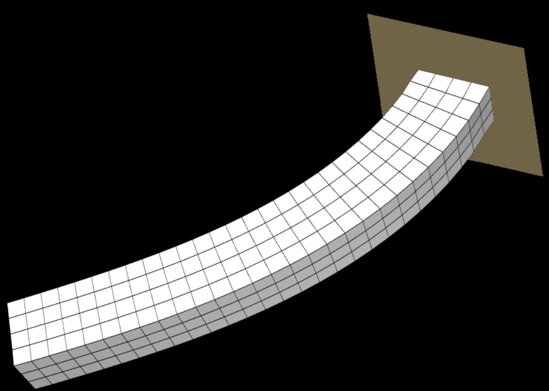
r decoupled oscillators!

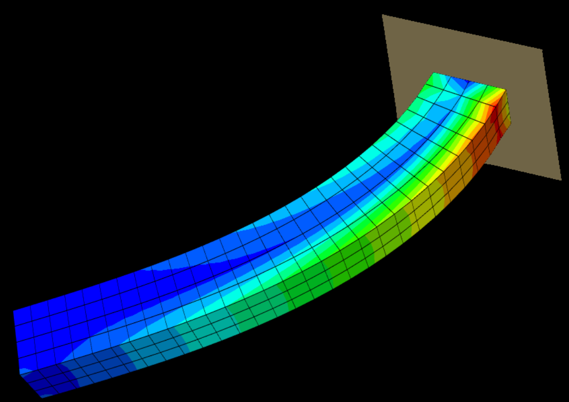
*Time-stepping is fast using **IIR filters***
[James and Pai 2002; van den Doel and Pai]

Background (solid mechanics):

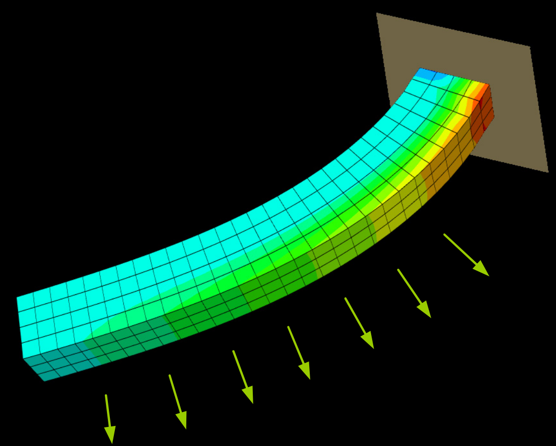
Internal Forces

mesh deformation → (nonlinear) → strain → (nonlinear) → stress



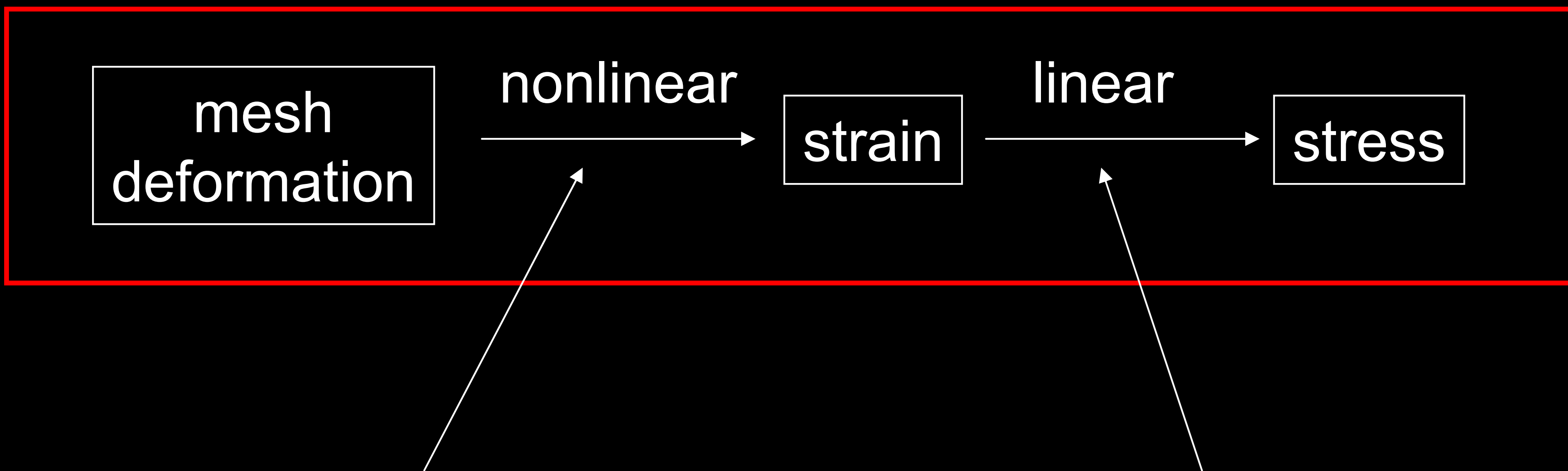


change in length vs. original length



internal forces per area

Def’n: St.Venant-Kirchhoff Model



“geometric nonlinearity”

- handles large deformations

“material linearity”

- approximation for speed

Reduced Internal Forces: St.-Venant Kirchhoff (StVK)

$$\tilde{R}(q) = U^T R(Uq)$$

$$= \begin{pmatrix} p_1(q) \\ p_2(q) \\ \vdots \\ \\ p_r(q) \end{pmatrix}$$

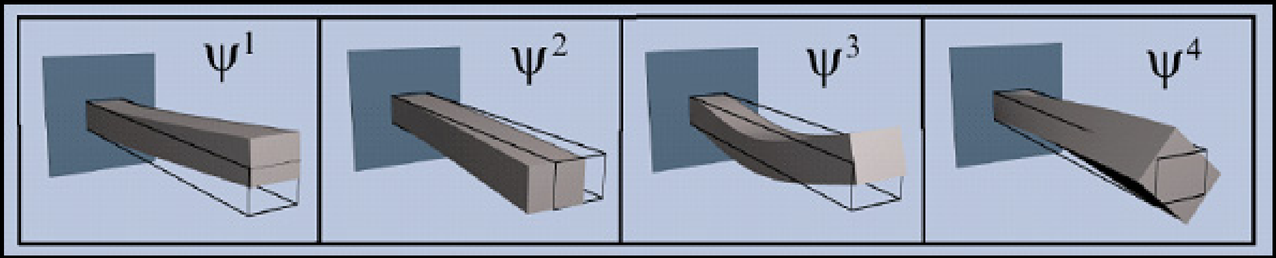
for StVK only

r cubic polynomials in components of q

$O(r^4)$ complexity

[Barbic and James 2005]

Large deformation basis?

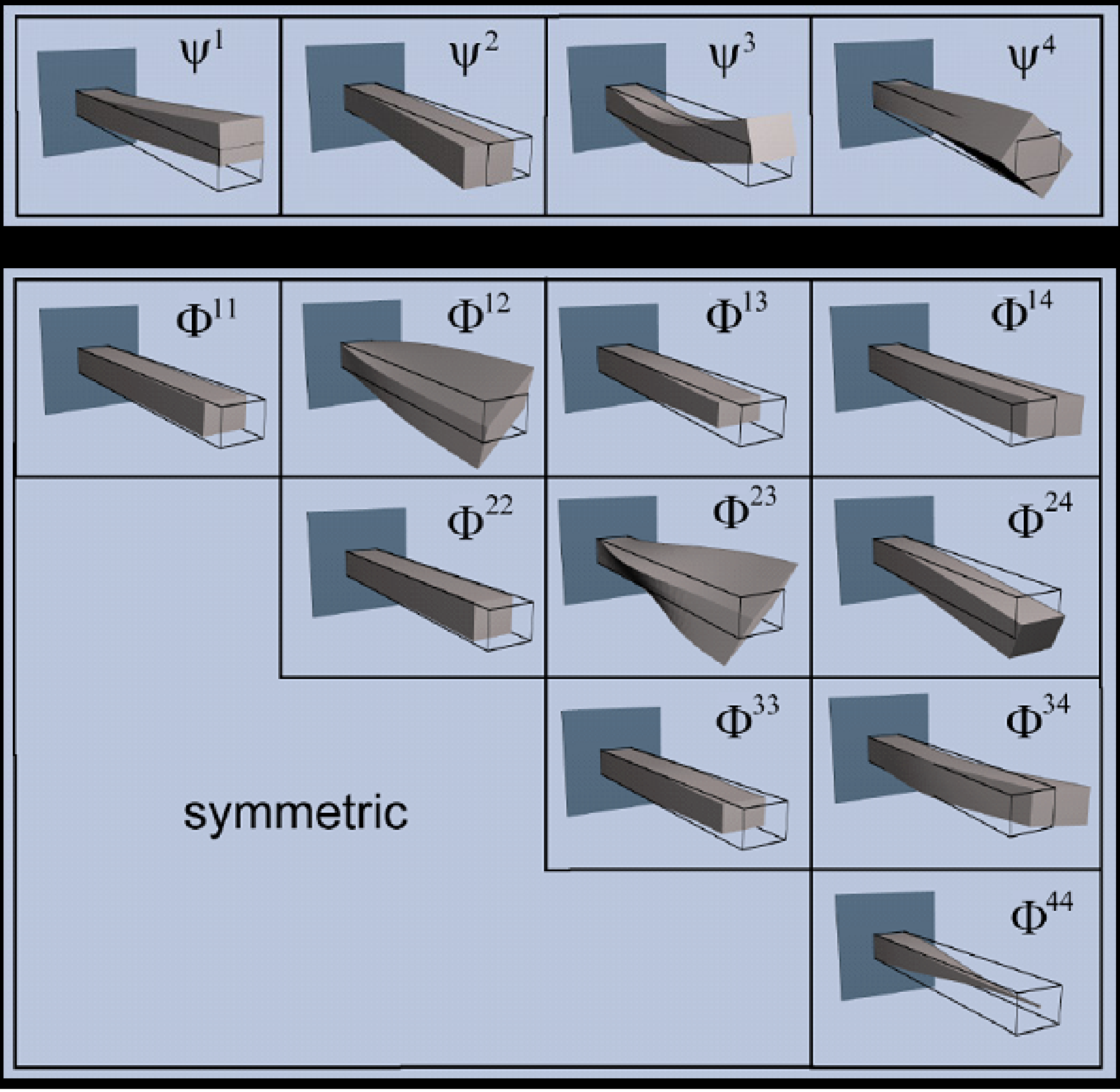


Linear Modes
k=4 shown

- Linear modes for small deformation only
- Large deformations don’t work
  - (although see Modal Warping **[Choi and Ko 2005]**)

Improving the basis:

Add Modal Derivatives



Linear Modes

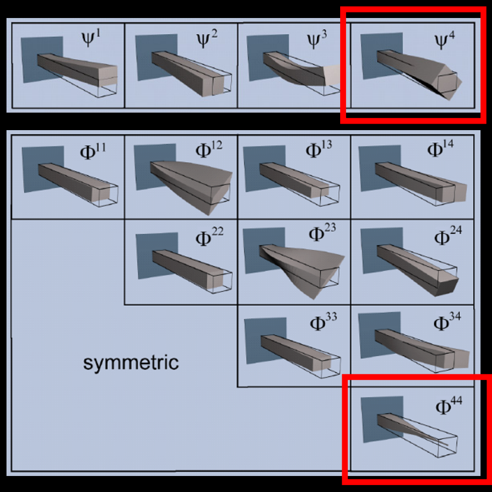
k=4 shown

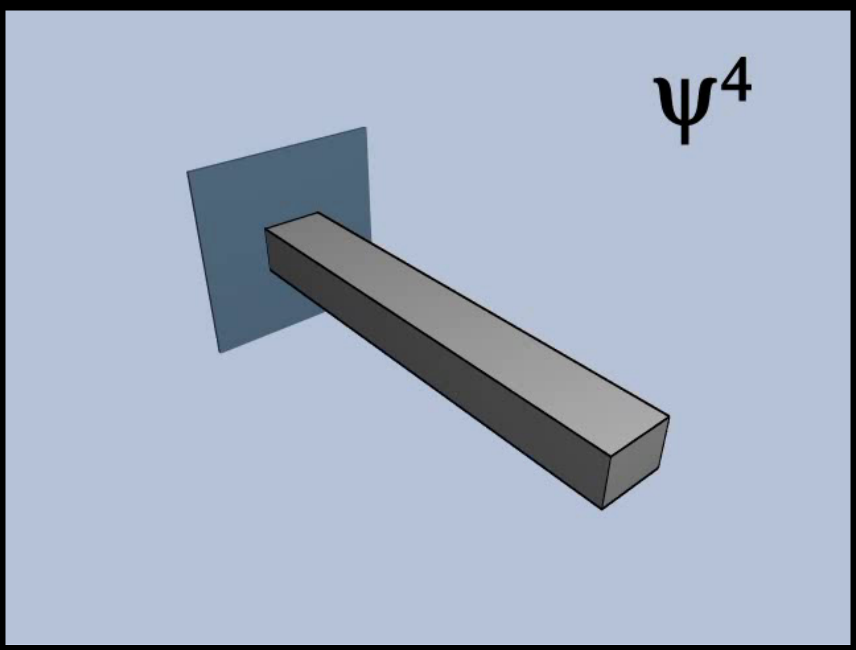
Modal Derivatives

k(k+1)/2 vectors

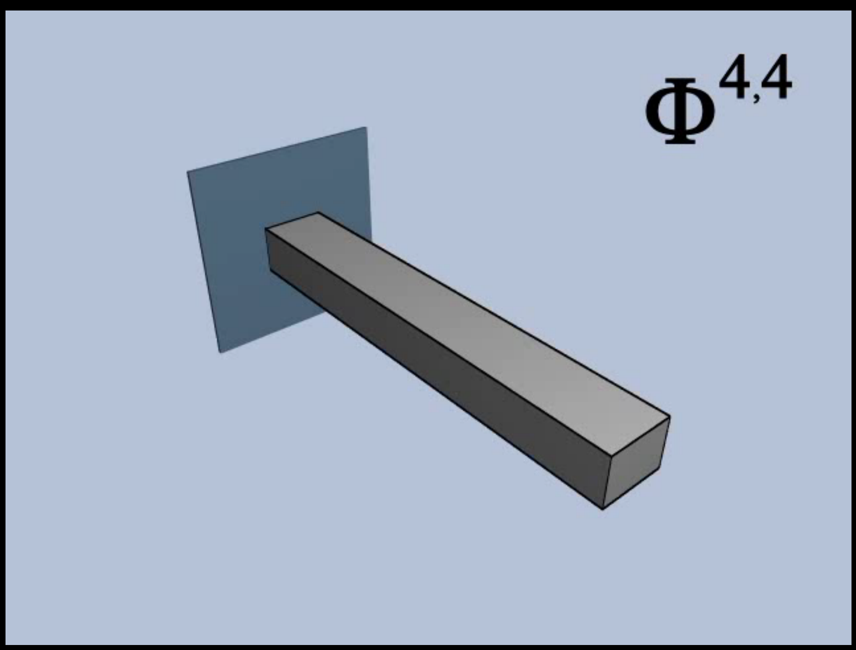
[Idelsohn and Cardona 1985]

Modal derivatives are nonlinear corrections to linear modes





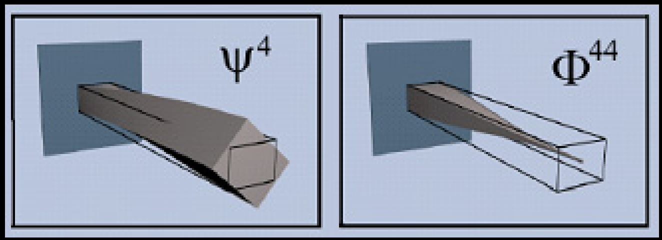
The “twist” linear mode and artifacts for large deformations

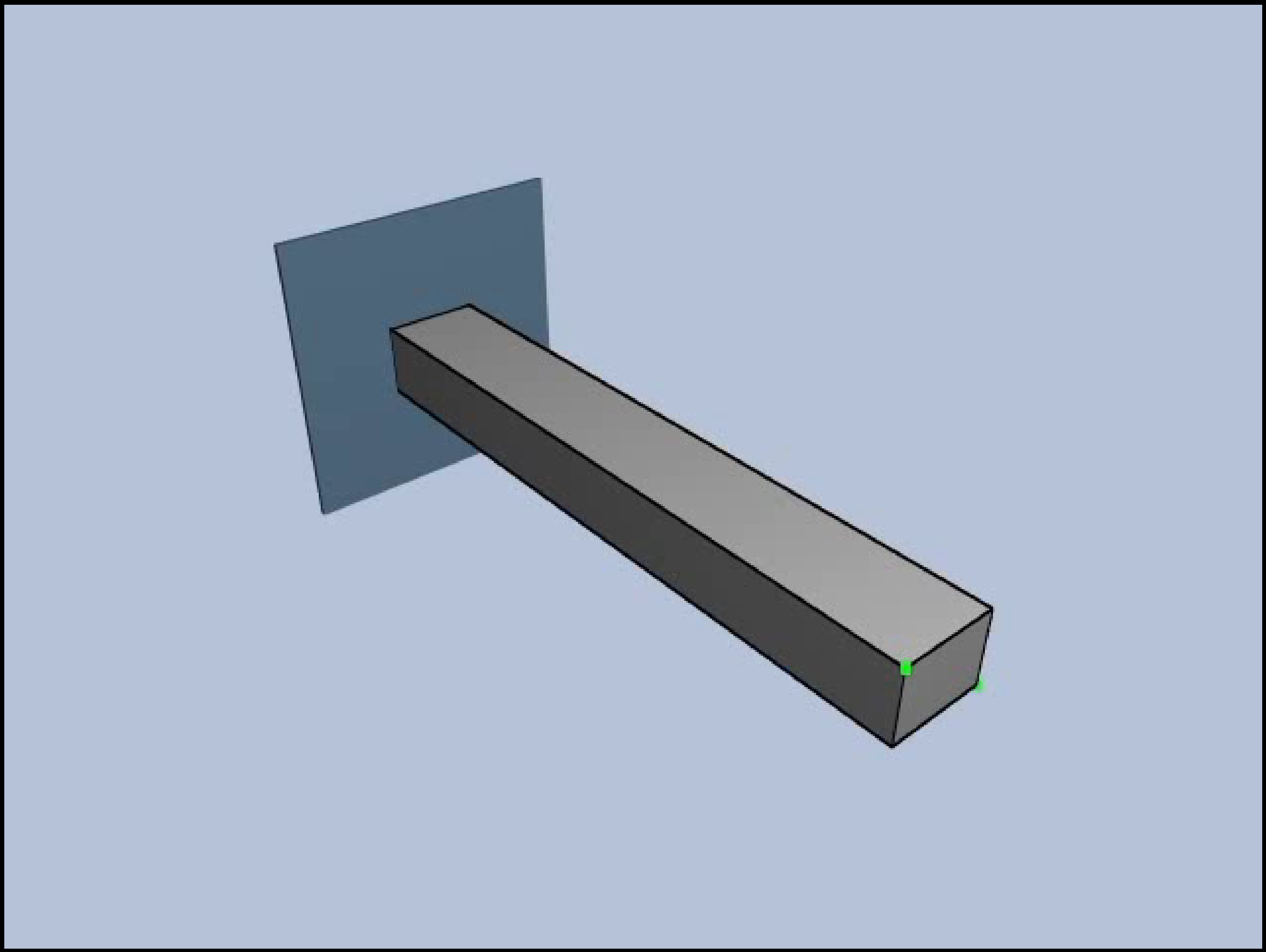


Modal derivative cancels volume growth

Runtime simulation: Modal Derivatives, r=2

Basis

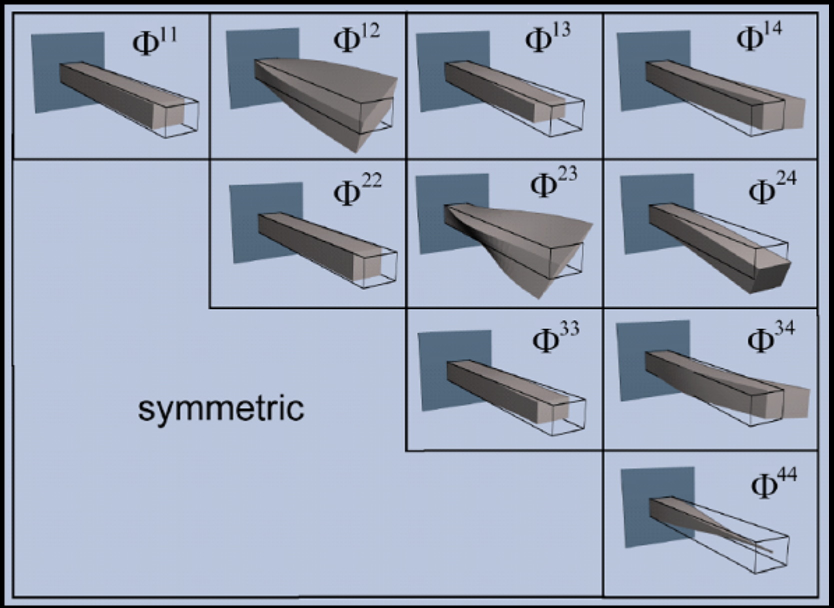




Precomputing Modal Derivatives

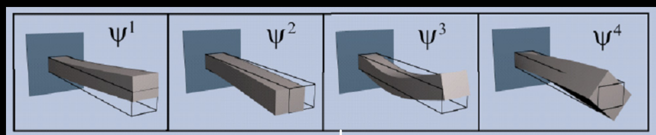
[Idelsohn and Cardona 1985]

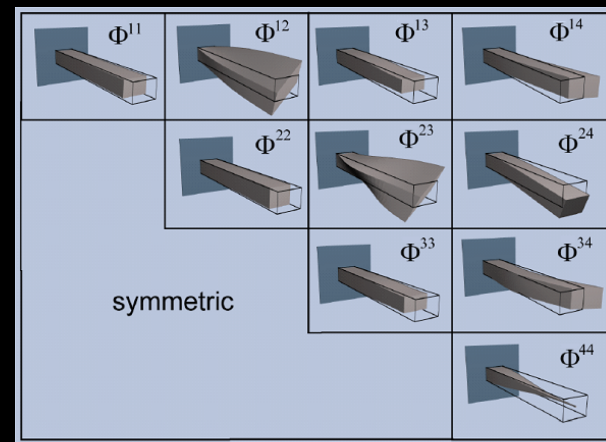
- Solutions to sparse linear systems of 3n equations



- 30 minutes total for models with 14000 elements (k=20)

Constructing the Motion Basis





Scale by frequency

Scale by frequency pair

Mass-scaled PCA

Basis of motion U

Why PCA?

- Redundancy
- Smaller r → faster

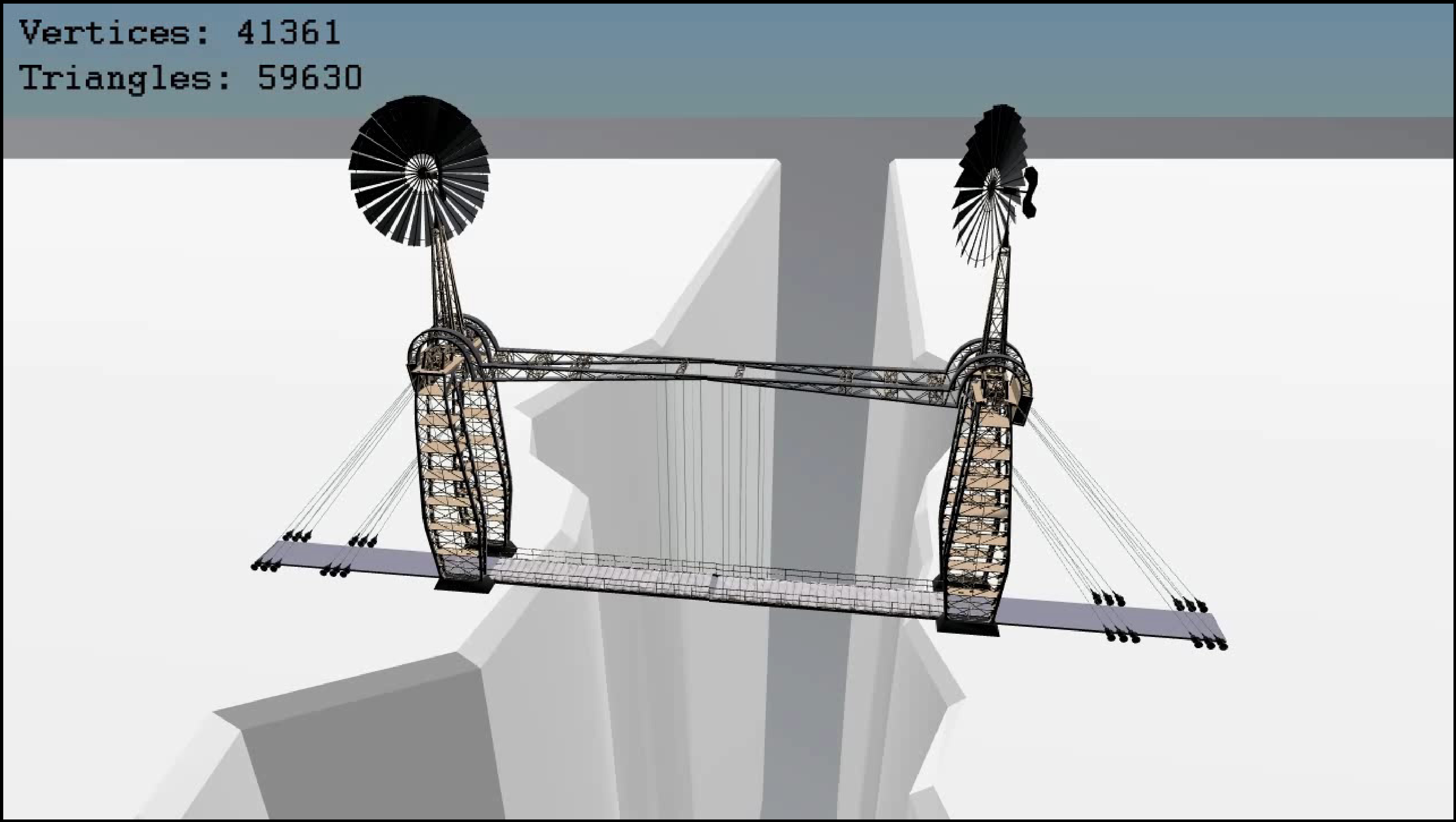
Deformable Spoon (modal derivatives)

r =12; 30 μsec; speedup $\geq$ 130,000x

Speedup estimate is conservative (all examples):
unreduced linear system solve time not included in timings

Bridge

r=15; 65 μsec; speedup ≥ 200,000x



Reduced Internal Forces: St.-Venant Kirchhoff (StVK)

$$\tilde{R}(q) = U^T R(Uq)$$

$$= \begin{pmatrix} p_1(q) \\ p_2(q) \\ \vdots \\ \\ p_r(q) \end{pmatrix}$$

for StVK only

r cubic polynomials in components of q

$O(r^4)$ complexity

[Barbic and James 2005]

CURRENT WORK

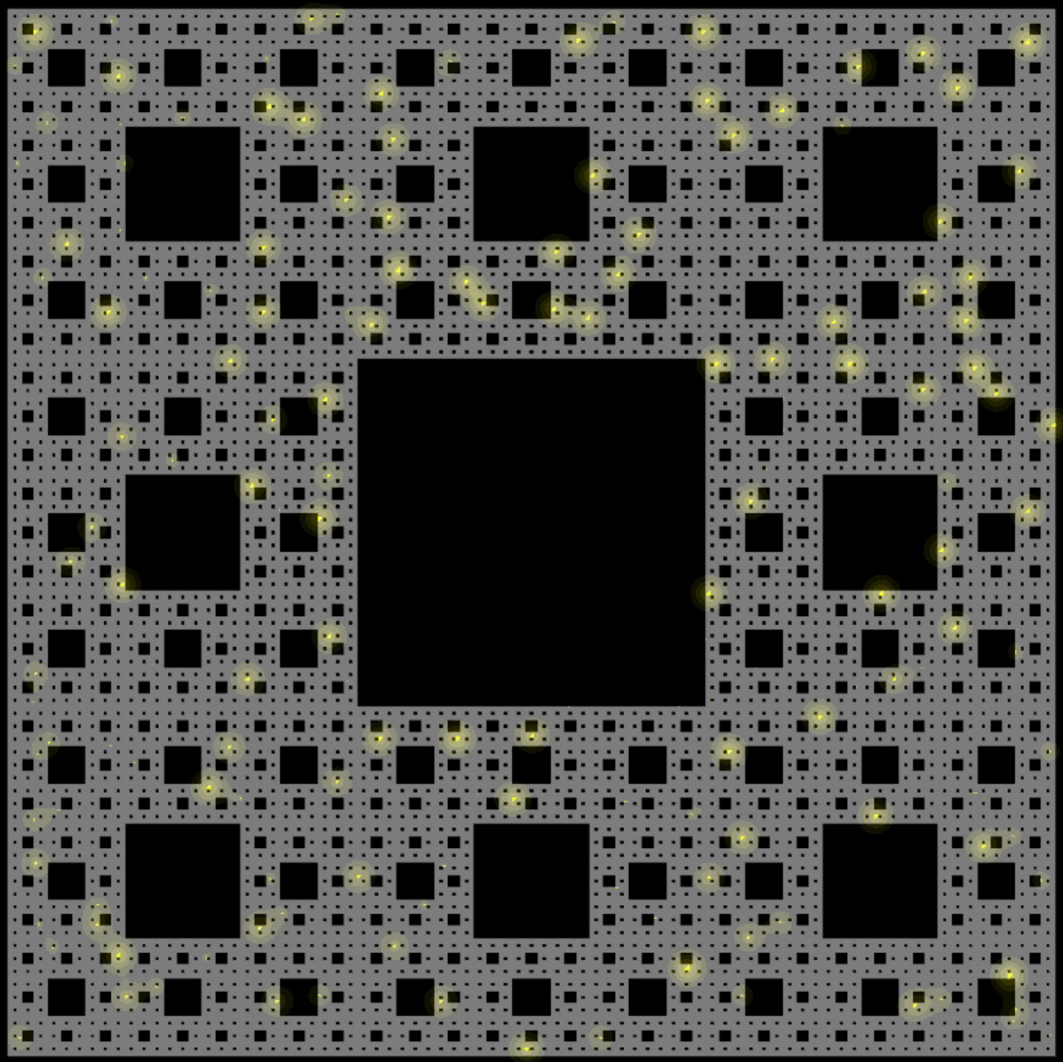
Cubature Optimization

[An et al. 2008] (to appear at SIGGRAPH Asia 2008)

$$\mathbf{f}(\mathbf{q}) = -\nabla_{\mathbf{q}} E(q)$$

$$= \int_{\Omega} \mathbf{g}(X\,;\mathbf{q})\ d\Omega_X$$

$$\approx \sum_{i=1}^{n} w_i\ \ \mathbf{g}(X_i;\mathbf{q})$$



Graphical Rendering

Hardware Rendering using q

- Geometry deformed & lit by shaders
- Vertex displacement [James and Pai 2002]

$$\mathbf{u}_i = \sum_{m=1}^{r} \mathbf{U}_{im}\, q_m$$

- Vertex normals [Rivers and James 2007]

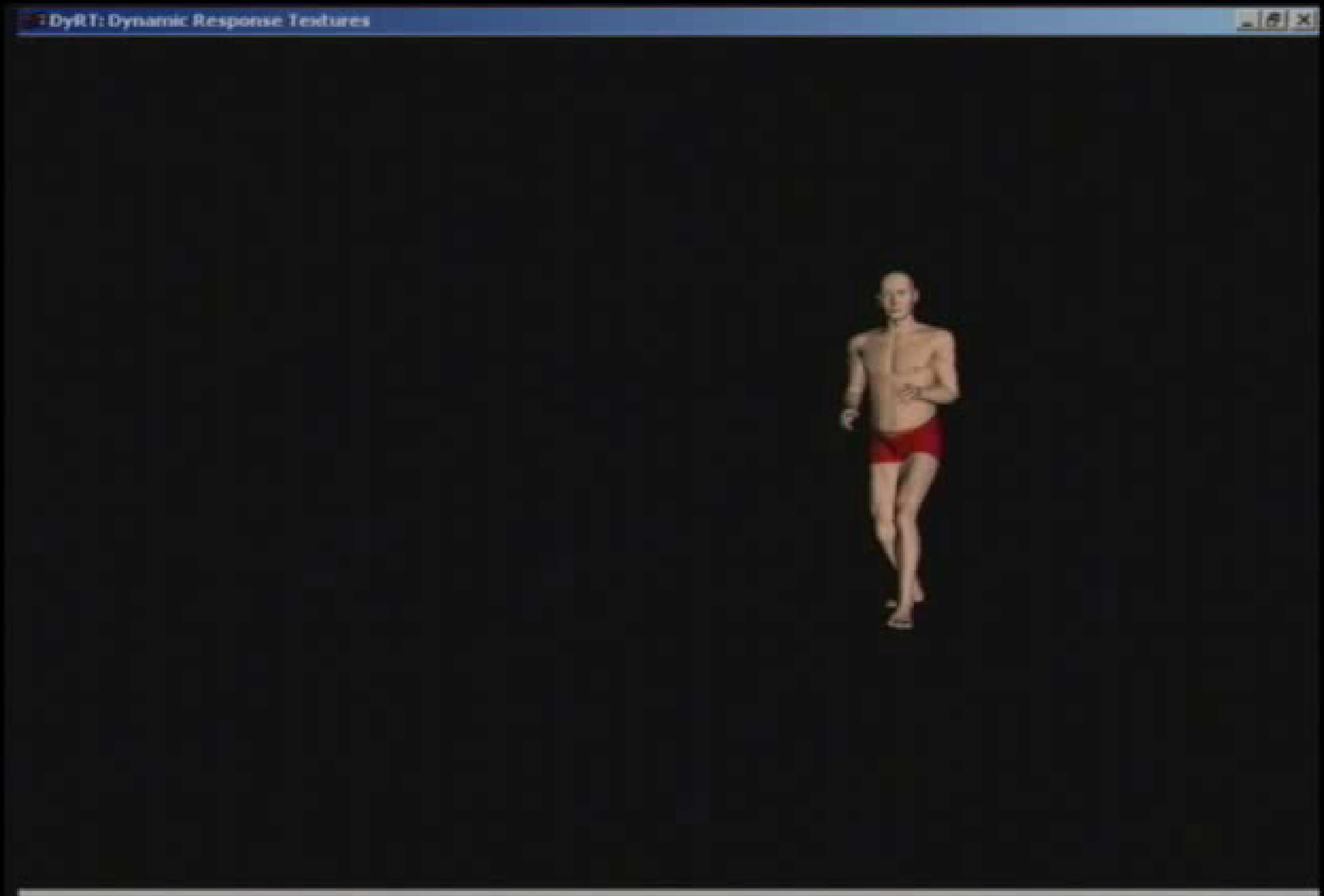
$$\mathbf{n}_i = \mathbf{s}_i \times \mathbf{t}_i$$

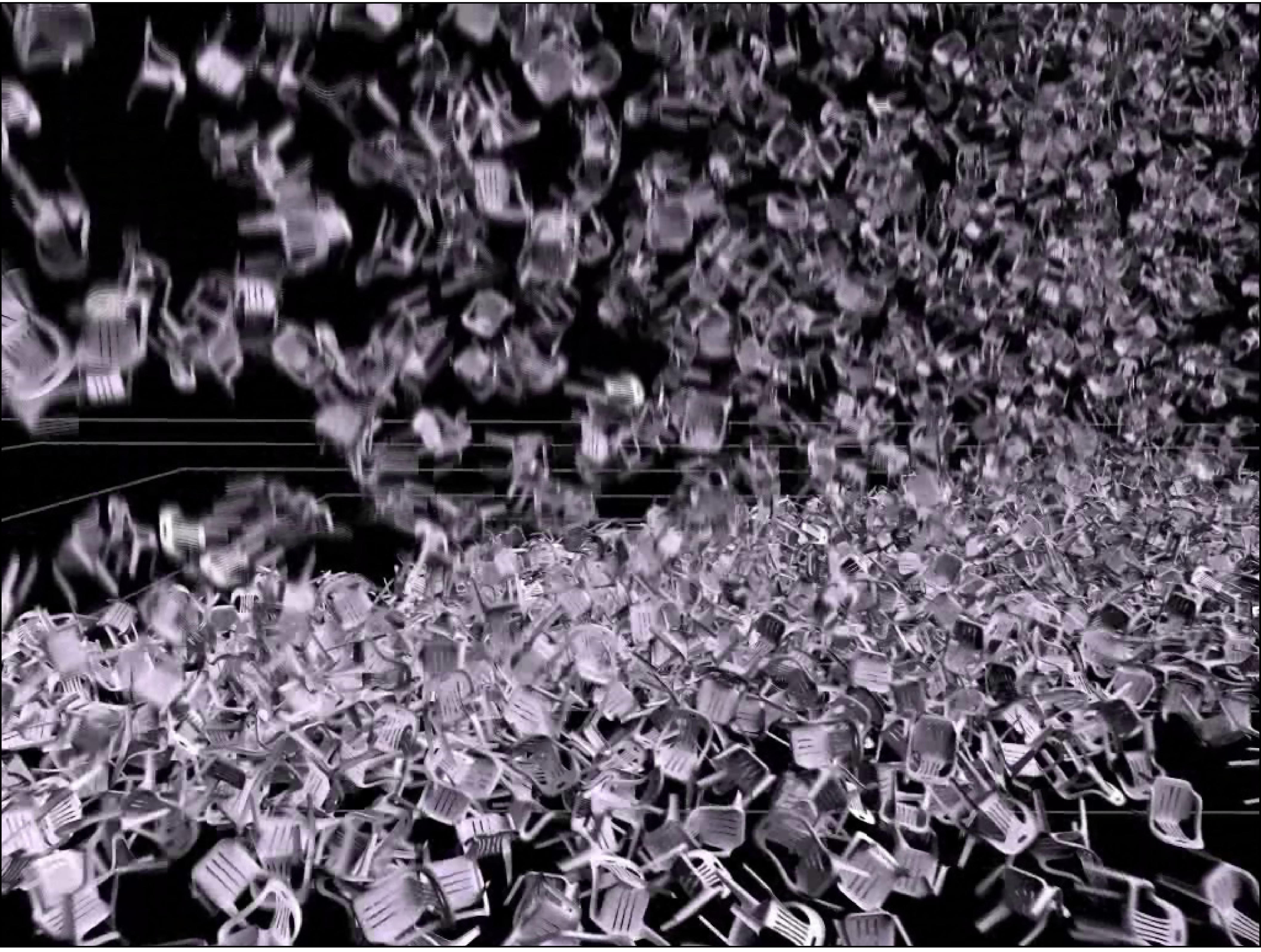
$$\mathbf{s}_i = \bar{\mathbf{s}}_i + \sum_{m=1}^{r} \mathbf{S}_{im}\, q_m$$

$$\mathbf{t}_i = \bar{\mathbf{t}}_i + \sum_{m=1}^{r} \mathbf{T}_{im}\, q_m$$

- Deformable PRT

Hardware Rendering: DyRT [James and Pai 2002]





Collision Processing

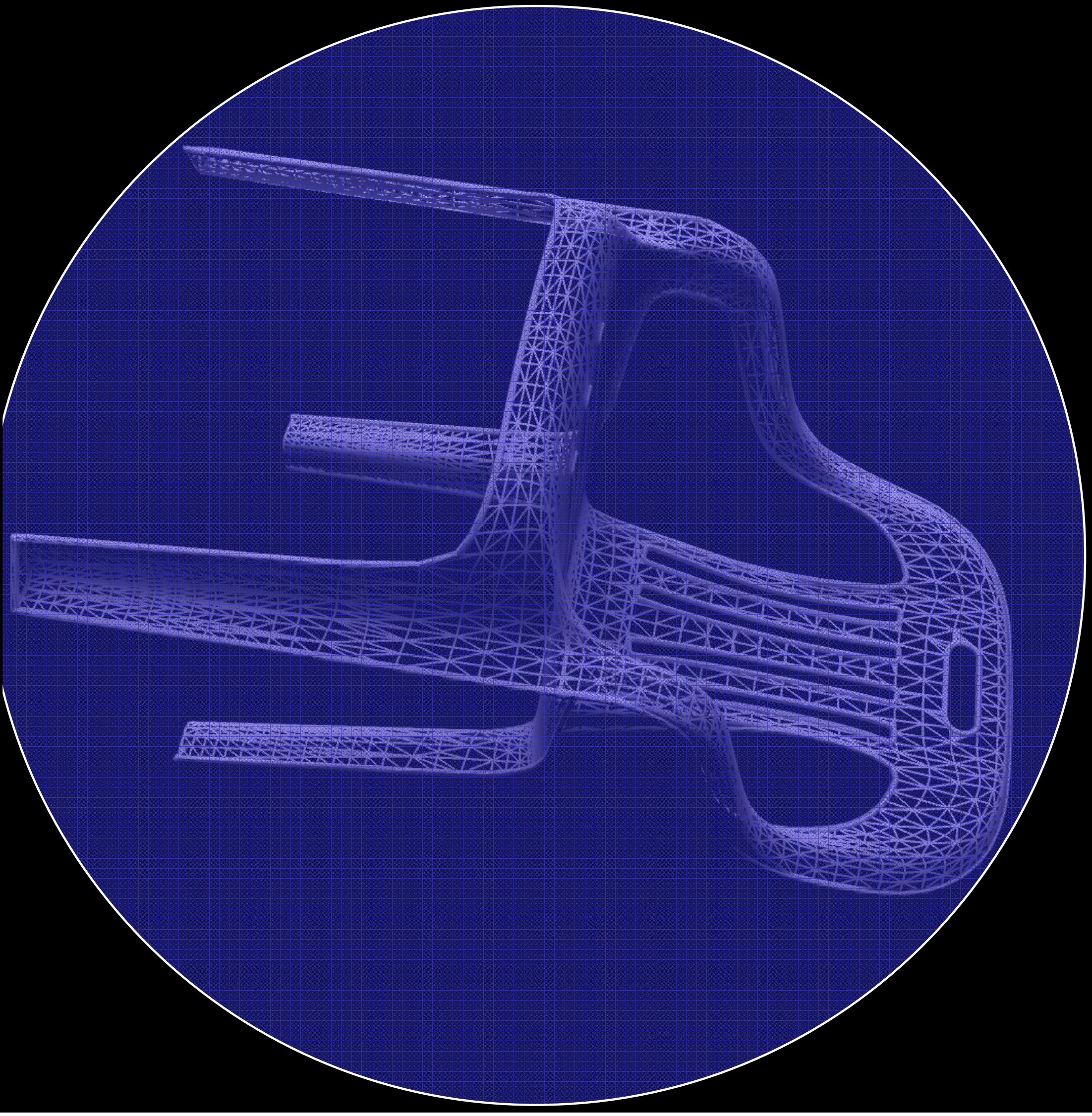
Bounded Deformation Trees

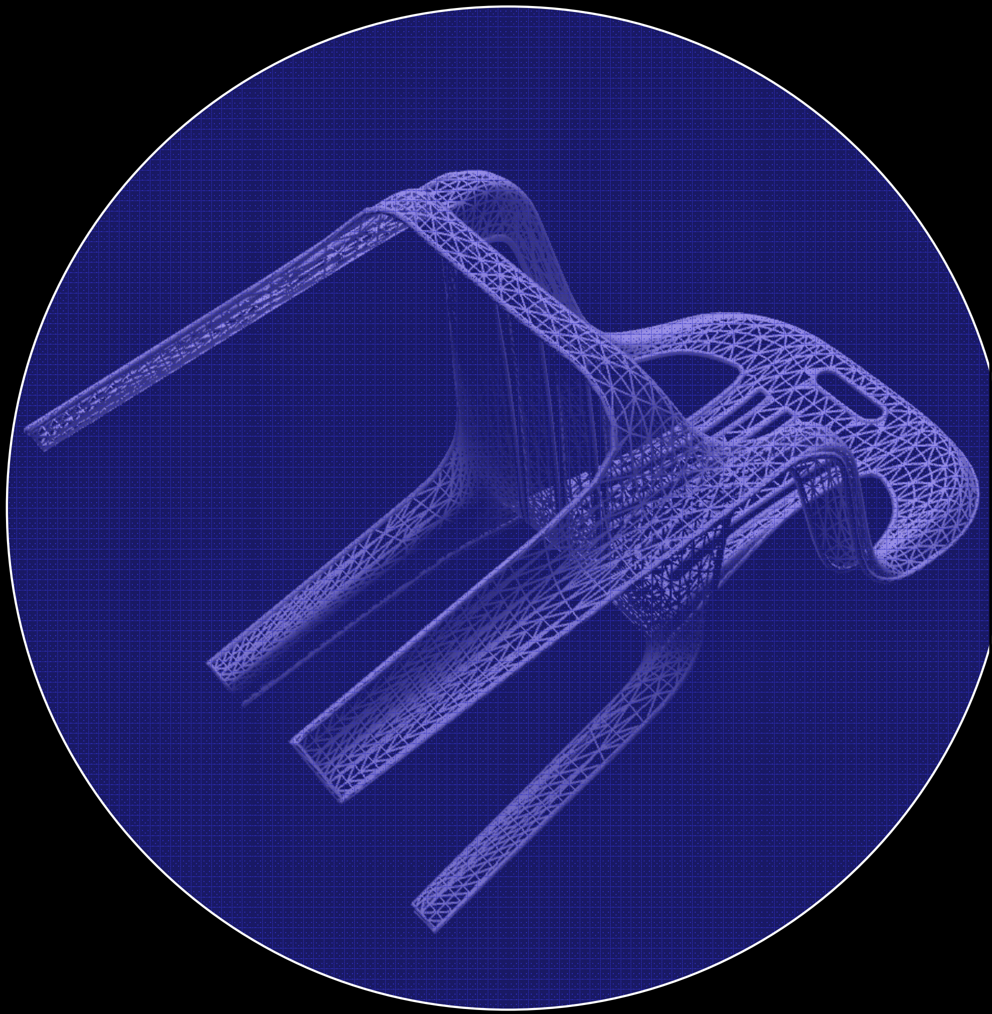
Doug L. James

Dinesh K. Pai

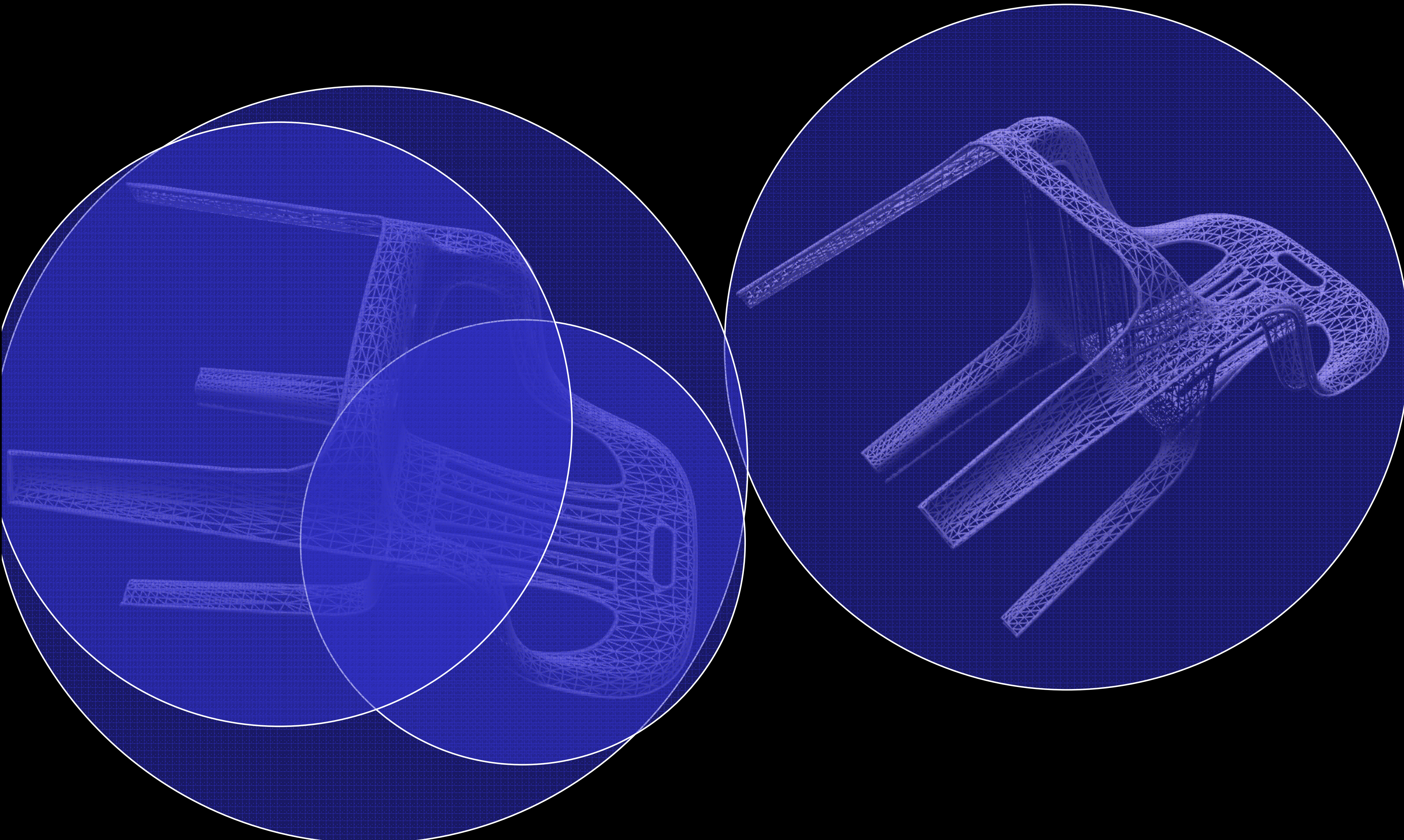
SIGGRAPH2004

Bounding Spheres

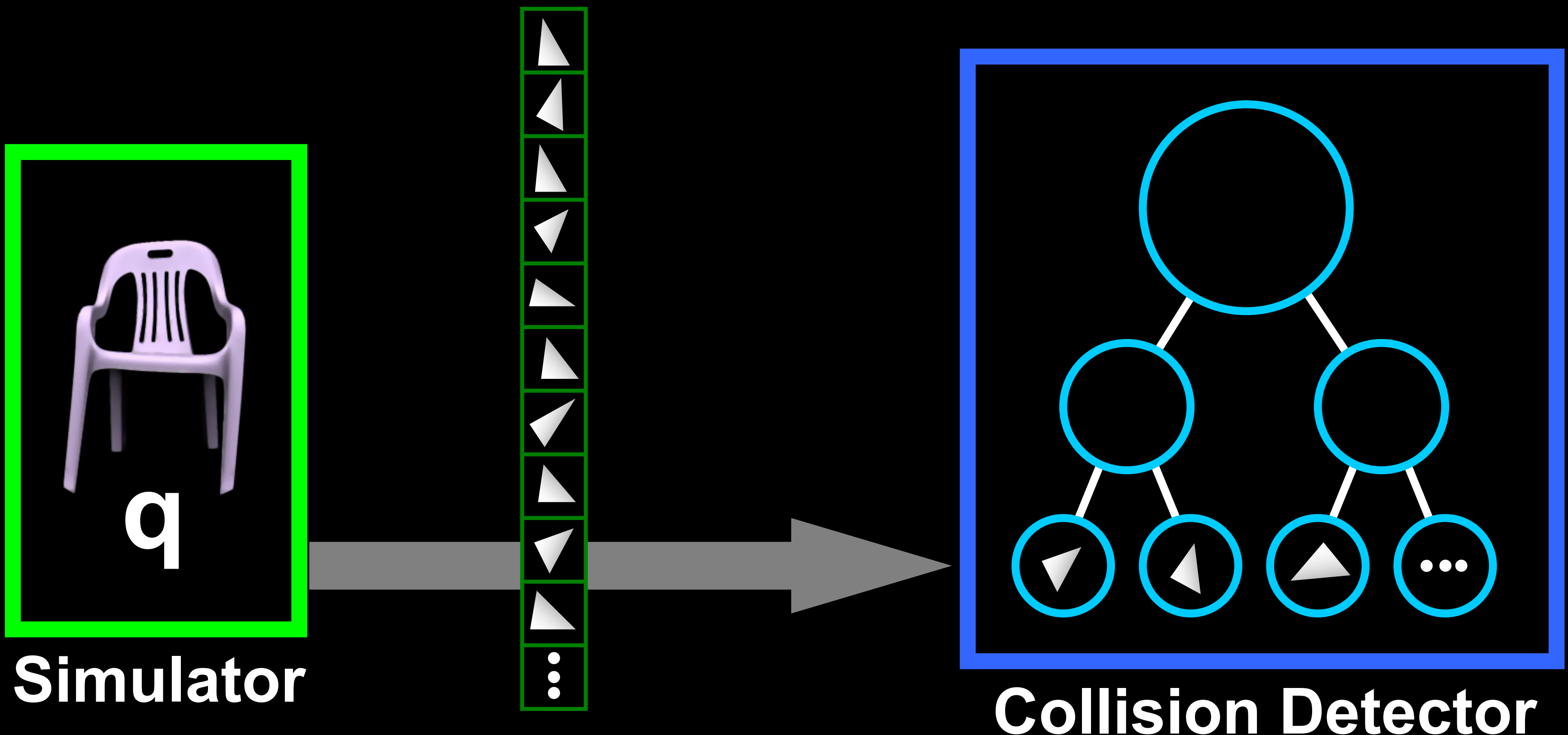




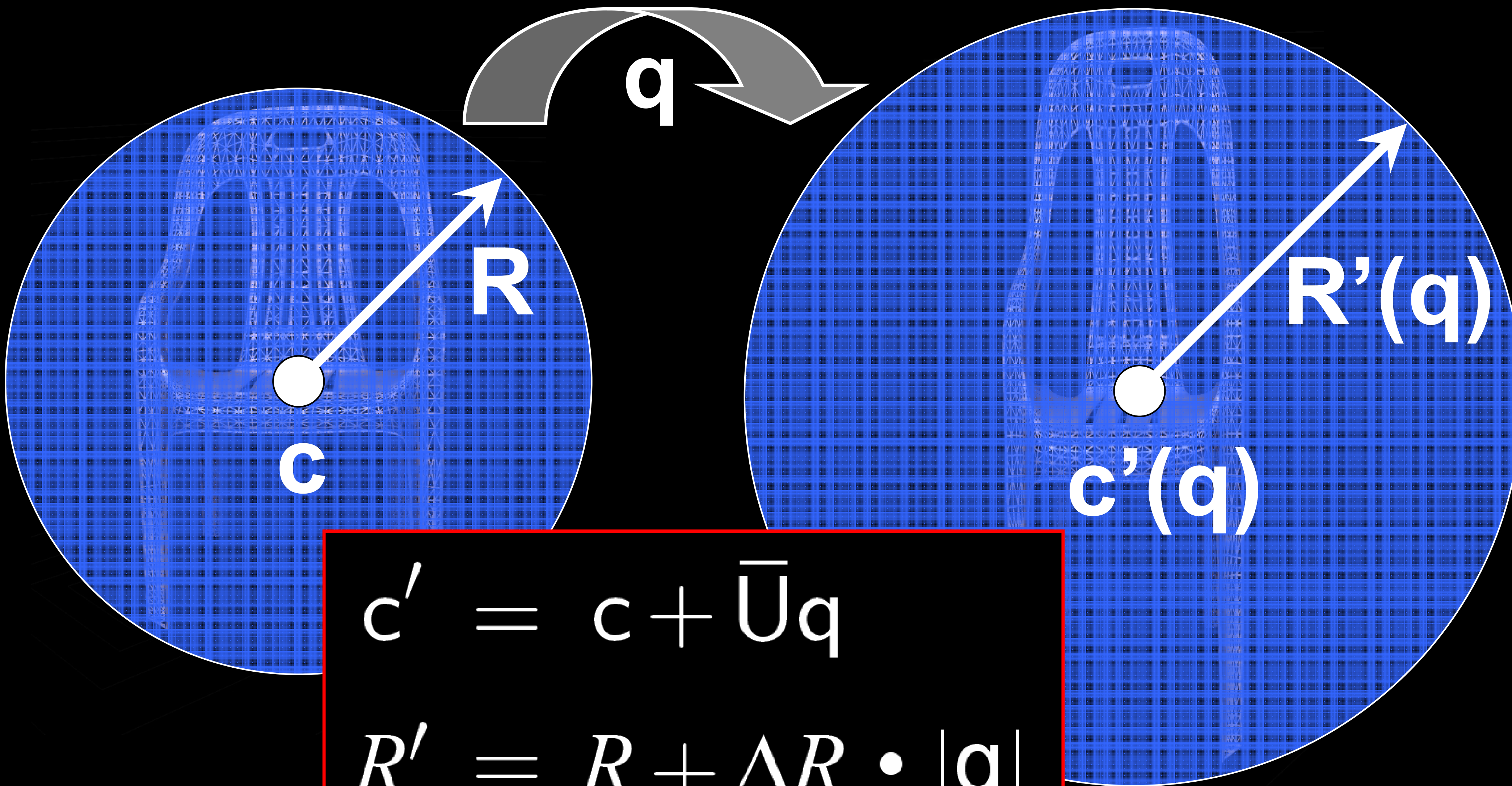
Bounding Sphere Hierarchy



Hierarchical Updates



Bounded Deformations



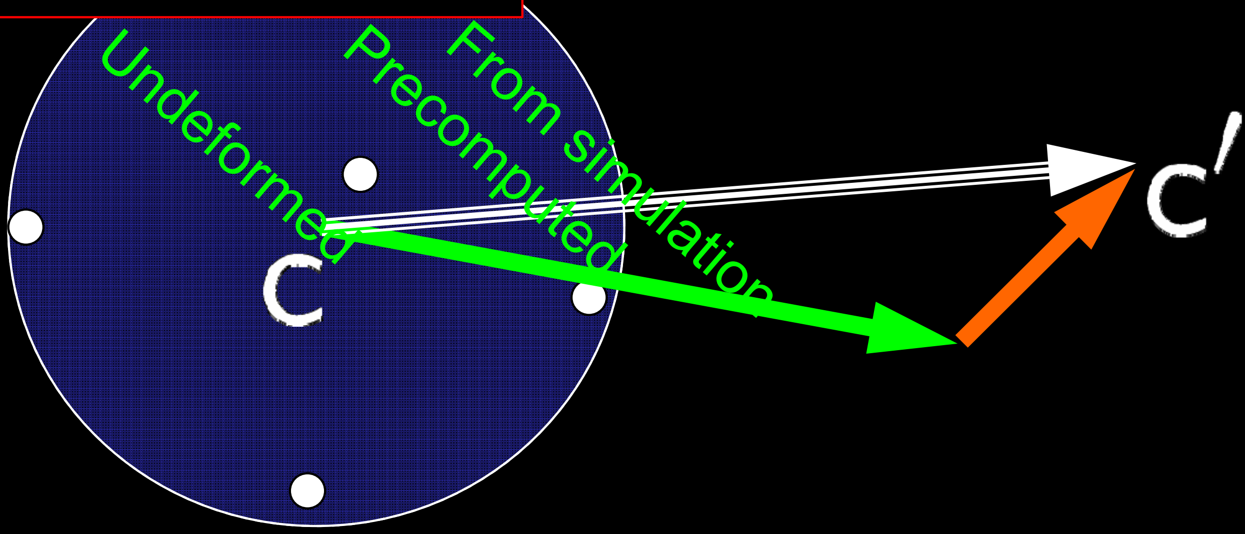
$$c' = c + \bar{U}q$$

$$R' = R + \Delta R \bullet |q|$$

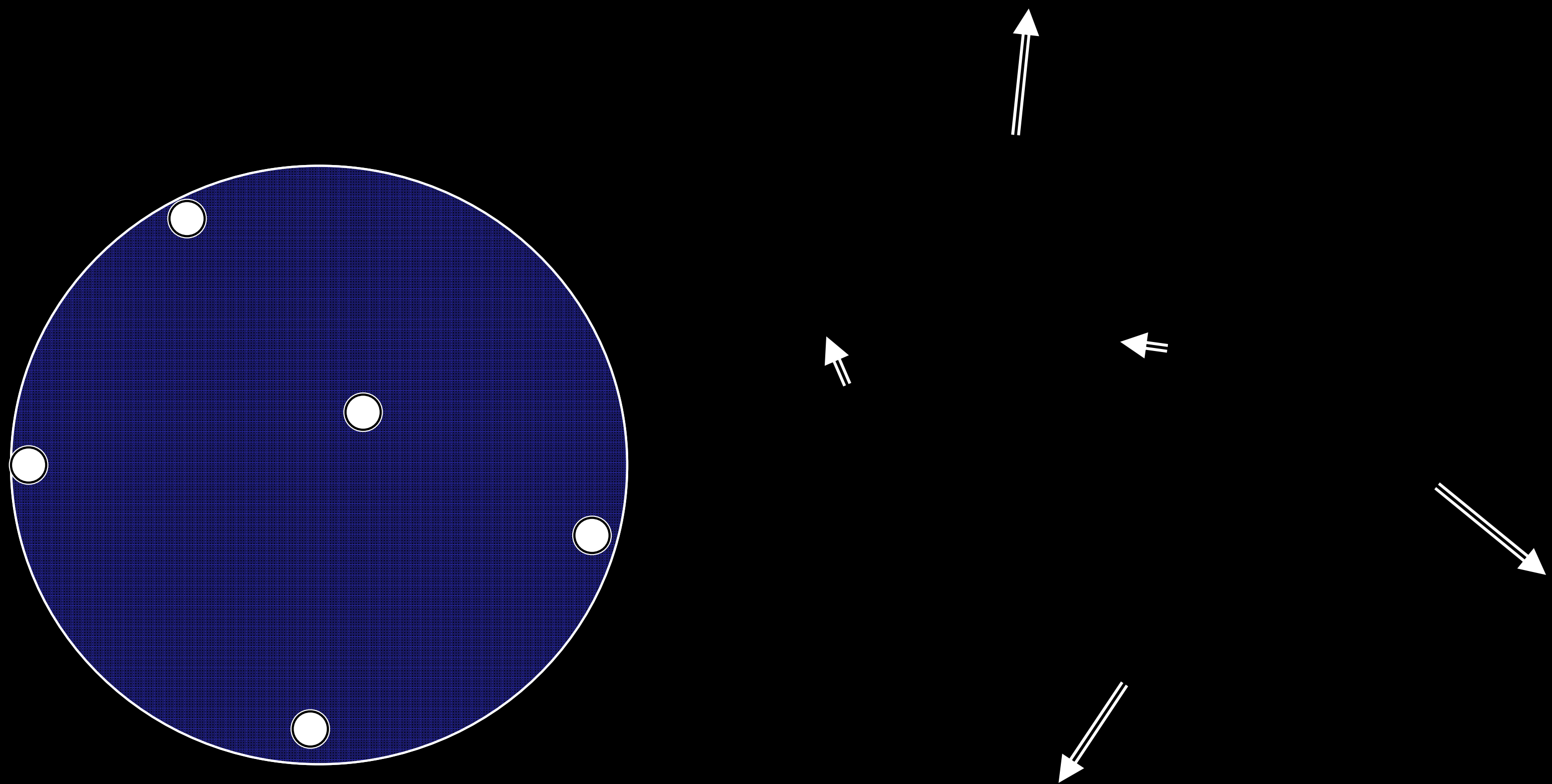
Center Formula, $c'(q)$

- **BD-Tree Approach:**
  Center tracks average displacement

$$c' = c + \bar{U}q$$



Radius Formula, R'(q)



- **BD-Tree Approach:**
  Increase radius by approximation of largest relative vertex movement

Radius Formula, $R'(\mathrm{q})$

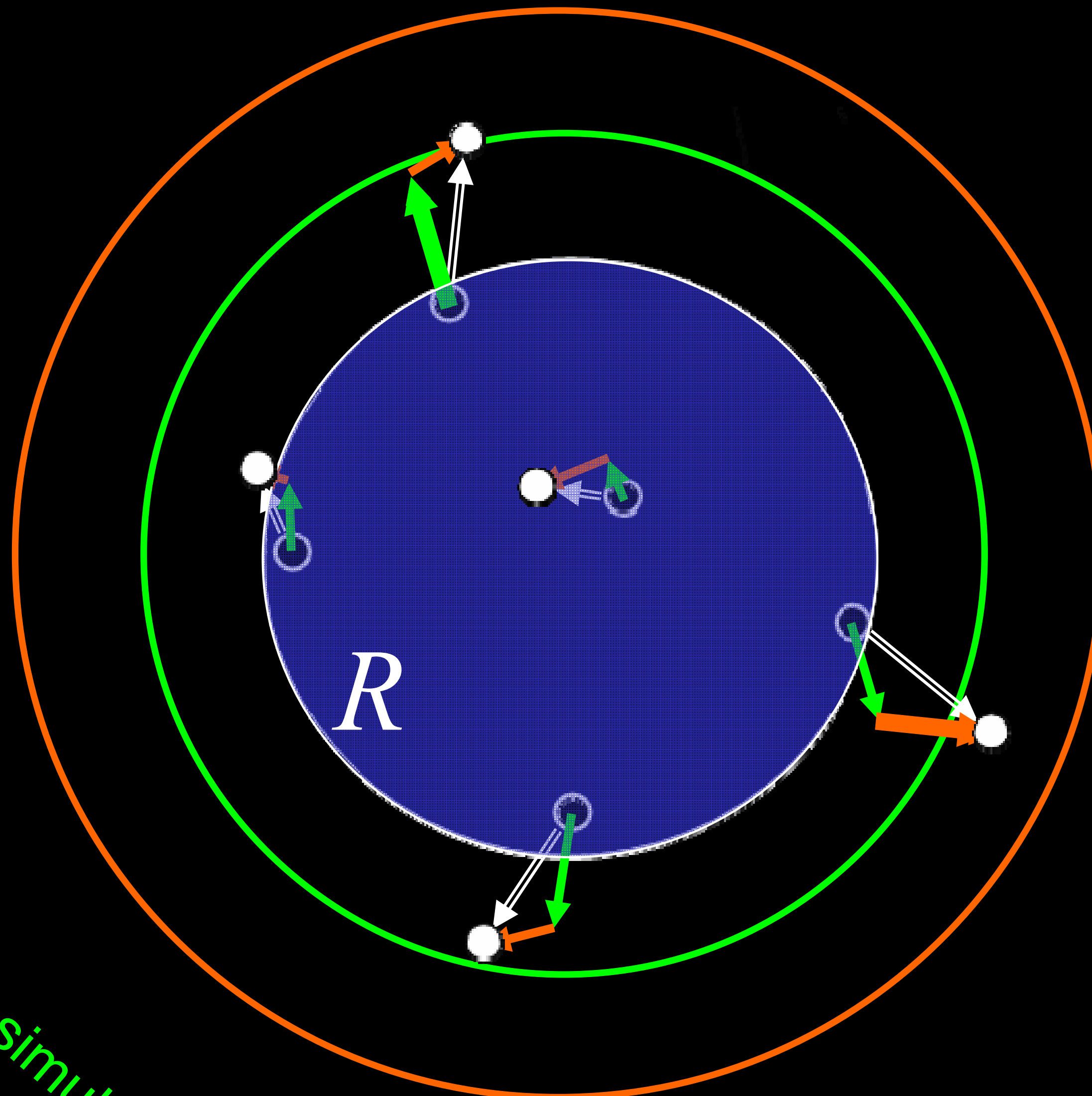
$$R' = R + \Delta R_1 |\mathrm{q}_1| + \Delta R_2 |\mathrm{q}_2|$$

$$R' = R + \Delta R \cdot |\mathrm{q}|$$

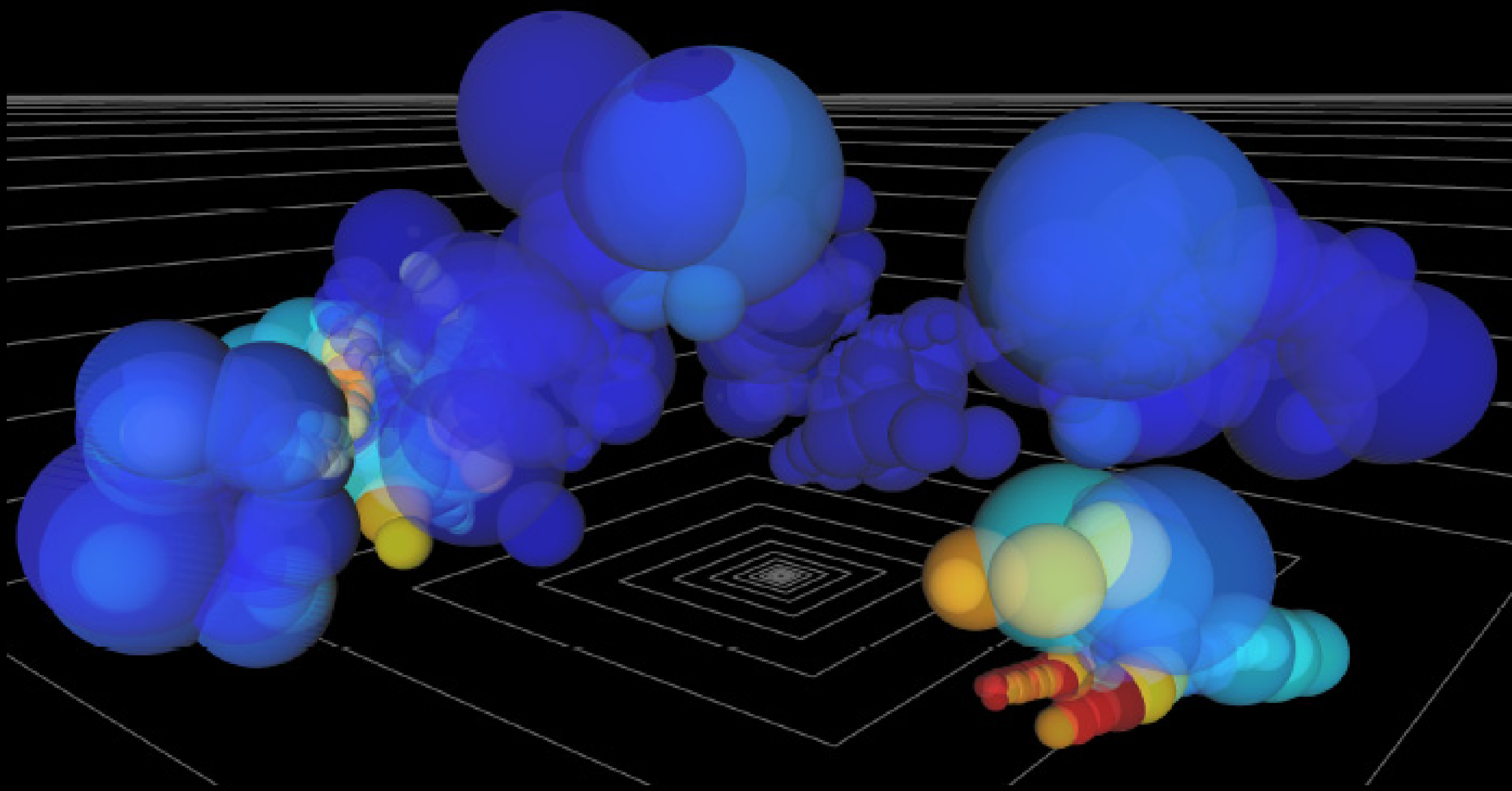
Undeformed

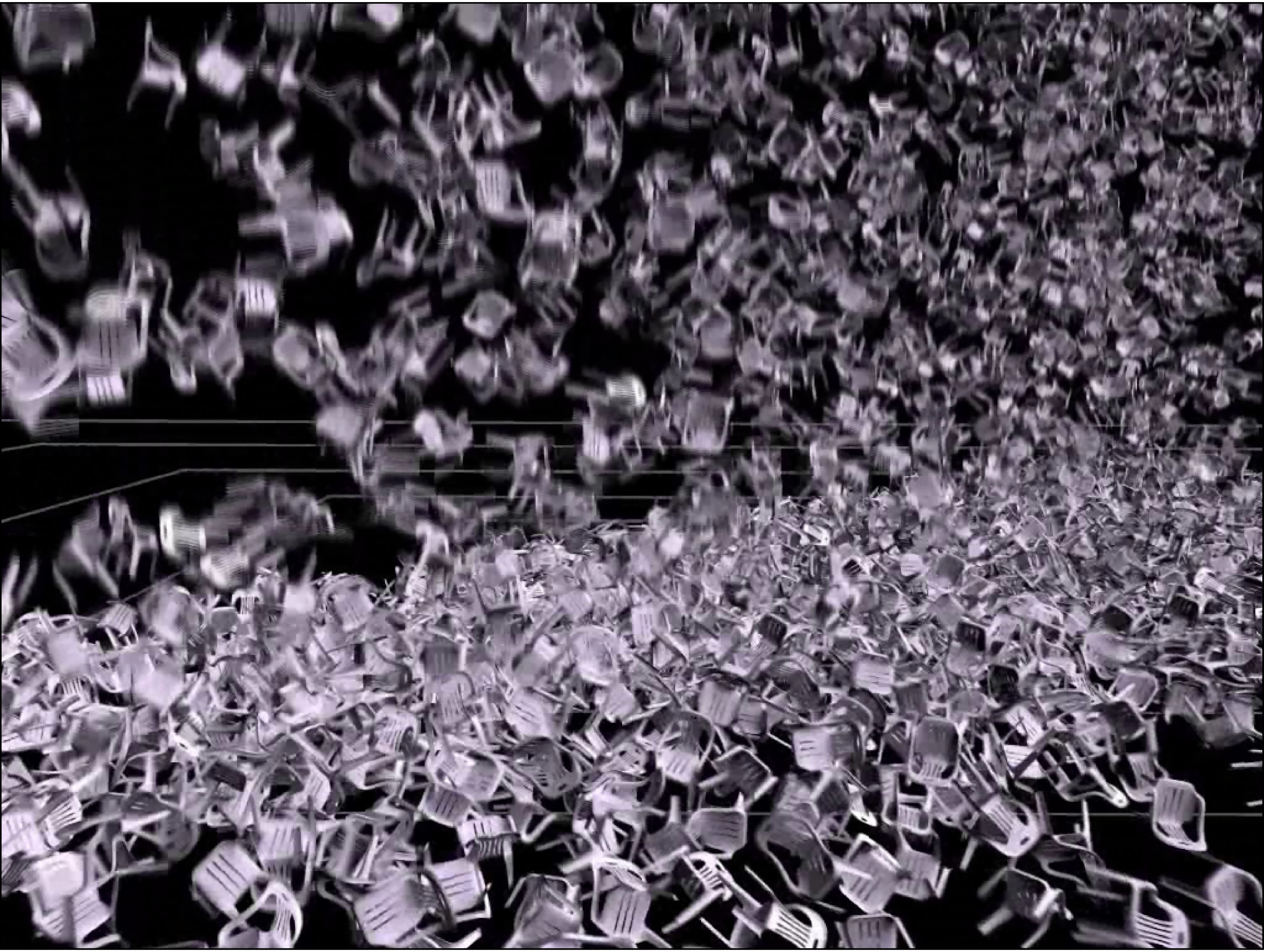
Precomputed

From simulation

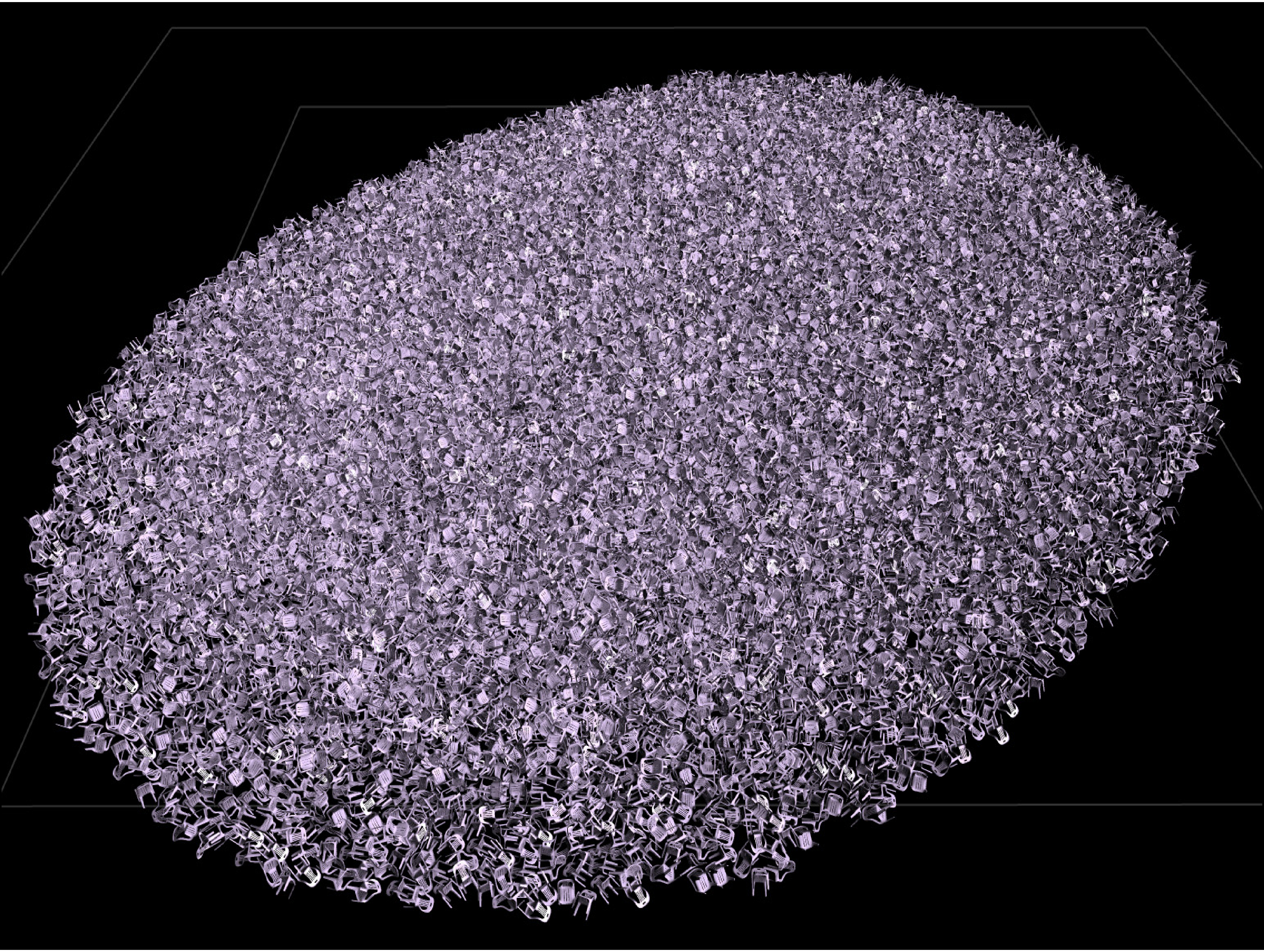


Output-sensitive evaluation





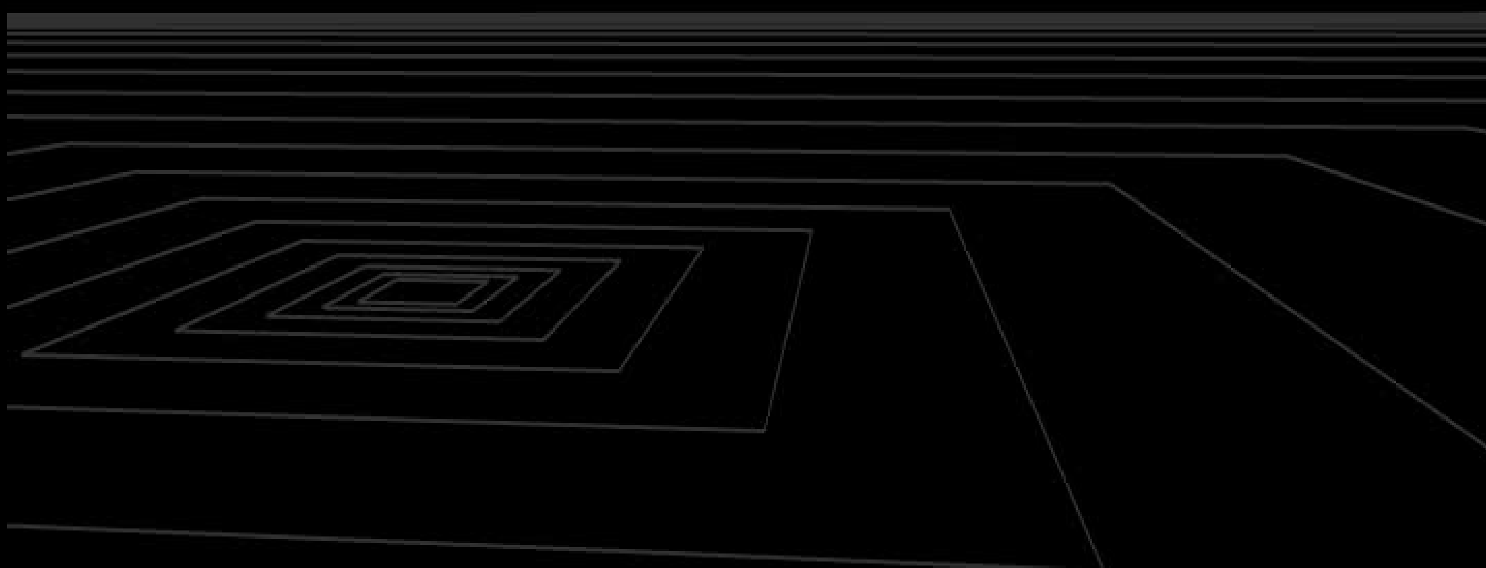
C:\Documents and Settings\djames\Desktop\videos\niagara2_1p1min_gam15_divx.avi



Altogether now…

Multibody Dynamics

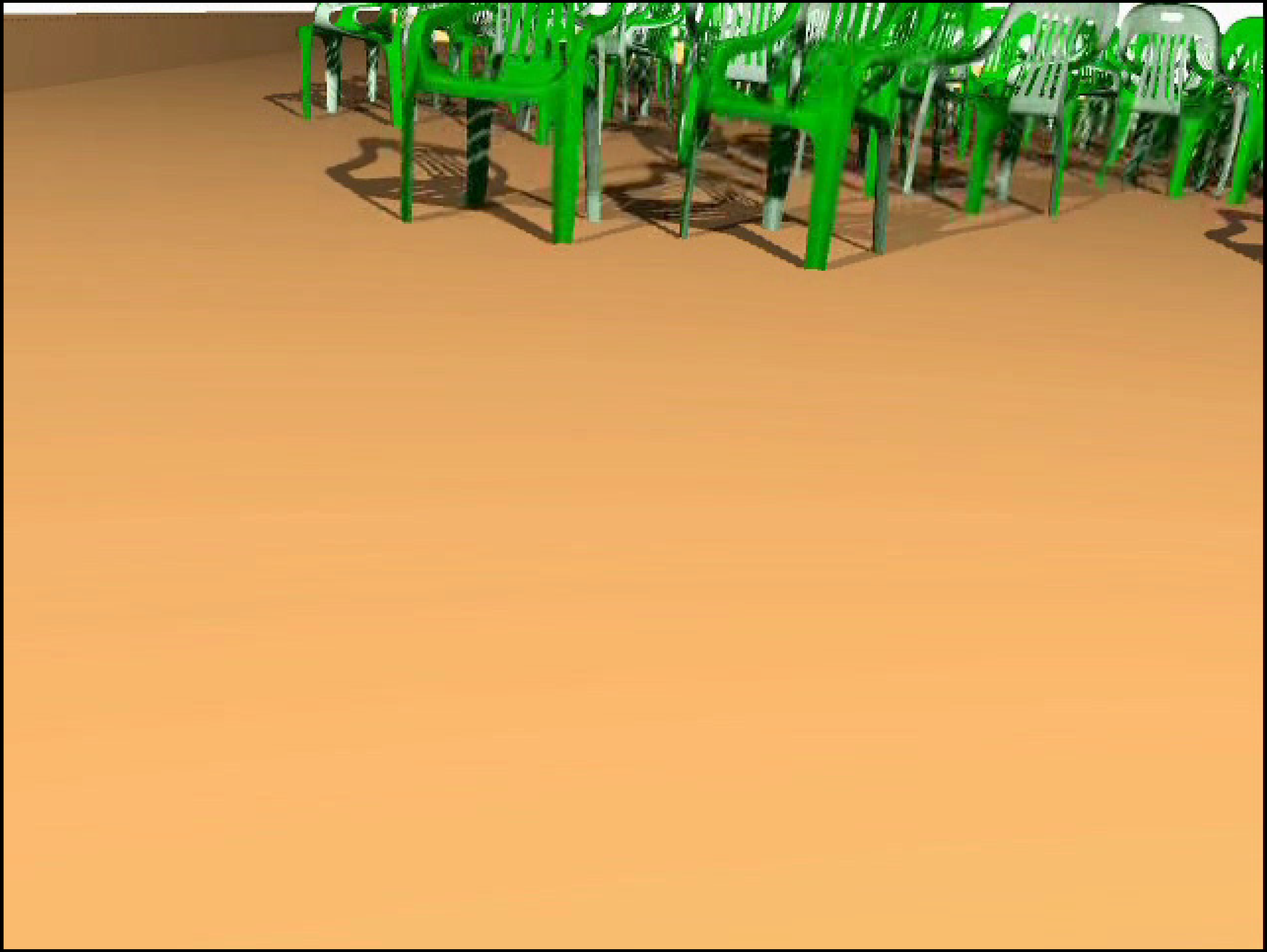
Multibody Dynamics



CURRENT WORK

Reduced Frictional Contact

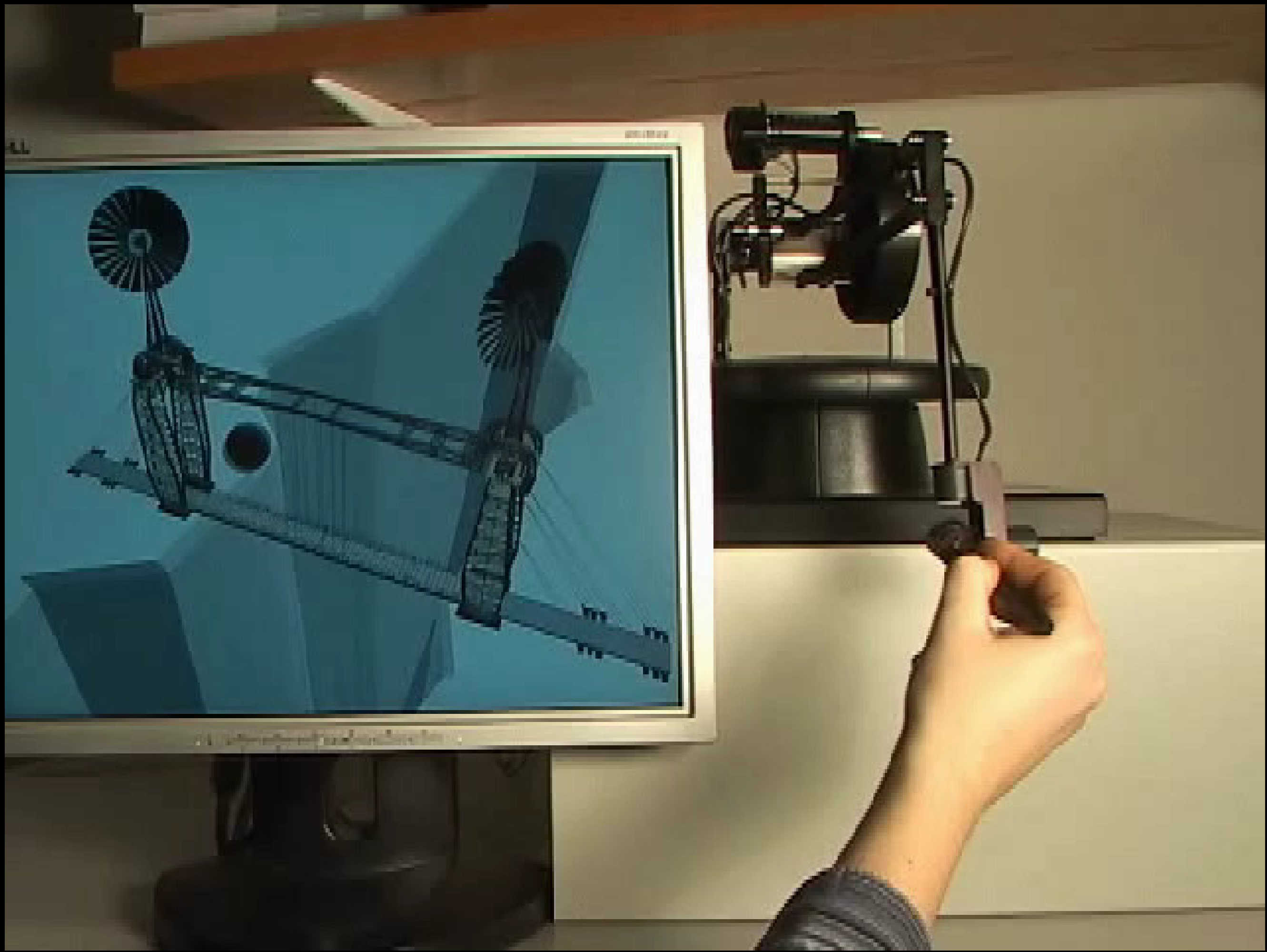
[Kaufman et al. 2008] (to appear at SIGGRAPH Asia 2008)



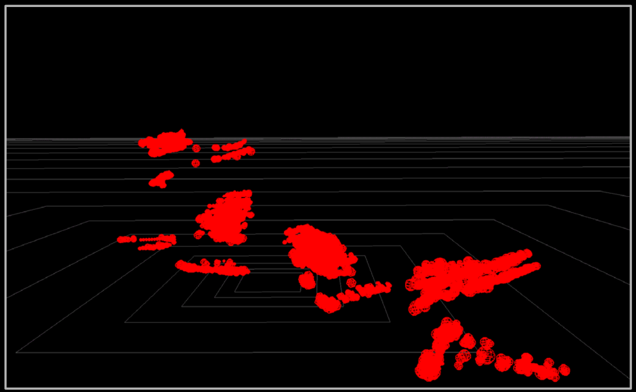
Class Overview

1. Reduced-order deformable models
2. **Haptic force-feedback rendering**
3. Sound synthesis
4. Interactive motion design
    - Many-Worlds Browsing

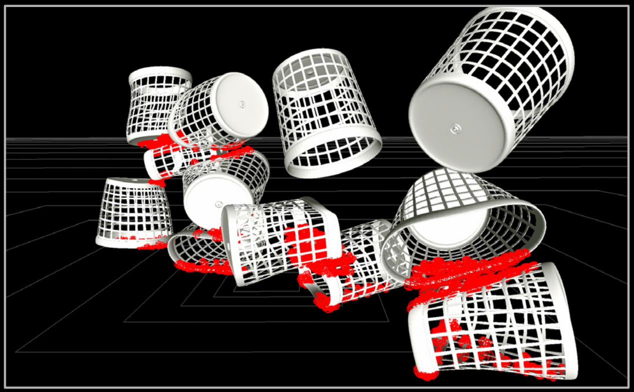
Haptic Rendering



Bottlenecks: Close proximity and conforming contact



Output-sensitive Processing and Communication



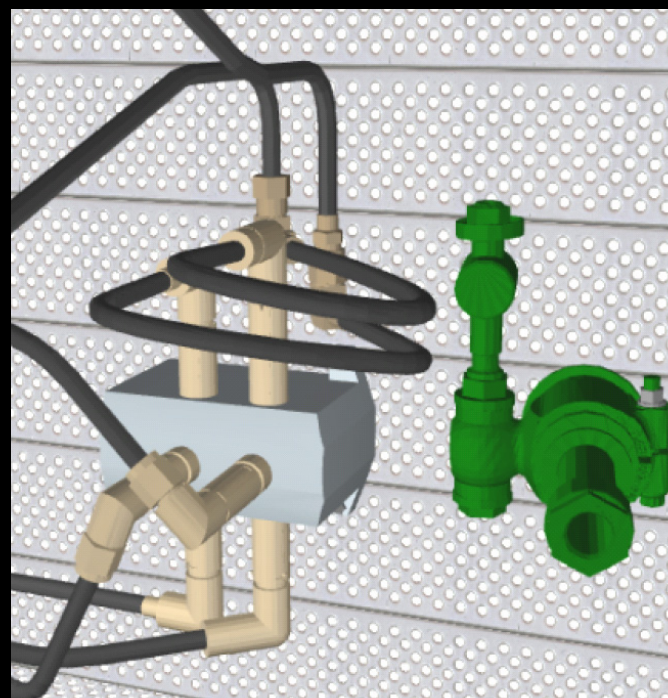
Synthesis

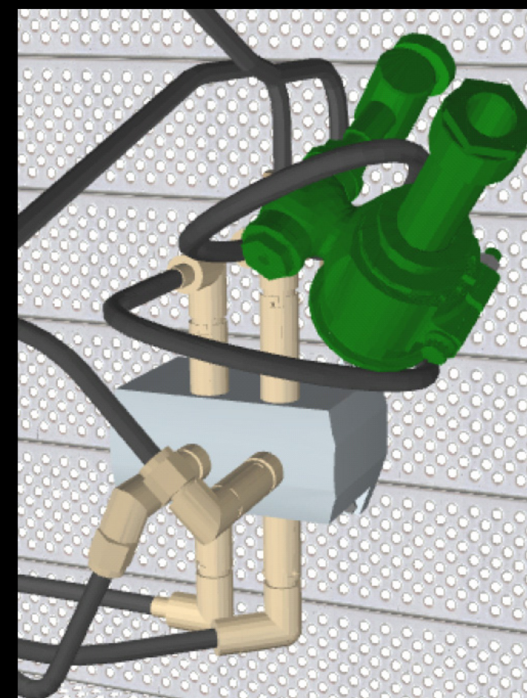
Time-critical multi-point deformable contact

… with Jernej Barbič (CMU□MIT)

Boeing Application: Interactive Assembly Planning

Haptics





CAD geometry and rigid green mechanical component

Four-level hierarchical pointshell

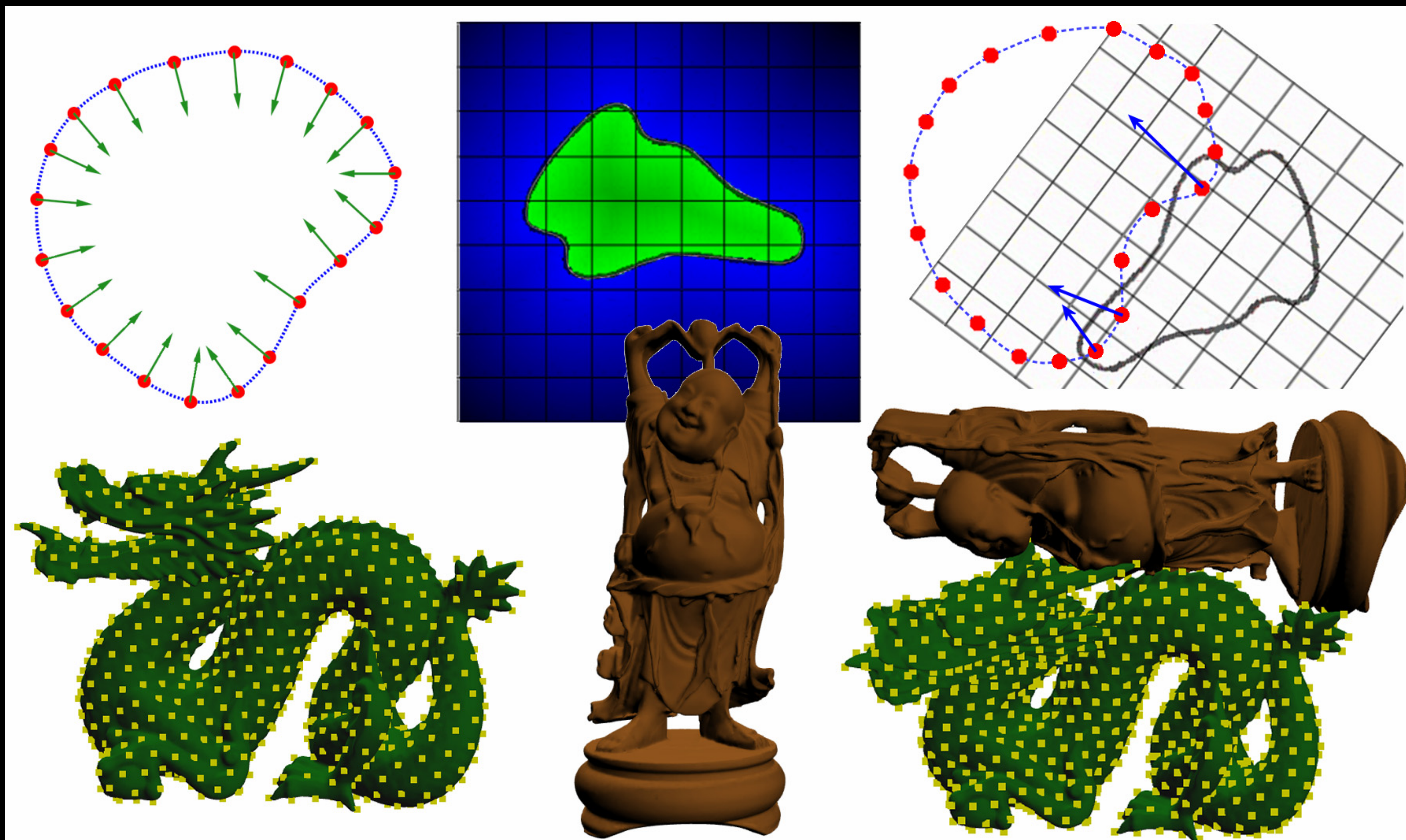
81920 points

15-dim deformation basis

Distance field

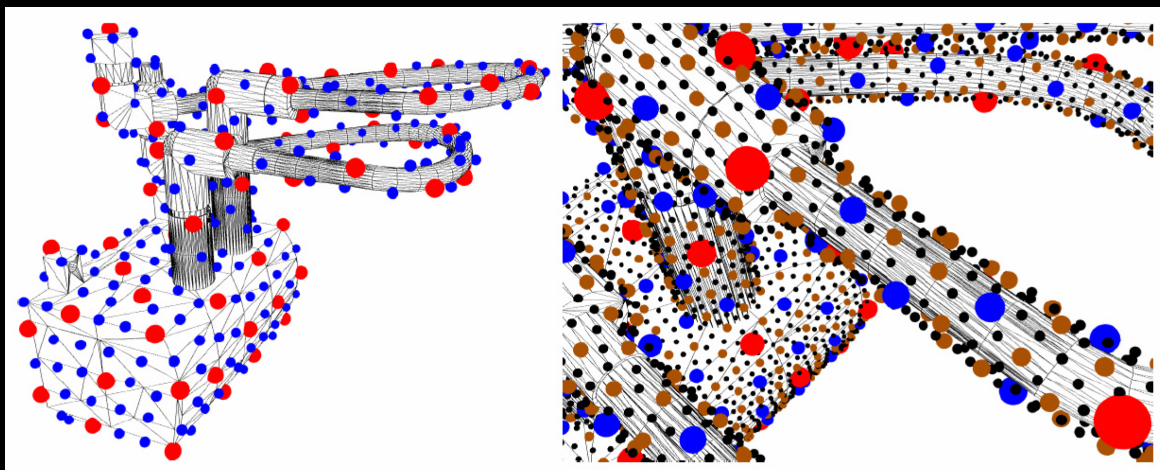
128x128x128

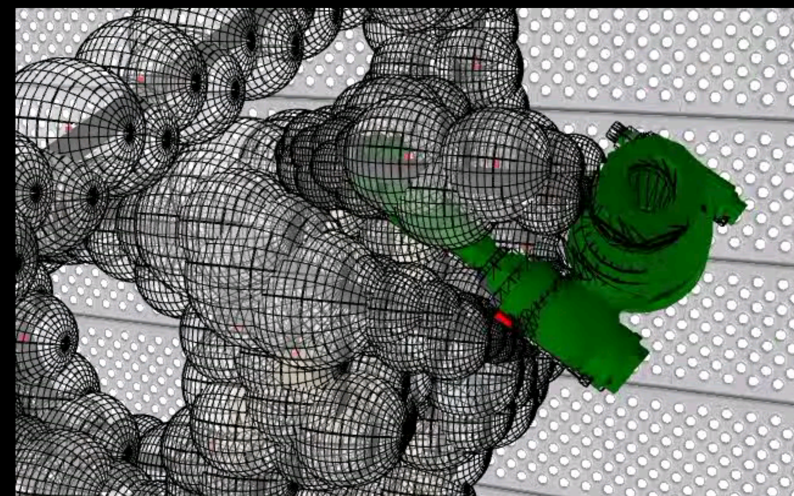
Multi-point penalty contact

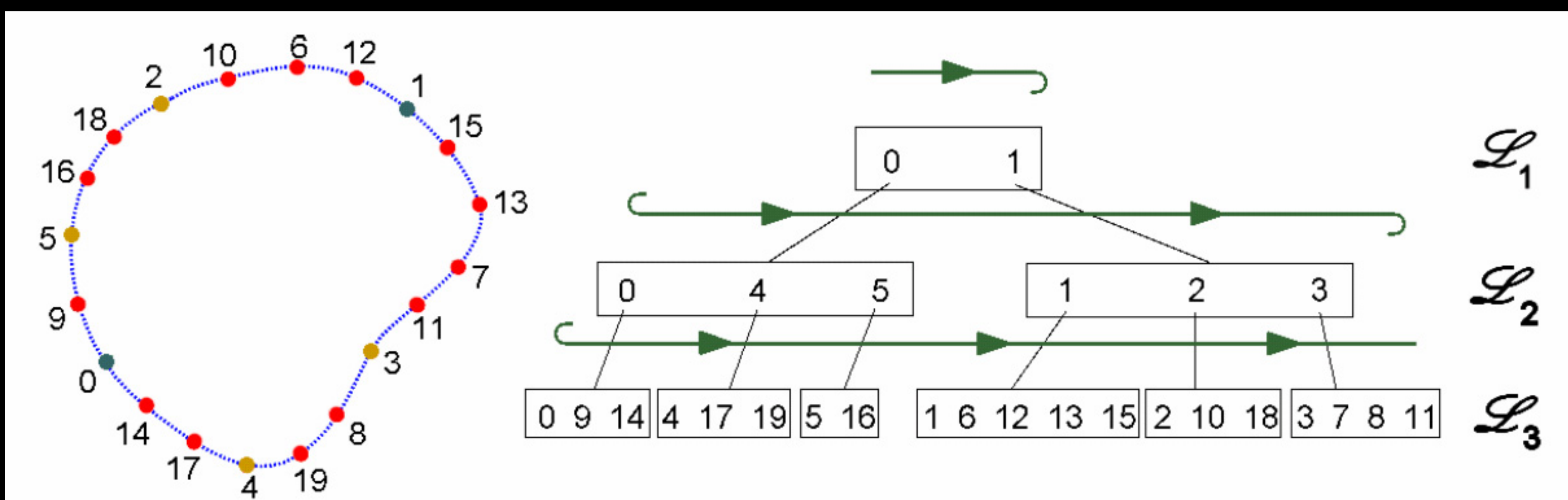


Pointshell Points

Pointshell BD-Tree







Graceful Degradation

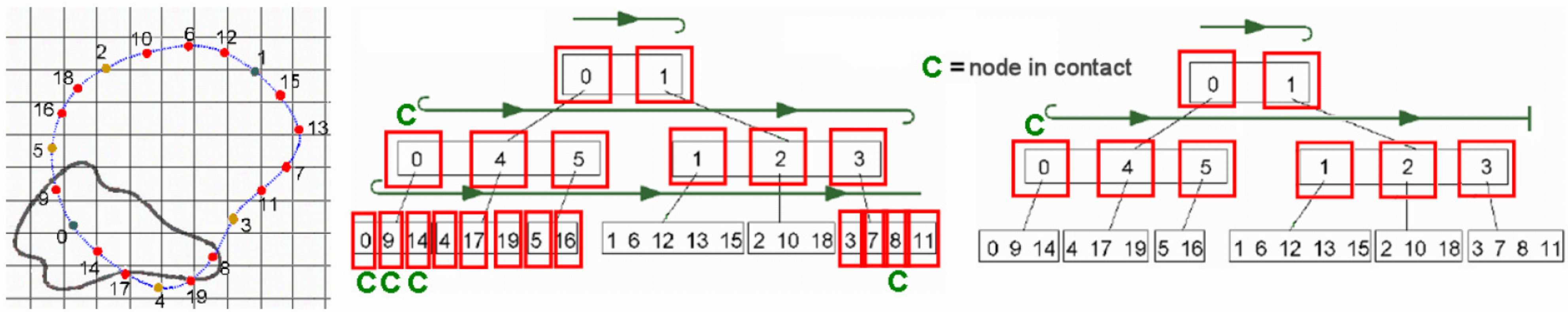
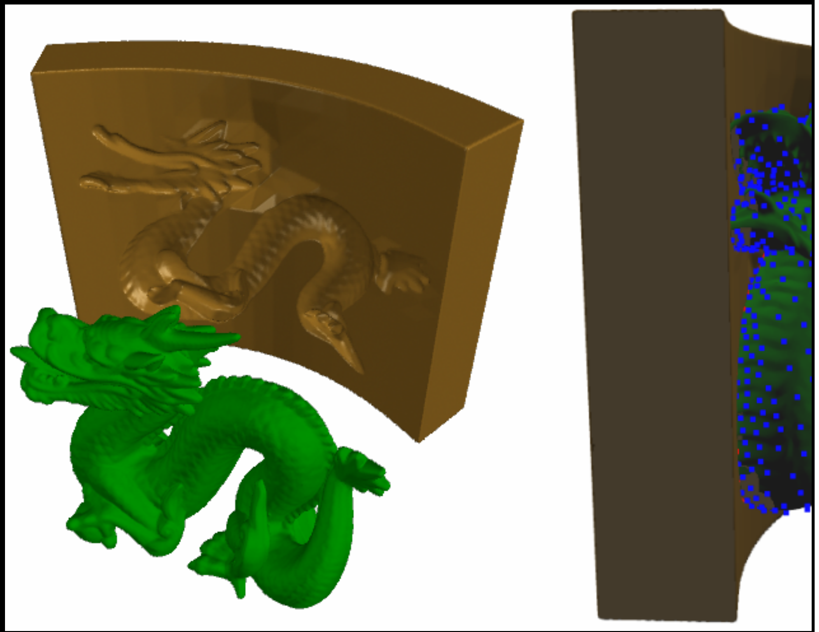
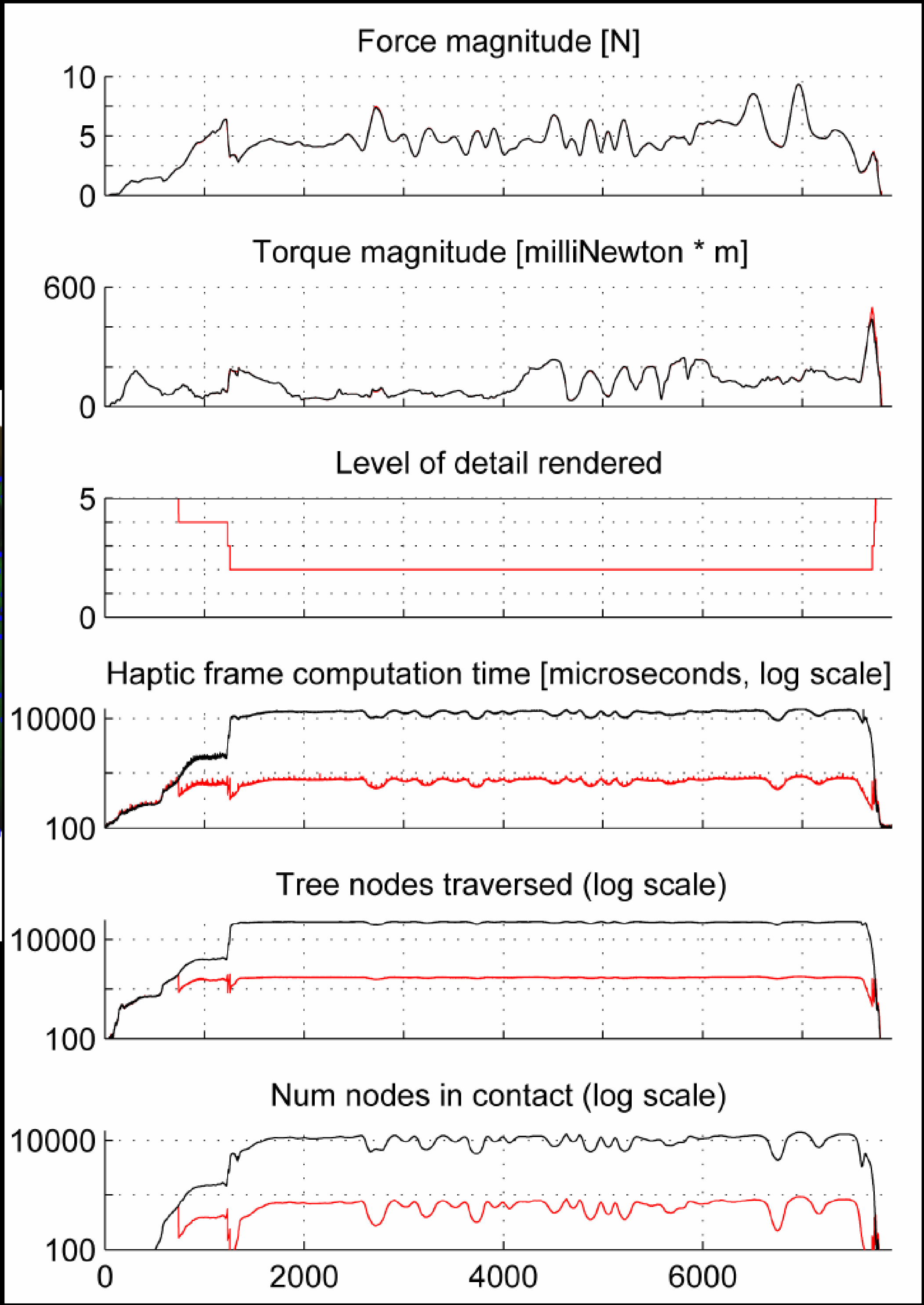


Figure 7: **Graceful degradation:** *With involved contact scenarios (Left), many tree nodes need to be traversed during a complete hierarchy traversal (Middle). Under graceful degradation (Right), tree is traversed only up to a shallow depth, as permitted by available CPU power. Squares denote traversed nodes.*

- **Exploit *temporal coherence*!**

Graceful Degradation





Rigid—vs—Rigid

Rigid knight and rigid axe

Six-level hierarchical pointshell

1,024,000 points

Distance field

256x256x256

Deformable—vs—Rigid

Deformable bridge and rigid dinosaur

Four-level hierarchical pointshell

85018 points

15-dim deformation basis

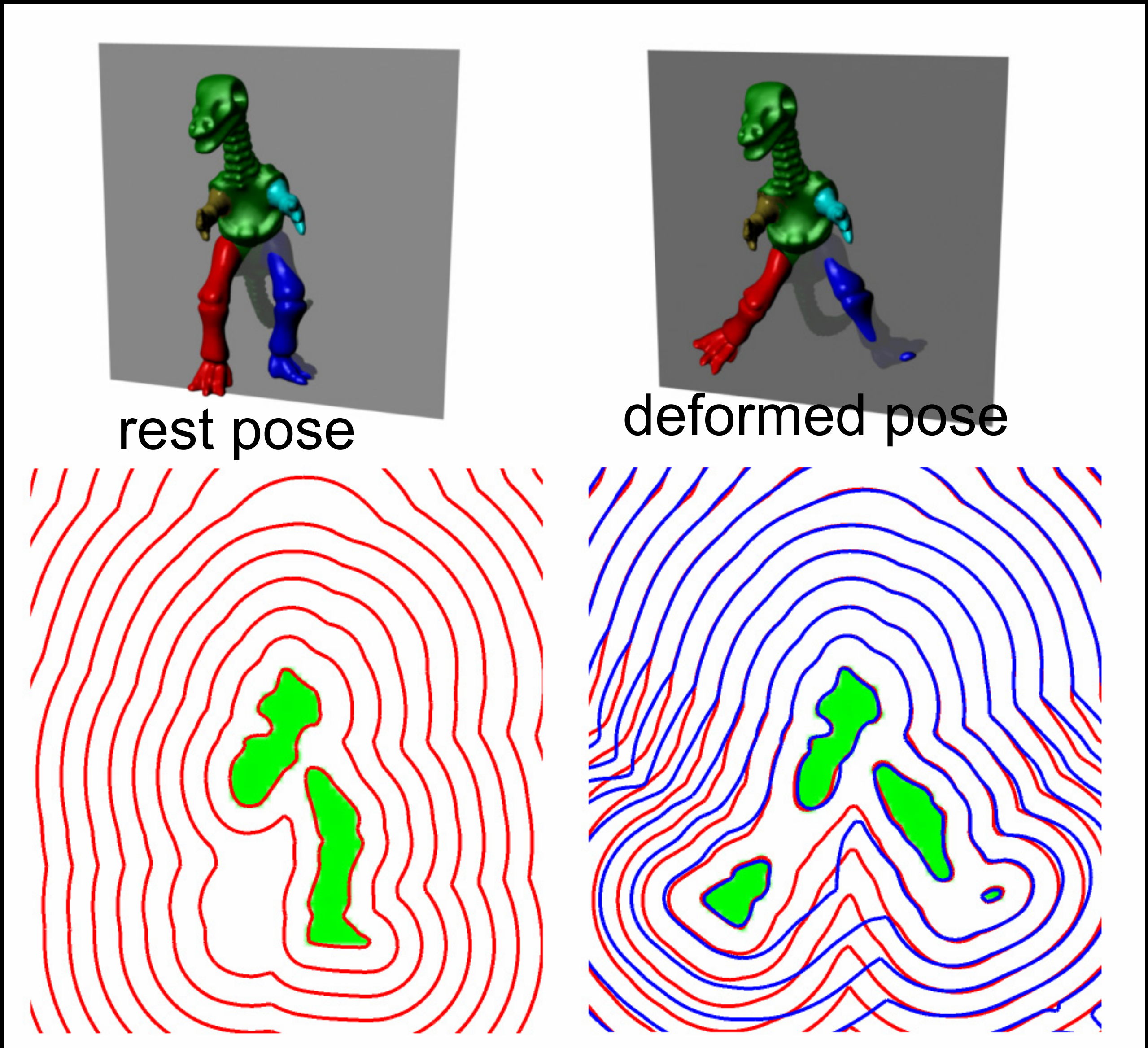
Distance field

128x128x128

Deformable—vs—Deformable

?!

Deformed Distance Fields



precomputed rest distance field isocontours

deformed distance field isocontours

Red = exact (offline)
Blue = approximate (haptic rates)

Deformable CAD geometry and deformable hose

Five-level hierarchical pointshell

256,000 points

20-dim deformation basis

Deformable distance field

256x256x256

1 domain, 20 proxies total

20-dim deformation basis

[redacted]

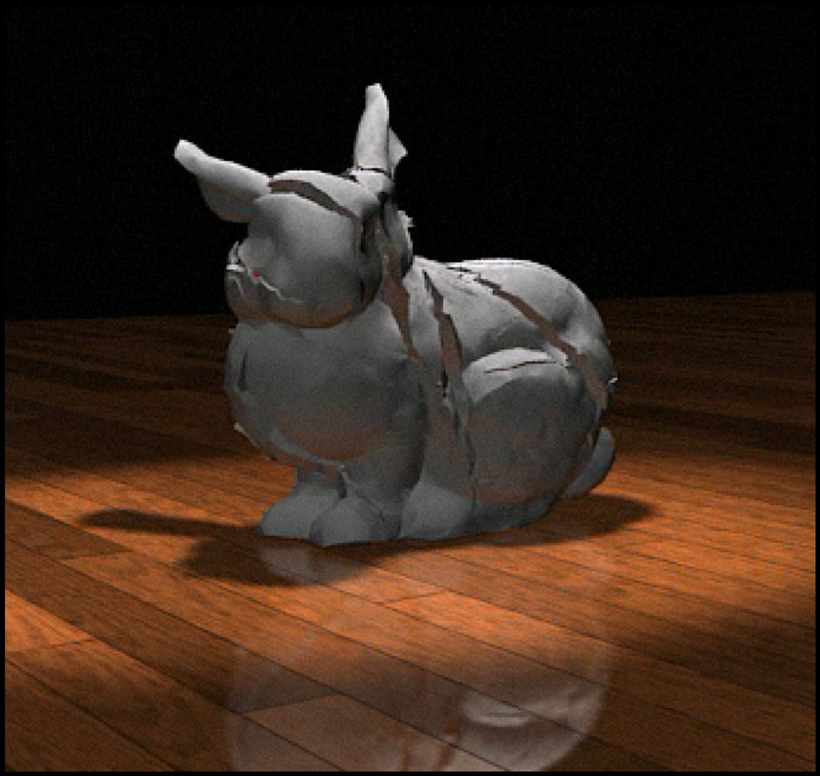
Class Overview

1. Reduced-order deformable models
2. Haptic force-feedback rendering
3. **Sound synthesis**
4. Interactive motion design
   - Many-Worlds Browsing

Physically Based Sound Rendering

Sound: A grand challenge for computer animation

[Enright et al. 2001]



[O'Brien & Hodgins 1999]

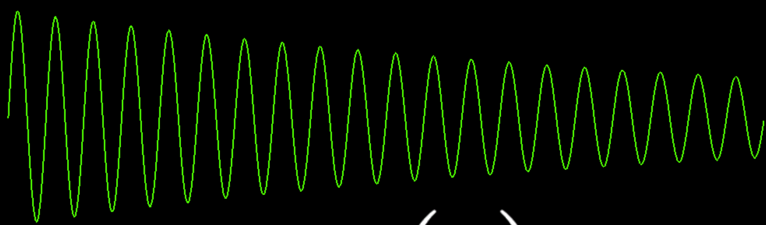
[James & Pai 2004]

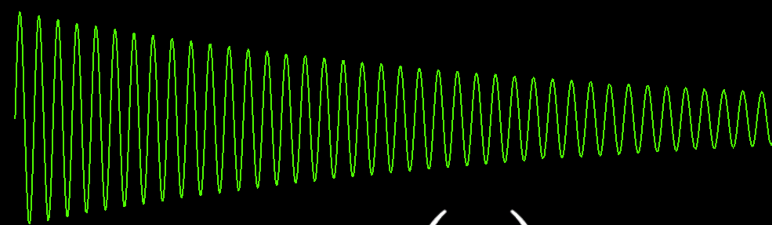
Precomputed Acoustic Transfer
(with Jernej Barbic and Dinesh Pai)

- Real-time sound radiation
  - FEM-based modal vibrations
  - Precompute *acoustic transfer*
- Equivalent multipole source approximations
- Output-sensitive sound synthesis

Modal Analysis (FEM)

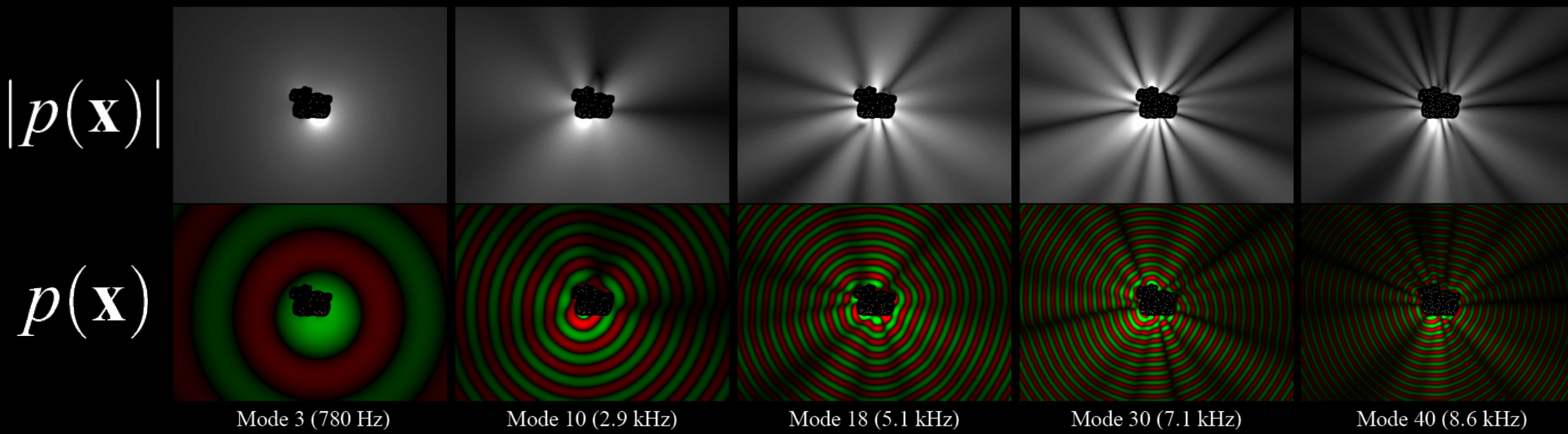
Modal Sound Synthesis

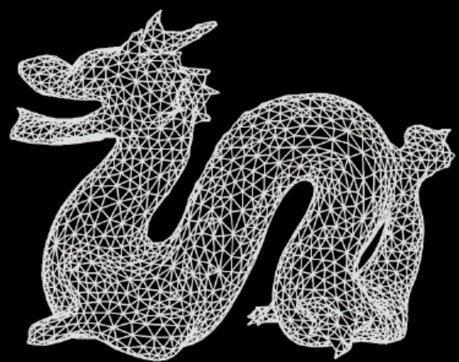




$$\frac{q_1(t) \quad + \quad q_2(t) \quad + \cdots}{r}$$

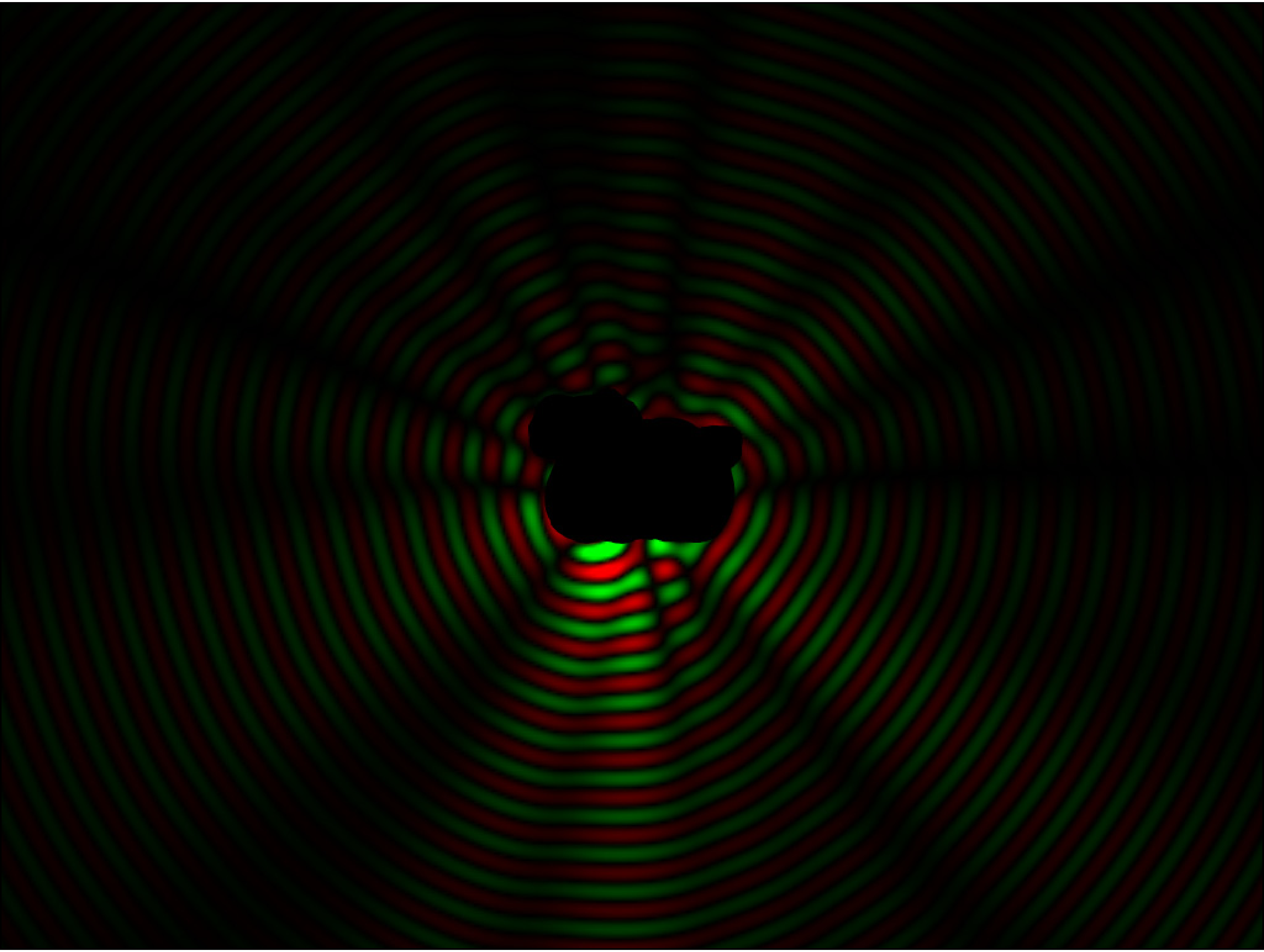
Acoustic Transfer Functions



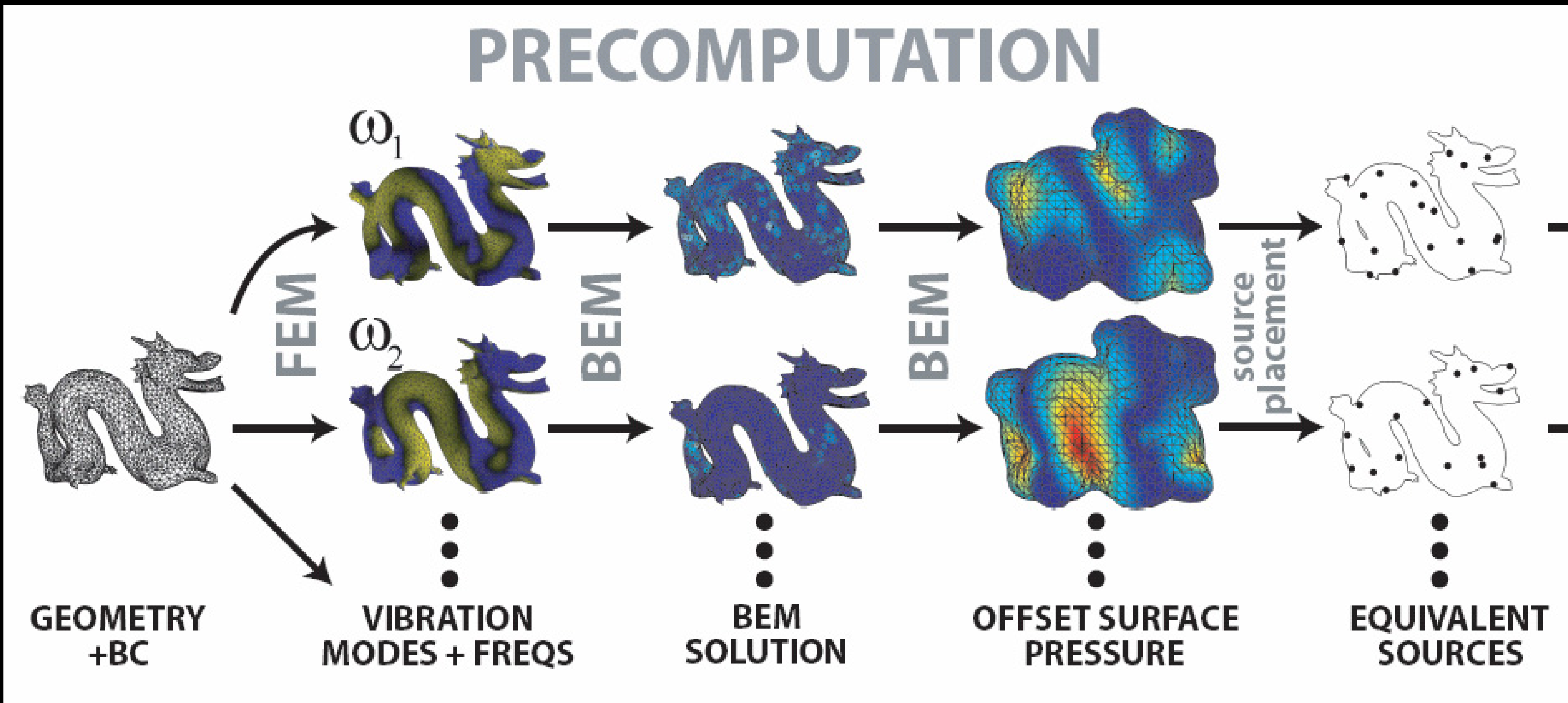


Bronze Dragon (20 cm)

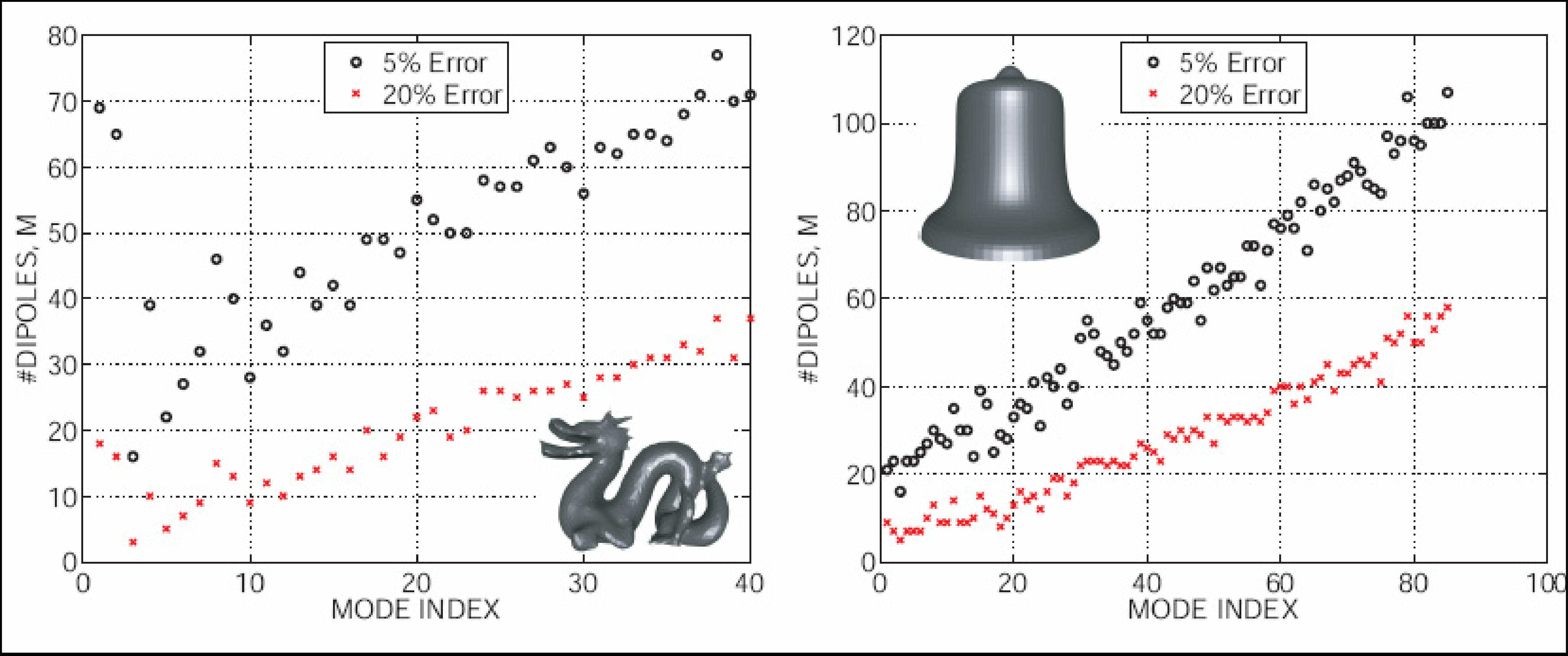

MODE 40



Precomputed Acoustic Transfer



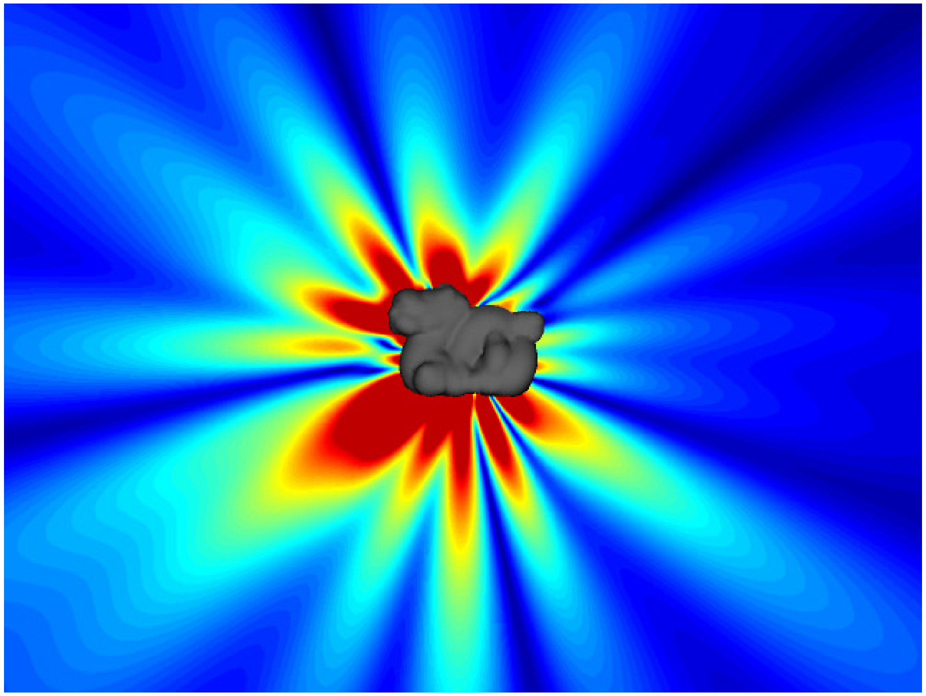
How many dipoles?



Evaluation cost ≈ 0.3 μsec/dipole

BEM vs PAT

Dragon : absolute pressure, 640x480 grid, 307200 samples, mode 30



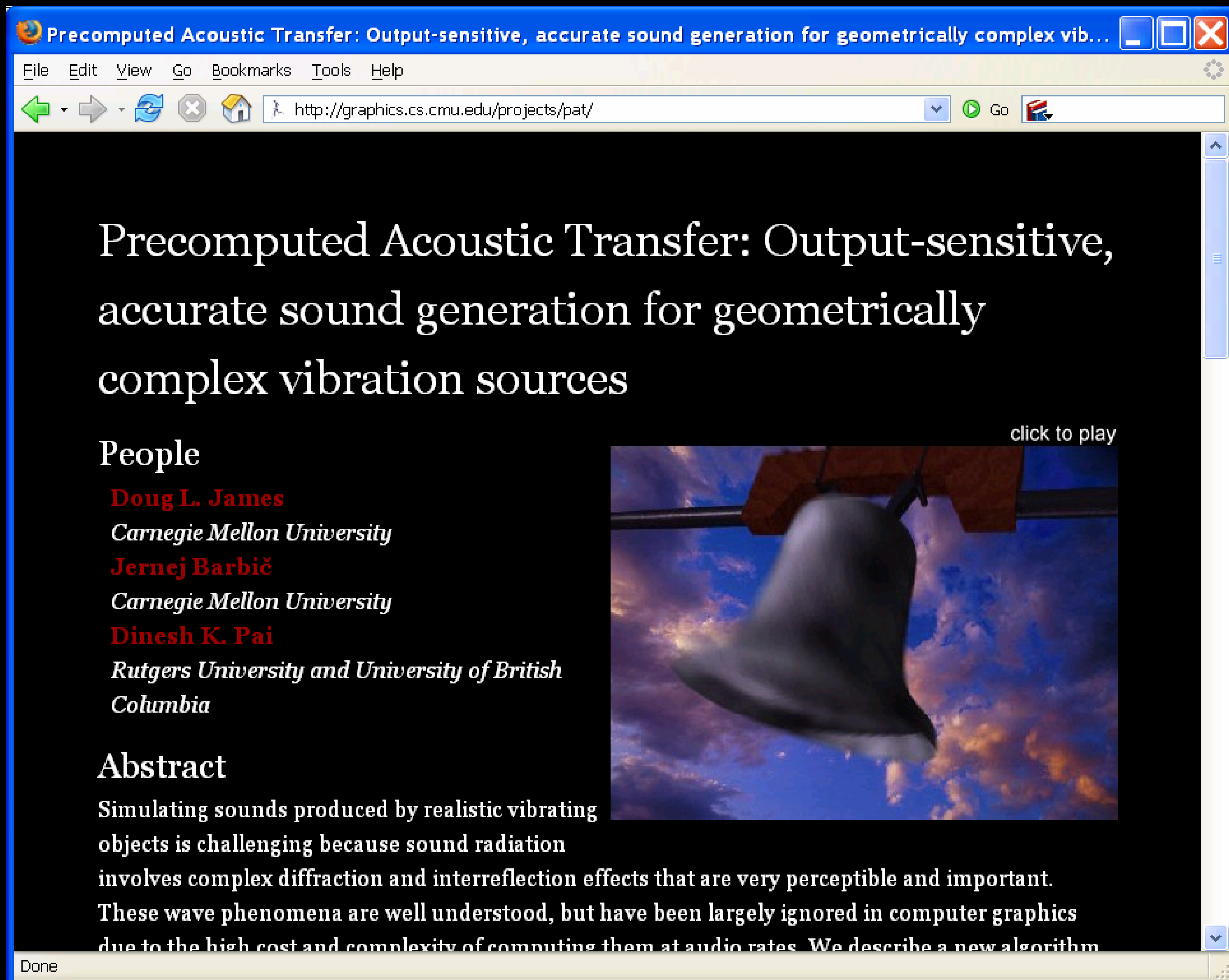
BEM: computation time: 8h 30min

Example: Bell

Bell

PAT renderer

Try the demo!

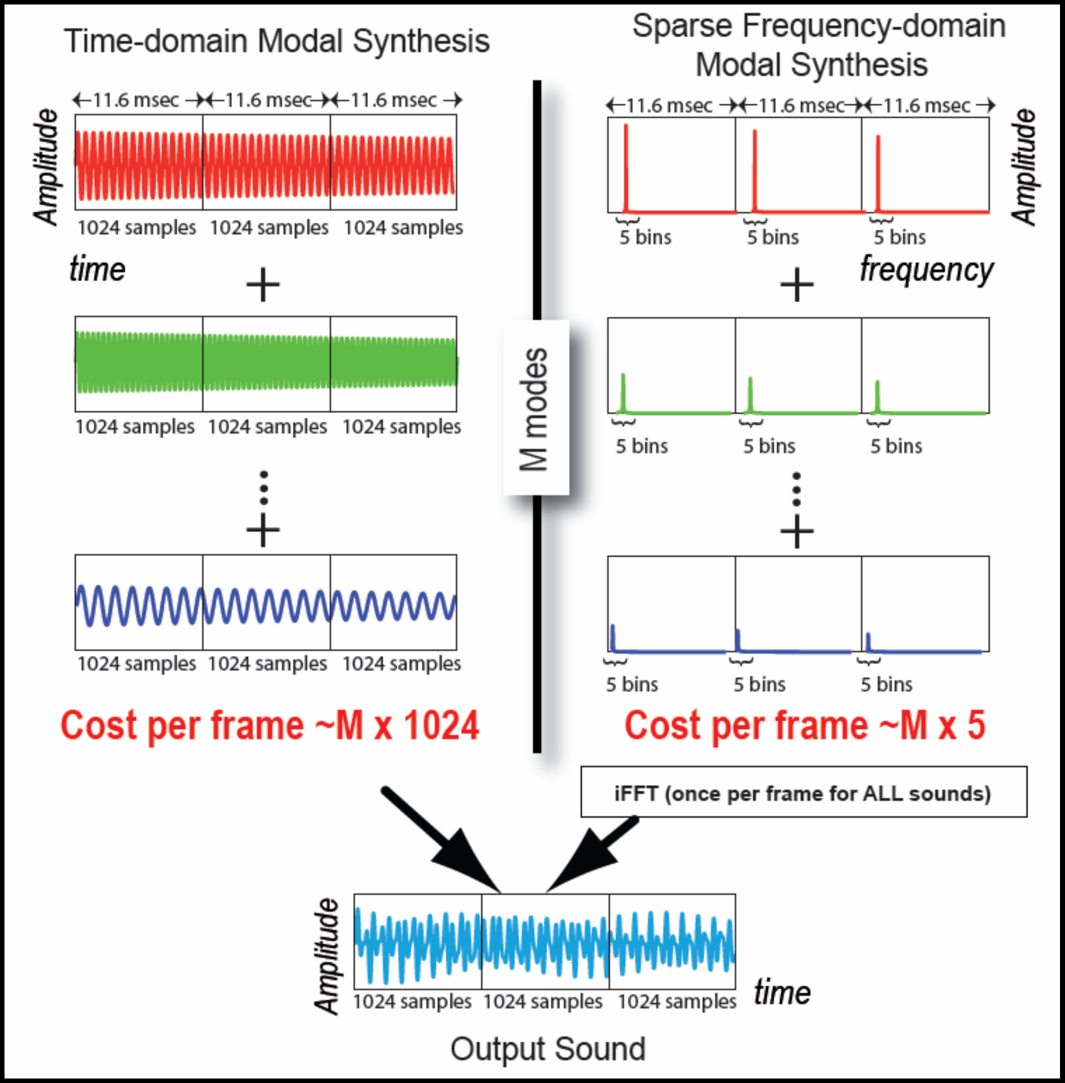


C:\Documents and Settings\djames\Desktop\pat-demo-v1.3

CURRENT WORK

Fast Modal Sounds with Scalable Frequency-Domain Synthesis

[Bonneel et al. 2008] (SIGGRAPH 2008)



[redacted]

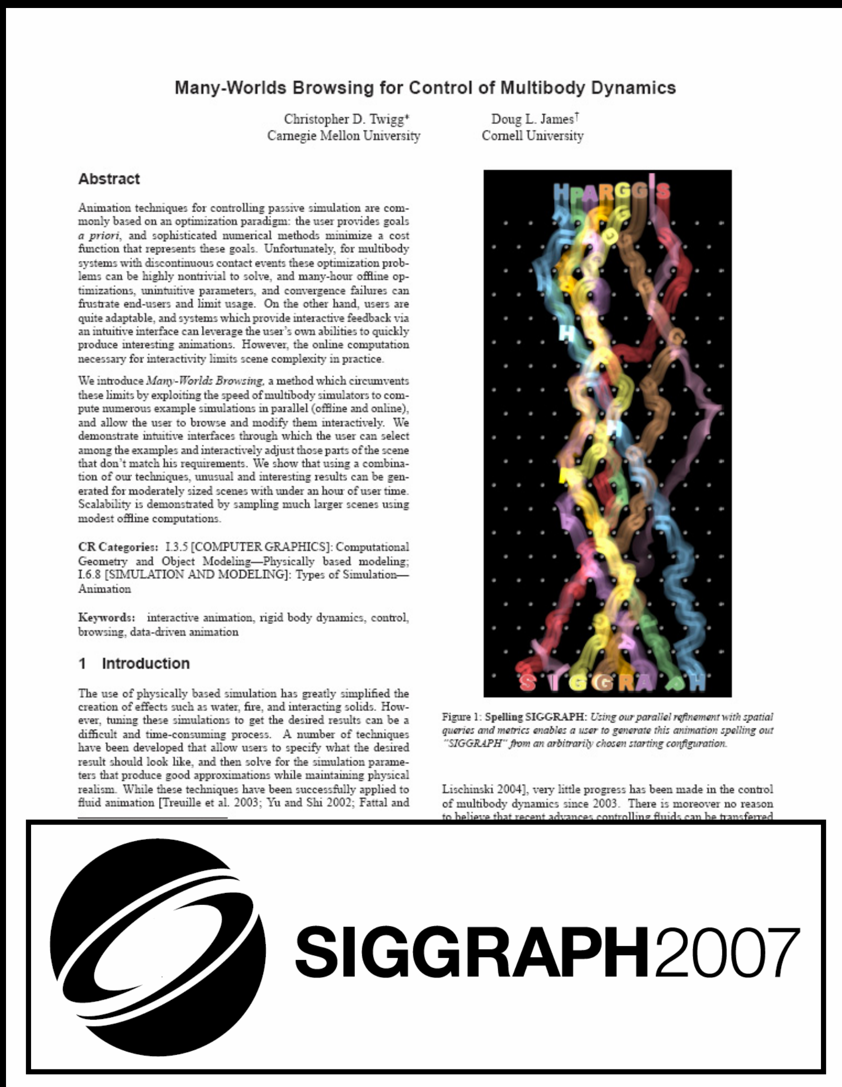
Class Overview

1. Reduced-order deformable models
2. Haptic force-feedback rendering
3. Sound synthesis
4. **Interactive motion design**
    - **Many-Worlds Browsing**

Many-Worlds Browsing

Controlling physics via interactive search

Christopher Twigg


Many-Worlds Browsing for Control of Multibody Dynamics

Christopher D. Twigg* Carnegie Mellon University
Doug L. James† Cornell University

Abstract

Animation techniques for controlling passive simulation are commonly based on an optimization paradigm: the user provides goals *a priori*, and sophisticated numerical methods minimize a cost function that represents these goals. Unfortunately, for multibody systems with discontinuous contact events these optimization problems can be highly nontrivial to solve, and many-hour offline optimizations, unintuitive parameters, and convergence failures can frustrate end-users and limit usage. On the other hand, users are quite adaptable, and systems which provide interactive feedback via an intuitive interface can leverage the user's own abilities to quickly produce interesting animations. However, the online computation necessary for interactivity limits scene complexity in practice.

We introduce *Many-Worlds Browsing*, a method which circumvents these limits by exploiting the speed of multibody simulators to compute numerous example simulations in parallel (offline and online), and allow the user to browse and modify them interactively. We demonstrate intuitive interfaces through which the user can select among the examples and interactively adjust those parts of the scene that don't match his requirements. We show that using a combination of our techniques, unusual and interesting results can be generated for moderately sized scenes with under an hour of user time. Scalability is demonstrated by sampling much larger scenes using modest offline computations.

CR Categories: I.3.5 [COMPUTER GRAPHICS]: Computational Geometry and Object Modeling—Physically based modeling; I.6.8 [SIMULATION AND MODELING]: Types of Simulation—Animation

Keywords: interactive animation, rigid body dynamics, control, browsing, data-driven animation

1 Introduction

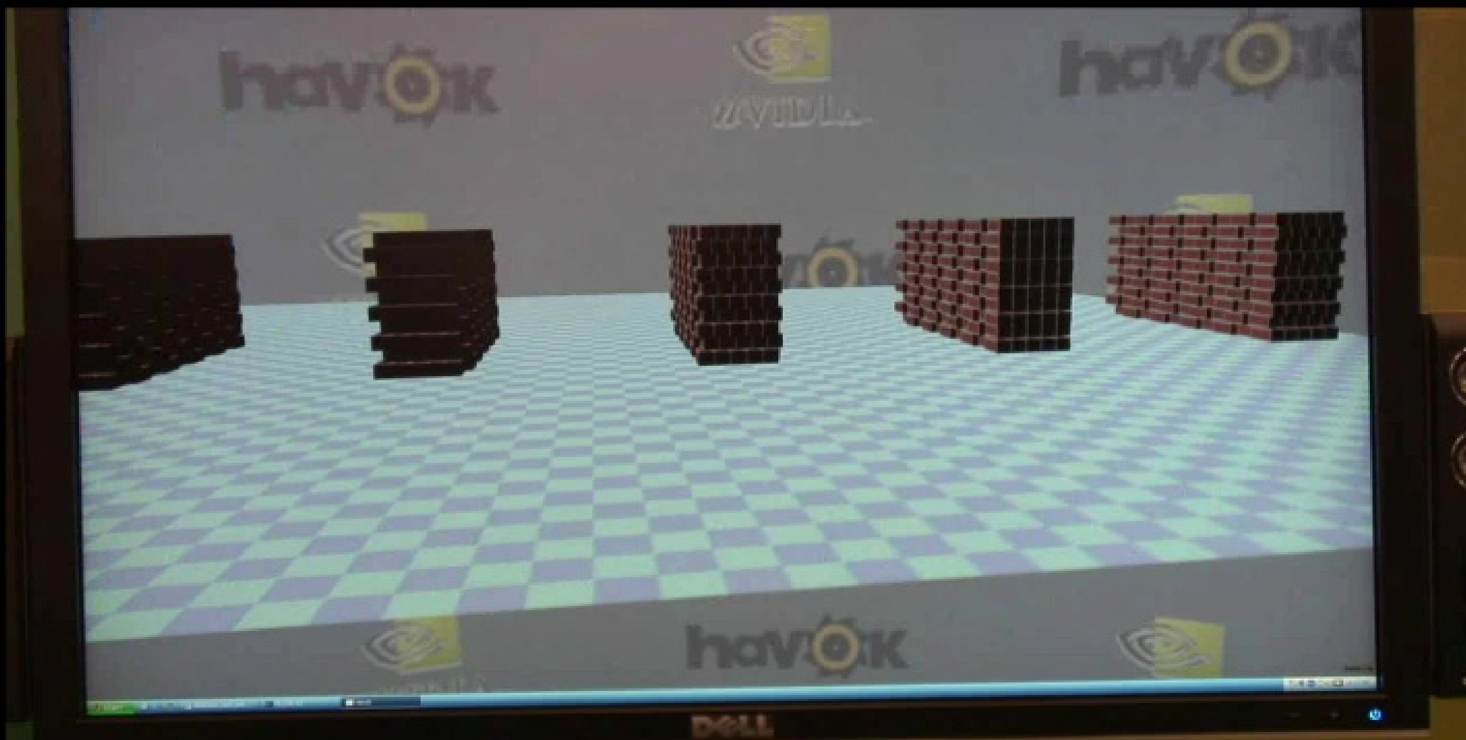
The use of physically based simulation has greatly simplified the creation of effects such as water, fire, and interacting solids. However, tuning these simulations to get the desired results can be a difficult and time-consuming process. A number of techniques have been developed that allow users to specify what the desired result should look like, and then solve for the simulation parameters that produce good approximations while maintaining physical realism. While these techniques have been successfully applied to fluid animation [Treuille et al. 2003; Yu and Shi 2002; Fattal and Lischinski 2004], very little progress has been made in the control of multibody dynamics since 2003. There is moreover no reason to believe that recent advances controlling fluids can be transferred

Figure 1: **Spelling SIGGRAPH:** *Using our parallel refinement with spatial queries and metrics enables a user to generate this animation spelling out "SIGGRAPH" from an arbitrarily chosen starting configuration.*

SIGGRAPH2007

Many-Worlds Browsing
is Chris's Ph.D. thesis work

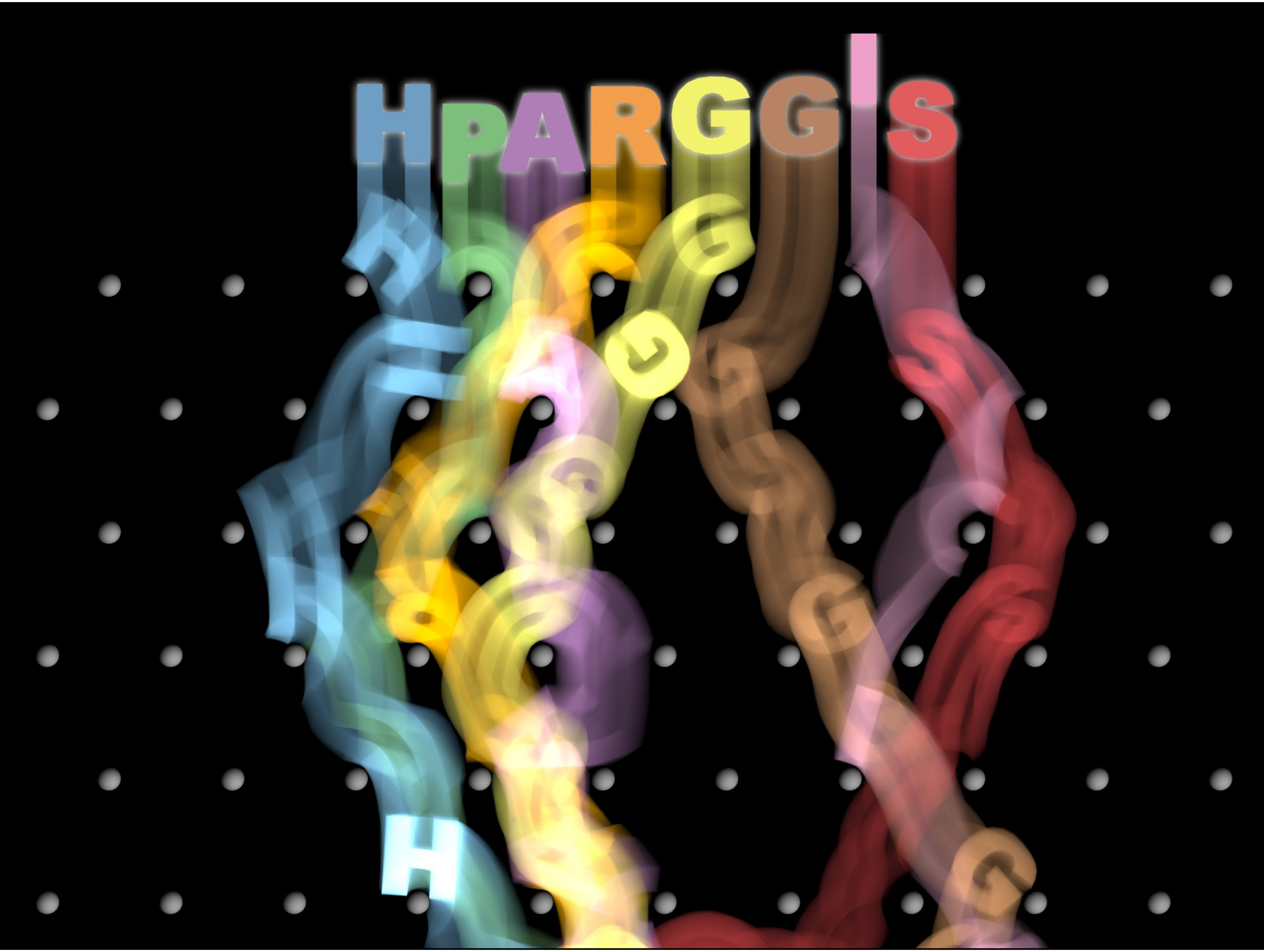
Rigid Body Simulation is Fast



Havok/NVIDIA Demo

Control!

Pixar


HPARGGIS

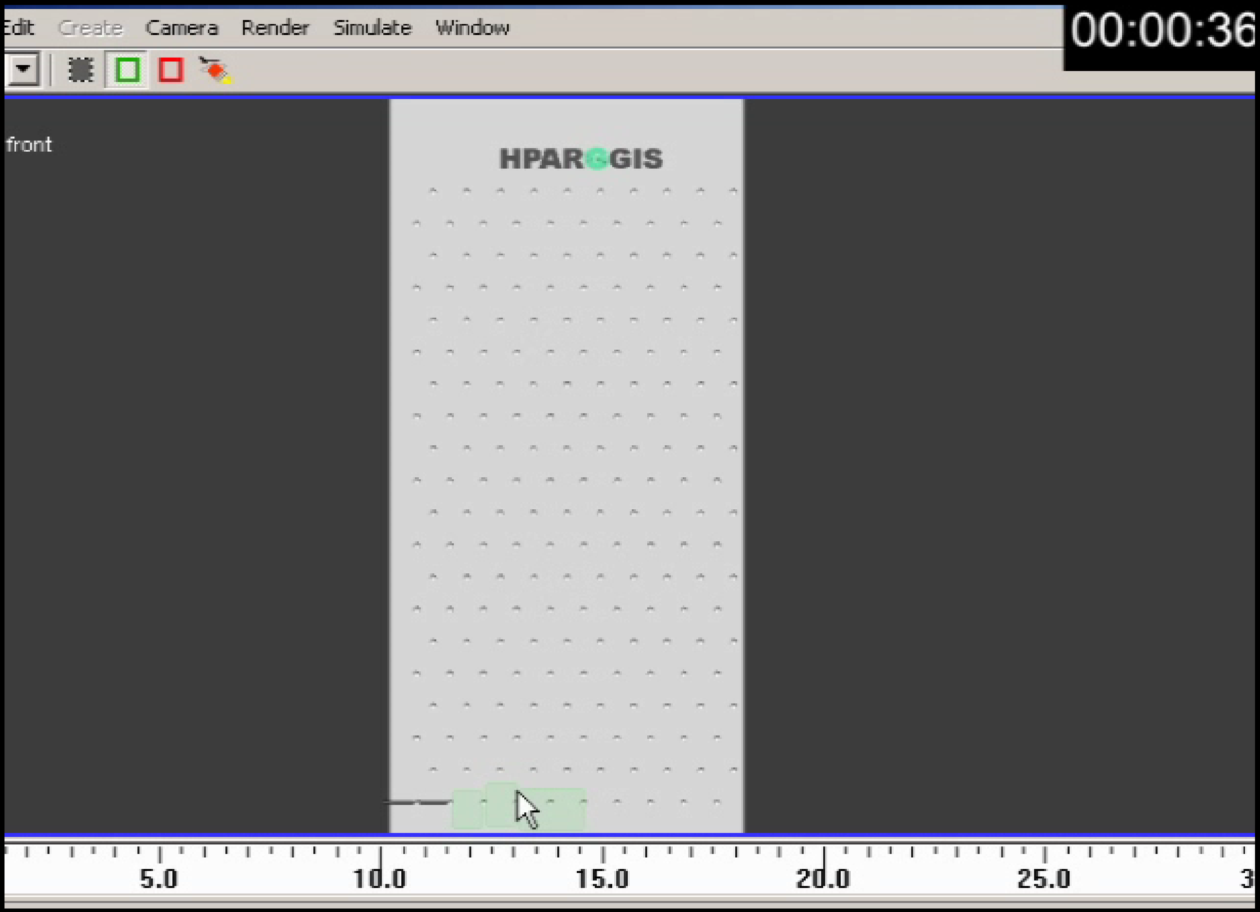
Google™

Google Search | I'm Feeling Lucky

Advanced Search
Preferences
Language Tools

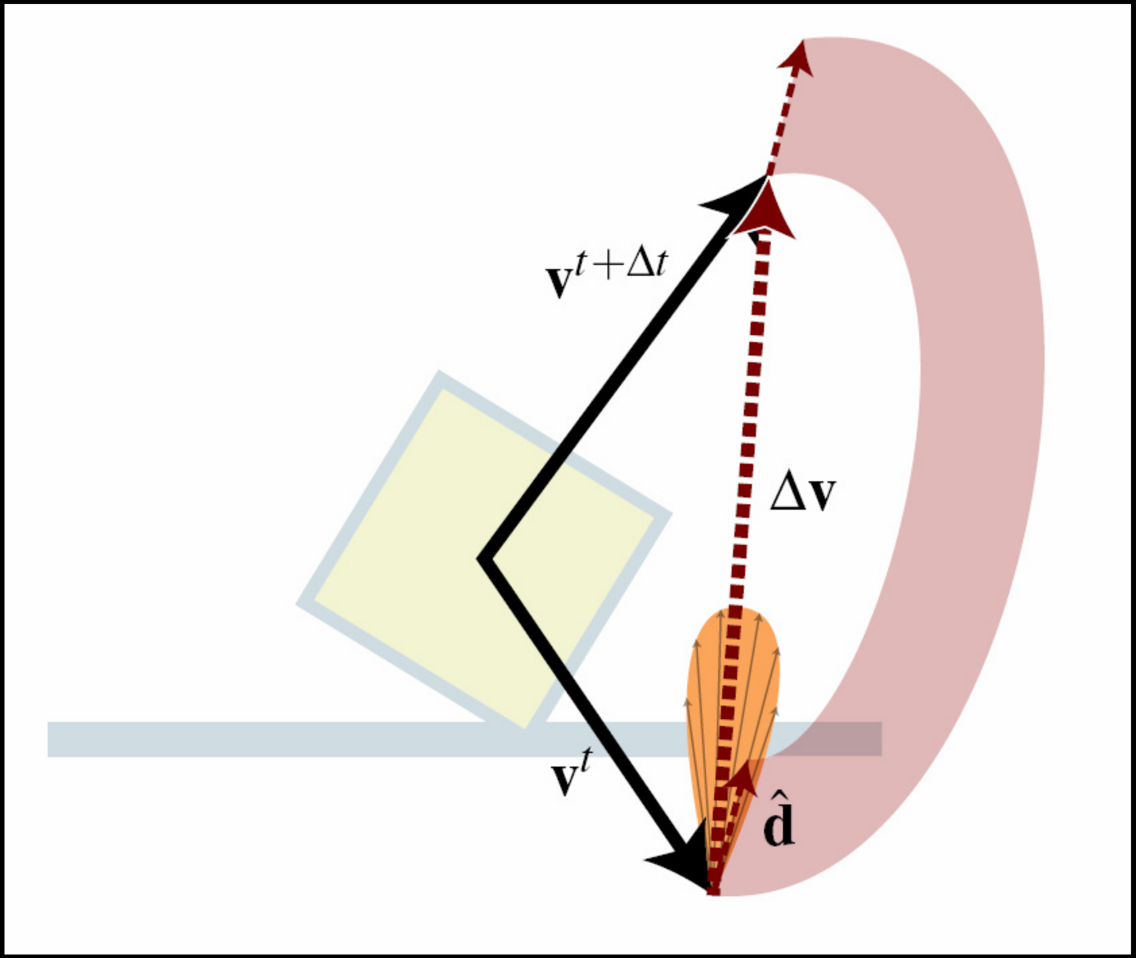
Advertising Programs - Business Solutions - About Google

Many-Worlds Browsing

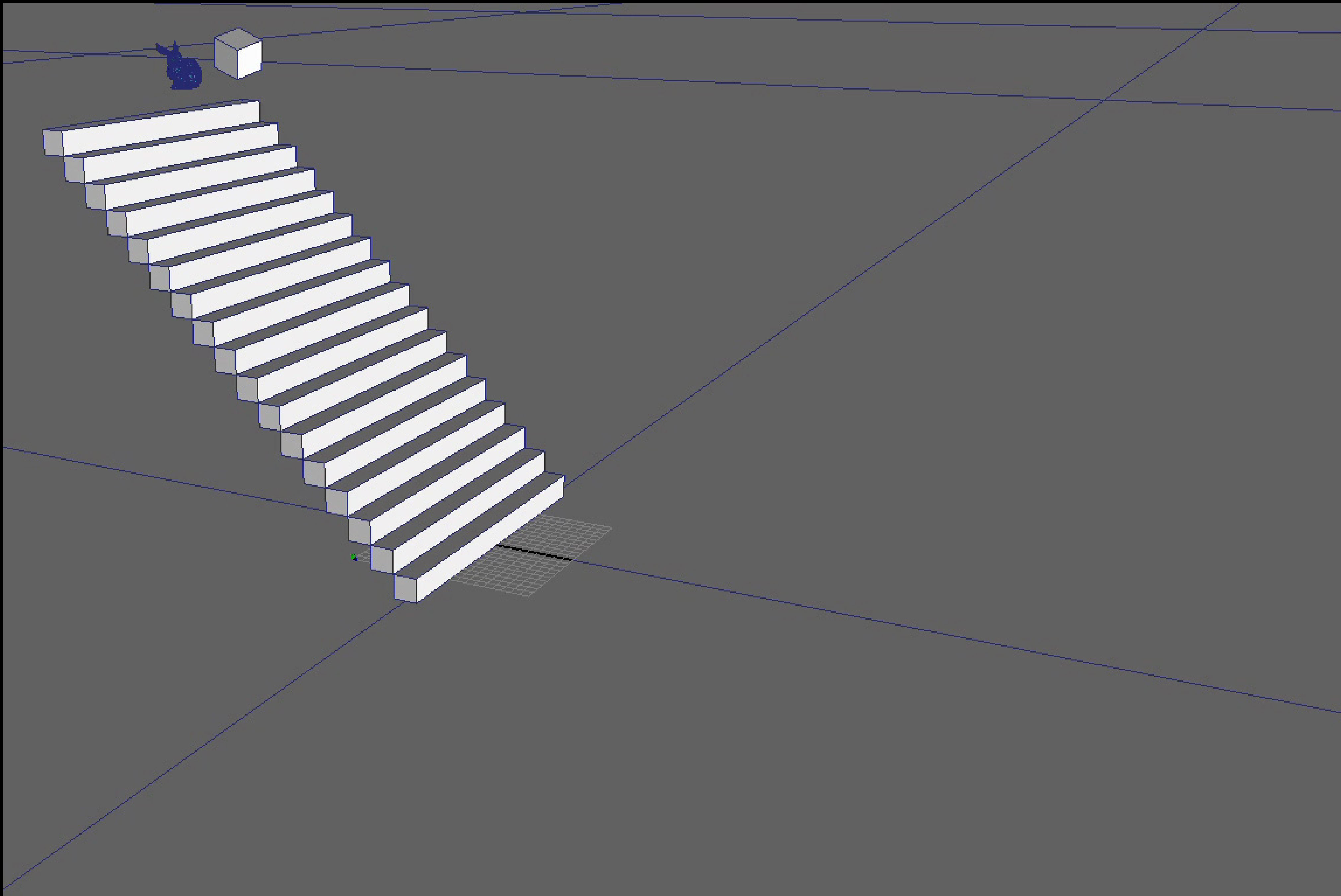


Browse What?!?

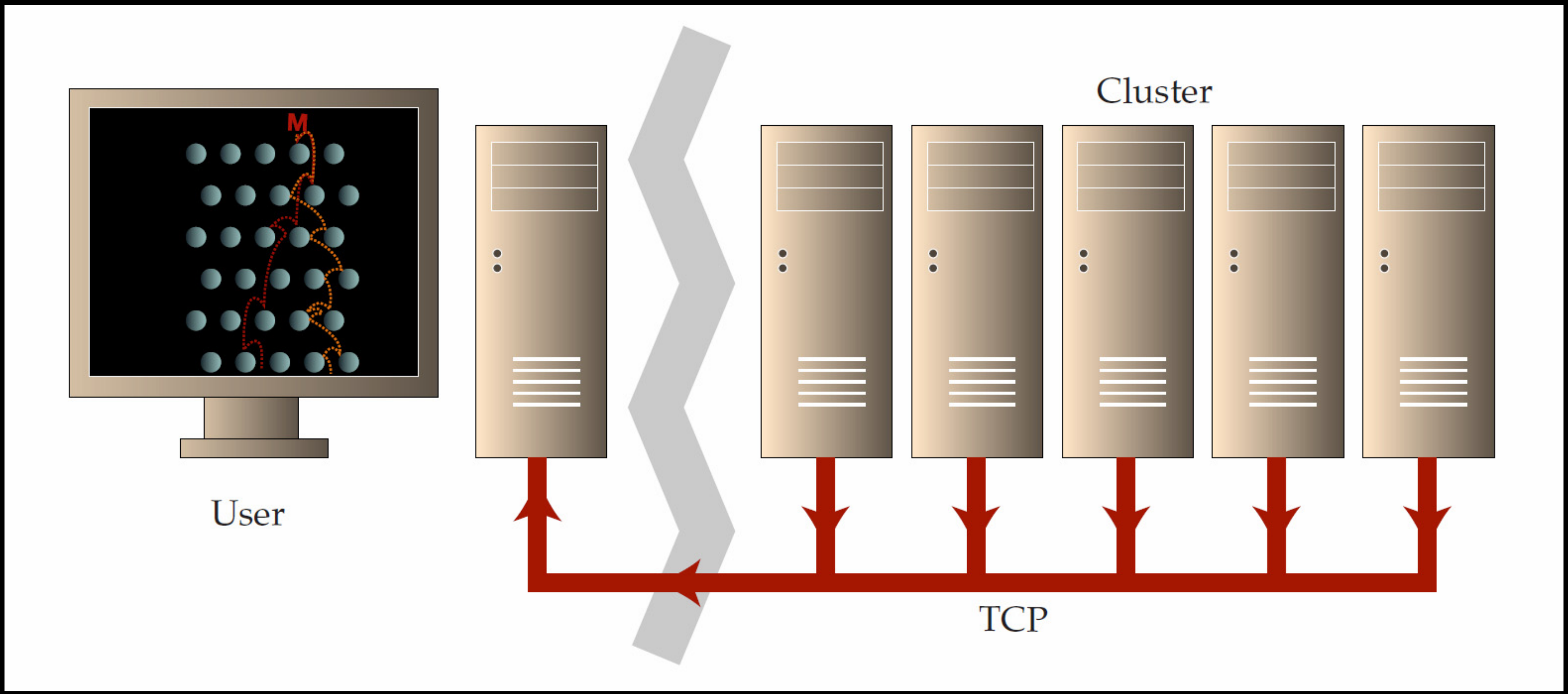
Sampling Possible Worlds



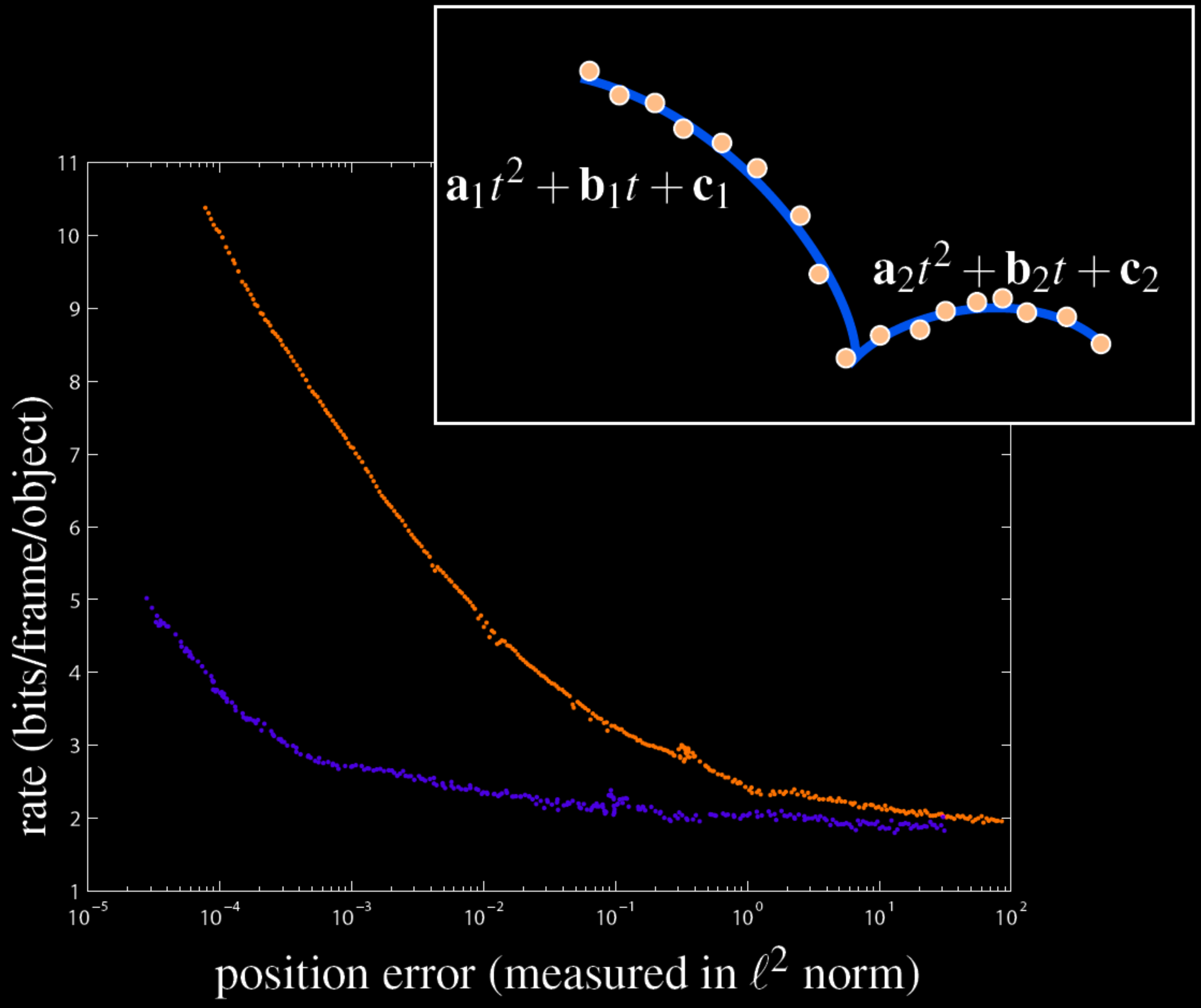
- Data-driven approach
- Sample "many worlds"
- Parallel computation
- Compress motion data for transmission, storage, and browsing



Parallel Simulation



Compression

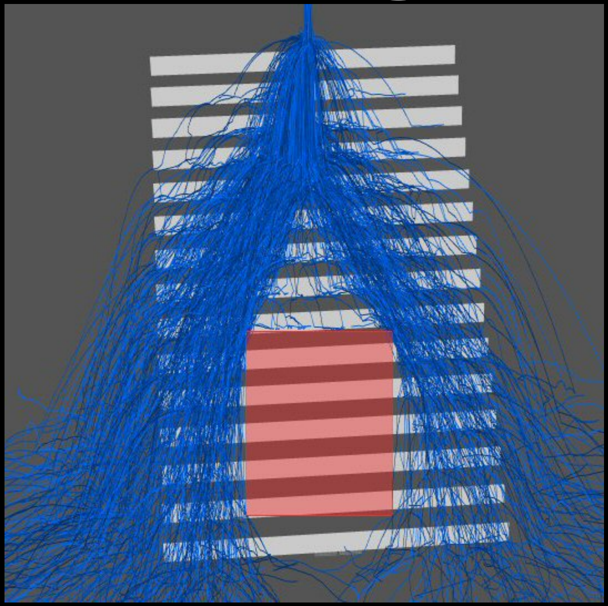


Browsing simulations

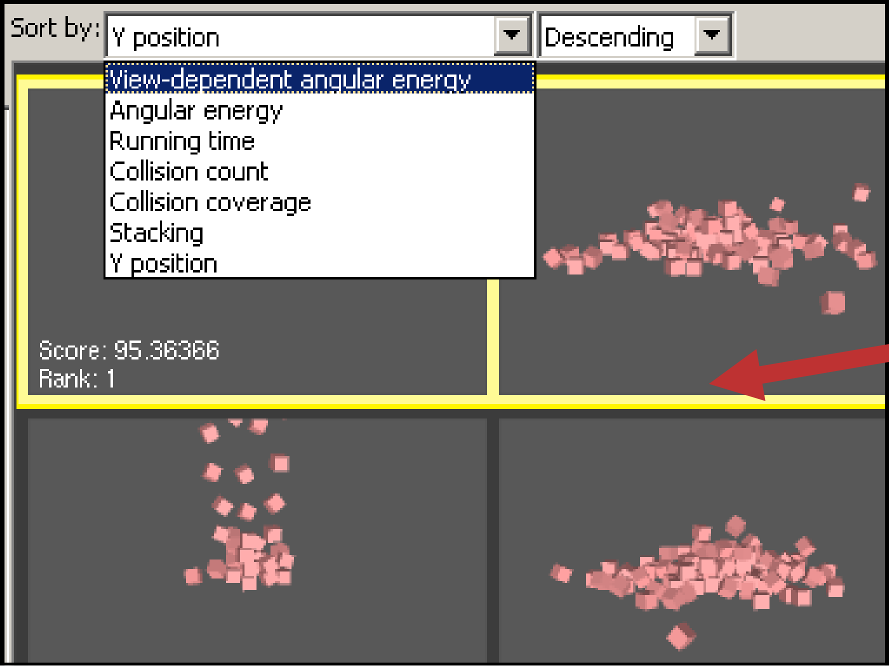
Constraints

Ranking

Browsing simulations

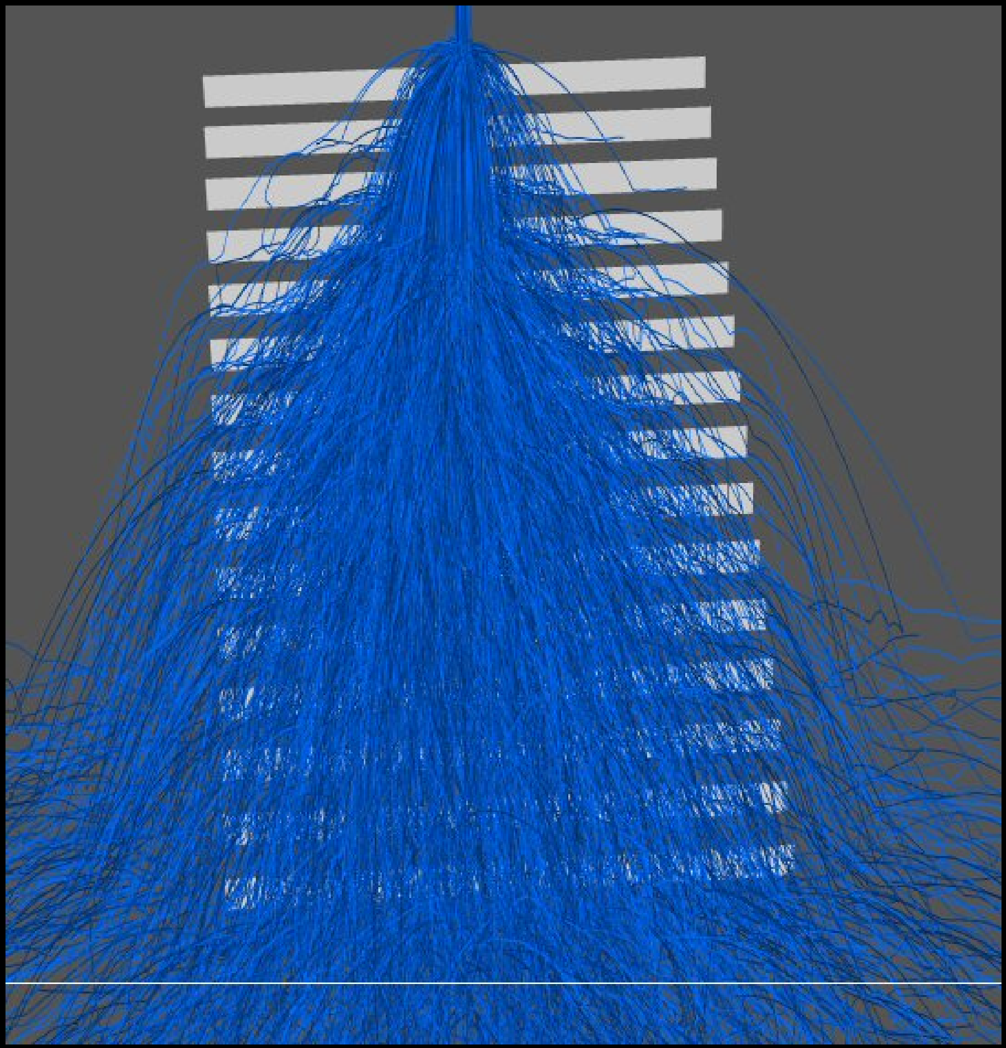


Constraints

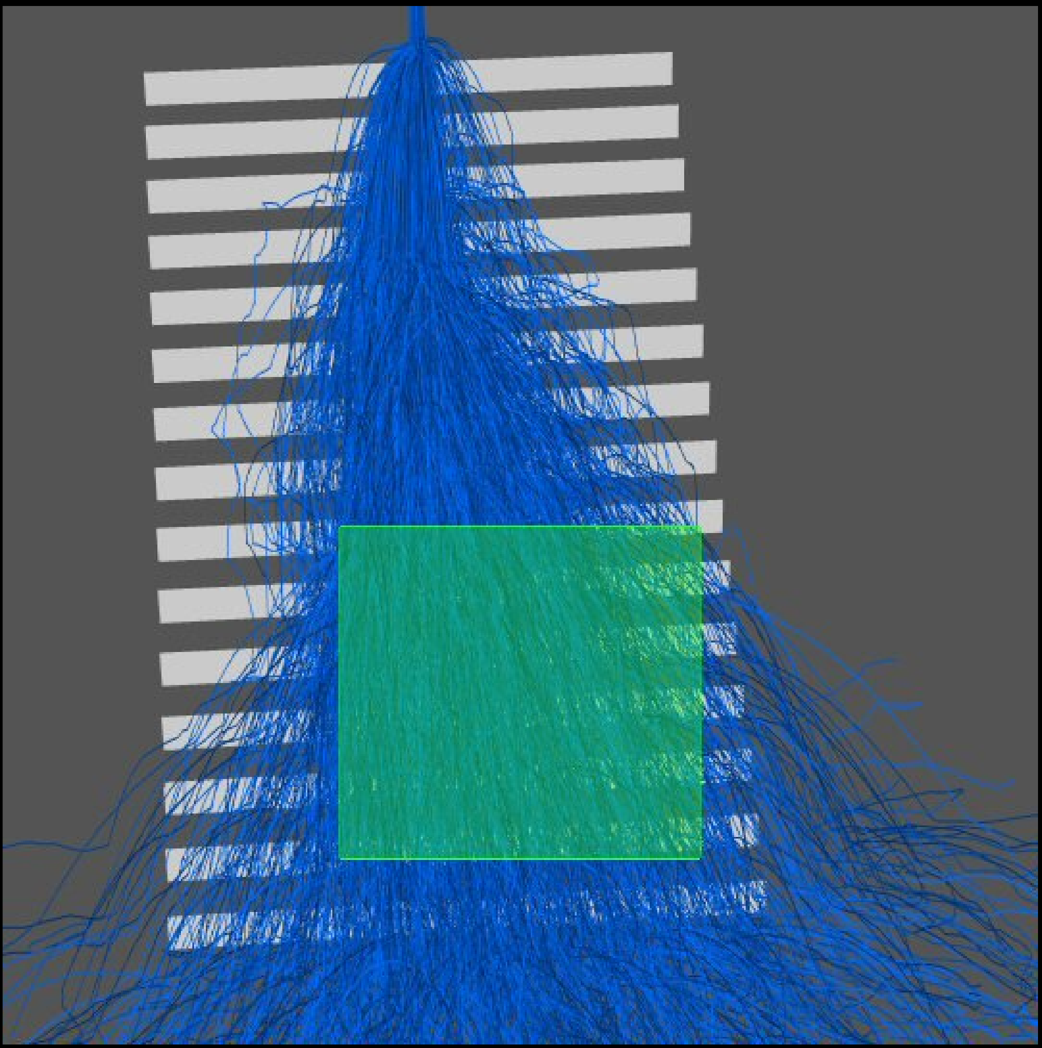
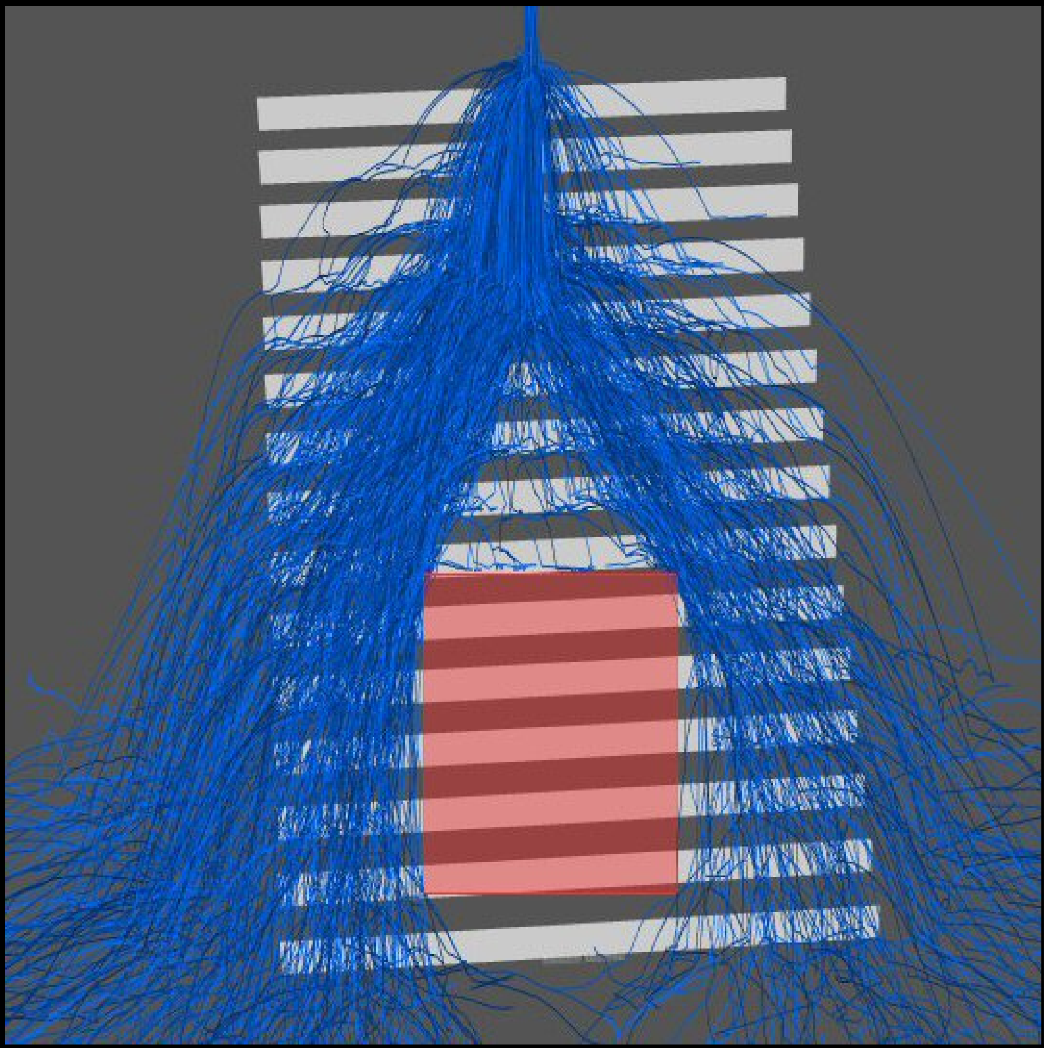


Ranking

Constraints

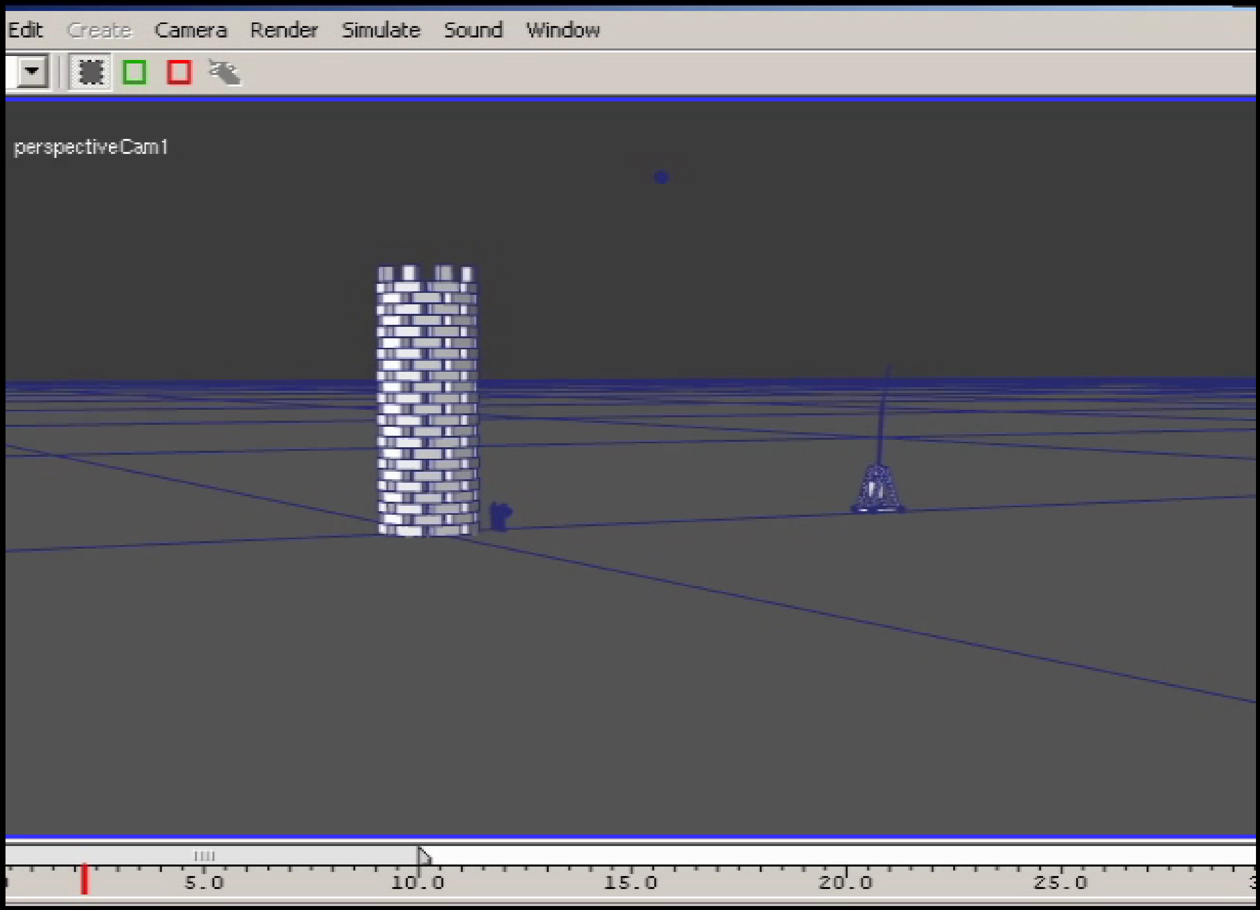


Constraints

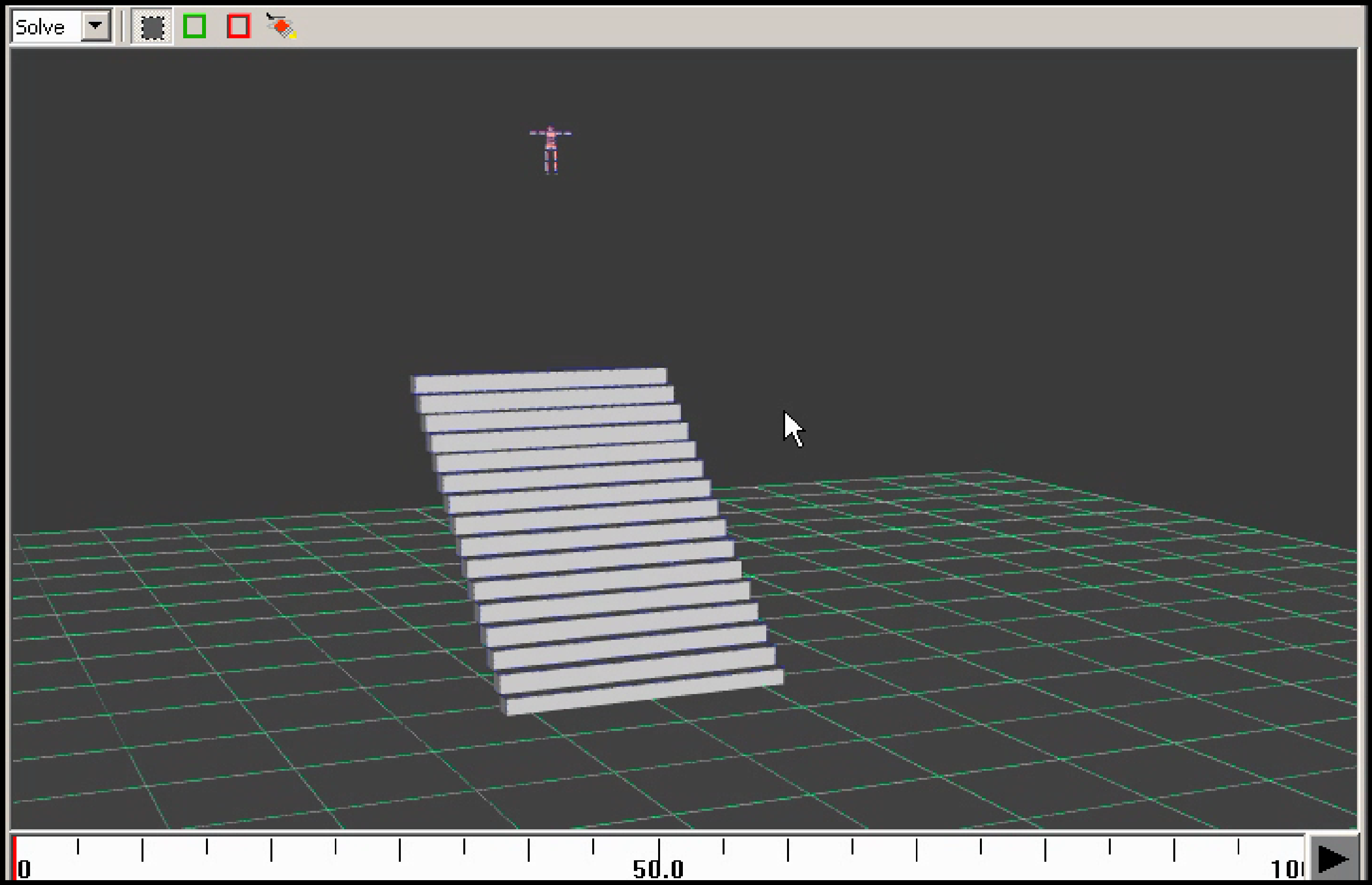
Positive **Negative**

Constraints

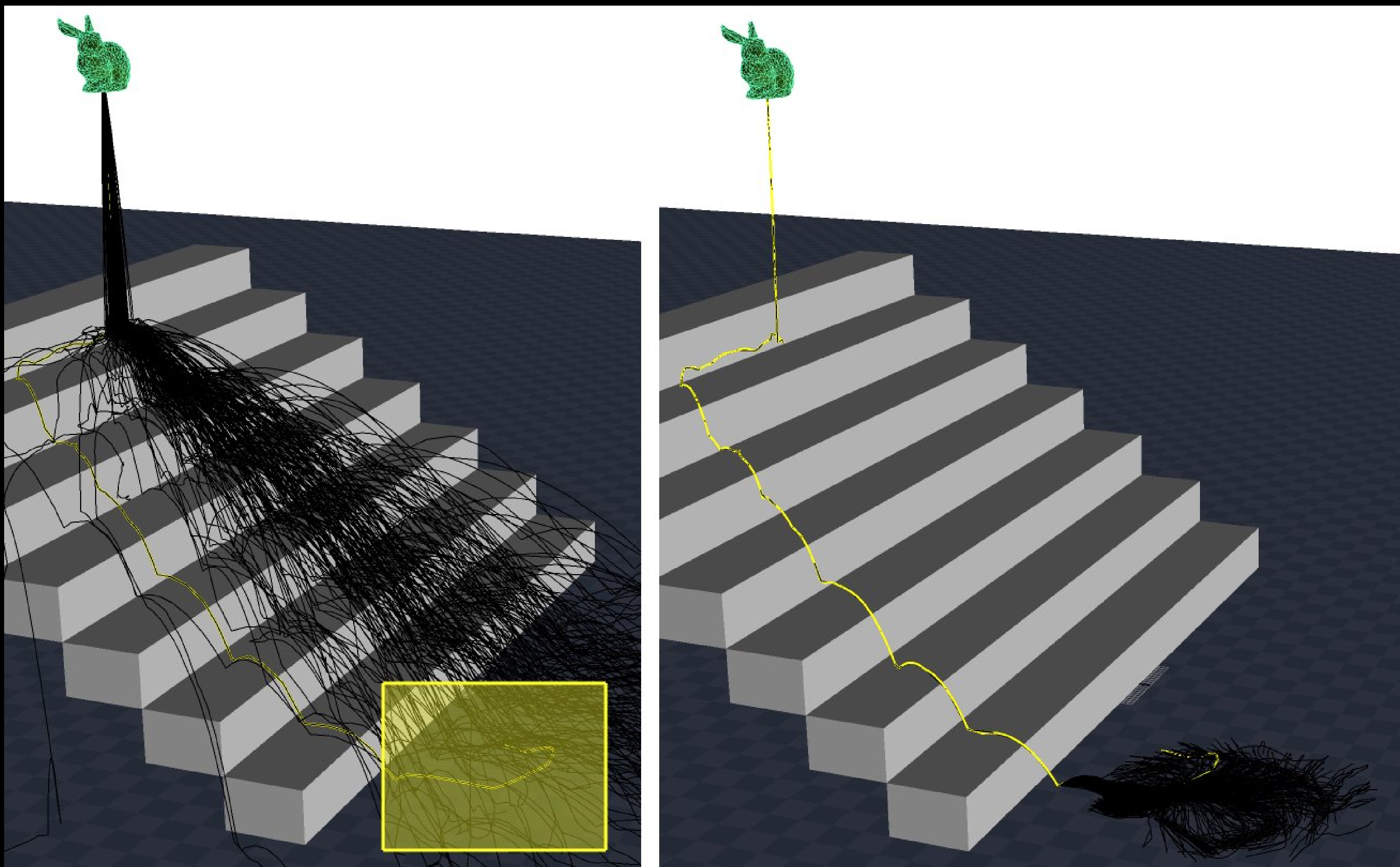


Spatial queries accelerated using fast OBB-Tree---frustum tests [Assarsson and Moller 2000]

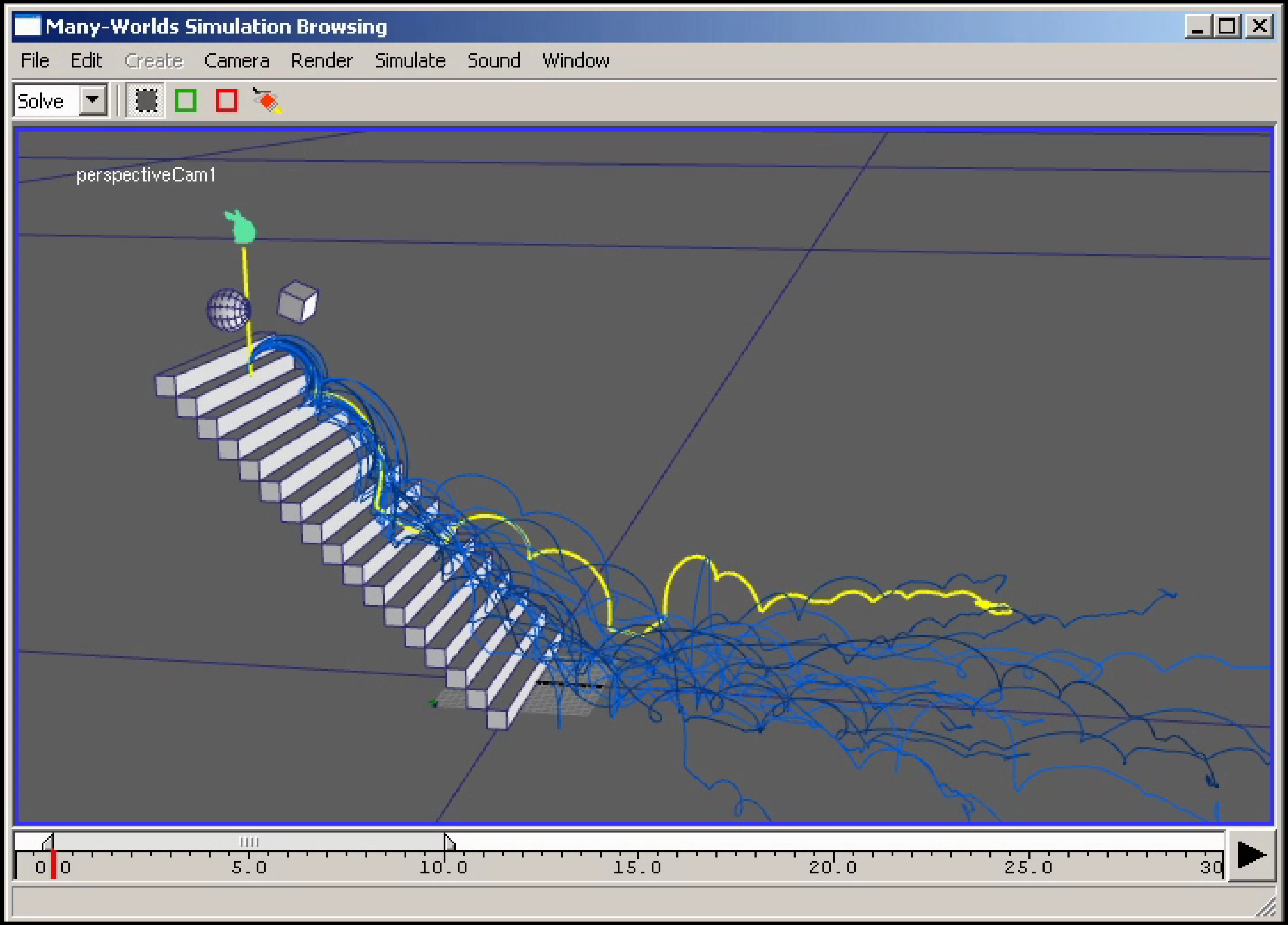
Ranking Metrics



Parallel Refinement



Refinement


Many-Worlds Simulation Browsing
File Edit Create Camera Render Simulate Sound Window
Solve
perspectiveCam1
0.0 5.0 10.0 15.0 20.0 25.0 30

Teaser

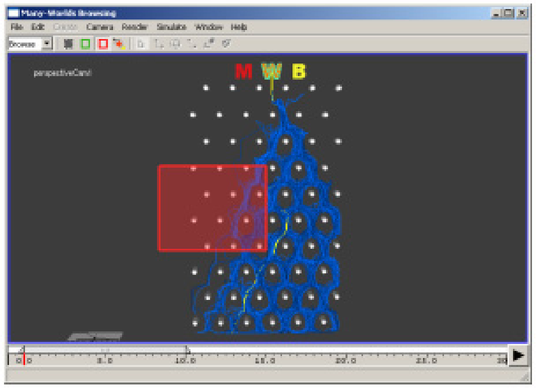
- C:\Documents and Settings\djames\Desktop\07 PAPERS\ManyWorldsBrowsing_caf2007_divx.avi

Getting started with the MWB demo - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://graphics.cs.cmu.edu/projects/mwb/demo/

Many-Worlds Browsing demo



To spur future research we are providing our entire system for others to experiment with. The executables are provided at this page; source code will shortly be appearing in a SourceForge project. Tips for getting started with the application can be found **here**; please read them before sending me questions, but let me know if there are any omissions.

- **The demo** (.zip file, 11MB)
- **Visual C++ 2005 SP1 redistributable package**: you will need this if you don't have it already. Note that this is **not** compatible with the pre-SP1 redistributable package.
- **Getting started with the MWB demo**

Note that this is a pretty sizeable block of research code, so it definitely has bugs in it. Please send **me** email if you run into any and I'll try to get to them as soon as I can. Thanks!

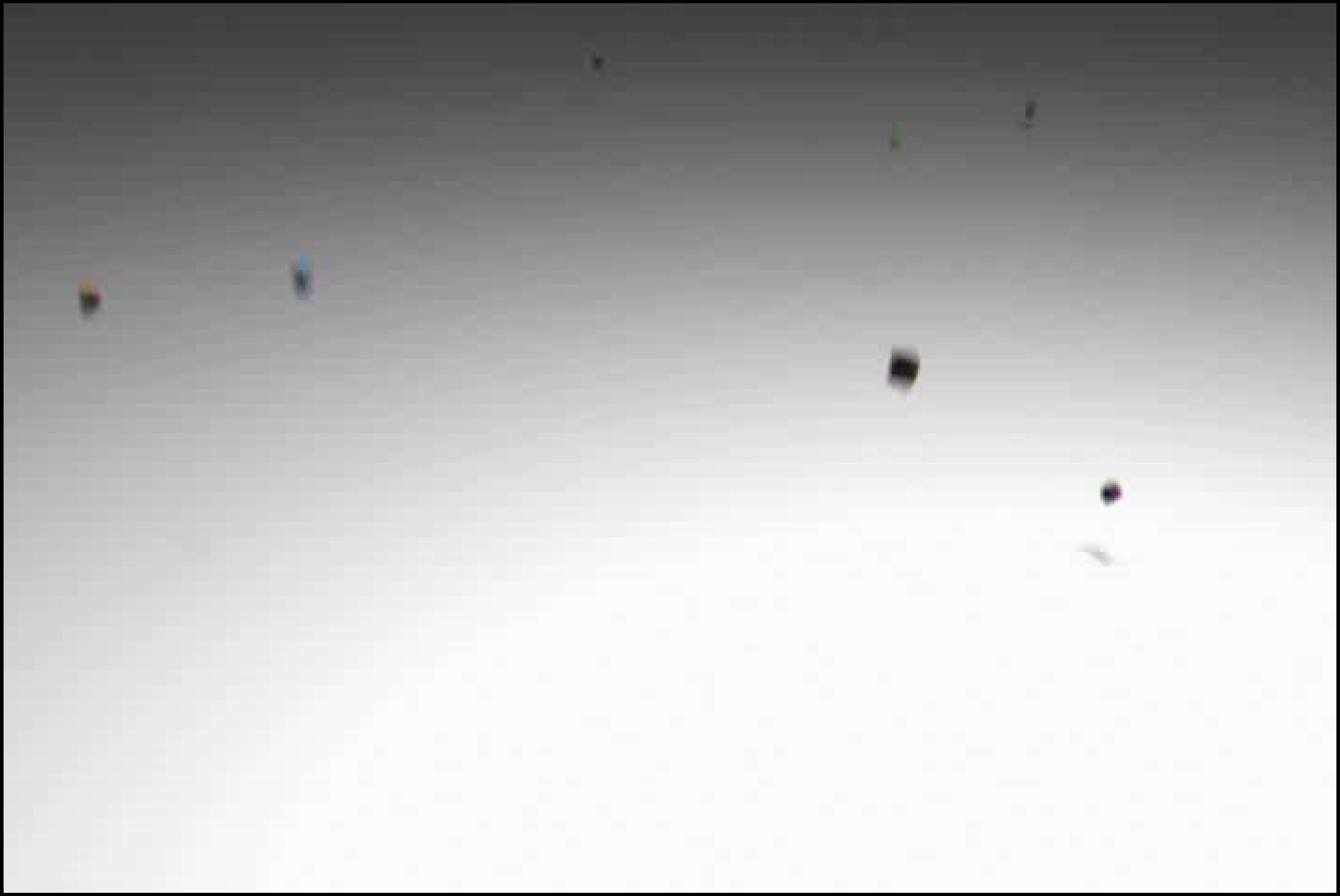
Carnegie Mellon GRAPHICS Cornell University

Comments, questions to **Christopher Twigg**. We support **web standards**; this page is valid **XHTML 1.0 Transitional** and **CSS**.

Done

CURRENT WORK

Backward Steps for Rigid Bodies



with Chris Twigg

CURRENT WORK

Yarn-level Simulation of Knitted Cloth

with Jonathan Kaldor and Steve Marschner

CURRENT WORK

Yarn-level Simulation of Knitted Cloth

with Jonathan Kaldor and Steve Marschner

Acknowledgements

- Funding and support:
  - National Science Foundation (CAREER, EMT)
  - Alfred P. Sloan Foundation
  - NIH
  - Pixar
  - The Boeing Company
  - NVIDIA (hardware, graduate fellowship)
  - Intel
  - Autodesk
  - The Link Foundation

Thanks!

Questions?

Untitled di Doris Salcedo. L'artista colombiana espone al Castello di Rivoli