Harmonic Shells
A Practical Nonlinear Sound Model for Near-Rigid Thin Shells

Jeffrey Chadwick, Steven An and Doug James

Cornell University
Linear Modal Sound Synthesis

\[ \text{Cymbals} = \text{Drum 1} + \text{Drum 2} + \ldots \]
Linear modal sound
Linear modal sound
Linear modal sound + transfer
Linear modal sound + transfer
Harmonic Shells
Harmonic Shells
Motivation
Motivation

Rigid objects: vibrations approximated well by linear dynamics
Motivation

Rigid objects: vibrations approximated well by linear dynamics

Shell structures: exhibit noisy nonlinear behavior (even under modest forcing)
Motivation

Linear modal sound simulation (side impact):
Motivation

Linear modal sound simulation (side impact):
Motivation

Nonlinear sound simulation:
Motivation

Nonlinear sound simulation:
Motivation

Nonlinear sound simulation:

(... but this took about 19 days to synthesize)
Harmonic Shells
Harmonic Shells

- A practical approach to computing nonlinear vibrations for thin shells
Harmonic Shells

- A practical approach to computing nonlinear vibrations for thin shells
- Extend standard linear modal sounds by introducing nonlinear mode coupling and force response
Harmonic Shells

• A practical approach to computing nonlinear vibrations for thin shells

• Extend standard linear modal sounds by introducing nonlinear mode coupling and force response
  • Richer sounds than linear models
Harmonic Shells

• A practical approach to computing nonlinear vibrations for thin shells

• Extend standard linear modal sounds by introducing nonlinear mode coupling and force response
  • Richer sounds than linear models

• A texture-based method for fast (O(1) per mode) acoustic transfer computation
Related Work

Linear Modal Sounds

Linear Modal Sounds:
eg. [van den Doel et al. 1996]

Frequently used in graphics, eg:

"FoleyAutomatic"
[van den Doel et al. 2001]

"Synthesizing Sounds from Rigid-Body Simulations"
[O’Brien et al. 2002]
Related Work

Linear Modal Sounds
Related Work

Linear Modal Sounds

- Fails to capture a lot of interesting sound behavior
Related Work
Linear Modal Sounds

- Fails to capture a lot of interesting sound behavior
- Simple example: sound characteristics (not just volume) change with impact magnitude
Related Work

Linear Modal Sounds

- Fails to capture a lot of interesting sound behavior
- Simple example: sound characteristics (not just volume) change with impact magnitude
Related Work
Linear Modal Sounds

- Fails to capture a lot of interesting sound behavior
- Simple example: sound characteristics (not just volume) change with impact magnitude
Related Work

Linear Modal Sounds

- Fails to capture a lot of interesting sound behavior
- Simple example: sound characteristics (not just volume) change with impact magnitude
Related Work
Linear Modal Sounds

• Fails to capture a lot of interesting sound behavior
• Simple example: sound characteristics (not just volume) change with impact magnitude
Related Work
Linear Modal Sounds

- Fails to capture a lot of interesting sound behavior
- Simple example: sound characteristics (not just volume) change with impact magnitude
- Linear model does not capture this
Related Work
Nonlinear vibrations and sound

“Synthesizing Sounds from Physically Based Motion”
[O’Brien et al. 2001]

Efficient, conservative numerical schemes for nonlinear plates and strings
[Bilbao 2005, 2008]

“Nonlinear vibrations and chaos in gongs and cymbals”
[Chaigne et al. 2005]

No efficient nonlinear synthesis methods for sound in animation
Algorithm Overview
Algorithm Overview
Algorithm Overview

Geometry → Vibration basis U
Algorithm Overview

Geometry → Vibration basis $U$ → Training poses
Algorithm Overview

Geometry $\rightarrow$ Vibration basis $U$ $\rightarrow$ Training poses $\rightarrow$ Cubature scheme
Algorithm Overview

Geometry → Vibration basis $U$ → Training poses → Cubature scheme
Algorithm Overview

- Geometry
- Vibration basis U
- Training poses
- Cubature scheme
- Acoustic pressure
Algorithm Overview

Geometry → Vibration basis U → Training poses → Cubature scheme → Acoustic pressure → Far-field acoustic transfer maps
Algorithm Overview

- Geometry
- Vibration basis $U$
- Training poses
- Acoustic pressure
- Far-field acoustic transfer maps
- Cubature scheme
Algorithm Overview

Geometry\rightarrow Vibration basis U\rightarrow Training poses\rightarrow Cubature scheme\rightarrow Acoustic pressure\rightarrow Far-field acoustic transfer maps

Rigid body simulation
Algorithm Overview

Geometry → Vibration basis $U$ → Training poses → Cubature scheme → Acoustic pressure → Far-field acoustic transfer maps

Rigid body simulation → Project impulse forces
Algorithm Overview

Geometry \rightarrow Vibration basis U \rightarrow Training poses \rightarrow Cubature scheme

Acoustic pressure \rightarrow Far-field acoustic transfer maps

Rigid body simulation \rightarrow Project impulse forces
Algorithm Overview

1. Geometry
2. Vibration basis $U$
3. Training poses
4. Cubature scheme
5. Acoustic pressure
6. Far-field acoustic transfer maps
7. Simulate vibrations
8. Project impulse forces
9. Rigid body simulation
Algorithm Overview

Geometry → Vibration basis U → Training poses → Cubature scheme → Project impulse forces → Simulate vibrations

Acoustic pressure

Far-field acoustic transfer maps

Simulate sound

Rigid body simulation
Model Reduction

Precompute exterior acoustic pressure

Far-field acoustic transfer maps

Train cubature scheme

Train cubature scheme

Simulate vibrations

Synthesize sound

Geometry, physical parameters

Vibration basis U

Training poses

Project impulse forces

Rigid body simulation

Sunday, December 13, 2009
Model Reduction

Related Work

Classical subspace integration, eg. [Bathe, 1996]

[Krysl et al. 2001] - Dimensional model reduction in non-linear finite element dynamics; “POD”/PCA


[An et al. 2008] - Accelerated reduced force computation for general nonlinear materials
Model Reduction

Strain energy density (constant over triangle) [Gingold et al. 2004]:

\[ W(X, x) = + \]
Model Reduction

Strain energy density (constant over triangle) [Gingold et al. 2004]:

\[ W(X, x) = S \int_{S} W(X, x) dS(X) \]

Strain Energy:

\[ E(x) = \int_{S} W(X, x) dS(X) \]
Model Reduction

Strain energy density (constant over triangle) [Gingold et al. 2004]:

\[ W(X, x) = \]

\[ E(x) = \int_S W(X, x) dS(X) \]

Strain Energy:

\[ f(x) = \nabla_x E(x) = \int_S \nabla_x W(X, x) dS(X) = \int_S G(X, x) dS(X) \]
Model Reduction

Strain energy density (constant over triangle) [Gingold et al. 2004]:

\[ W(X, x) = \]

\[ E(x) = \int_S W(X, x) dS(X) \]

Strain Energy:

Internal forces:

\[ f(x) = \nabla_x E(x) = \int_S \nabla_x W(X, x) dS(X) = \int_S G(X, x) dS(X) \]

Force density
Model Reduction
Model Reduction

Nonlinear system of equations in displacements $u$

\[ M \ddot{u} + f(u) = f_{external} \]
Model Reduction

Nonlinear system of equations in displacements $u$

\[ M\ddot{u} + f(u) = f_{\text{external}} \]

Internal forces
Model Reduction

Nonlinear system of equations in displacements $u$

\[ M\ddot{u} + f(u) = f_{\text{external}} \]

Suppose some displacement basis given:

\[ u = Uq \quad U \in \mathbb{R}^{3N \times r} \quad U = \text{displacement basis} \]
Model Reduction

Nonlinear system of equations in displacements \( u \)

\[
M \ddot{u} + f(u) = f_{\text{external}}
\]

Suppose some displacement basis given:

\[
u = Uq \quad U \in \mathbb{R}^{3N \times r} \quad U = \text{displacement basis}
\]

\[
q \in \mathbb{R}^{r} \quad r \ll 3N \quad q = \text{modal coordinates}
\]

\[
3N \sim 100K \quad q \sim \text{hundreds}
\]
Model Reduction

\[ M \ddot{u} + f(u) = f_{\text{external}} \quad u = Uq \]

Eigen-modes and frequencies from linear modal analysis
Model Reduction

\[ \ddot{\mathbf{u}} + f(\mathbf{u}) = f_{\text{external}} \quad \mathbf{u} = \mathbf{U}q \]

Eigen-modes and frequencies from linear modal analysis

\[ U_{:,1} \quad U_{:,2} \quad U_{:,3} \]

\[ U_{:,4} \quad U_{:,5} \quad U_{:,6} \]

\[ U_{:,7} \quad U_{:,8} \quad U_{:,9} \]
Model Reduction

\[ \mathbf{M} \ddot{\mathbf{u}} + f(u) = f_{\text{external}} \quad u = U_q \]
Model Reduction

\[ \ddot{u} + f(u) = f_{\text{external}} \]

\[ u = Uq \]
Model Reduction

\[
\begin{align*}
M\ddot{u} + f(u) &= f_{\text{external}} \\
u &= Uq \\
U^T M U \ddot{q} + U^T f(Uq) &= U^T f_{\text{external}}
\end{align*}
\]
Model Reduction

\[ M\ddot{u} + f(u) = f_{\text{external}} \]

\[ u = Uq \]

\[ U^T M U \ddot{q} + U^T f(Uq) = U^T f_{\text{external}} \]

\[ \tilde{M} \ddot{q} + \tilde{f}(q) = \tilde{f}_{\text{external}} \]
Model Reduction

\[ M\ddot{u} + f(u) = f_{\text{external}} \]

\[ u = Uq \]

\[ U^TMU\ddot{q} + U^Tf(Uq) = U^Tf_{\text{external}} \]

\[ \ddot{\tilde{M}}q + \tilde{f}(q) = \tilde{f}_{\text{external}} \]

Reduced internal forces
Model Reduction

\[ M\ddot{u} + f(u) = f_{external} \]

\[ u = Uq \]

\[ U^T M U \ddot{q} + U^T f(Uq) = U^T f_{external} \]

\[ \tilde{M}\ddot{q} + \tilde{f}(q) = \tilde{f}_{external} \]

Question: How to compute \( \tilde{f}(q) \)?
Model Reduction
Model Reduction

Recall: Internal forces

\[ f(x) = \int_S G(X, x) dS(X) \]
Model Reduction

Recall: Internal forces

\[ f(x) = \int_S G(X, x) dS(X) \]

\[ \tilde{f}(q) = \int_S U^T G(X, Uq) dS(X) = \int_S g(X, q) dS(X) \]
Model Reduction

Recall: Internal forces

\[ f(x) = \int_S G(X, x) dS(X) \]

Forced density

\[ \tilde{f}(q) = \int_S U^T G(X, Uq) dS(X) = \int_S g(X, q) dS(X) \]

Problem: Matrix multiplies are \( O(rN) \)
Model Reduction

Recall: Internal forces

\[ f(x) = \int_S G(\mathbf{X}, x) \, dS(\mathbf{X}) \]

Reduced force density

\[ \tilde{f}(q) = \int_S \mathbf{U}^T G(\mathbf{X}, \mathbf{U}q) \, dS(\mathbf{X}) = \int_S g(\mathbf{X}, q) \, dS(\mathbf{X}) \]

Problem: Matrix multiplies are \( O(rN) \)

Want: Reduced force evaluation independent of \( N \) (dependent only on \( r \))
Model Reduction

\[ \ddot{M}\ddot{q} + \tilde{f}(q) = \tilde{f}_{external} \]

\[ \tilde{f}(q) = \int_S U^T G(X, Uq) dS(X) = \int_S g(X, q) dS(X) \]

Classical model reduction approach, eg. [Bathe 1996]

Individual explicit time steps more expensive (\(O(rN)\) instead of \(O(N)\))

Has potential to significantly improve stability in explicit integration (larger time steps)
Optimized Cubature

Previous work
Optimized Cubature

Previous work

• Introduced in [An et al. 2008]; tetrahedral models
Optimized Cubature

Previous work

• Introduced in [An et al. 2008]; tetrahedral models

• Approximate integral:

\[ \tilde{f}(\mathbf{q}) = \int_{S} g(\mathbf{X}, \mathbf{q}) dS(\mathbf{X}) \approx \sum_{i=1}^{M} w_i g(\mathbf{X}_i, \mathbf{q}) \]
Optimized Cubature

Previous work

• Introduced in [An et al. 2008]; tetrahedral models

• Approximate integral:

\[ \tilde{f}(q) = \int_S g(X, q) dS(X) \approx \sum_{i=1}^{M} w_i g(X_i, q) \]

• Input: Training poses and forces

\[ q_1, q_2, \ldots, q_{NT}, \quad \tilde{f}(q_1), \tilde{f}(q_2), \ldots, \tilde{f}(q_{NT}) \]
Optimized Cubature

Previous work

- Introduced in [An et al. 2008]; tetrahedral models
- Approximate integral:

\[ \tilde{f}(q) = \int_S g(X, q)dS(X) \approx \sum_{i=1}^{M} w_i g(X_i, q) \]

- Input: Training poses and forces

\[ q_1, q_2, \ldots, q_{NT}, \quad \tilde{f}(q_1), \tilde{f}(q_2), \ldots, \tilde{f}(q_{NT}) \]

- Output: points \( X_i \) and optimized weights \( w_i \)
Optimized Cubature

Previous work

\[ \tilde{f}(q) = \int_S g(X, q) dS(X) \approx \sum_{i=1}^{M} w_i g(X_i, q) \]
Optimized Cubature

Previous work

\[ \tilde{f}(q) = \int_S g(X, q) \, dS(X) \approx \sum_{i=1}^{M} w_i g(X_i, q) \]

Result: \( O(r^2) \) approximation of \( \tilde{f}(q) \)
Optimized Cubature

Previous work

\[ \tilde{f}(q) = \int_S g(X, q) dS(X) \approx \sum_{i=1}^M w_i g(X_i, q) \]

Result: \( O(r^2) \) approximation of \( \tilde{f}(q) \)

\( O(r^2) \) explicit time steps for system - reduced from \( O(rN) \)

\[ \tilde{M}\ddot{q} + \tilde{f}(q) = \tilde{f}_{external} \]
Optimized Cubature
Applying Cubature to Thin Shells

\[ \tilde{f}(q) = \int_S g(X, q) dS(X) \]
Optimized Cubature
Applying Cubature to Thin Shells

\[ \tilde{f}(q) = \int_{S} g(X, q) dS(X) \]

Strain energy density: constant over each triangle
(same is true for reduced force density)

\[ W(X, x) = \]
Optimized Cubature
Applying Cubature to Thin Shells

Internal forces: sum over triangles

\[ \tilde{f}(q) = \int_S g(X, q) dS(X) = \sum_{i=1}^{N_T} A_i g(X_{T_i}, q) \]

\[ g(X_{T_i}, q) = \]

Sunday, December 13, 2009
Optimized Cubature
Applying Cubature to Thin Shells

\[ \tilde{f}(q) = \sum_{i=1}^{N_T} A_i g(X_{T_i}, q) \]

\[ g(X_{T_i}, q) = \]
Optimized Cubature
Applying Cubature to Thin Shells

Internal forces: sum over triangles

\[ \tilde{f}(q) = \sum_{i=1}^{N_T} A_i g(X_{T_i}, q) \]

\[ g(X_{T_i}, q) = + \]
Optimized Cubature
Applying Cubature to Thin Shells

Internal forces: sum over triangles

\[ \tilde{f}(q) = \sum_{i=1}^{N_T} A_i g(X_{T_i}, q) \]

Choose subset and weights:

\[ \{t_1, \ldots, t_C\} \subset \{T_1, \ldots, T_{N_T}\} \]
\[ \{w_1, \ldots, w_C\} \]

\[ g(X_{T_i}, q) = + \]

\[ C \ll N_T \]
Optimized Cubature
Applying Cubature to Thin Shells

Internal forces: sum over triangles

\[ \tilde{f}(q) = \sum_{i=1}^{N_T} A_i g(X_{T_i}, q) \]

Choose subset and weights:

\[ \{t_1, \ldots, t_C\} \subset \{T_1, \ldots, T_{N_T}\} \]
\[ \{w_1, \ldots, w_C\} \]

\[ \sum_{i=1}^{N_T} A_i g(X_{T_i}, q) \approx \sum_{i=1}^{C} w_i A_i g(X_{t_i}, q) \]
Optimized Cubature
Applying Cubature to Thin Shells

Internal forces: sum over triangles

\[ \tilde{f}(q) = \sum_{i=1}^{N_T} A_i g(X_{T_i}, q) \]

Choose subset and weights:

\[ \{ t_1, \ldots, t_C \} \subset \{ T_1, \ldots, T_{N_T} \} \]
\[ \{ w_1, \ldots, w_C \} \]

\[ \sum_{i=1}^{N_T} A_i g(X_{T_i}, q) \approx \sum_{i=1}^{C} w_i A_i g(X_{t_i}, q) \]

Use cubature training to choose subset/weights
Optimized Cubature

800 element cubature scheme (78K triangles)
Model Reduction

Summary
Model Reduction

Summary

• What we keep from linear modal sound synthesis:
Model Reduction

Summary

• What we keep from linear modal sound synthesis:
  • Small displacement assumption
Model Reduction

Summary

- What we keep from linear modal sound synthesis:
  - Small displacement assumption
  - Linear shape model

\[ u = Uq \]
Model Reduction

Summary

• What we keep from linear modal sound synthesis:
  • Small displacement assumption
  • Linear shape model

\[ u = Uq \]

• Differences from linear modal synthesis

\[ \tilde{M}\ddot{q} + \tilde{K}q = U^Tf_{ext} \]
Model Reduction

Summary

• What we keep from linear modal sound synthesis:
  • Small displacement assumption
  • Linear shape model

\[ u = Uq \]

• Differences from linear modal synthesis

\[ \ddot{M}\ddot{q} + \ddot{f}_{int}(q) = U^Tf_{ext} \]
Model Reduction

Summary

\[ \tilde{\dot{M}} \ddot{q} + \tilde{f}_{int}(q) = U^T f_{ext} \]
Model Reduction

Summary

\[ \ddot{\textbf{M}} \dot{\textbf{q}} + \tilde{\textbf{f}}_{int}(\textbf{q}) = \textbf{U}^T \textbf{f}_{ext} \]

Dimensional model reduction:
Significantly increases stable time step size
Model Reduction

Summary

\[ \ddot{\tilde{M}}\ddot{q} + \tilde{f}_{int}(q) = U^T f_{ext} \]

Dimensional model reduction:
Significantly increases stable time step size

Full simulation: \( \sim 11\text{M time steps per second} \)

Reduced simulation: 44100 time steps per second
Model Reduction

Summary

\[ \tilde{M} \ddot{q} + \tilde{f}_{int}(q) = U^T f_{ext} \]

Dimensional model reduction:
Significantly increases stable time step size

Full simulation: \(~11M\) time steps per second

Reduced simulation: 44100 time steps per second

19 days vs. 15 hours for 5s of audio
Model Reduction

Summary

\[ \tilde{M} \ddot{q} + \tilde{f}_{\text{int}}(q) = U^T f_{\text{ext}} \]
Model Reduction

Summary

\[ \tilde{M}\ddot{q} + \tilde{f}_{\text{int}}(q) = U^T f_{\text{ext}} \]

Cubature algorithm:
Reduces time step cost from \( O(rN) \) to \( O(r^2) \)
Cubature algorithm:
Reduces time step cost from $O(rN)$ to $O(r^2)$

15 hours vs. 1.5 hours for 5s of audio
Model Reduction

Summary

\[ \ddot{\mathbf{M}} \ddot{\mathbf{q}} + \tilde{\mathbf{f}}_{\text{int}}(\mathbf{q}) = \mathbf{U}^T \mathbf{f}_{\text{ext}} \]

Cubature algorithm:
Reduces time step cost from \( O(rN) \) to \( O(r^2) \)

15 hours vs. 1.5 hours for 5s of audio

Overall: Larger, cheaper time steps
Approximating Acoustic Transfer

- Precompute exterior acoustic pressure
- Far-field acoustic transfer maps
- Geometry, physical parameters → Vibration basis $U$ → Training poses → Train cubature scheme → Simulate vibrations → Synthesize sound
- Rigid body simulation → Project impulse forces
Approximating Acoustic Transfer
Approximating Acoustic Transfer

Sum of modal amplitudes:

\[ p(x, t) = \frac{q_1(t)}{r} + \frac{q_2(t)}{r} + \ldots \]
Approximating Acoustic Transfer

Sum of modal amplitudes:

\[ p(x, t) = \sum_{i=1}^{N_{\text{modes}}} q_i(t) |p_i(x)| + q_1(t) + q_2(t) + \ldots \]

Or, weighted sum:

\[ p(x, t) = \sum_{i=1}^{N_{\text{modes}}} q_i(t) |p_i(x)| \]
Approximating Acoustic Transfer

Sum of modal amplitudes:

\[ p(\mathbf{x}, t) = \frac{q_1(t) + q_2(t) + \ldots}{r} \]

Or, weighted sum:

\[ p(\mathbf{x}, t) = \sum_{i=1}^{N_{\text{modes}}} q_i(t) |p_i(\mathbf{x})| \]

\[ p_i(\mathbf{x}) \propto \frac{k_i}{|\mathbf{x}|} \]

“Acoustic transfer function” (far-field, low frequency, monopole approximation)
Approximating Acoustic Transfer

Sum of modal amplitudes:

\[ p(x, t) = \frac{q_1(t)}{r} + \frac{q_2(t)}{r} + \ldots \]

Or, weighted sum:

\[ p(x, t) = \sum_{i=1}^{N_{\text{modes}}} q_i(t) \left| p_i(x) \right| \]

\[ p_i(x) \propto \frac{k_i}{|x|} \]

“Acoustic transfer function” (far-field, low frequency, monopole approximation)

In general: \( (\nabla^2 + k_i^2) p_i(x) = 0 \)
Approximating Acoustic Transfer

Acoustic Transfer function: $p(x)$

Amplitude of unit vibration: $|p(x)|$

Modal sound contribution: $|p(x)|q(t)$

**Problem:** Must evaluate $p(x)$ for each time sample, mode and object

Standard solution techniques (eg. BEM) too expensive
Approximating Acoustic Transfer
Approximating Acoustic Transfer

• “Precomputed Acoustic Transfer” [James et al. 2006]
  • Approximate $p_i(x)$ with sum of simple source functions
Approximating Acoustic Transfer

- “Precomputed Acoustic Transfer” [James et al. 2006]

  - Approximate $p_i(x)$ with sum of simple source functions

- Problems with this approach:
  - Difficult fitting problem for high frequencies
  - Increasingly costly transfer evaluations with higher frequencies (more sources needed)
Approximating Acoustic Transfer

Exploiting radial structure
Approximating Acoustic Transfer
Exploiting radial structure

Ignore behavior near to the object (eg. within 2-3 bounding sphere radii)
Approximating Acoustic Transfer
Exploiting radial structure

Ignore behavior near to the object (eg. within 2-3 bounding sphere radii)

Look for structure in far field pressure behavior
Approximating Acoustic Transfer
Exploiting radial structure
Approximating Acoustic Transfer
Exploiting radial structure
Approximating Acoustic Transfer
Exploiting radial structure
Approximating Acoustic Transfer
Exploiting radial structure
Approximating Acoustic Transfer

Exploiting radial structure
Approximating Acoustic Transfer
Exploiting radial structure
Approximating Acoustic Transfer

Exploiting radial structure
Approximating Acoustic Transfer

Exploiting radial structure
Approximating Acoustic Transfer
Exploiting radial structure
Approximating Acoustic Transfer

Suppose the pressure field surrounding an object is known:
Approximating Acoustic Transfer

Suppose the pressure field surrounding an object is known:
Approximating Acoustic Transfer

Fix radial direction:

Pre-compute estimate in this direction
Consider an M-term asymptotic expansion

\[ p(x) \sim h_0(kR) \left\{ \Psi_1(\theta, \phi) + \frac{\Psi_2(\theta, \phi)}{kR} + \cdots + \frac{\Psi_M(\theta, \phi)}{(kR)^{M-1}} \right\} \]

\[ -\frac{ie^{-ikR}}{kR} \]
Approximating Acoustic Transfer

Consider an M-term asymptotic expansion

\[
p(x) \sim h_0(kR) \left\{ \Psi_1(\theta, \phi) + \frac{\Psi_2(\theta, \phi)}{kR} + \ldots + \frac{\Psi_M(\theta, \phi)}{(kR)^{M-1}} \right\}
\]

\[-\frac{i e^{-i kR}}{kR}\] Unknowns
Approximating Acoustic Transfer

\[ p(x) \sim h_0(kR) \left\{ \Psi_1(\theta, \phi) + \frac{\Psi_2(\theta, \phi)}{kR} + \ldots + \frac{\Psi_M(\theta, \phi)}{(kR)^{M-1}} \right\} \]
Approximating Acoustic Transfer

\[ p(x) \sim h_0(kR) \left\{ \Psi_1(\theta, \phi) + \frac{\Psi_2(\theta, \phi)}{kR} + \cdots + \frac{\Psi_M(\theta, \phi)}{(kR)^{M-1}} \right\} \]

Precompute pressure samples on concentric spherical shells using fast multipole BEM

[Greengard and Rokhlin 1987; Gumerov and Duraiswami 2005]

(FastBEM implementation [Liu 2009])
Approximating Acoustic Transfer

\[ p(x) \sim h_0(kR) \left\{ \Psi_1(\theta, \phi) + \frac{\Psi_2(\theta, \phi)}{kR} + \cdots + \frac{\Psi_M(\theta, \phi)}{(kR)^{M-1}} \right\} \]

Precompute pressure samples on concentric spherical shells using fast multipole BEM
[Greengard and Rokhlin 1987; Gumerov and Duraiswami 2005]
(FastBEM implementation [Liu 2009])
Approximating Acoustic Transfer

\[ p(x) \sim h_0(kR) \left\{ \Psi_1(\theta, \phi) + \frac{\Psi_2(\theta, \phi)}{kR} + \cdots + \frac{\Psi_M(\theta, \phi)}{(kR)^{M-1}} \right\} \]

Precompute pressure samples on concentric spherical shells using fast multipole BEM
[Greengard and Rokhlin 1987; Gumerov and Duraiswami 2005]
(FastBEM implementation [Liu 2009])

Estimate terms \( \Psi_1(\Theta_l), \ldots, \Psi_M(\Theta_l) \)
Approximating Acoustic Transfer

\[ p(x) \sim h_0(kR) \left\{ \Psi_1(\theta, \phi) + \frac{\Psi_2(\theta, \phi)}{kR} + \cdots + \frac{\Psi_M(\theta, \phi)}{(kR)^{M-1}} \right\} \]

Precompute pressure samples on concentric spherical shells using fast multipole BEM
[Greengard and Rokhlin 1987; Gumerov and Duraiswami 2005]
(FastBEM implementation [Liu 2009])

Estimate terms \( \Psi_1(\Theta_l), \ldots, \Psi_M(\Theta_l) \)

\[ \sum_{j=1}^{M} \frac{h_0(kR_i)}{(kR_i)^{j-1}} \Psi_j(\Theta_l) = p(R_i, \Theta_l) \]

\( \iff \sum_{j=1}^{M} A_{ij} \Psi_{jl} = p_{il} \)
Approximating Acoustic Transfer

\[ p(x) \sim h_0(kR) \left\{ \Psi_1(\theta, \phi) + \frac{\Psi_2(\theta, \phi)}{kR} + \cdots + \frac{\Psi_M(\theta, \phi)}{(kR)^{M-1}} \right\} \]

Precompute pressure samples on concentric spherical shells using fast multipole BEM
[Greengard and Rokhlin 1987; Gumerov and Duraiswami 2005]
(FastBEM implementation [Liu 2009])

Estimate terms \( \Psi_1(\Theta_l), \ldots \Psi_M(\Theta_l) \)

\[ \sum_{j=1}^{M} \frac{h_0(kR_i)}{(kR_i)^{j-1}} \Psi_j(\Theta_l) = p(R_i, \Theta_l) \]

\( \iff \)

\[ \sum_{j=1}^{M} A_{ij} \Psi_{jl} = p_{il} \]

Unknowns

Precomputed pressures
Approximating Acoustic Transfer

\[ p(x) \sim h_0(kR) \left\{ \Psi_1(\theta, \phi) + \frac{\Psi_2(\theta, \phi)}{kR} + \cdots + \frac{\Psi_M(\theta, \phi)}{(kR)^{M-1}} \right\} \]

\[ \frac{ie^{-ikR}}{kR} \]
Approximating Acoustic Transfer

\[ p(x) \sim \left\{ \frac{ie^{-ikR}}{kR} + \frac{1}{kR} + \cdots + \frac{1}{(kR)^{M-1}} \right\} \]
Approximating Acoustic Transfer

$p(x) \sim \frac{ie^{-ikR}}{kR} + \frac{kR}{(kR)^{M-1}}$

Far Field Acoustic Transfer (FFAT) Maps

- Low-error transfer, e.g., $M=4$
- $O(1)$ transfer evaluation cost
Results
<table>
<thead>
<tr>
<th>Model</th>
<th>Dimensions</th>
<th># of triangles</th>
<th># of modes</th>
<th>Freq. range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trash can</td>
<td>0.75m tall</td>
<td>78k triangles</td>
<td>200 modes</td>
<td>0.071-4.43 kHz</td>
</tr>
<tr>
<td>Cymbal</td>
<td>0.50m diameter</td>
<td>62k triangles</td>
<td>500 modes</td>
<td>0.061-9.94 kHz</td>
</tr>
<tr>
<td>Water bottle</td>
<td>0.46m tall</td>
<td>29k triangles</td>
<td>300 modes</td>
<td>0.116-3.59 kHz</td>
</tr>
<tr>
<td>Recycling bin</td>
<td>0.61m wide</td>
<td>110k triangles</td>
<td>300 modes</td>
<td>0.062-2.21 kHz</td>
</tr>
<tr>
<td>Trash can lid</td>
<td>0.55m diameter</td>
<td>34k triangles</td>
<td>200 modes</td>
<td>0.112-6.79 kHz</td>
</tr>
</tbody>
</table>
Results

• 500 modes
• 1500 cubature features (10.7% error)
• Timestep: \(\frac{1}{88200}\)s
• Simulation cost: 3900s per second of audio
Results

• 500 modes
• 1500 cubature features (10.7% error)
• Timestep: \(\frac{1}{88200}\)s
• Simulation cost: 3900s per second of audio
Results

• 300 modes
• 1200 cubature features (15.7% error)
• Timestep: $\frac{1}{44100}$ s
• Simulation cost: 1224 s per second of audio
Results

- 300 modes
- 1200 cubature features (15.7% error)
- Timestep: \( \frac{1}{44100} \)s
- Simulation cost: 1224s per second of audio
Results

• 200 modes
• 800 cubature features (11.5% error)
• Timestep: \( \frac{1}{44100} \) seconds
• Simulation cost: 624 seconds per second of audio
Results

• 200 modes
• 800 cubature features (11.5% error)
• Timestep: (1 / 44100)s
• Simulation cost: 624s per second of audio
Results

• 200 modes
• 800 cubature features (10.3% error)
• Timestep: (1 / 44100)s
• Simulation cost: 714s per second of audio
Results

- 200 modes
- 800 cubature features (10.3% error)
- Timestep: $(1 / 44100)$s
- Simulation cost: 714s per second of audio
Results

- 300 modes
- 900 cubature features (10.7% error)
- Timestep: \( \frac{1}{44100} \) s
- Simulation cost: 1026 s per second of audio
Results

- 300 modes
- 900 cubature features (10.7% error)
- Timestep: \((1 / 44100)\)s
- Simulation cost: 1026s per second of audio
Comparisons
Comparisons: Linear vs. Nonlinear
Comparisons: Linear vs. Nonlinear

1. Nonlinear dynamics + Transfer
   (“Harmonic Shells”) (~1.5-3h per 10s of audio)
Comparisons: Linear vs. Nonlinear

1. Nonlinear dynamics + Transfer
   (“Harmonic Shells”) (~1.5-3h per 10s of audio)

2. Linear dynamics + Transfer
   (audio can be computed in real-time)
Comparisons: Linear vs. Nonlinear

1. Nonlinear dynamics + Transfer
   ("Harmonic Shells") (~1.5-3h per 10s of audio)

2. Linear dynamics + Transfer
   (audio can be computed in real-time)

3. Linear dynamics + Monopole
Comparisons: Linear vs. Nonlinear

1. Nonlinear dynamics + Transfer
   “Harmonic Shells”

2. Linear dynamics + Transfer

3. Linear dynamics + Monopole
**Comparisons:** Linear vs. Nonlinear

1. Nonlinear dynamics + Transfer
   “Harmonic Shells”

2. Linear dynamics + Transfer

3. Linear dynamics + Monopole
More Results
More Results
Limitations and Future Work
Limitations and Future Work

• All-frequency sound synthesis
Limitations and Future Work

- All-frequency sound synthesis
- Frequency range limited to ~4-5 kHz for moderately sized objects
Limitations and Future Work

- All-frequency sound synthesis
- Frequency range limited to ~4-5 kHz for moderately sized objects
- $O(r^2)$ does not scale to thousands of modes
Limitations and Future Work

- All-frequency sound synthesis
- Frequency range limited to ~4-5 kHz for moderately sized objects
- $O(r^2)$ does not scale to thousands of modes
- FFAT Map storage
Limitations and Future Work

- All-frequency sound synthesis
- Frequency range limited to ~4-5 kHz for moderately sized objects
- $O(r^2)$ does not scale to thousands of modes
- FFAT Map storage
- Typically 50-100MB for single term map (500MB for cymbal)
Limitations and Future Work

- All-frequency sound synthesis
- Frequency range limited to \(~4-5\) kHz for moderately sized objects
- \(O(r^2)\) does not scale to thousands of modes
- FFAT Map storage
  - Typically 50-100MB for single term map (500MB for cymbal)
  - Better sampling of angular space (not all directions as complex)
Limitations and Future Work
Limitations and Future Work

• Nonlinear vibrations but radiation model assumes linear vibrations
Limitations and Future Work

- Nonlinear vibrations but radiation model assumes linear vibrations
- Radiation model which takes into account mode coupling, etc.
Conclusions
Conclusions

• Practical nonlinear modal sound synthesis for objects with hundreds of modes
  • $O(r^2)$ cost per timestep
  • Larger timesteps
Conclusions

- Practical nonlinear modal sound synthesis for objects with hundreds of modes
  - $O(r^2)$ cost per timestep
  - Larger timesteps
- Richer sounds than linear modal models
Conclusions

- Practical nonlinear modal sound synthesis for objects with hundreds of modes
  - O($r^2$) cost per timestep
  - Larger timesteps
- Richer sounds than linear modal models
- Data-driven technique for O(1) computation of pressure contribution from each mode
  - O($r$) for all $r$ modes
Acknowlegdements

• Anonymous Reviewers
• The National Science Foundation:
  • CAREER-0430528, EMT-CompBio-0621999, HCC-0905506
• NIH (NIBIB/NIH R01EB006615)
• Alfred P. Sloan Foundation
• Pixar
• Intel
• Advanced CAE Research, LLC (FastBEM Acoustics)
• Autodesk
• NVIDIA