

Gamefest

MICROSOFT GAME TECHNOLOGY CONFERENCE 2011



Microsoft

Sound Synthesis in *Crackdown 2*

Nikunj Raghuvanshi

Researcher, Microsoft Research


Realism

- Graphics has made leaps and bounds
- Audio far behind: great potential

Sound in Games Today

- Recorded samples
- Realism requires tedious labor
- Reduces time for creativity

Physically-Based Sound

- Physics  Perception
automation "Physically-based" control
- Use auditory perception → compact, parametric models
 - Realistic-sounding starting point
 - Artistic control
 - Quality-efficiency trade-off
- More time for creative work



Overview

- Physics of sound
- Sound synthesis in *Crackdown 2*
 - Collaborators:
 - Brandon Lloyd, Microsoft Research
 - Nikunj Raghuvanshi, Microsoft Research
 - Naga K. Govindaraju , Microsoft Research
 - Guy Whitmore, Microsoft Game Studio
 - Kristofor Mellroth, Microsoft Game Studio

Sound Synthesis

- Collisions lead to surface vibrations
- Vibrations create pressure waves in air
- Pressure waves perceived as sound

Vibration



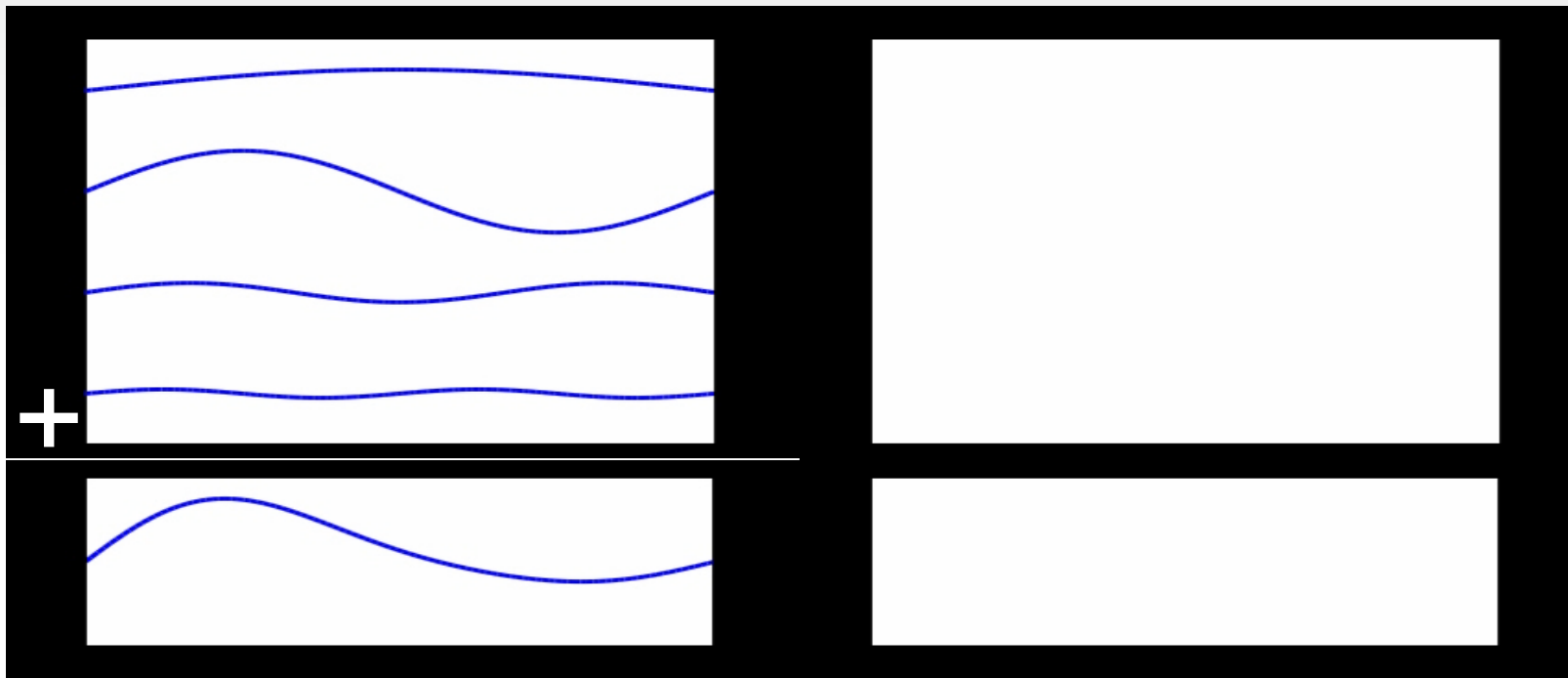
Pressure (sound) waves



Physical Model: Modal Synthesis

Modes

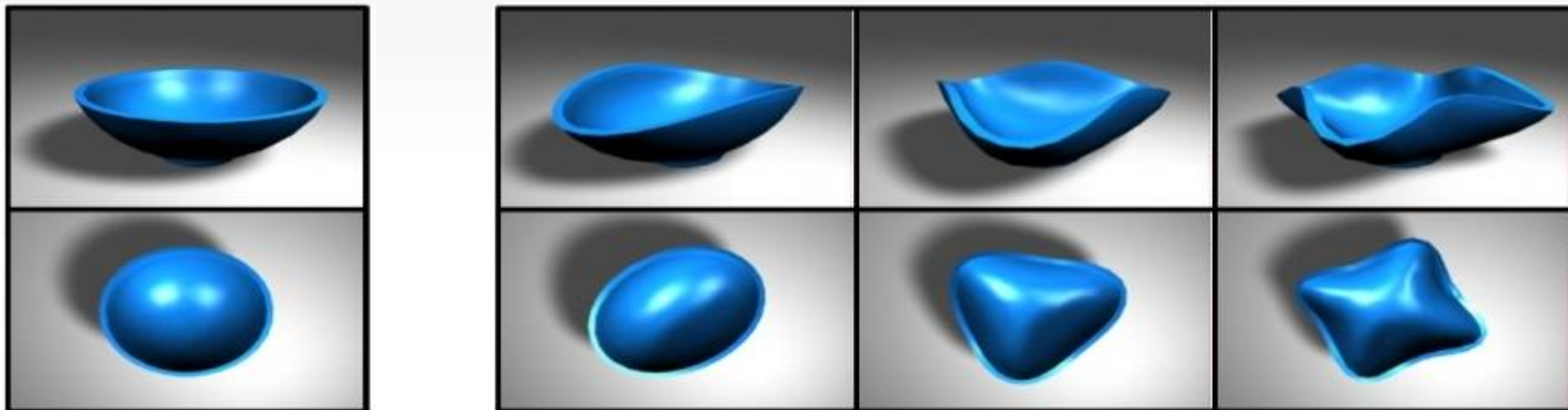
Amplitude



String

Waveform

Modal Synthesis



O'Brien et al. 2002

Forward Modal Synthesis

- geometric shape + material \rightarrow Sound
- Previous work in graphics:
 - van den Doel et al. 2001
 - O'Brien et al. 2002
 - Raghuvanshi and Lin 2007
 - Bonneel et al. 2008
- All assume ideal exponential decay

Forward Modal Synthesis: Pros and Cons

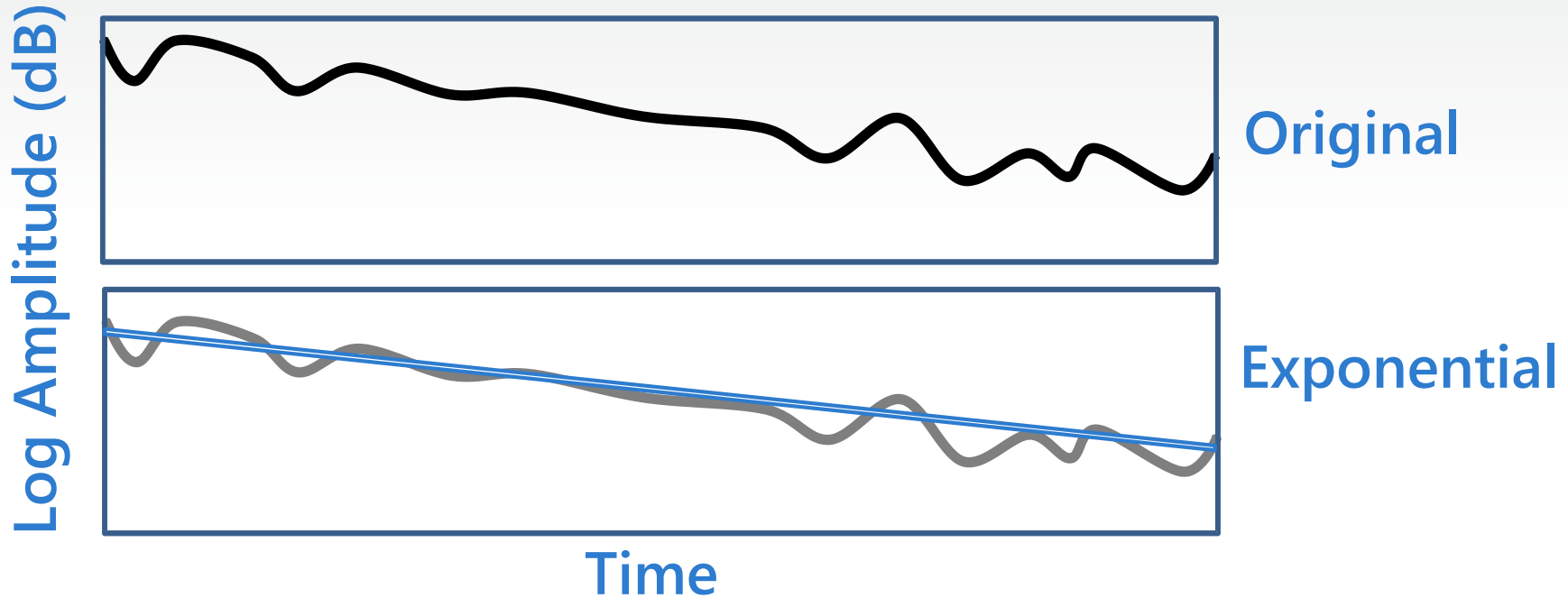
Pros:

- No need for recordings
- Very simple to implement

Cons:

- Ideal exponential decay doesn't sound realistic
- Most real-life sounds' transients have a non-modal component
 - Synthesized sounds are **too clean**

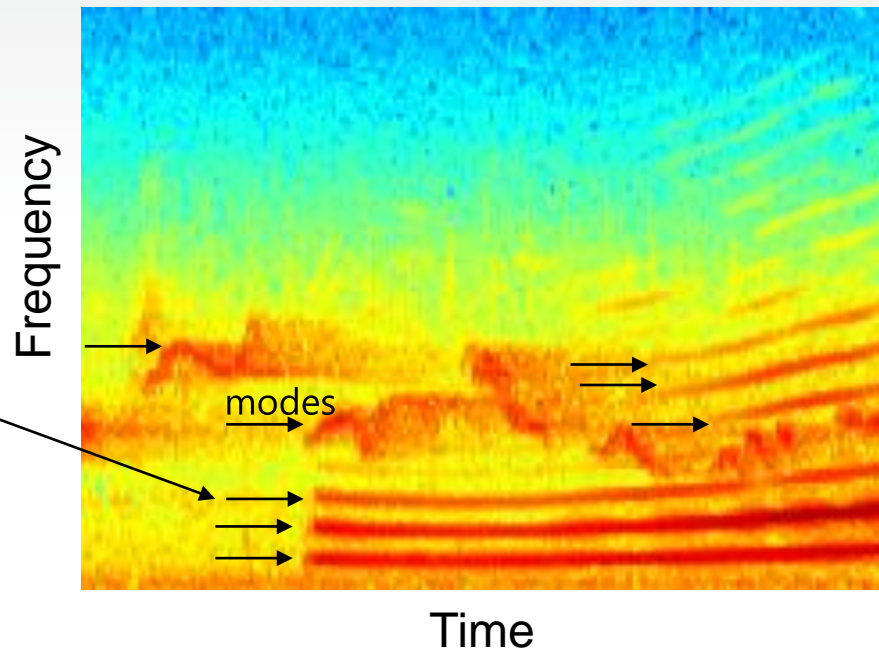
Amplitude Envelopes per Mode



Spectral Modeling Synthesis

- Model spectral content rather than physics
- Sinusoidal partials (modes)
 - Amplitude envelope + frequency
- Noise
 - Spectral envelope
- Good: Quite general and accurate
- Bad: Expensive to compute

Spectrogram



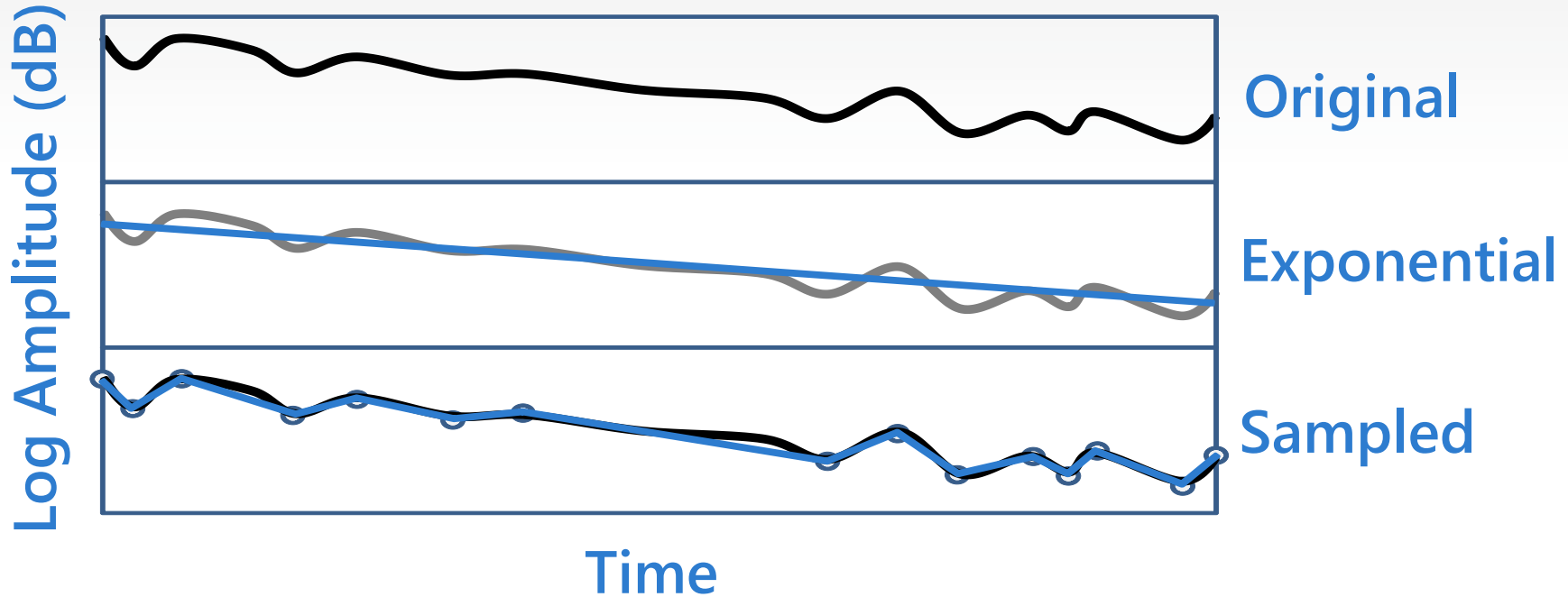
Implementation

- Xbox 360: 3 PowerPC cores @ 3.0 GHz
- Bounded CPU usage (10% of a hardware thread)
- Limited memory
 - ~25 MB for all audio
 - ~2 MB for all physics sound data (including nonprocedural)
- Easy-to-use internal tools
 - UI tool: generate procedural models, control efficiency vs. quality
 - Custom plug-ins for Audiokinetic's Wwise

Our Approach: Middle Ground

- Arbitrarily decaying sinusoids + residual waveform
- Much more realistic-sounding than forward synthesis
- Faster to compute than spectral modeling synthesis

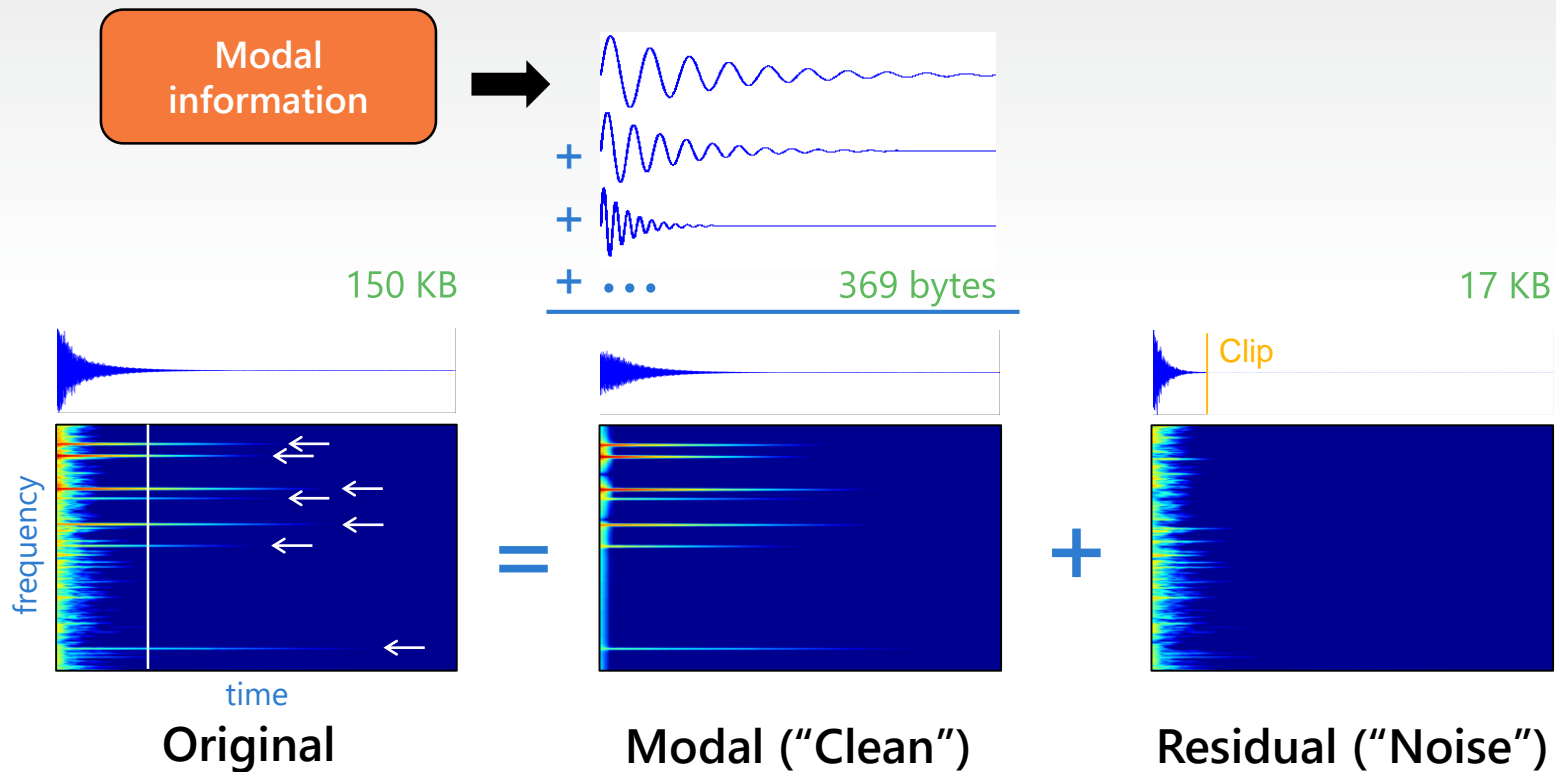
Amplitude Envelopes per Mode



Physically-Based Impact Sounds

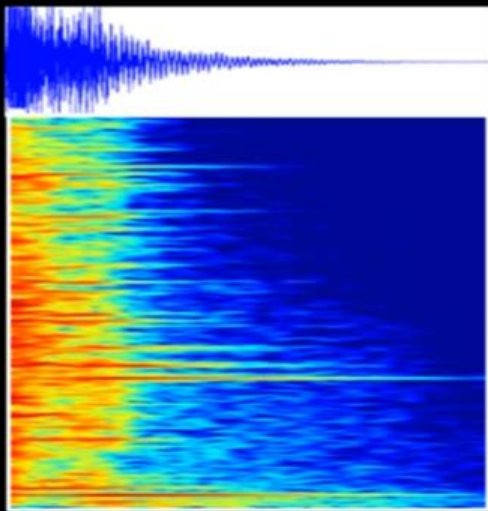
- Most games: Variation achieved through multiple clips (3 to 5) for each sound
 - Inflexible, memory-intensive
- Our approach:
 - Extract compact physical model from a single clip
 - Procedurally generate sound from model (costs compute)
 - **Infinite variation and artistic control**

Our Approach: Details

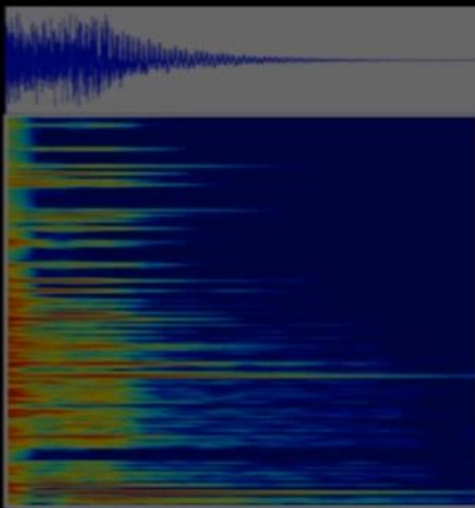


Amplitude Envelopes vs. Exponential (modal only)

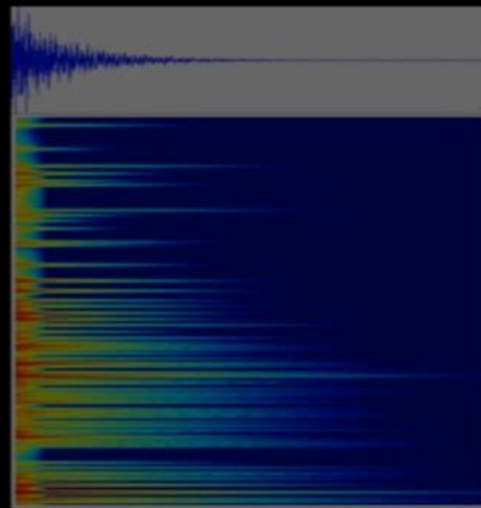
Armor Plate



Original



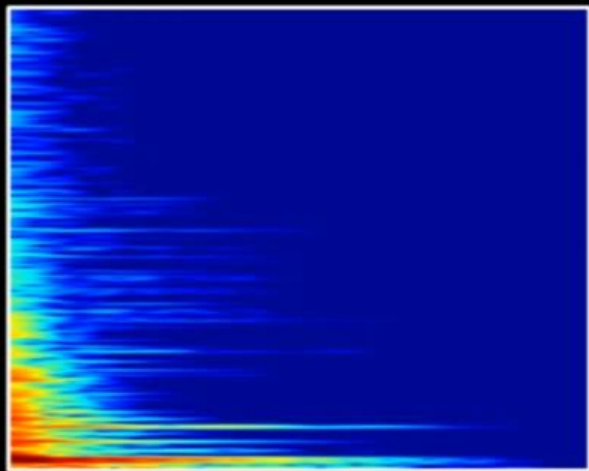
Ours



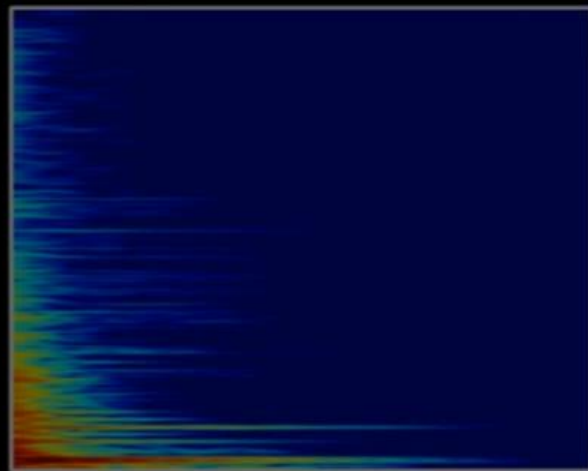
Exponential

Original vs. Synthesized Sound

Plastic Barrel



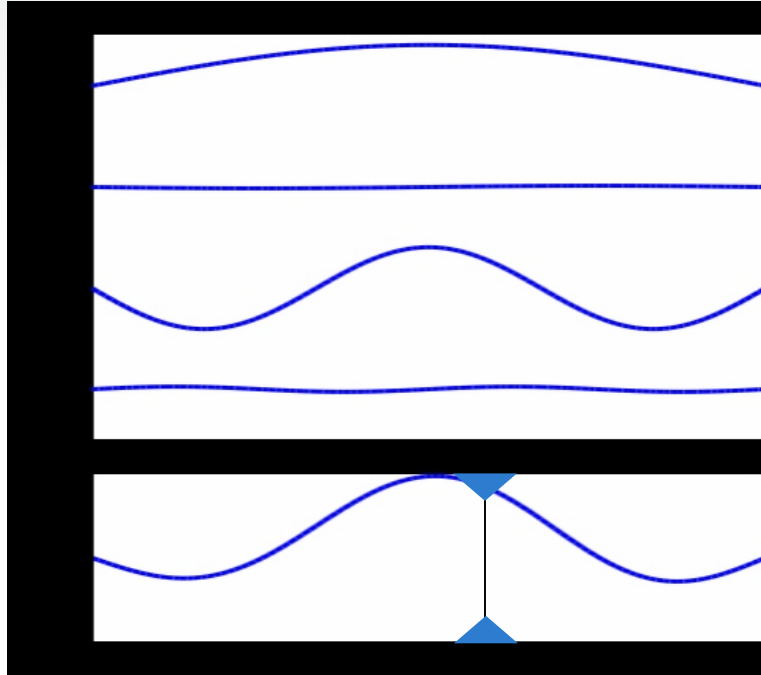
Original



Modal + Residual
(23 modes)

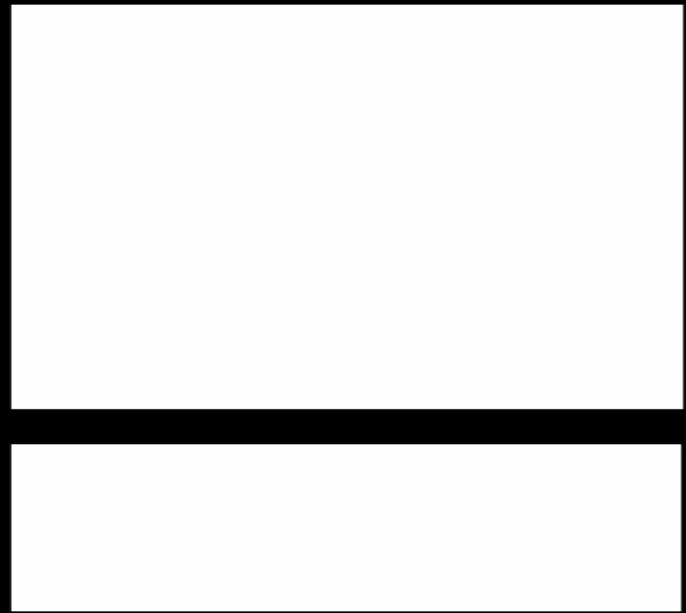
Variation

Modes



String

Amplitude

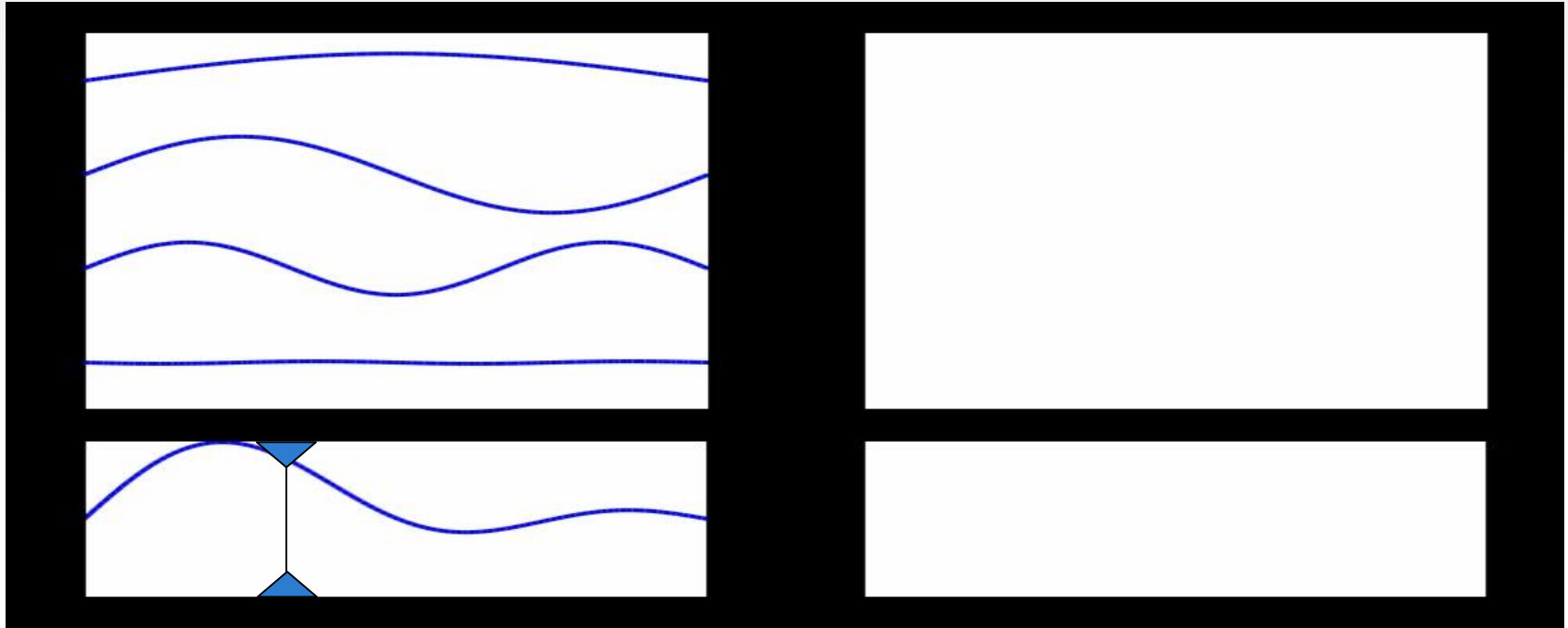


Waveform

Variation

Modes

Amplitude

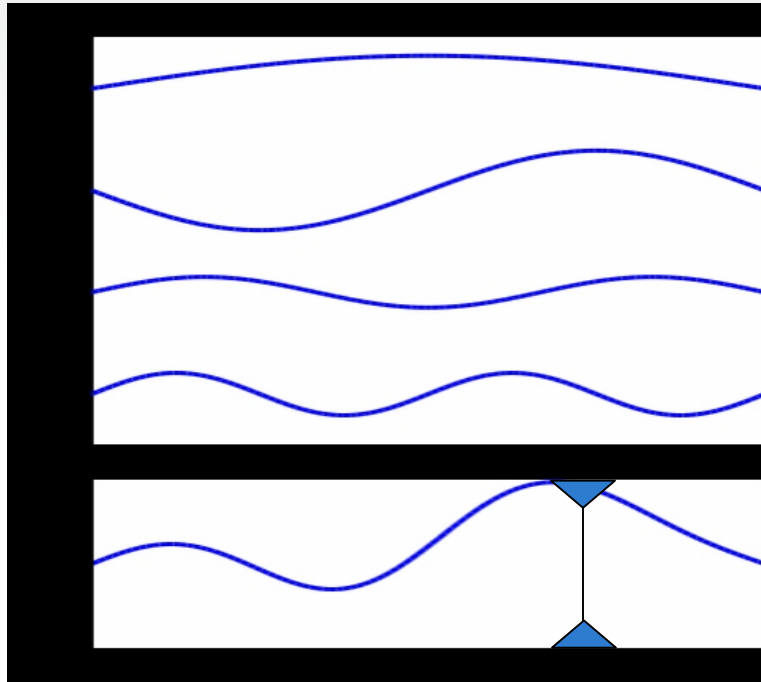


String

Waveform

Variation

Modes



String

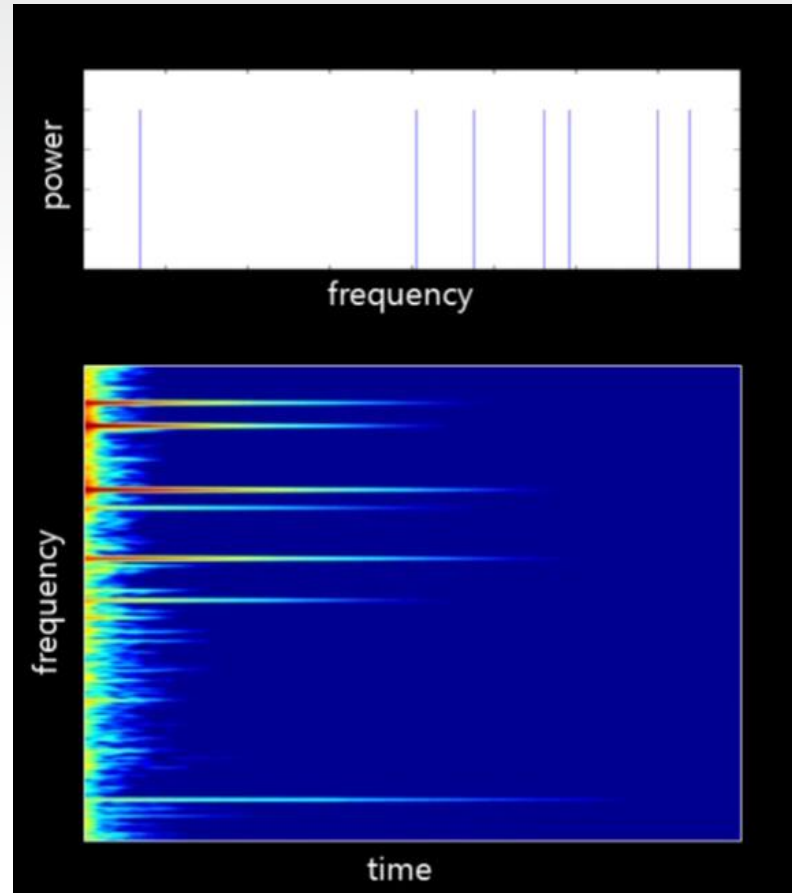
Amplitude



Waveform

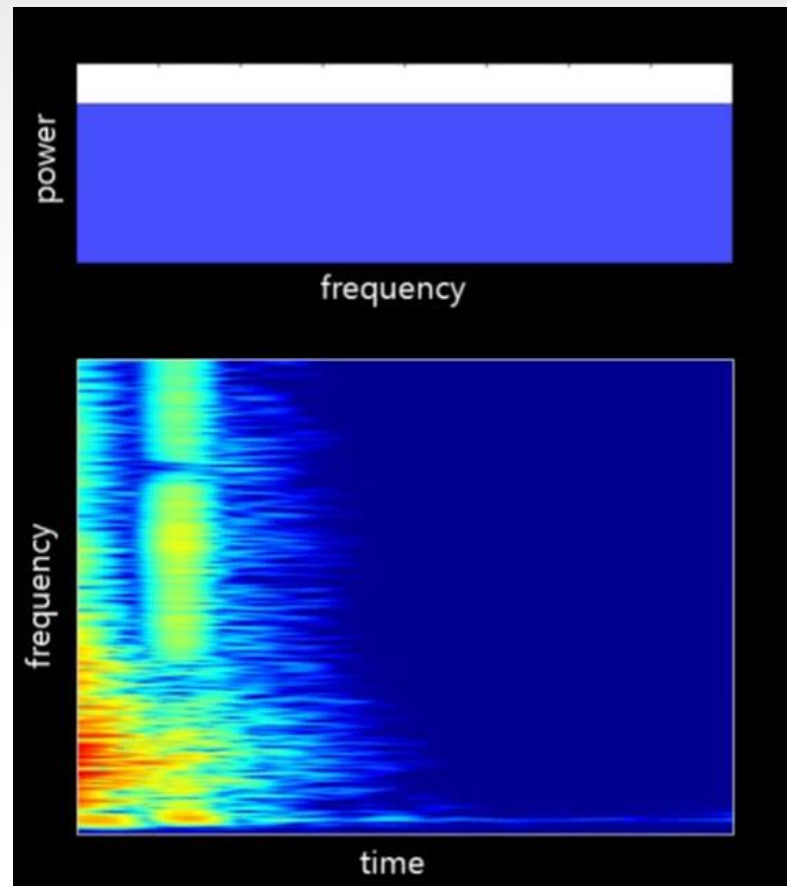
Variation for Modal Sounds

- Exact variation: mode shapes to determine mode amplitudes
 - Requires a lot of memory
- Simple yet effective: Randomize mode amplitudes
 - Degree of variation: artistic control



Variation for Non-Modal Sounds

- Attenuate regions of spectrum
- Cascade several dip filters (biquad)
- Randomize parameters
 - Location (center frequency)
 - Width (Q factor)
 - Depth (gain)
- Provides variation from a single clip



Variation Examples

Name

Variation

Plastic Barrel

Modal

Brass Bell

Modal

Road Cone

Modal

Wooden Stick

Modal

Door

Filter

Ground

Filter

Ice

Filter

Plastic Ball

Filter

Modal Synthesis

- single input clip
- no variation

Crackdown 2

Modal Synthesis

- single input clip
- with variation

Crackdown 2

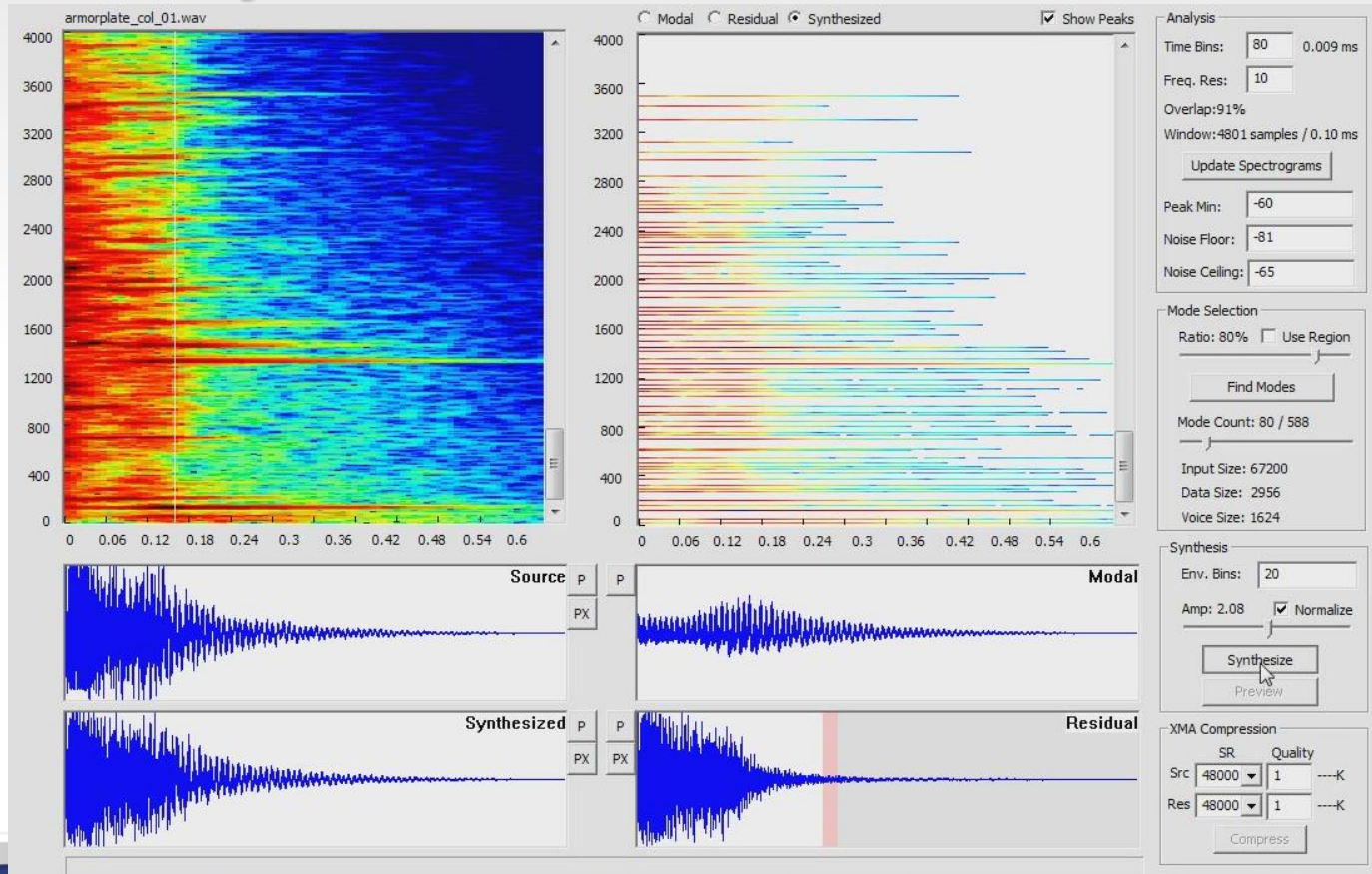
Variation Filter

- single clip

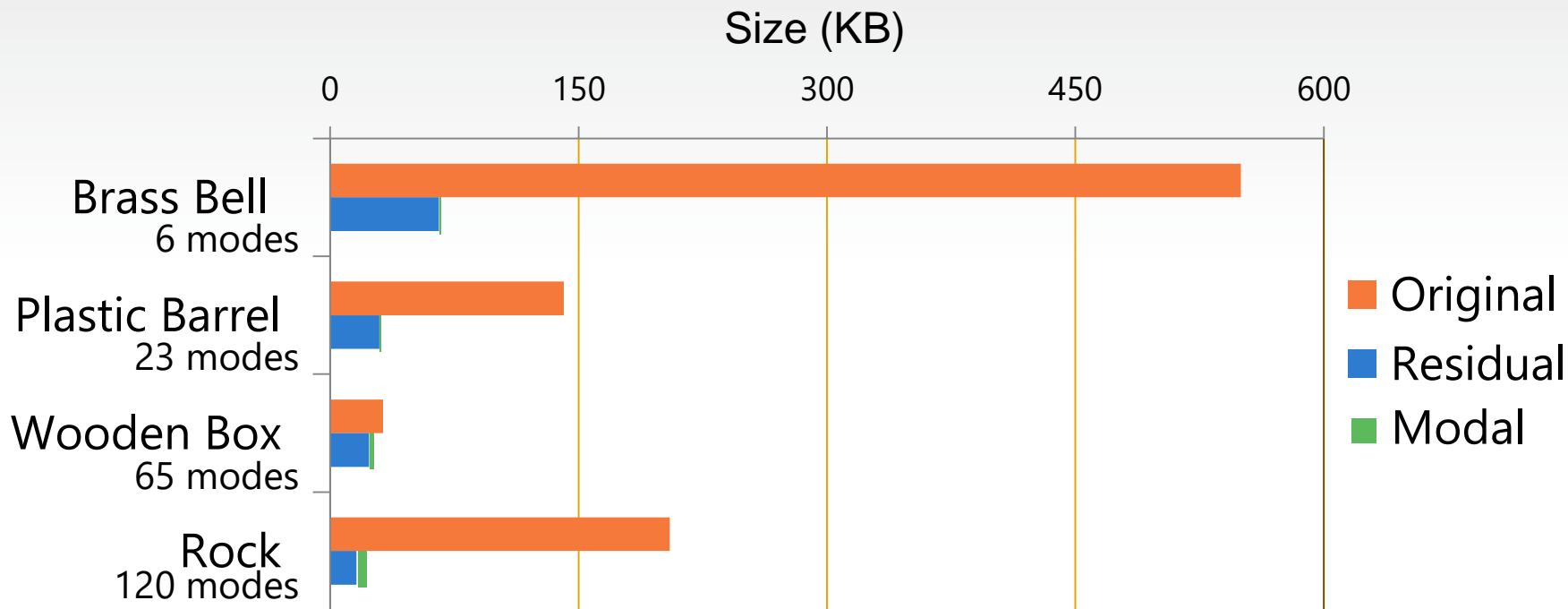
Single clip
(no variation)

Crackdown 2

Demo: Synthesis Tool

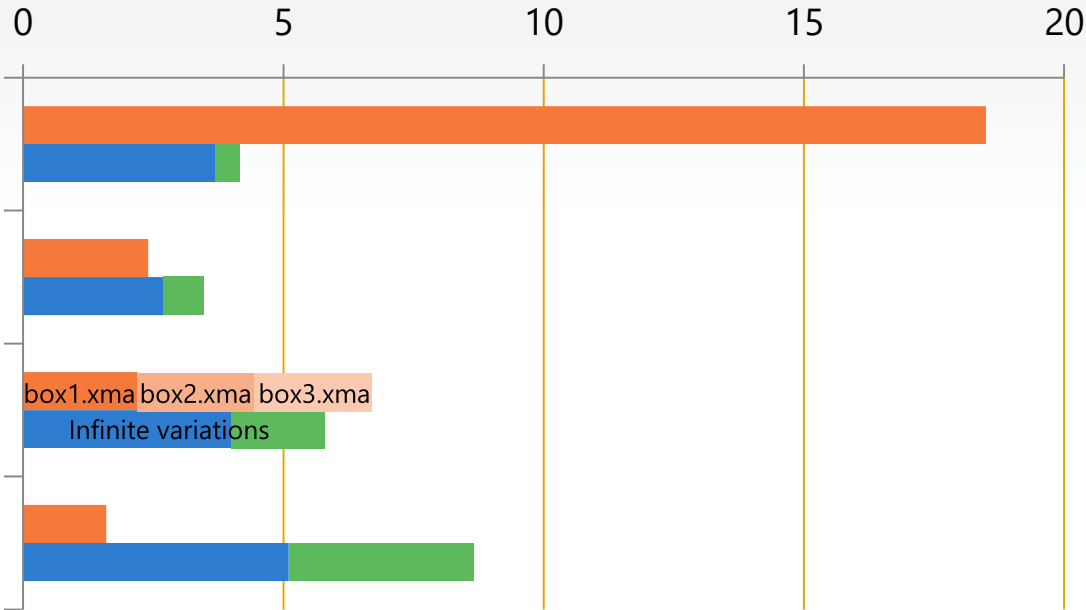


Memory Usage (PCM)



Memory Usage (XMA)

Size (KB)



- Original
- Residual
- Modal

Compressed with XMA2 at maximum setting

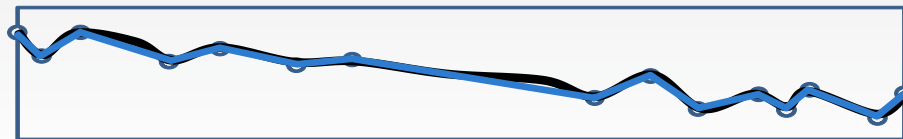
Optimizations on the Xbox 360

- Compact decay envelopes
- Runtime quality scaling
- Multirate mixing
- SIMD

Compact Decay Envelopes

- Simplification

- Noise threshold
- Adaptive sampling
 - Same sample points for all envelopes



- Quantization

- Store amplitudes in dB: 8 bits is sufficient

Runtime Quality Scaling

- User-specified, fixed-mode budget
 - Fixed memory overhead
 - Fixed CPU usage
- Prioritize modes by how much they contribute
- Remove low-priority modes

Quality Scaling: Reference

- high mode quota (5000)

Crackdown 2

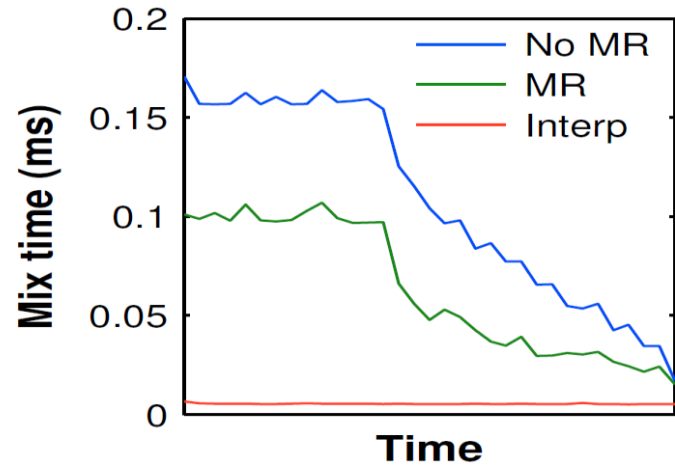
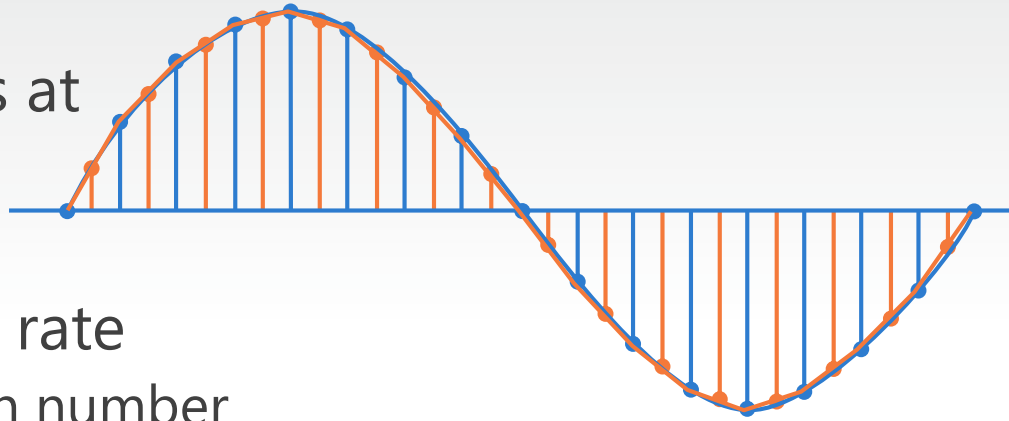
Quality Scaling

- low mode quota (1000)
- 5x savings in compute

Crackdown 2

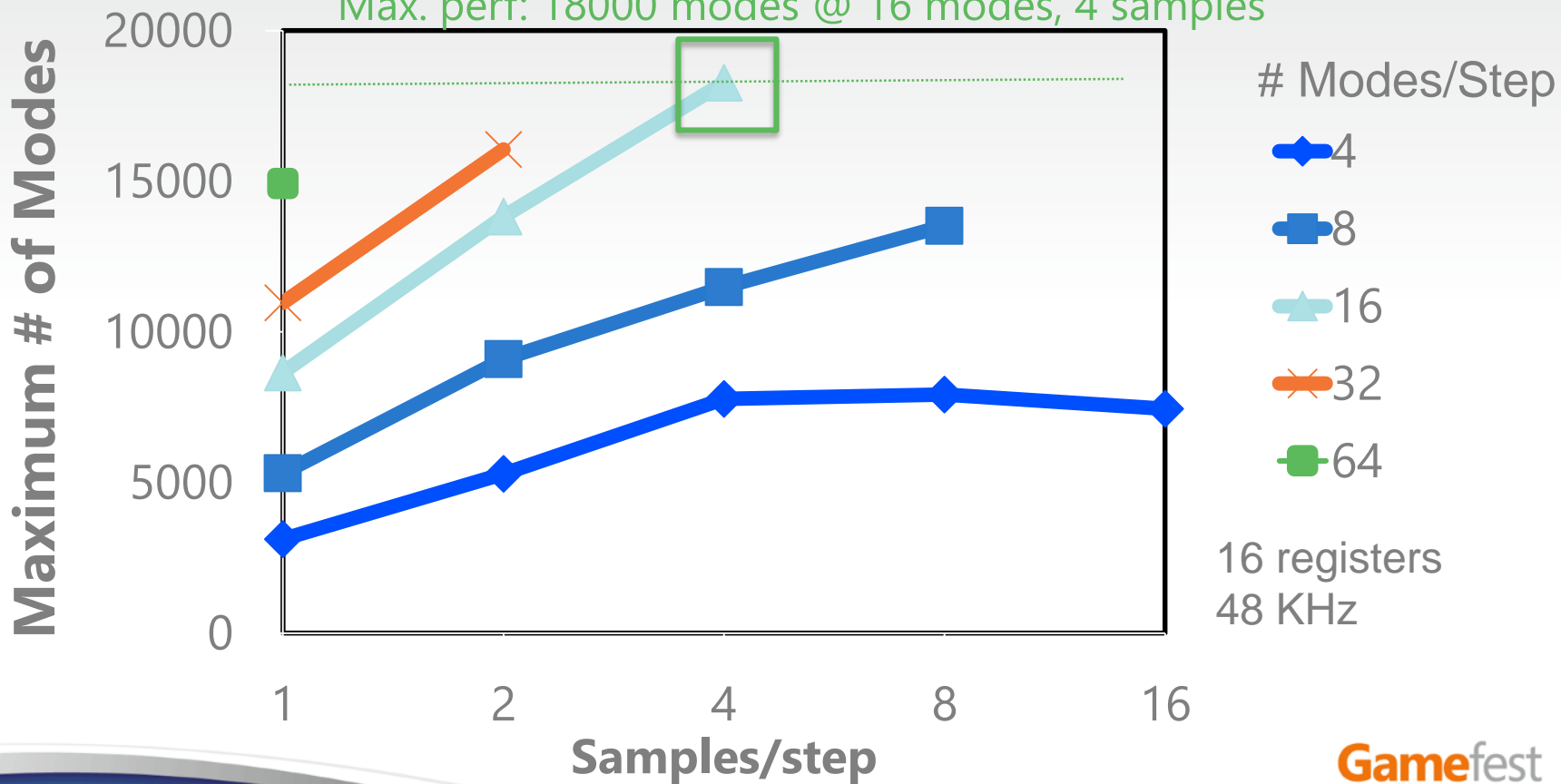
Multirate Mixing

- Mix low-frequency modes at lower rate
- Interpolate to next higher rate
 - Fixed overhead depends on number of bands (usually 4 bands)
 - 8-tap filter
- ~30% to 50% reduction in compute



VMX128

Max. perf: 18000 modes @ 16 modes, 4 samples



Further Work

- Mode evaluation currently in time domain
 - Frequency domain would be much faster
 - Hard to integrate with current audio middleware
- Use mesh or texture map for impact-strike variation
 - Edge strikes tend to act as high-pass filters.

Summary

- Sound synthesis for high quality impact sounds with variation in *Crackdown 2*
 - More compute
 - Less memory
 - Infinite variety
- Powerful physically-based representation
 - Rolling/sliding could potentially be modeled
 - Field is open for your experimentation!
- Physically-based techniques are a powerful way to deal with game sounds
 - Automation for the tedious part of audio design

Thank you!! Questions?

- <http://research.microsoft.com/people/nikunj/>
- Papers, demos, *etc.*
- nikunj@microsoft.com

Please Remember Your Feedback For All Talks

Physically-Based Sound: Scenarios

- Grenade rolling on wooden floor
- Simulated footsteps
- Rustling leaves
- ...

Conclusion

- Physically-based impact sounds in *Crackdown 2*
 - Infinite variation
 - Saves memory
- Physically-based techniques are a powerful way to deal with game sounds
 - Automation for the tedious part of audio design