Perceptual Scheduling in Real-time Music and Audio Applications

PhD Dissertation

Amar Chaudhary

Lawrence A. Rowe and David Wessel, Co-Chairs

University of California at Berkeley

April 18, 2001

Online Audio Examples

http://www.ptank.com/phdtalk/sounds.html

Supplemental audio material
for online PDF and PowerPoint Slides
(Arranged by slide #)

Collaboration with CNMAT

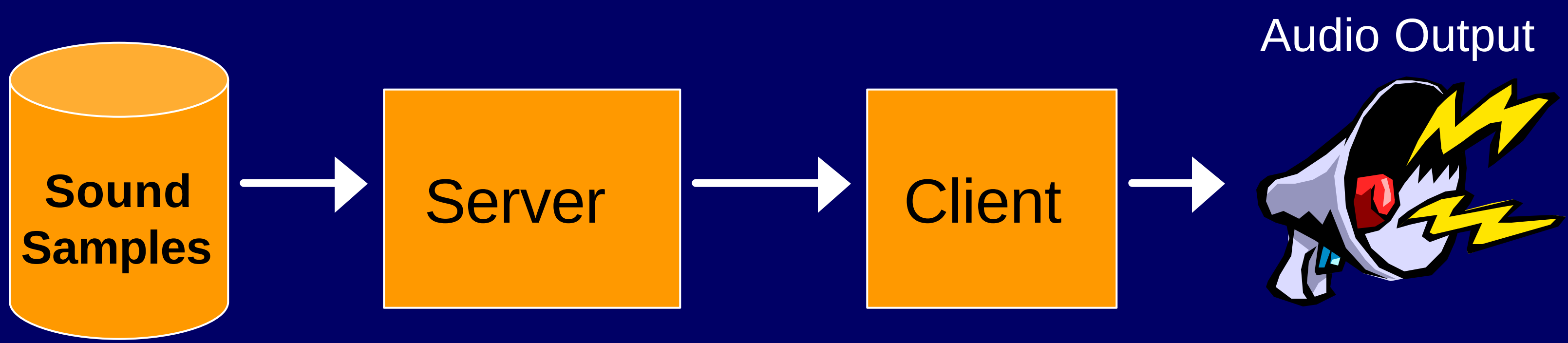
- Center for New Music and Audio Technologies
- Interdisciplinary
  - Music
  - EECS
  - Psychology
- Both research and artistic activities

Outline

- Overview of sound synthesis
  - Synthesis Servers
  - Additive synthesis and resonance modeling
- Computational Issues and Problems
- Perceptual Scheduling
- Computational Reduction Strategies
- Evaluation on Musical Examples
- Conclusions & Future Work

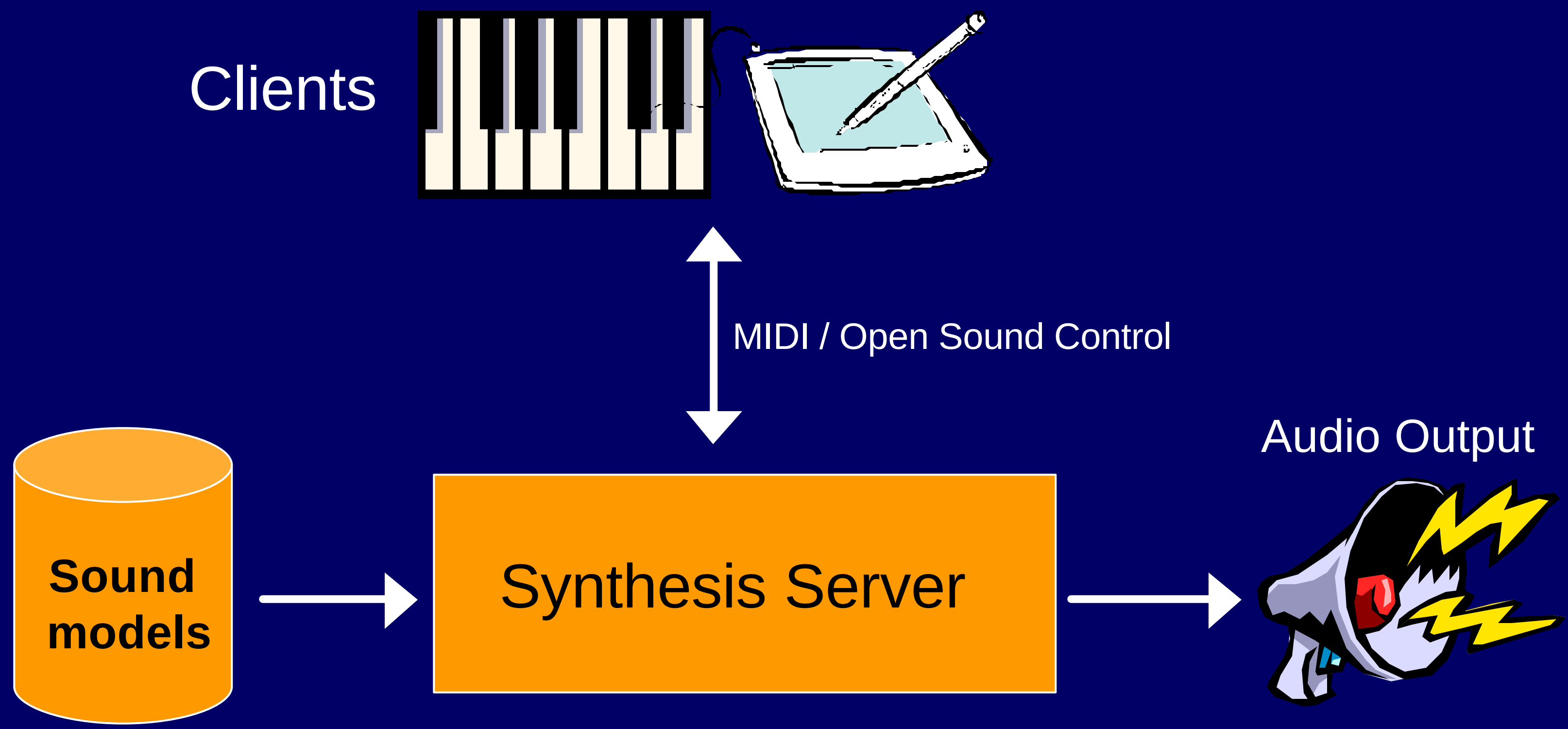
Playing Music on Computers

- Streaming Audio Servers
  - Internet Radio
  - Napster
  - Playing audio CDs on your computer



- All the system you need…if all you play is the stereo!

Synthesis Servers



Independent of hardware, OS and transport

What is a “Sound Model?”

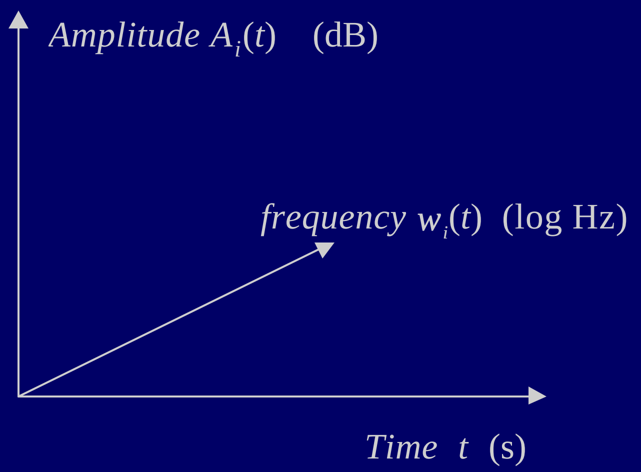
- Waveform representation of sound:
  - a sequence of samples *y*(n)
- *Synthesize* sound from parametric models
  - Example: a pure tone (i.e., “sine wave”)

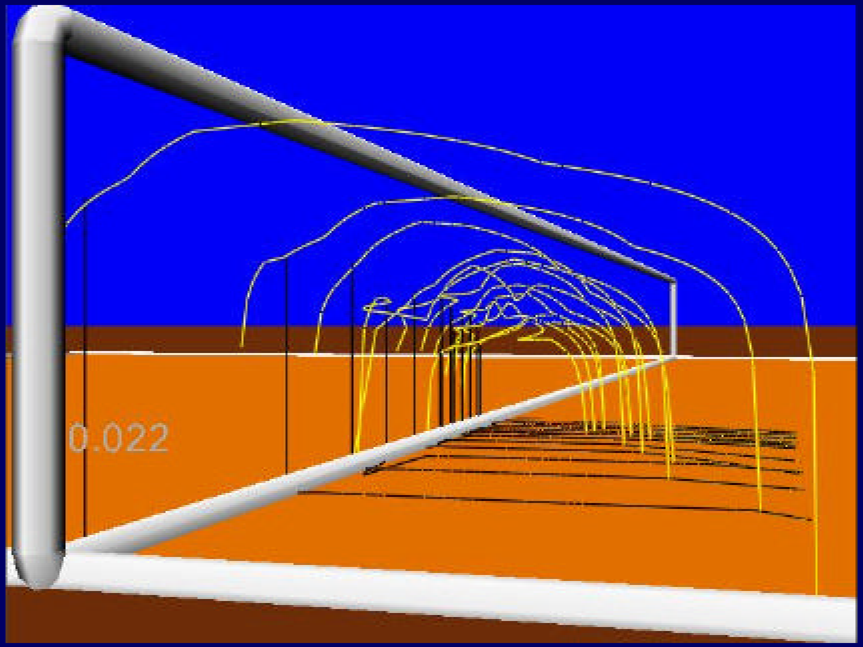
    $y(n) = A(n) \sin (f(n) + f(n))$
- Advantages of a sound model
  - Mutability (i.e., any pitch or amplitude)
  - Compression
- Example: A sine wave synthesis server

Sinusoidal Models

- Sum of time-varying sinusoids:

$$x(t) = \sum_{i=1}^{N} A_i(t)\cos\left(w_i(t)t + f_i(t)\right)$$





Phase is *not* shown

Sinusoidal Models

- Sum of time-varying sinusoids:

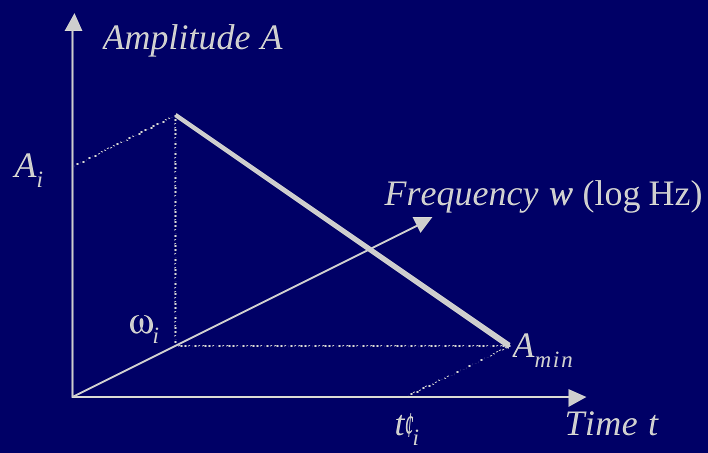
$$x(t) = \sum_{i=1}^{N} A_i(t) \cos(w_i(t)t + f_i(t))$$

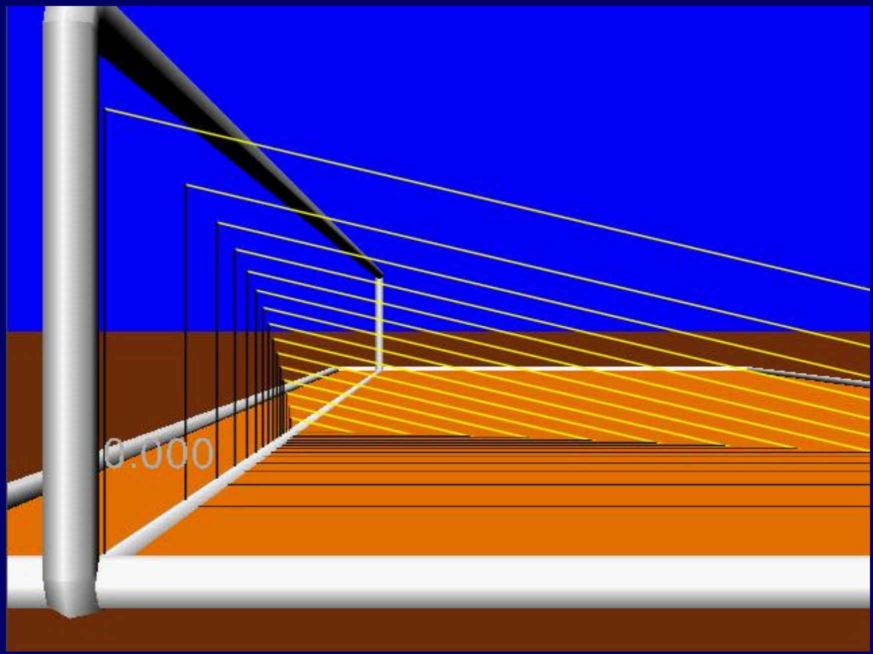
- Advantages:
  - Independent control of time and frequency
  - Control of timbre
- Disadvantages:
  - Large and expensive to compute

Resonance Models

- Exponentially-decaying sinusoids:

$$x(t) = \sum_{i=1}^{N} A_i e^{-p k_i t} \cos(w_i t + f_i)$$





Parameters are *not* time-varying

Resonance Models

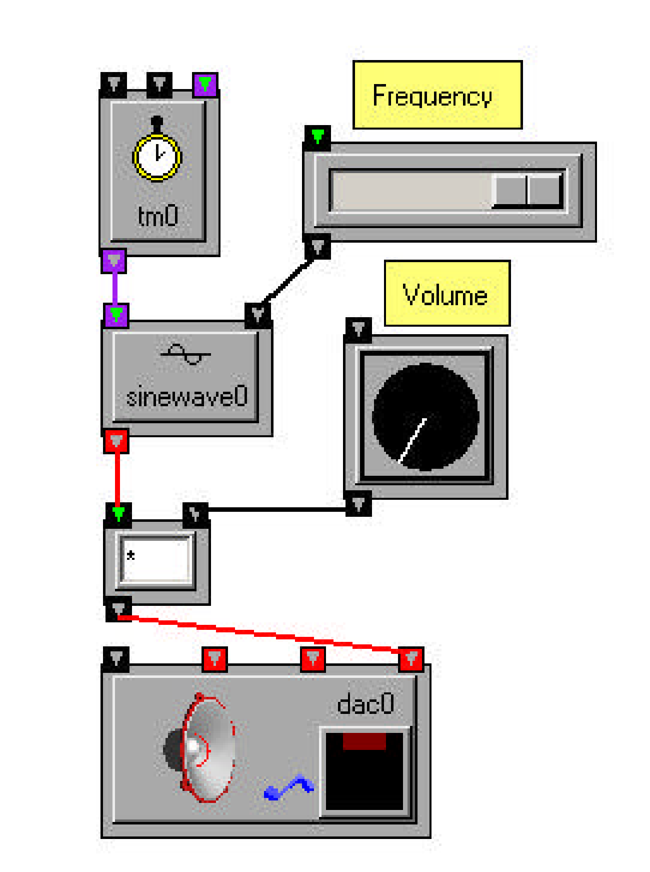
- Exponentially-decaying sinusoids:

$$x(t) = \sum_{i=1}^{N} A_i e^{-p k_i t} \cos(w_i t + f_i)$$

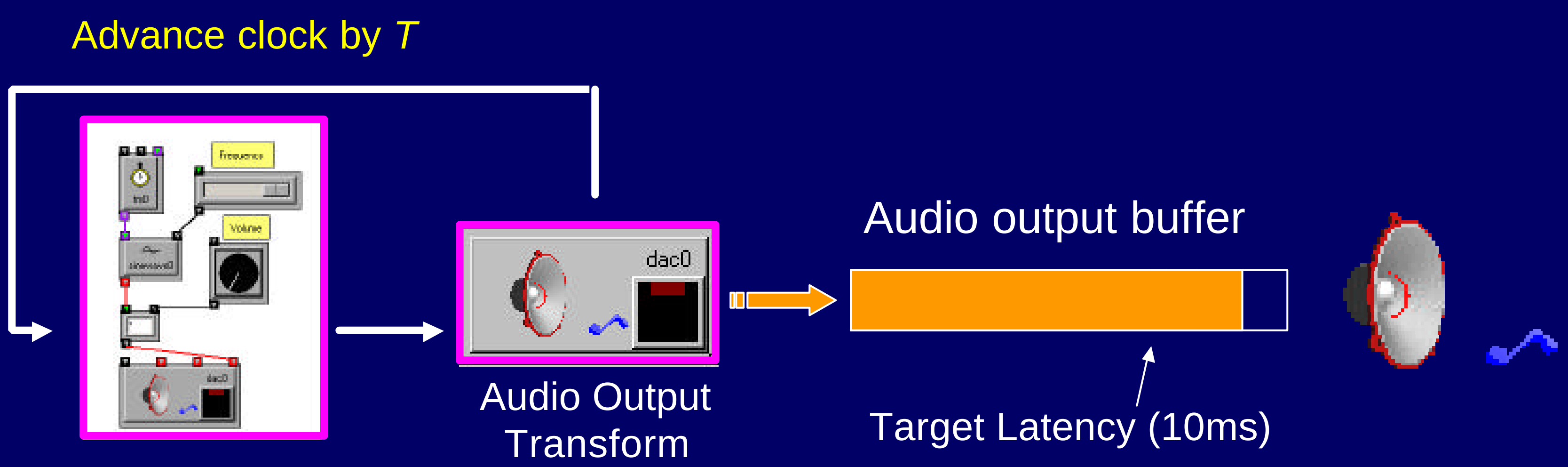
- Advantages:
  - Independent control of time and frequency
  - Perceptually meaningful control of timbre
  - Small (a few hundred numbers for entire sound)
- Disadvantages:
  - Expensive to compute

Open Sound World

- Language for synthesis servers
- Visual dataflow language
- Incremental development
- *Transforms* are connected to form *patches*
- Modern type system
- Nested patches
- Hierarchical name space
- Extensible set of transforms and data types
- Profiling Features

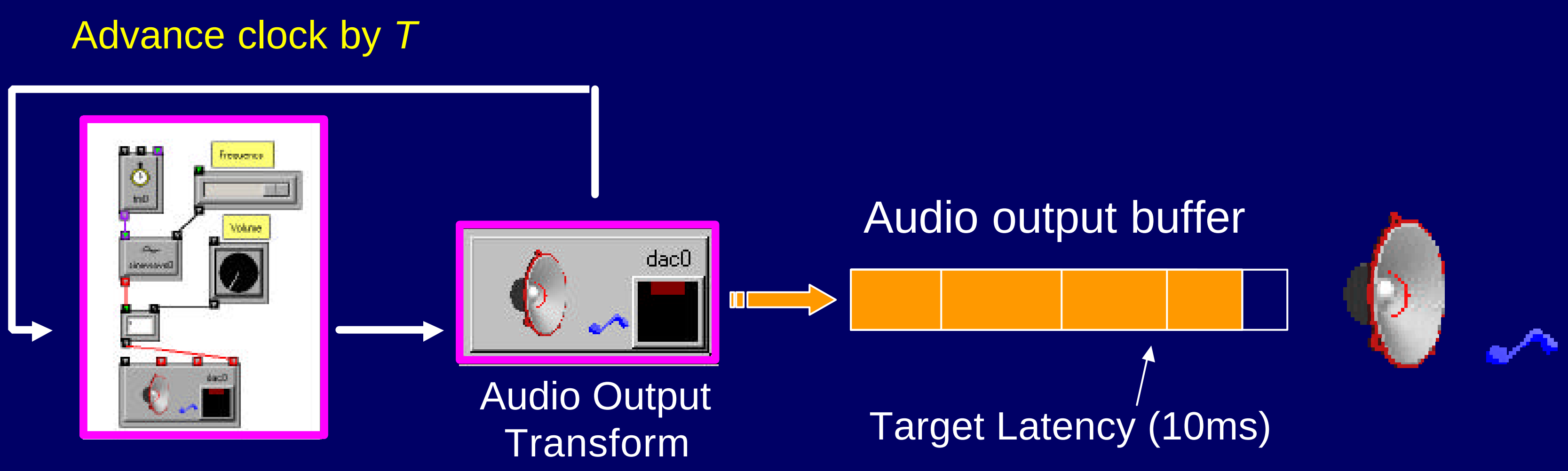


Synthesis Server Execution



- Maintain *quality of service* (QoS): audio continuity, bounded latency & jitter (10 ±1ms)
- Audio output every period *T* (For simplicity, *T* = 1 / sampling rate)
- Output samples
- Advance clock by *T*
- Execute patch
- Wait for output buffer to reach target latency, and repeat process

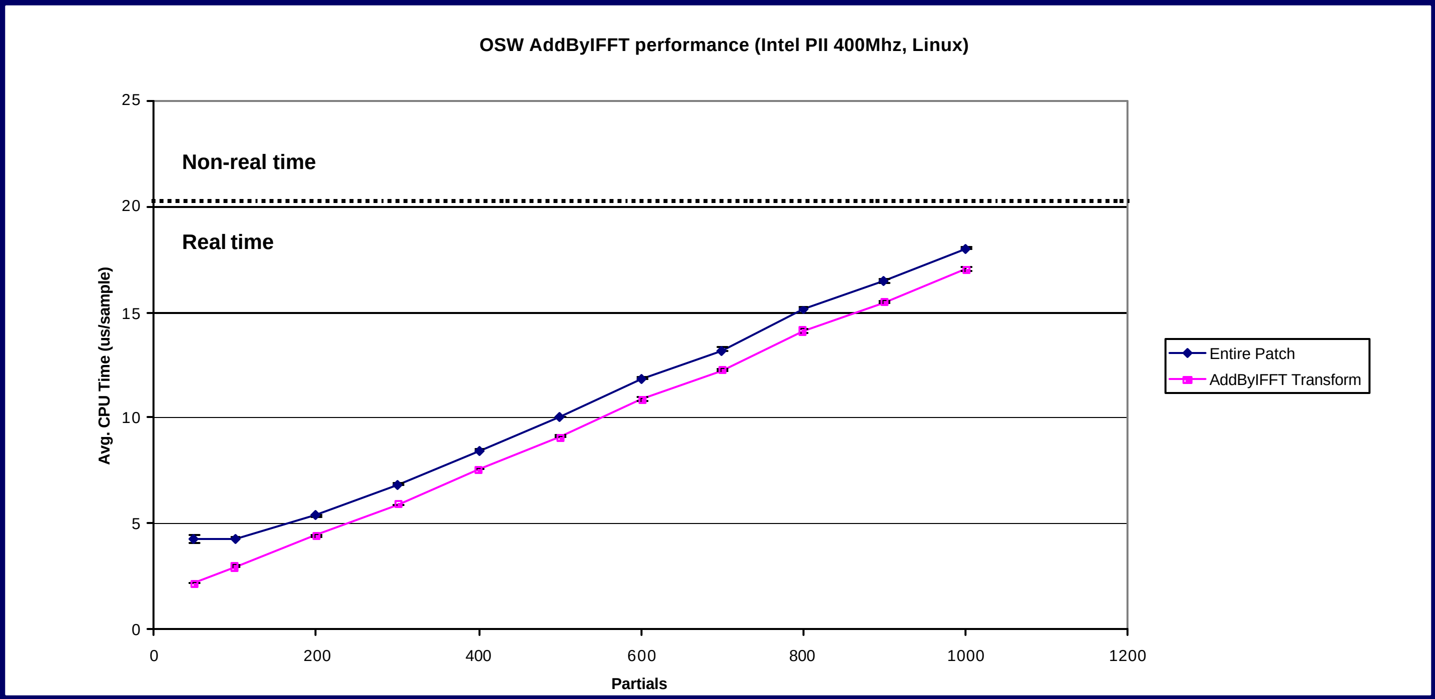
Missed QoS Guarantees



- The per-sample execution time of the patch must be less than *T* (20 μs/sample at 44.1kHz)
- If execution time is greater, the buffer will underflow (audible clicks)
- Increasing buffer size to avoid underflow increases latency

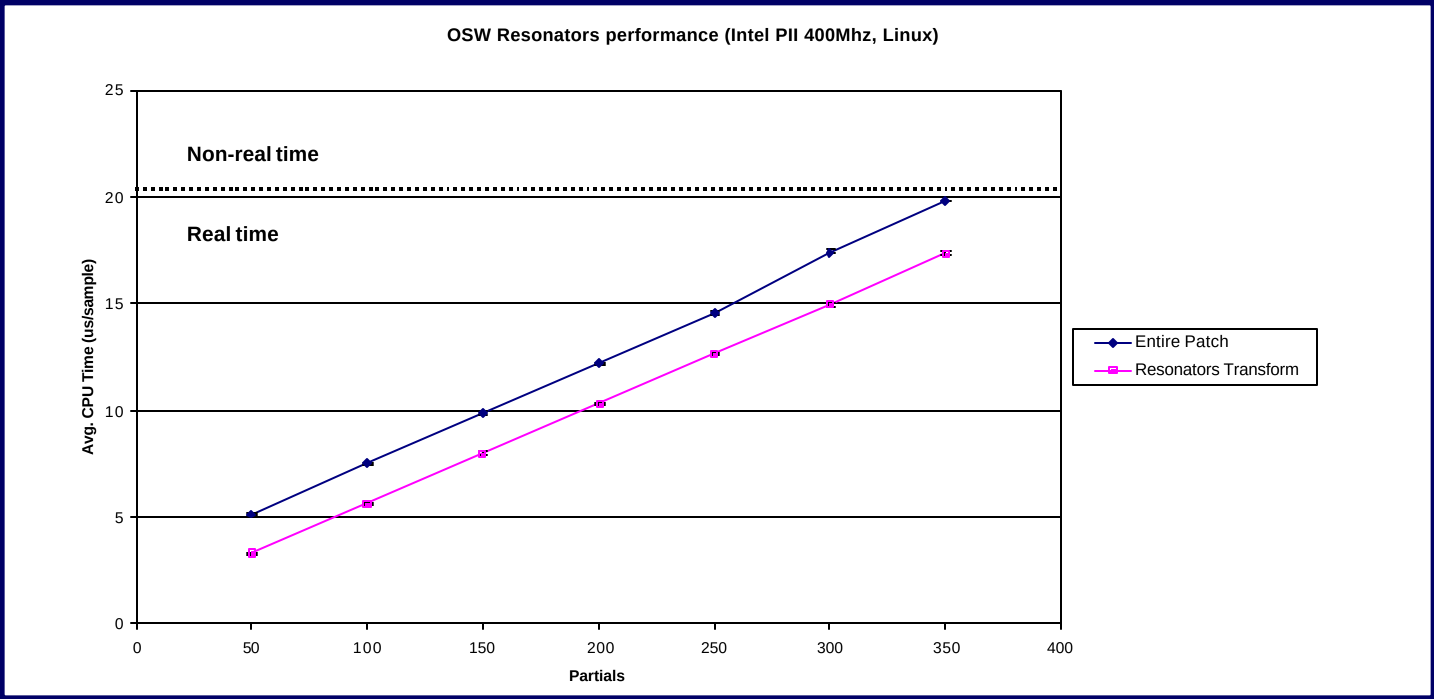
What can we do in 20μs?

- Measured performance of sinusoidal-modeling algorithm



What can we do in 20μs?

- Measured performance of resonance-modeling algorithm



Is this enough?

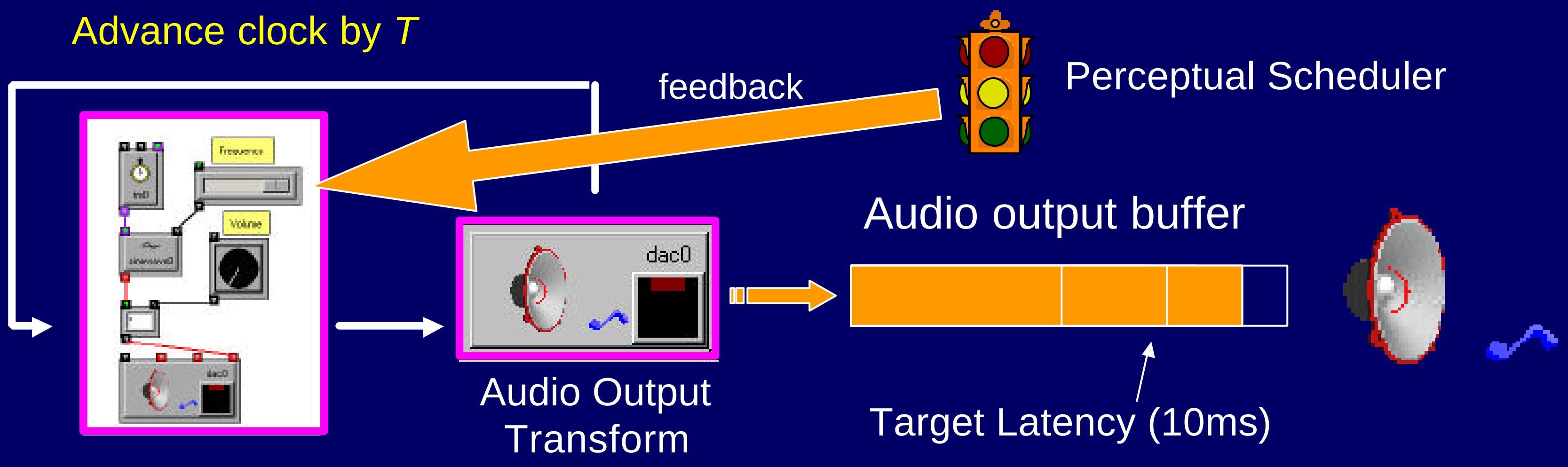
- Adequate for most individual models
- Multiple models
  - Polyphony
  - Multiple audio channels
  - Directional acoustics
- 96kHz Audio
  - Under 10 μs per sample

80 sinusoids

12 x 80 = 960 sinusoids

+ 8x channel overhead

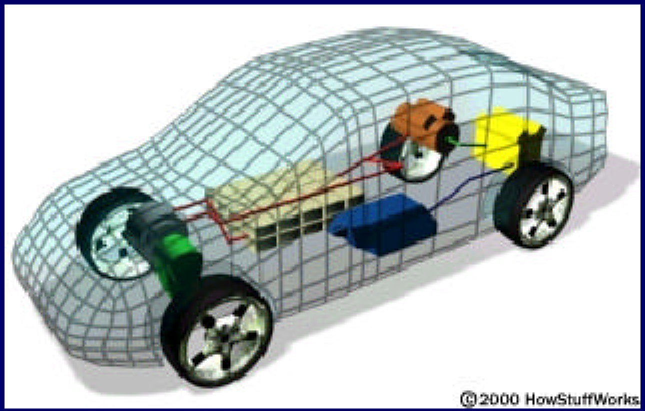
Perceptual Scheduling



- Detect potential QoS failures
- Provide feedback to transforms
- ***Transforms voluntarily reduce computation using measures of perceptual salience***

Analogy: Hybrid Cars

- Maintain QoS
  - Velocity
- Limited bandwidth
  - Smaller engine
  - Less power
- Dynamic adaptation
  - Electric motor assist
  - Regenerative breaking
  - Electric only at slow speed



http://www.howstuffworks.com/hybrid-car.htm

Perceptual Scheduling Details

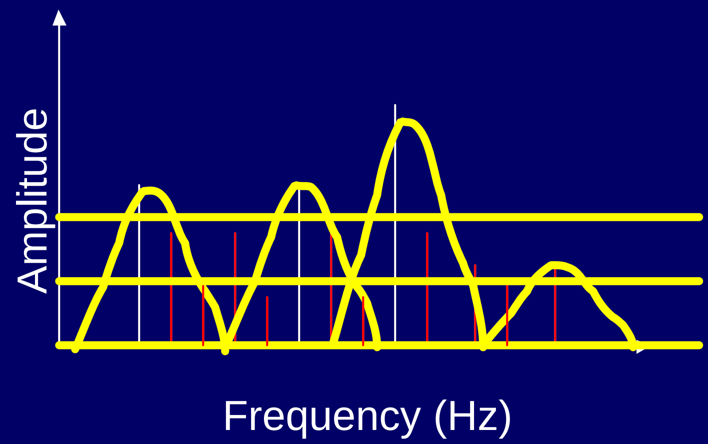
Given execution time E, target execution time E_{max} and reducible transform set R:

1. For each transform $r \in R$, calculate $c(r)$, the time saved by reducing r using an appropriate measure of perceptual salience
2. Find $R' \subseteq R$ such that $E - \sum_{r \in R} c(r) \leq E_{\max}$
3. Reduce computation of each transform in R'

A *reducible transform* requires a reduction strategy and measure of perceptual salience

Reduction Strategies

- Reduce the number of sinusoids in a model
- Graceful degradation by removing weakest sinusoids
- Amplitude threshold
- Masking
- Strategies also used for Resonance Models



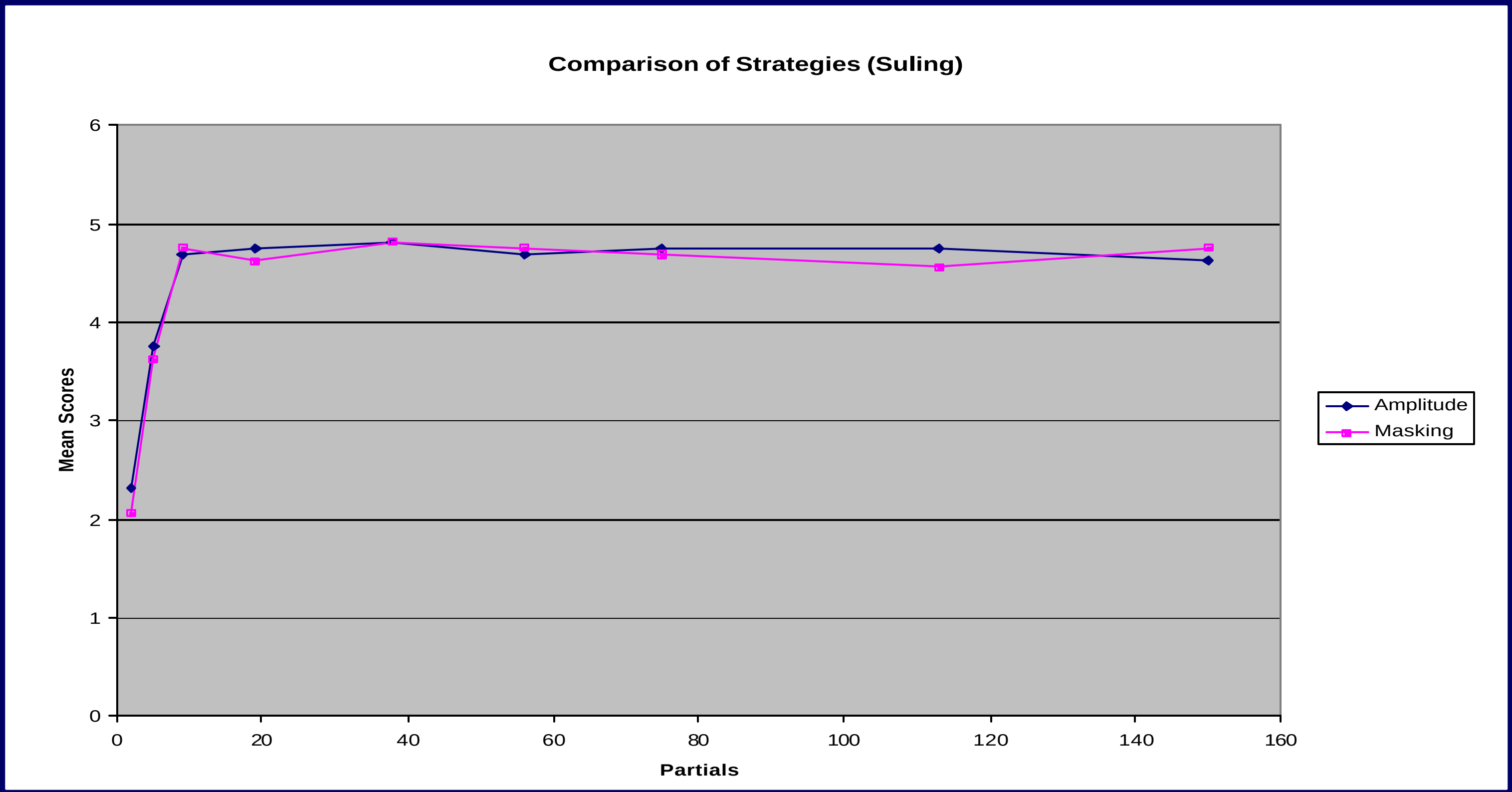
Listening Experiments (I)

- Measure effectiveness of reduction strategies
  - Perceived quality (1 thru 5) vs. model size.
- Summer and Fall, 2000
- Three sinusoidal models
  - Suling flute, berimbao, James Brown
- Three resonance models
  - Marimba, string bass, tam-tam
- Compare reduced and original versions

Suling Sinusoidal Model

150 75 38 19 9 3

Listener Score

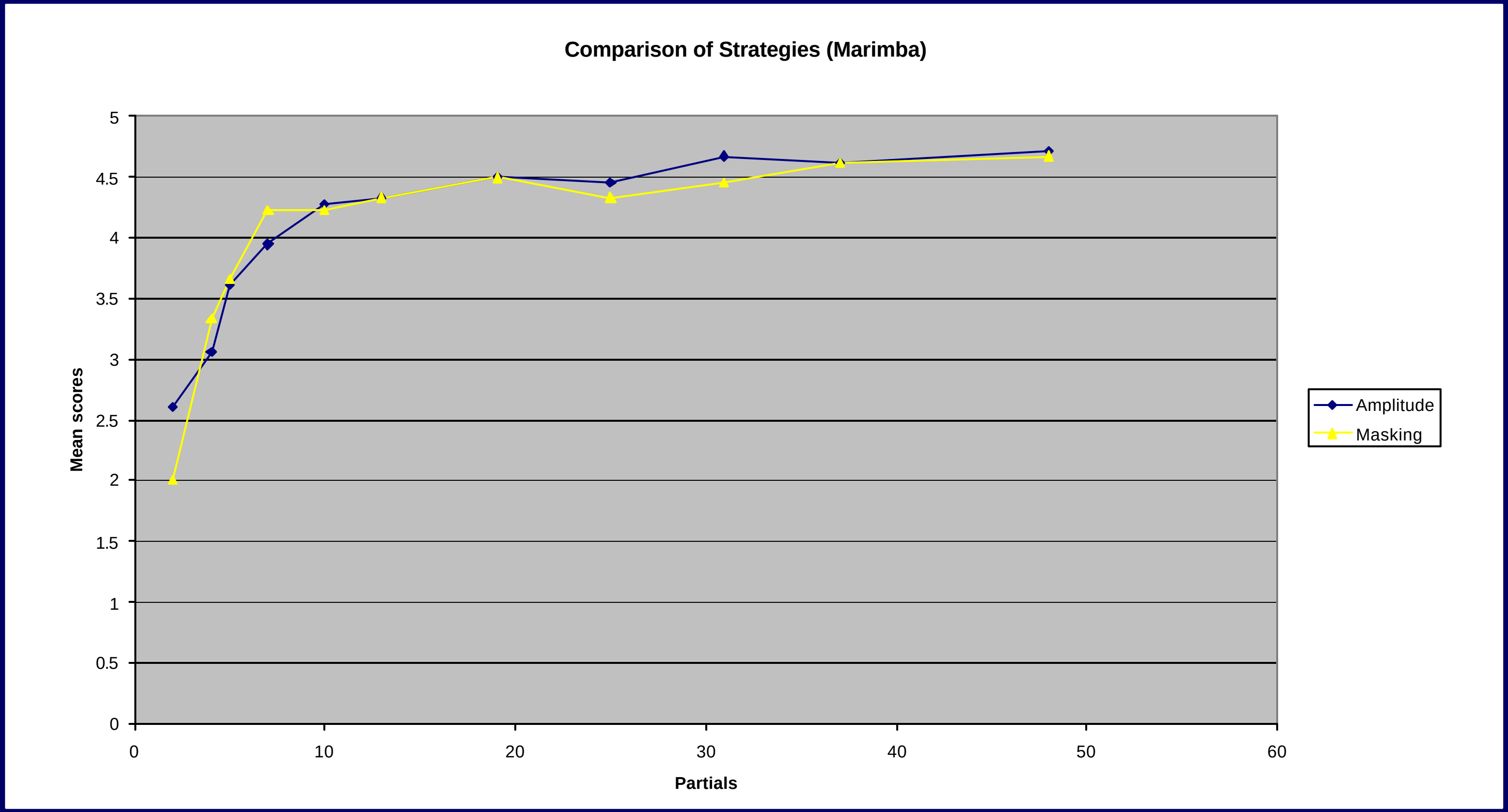


Partials

Marimba Resonance Model

48 25 13 7 5 2

Listener Score



Partials

Discussion

- Quality can be preserved in reduced models
- Little difference between amplitude and masking strategies
  - Few partials are masked
  - Remaining masked partials have low amplitude
  - Amplitude strategy is less computationally expensive!
- Prune partials by amplitude
  - In many models (e.g., suling, marimba), a few partials contribute most of the energy
  - Keep enough partials to maintain 75% of the original energy
  - For resonance models, integrate amplitude over time

Listening Experiments (II)

- Measure effectiveness of reduction strategies within perceptual scheduling framework
  - Perceived quality (1 thru 5) vs. average CPU time.
- Larger musical examples
- February-March, 2001

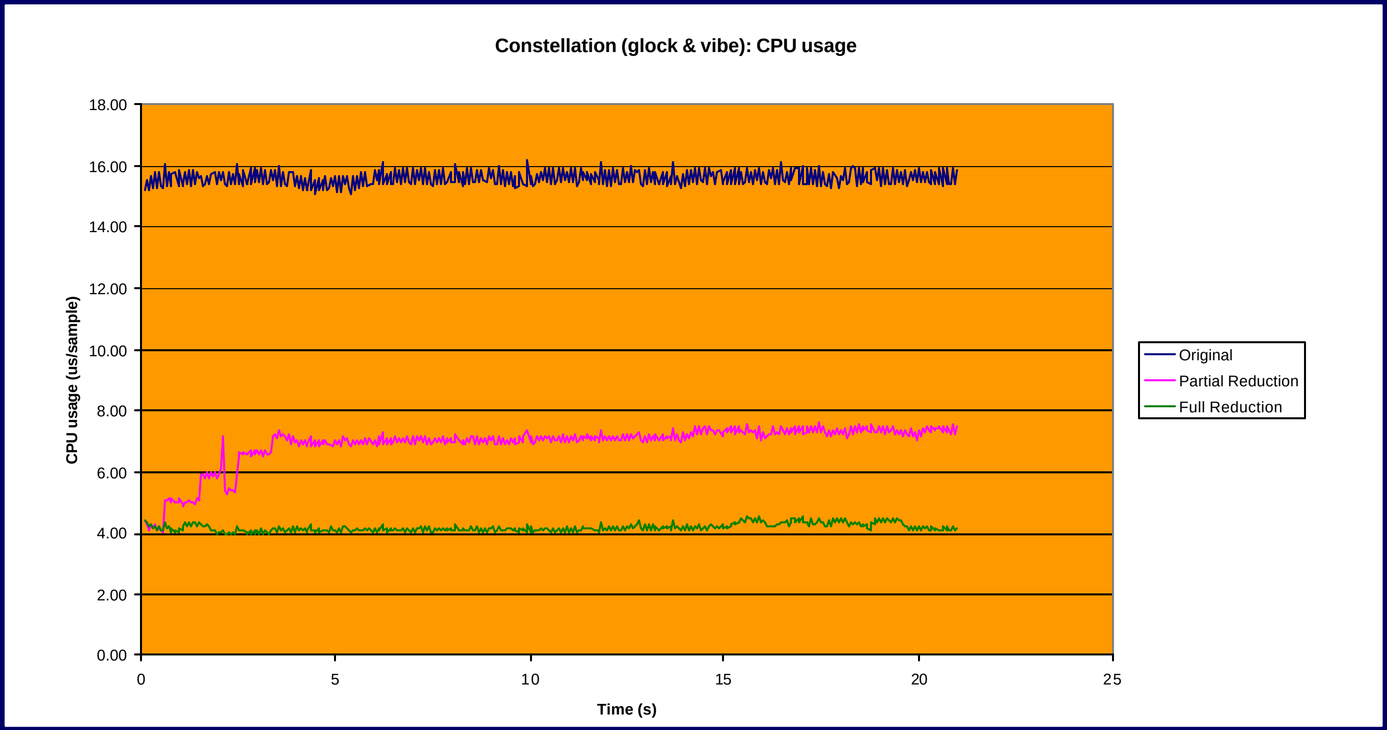
Results: *Constellation* (Glockenspiel and Vibes)

Original D.1

Reduction D.2

Reduction D.3

Mean CPU Time (µs/sample)

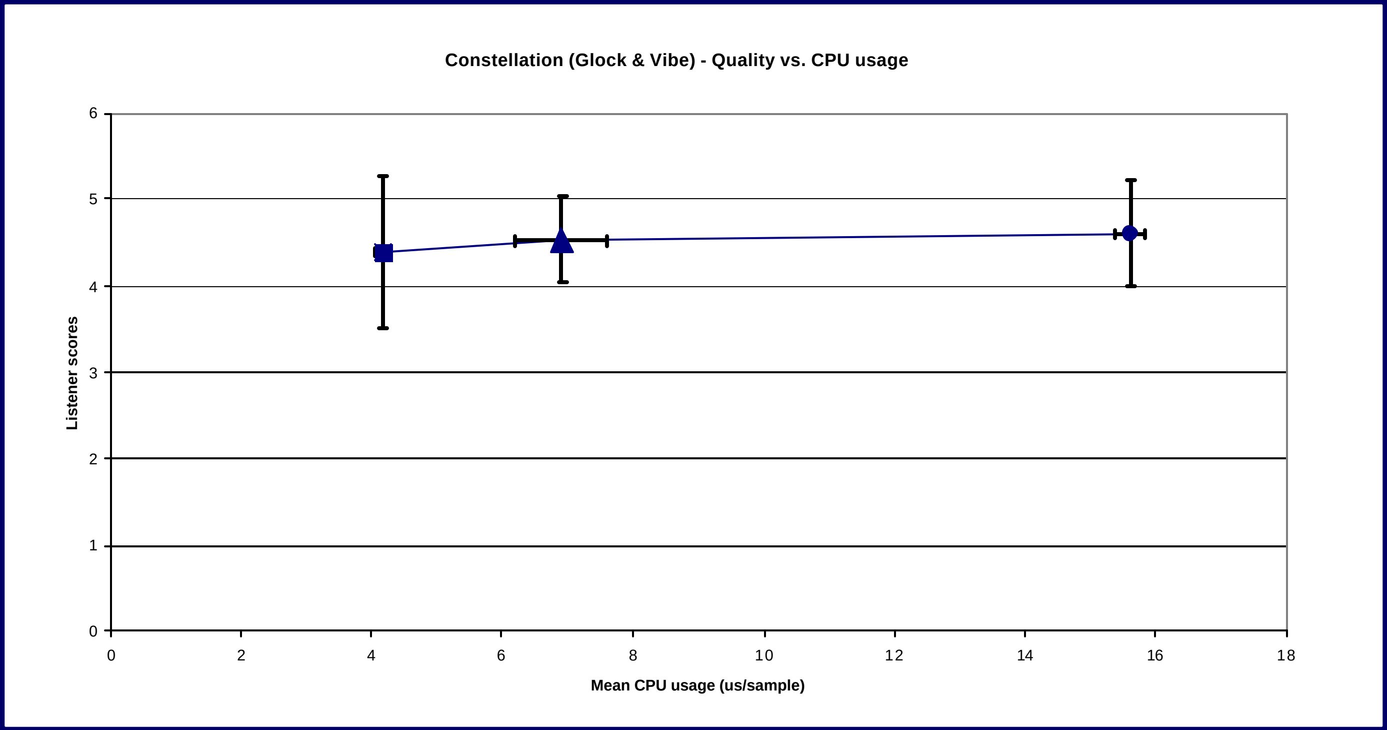


Time (s)

Results: *Constellation* (Glockenspiel and Vibes)

Original ● D.1 Reduction ▲ D.2 Reduction ■ D.3

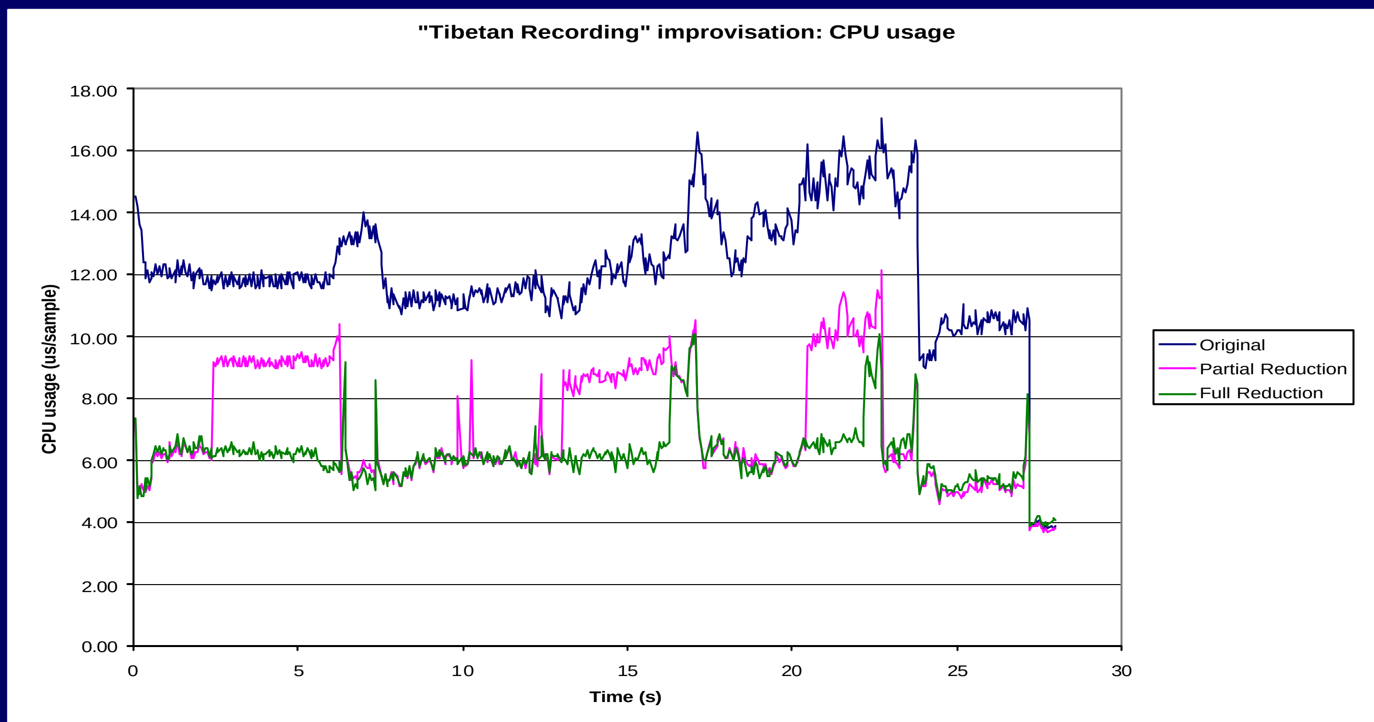
Listener Score



Mean CPU Time (µs/sample)

Results: Tibetan Singing

Original F.1 Reduction F.2 Reduction F.3

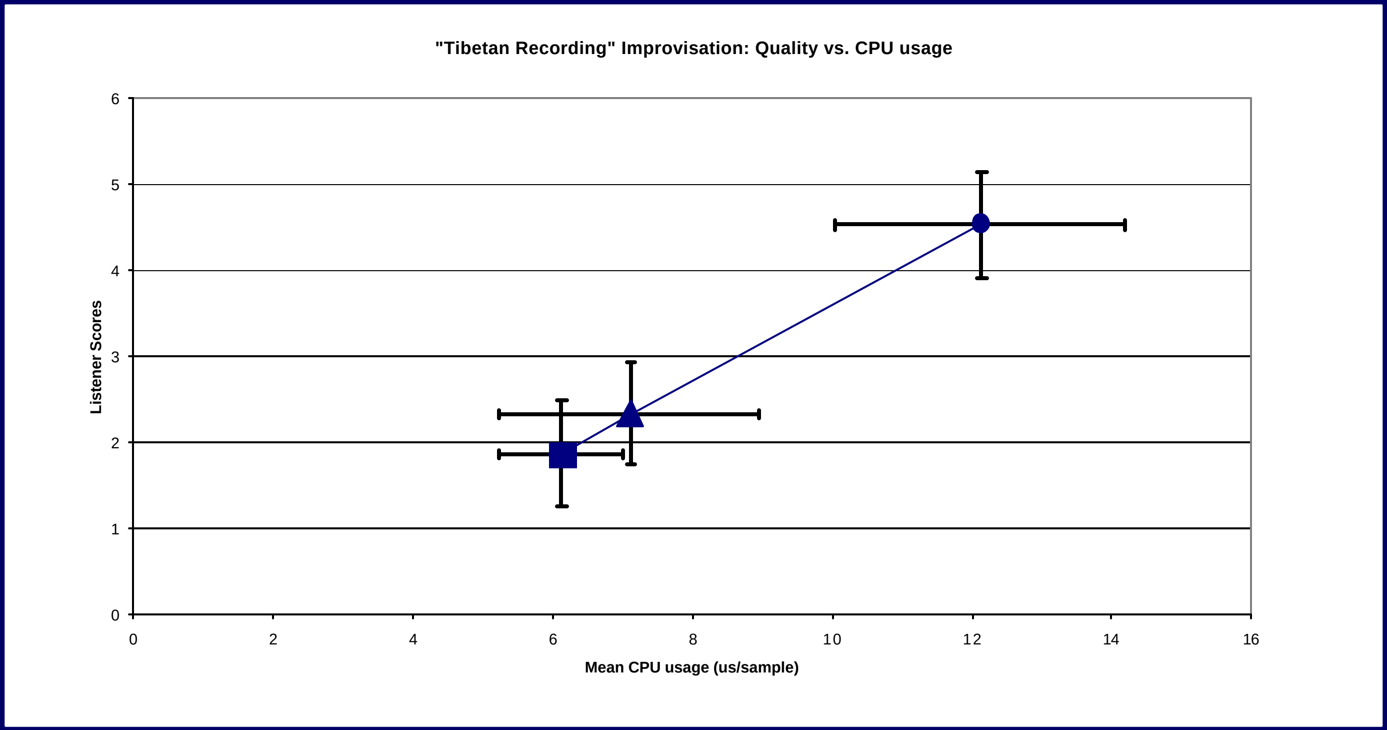


Mean CPU Time (µs/sample)

Time (s)

Results: Tibetan Singing

Original ● F.1 Reduction ▲ F.2 Reduction ■ F.3



Listener Score

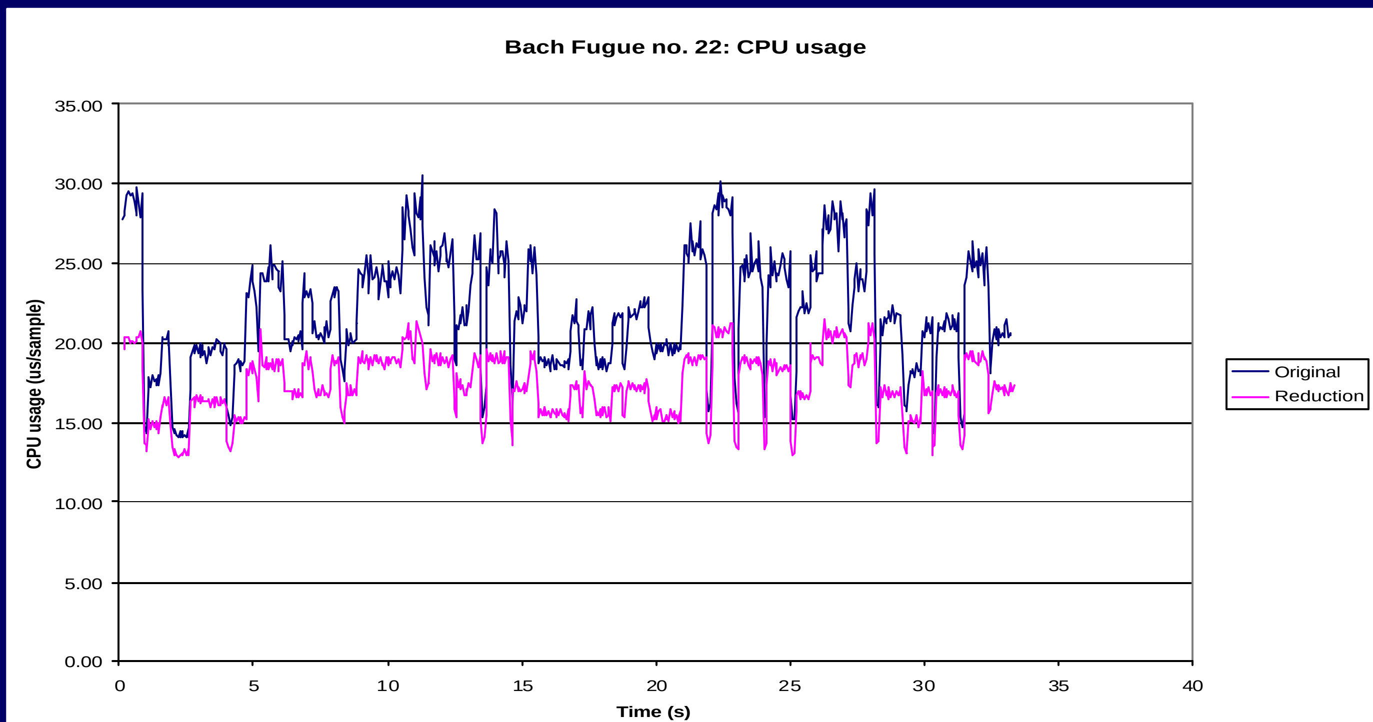
Mean CPU Time (μs/sample)

Results: Bach Fugue (bwv 867)

Original E.1

Reduction E.2

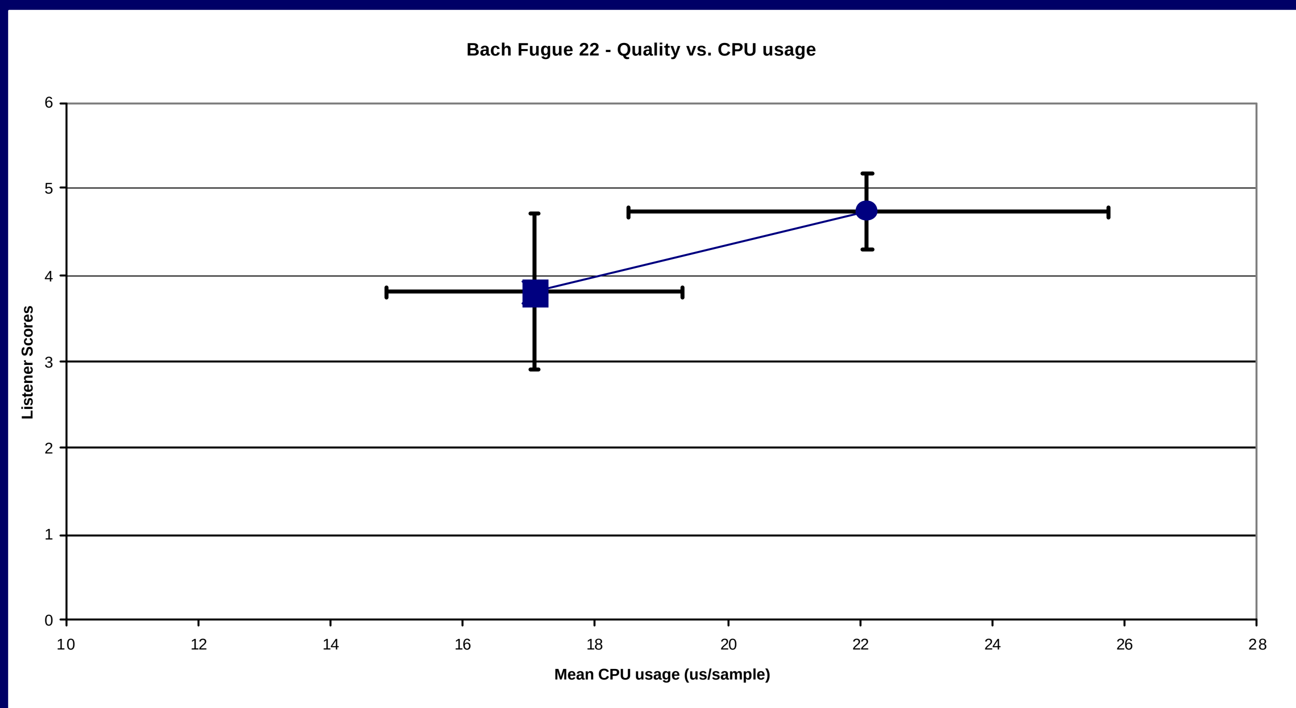
Mean CPU Time (µs/sample)



Time (s)

Results: Bach Fugue (bwv 867)

Original E.1 Reduction E.2



Listener Score

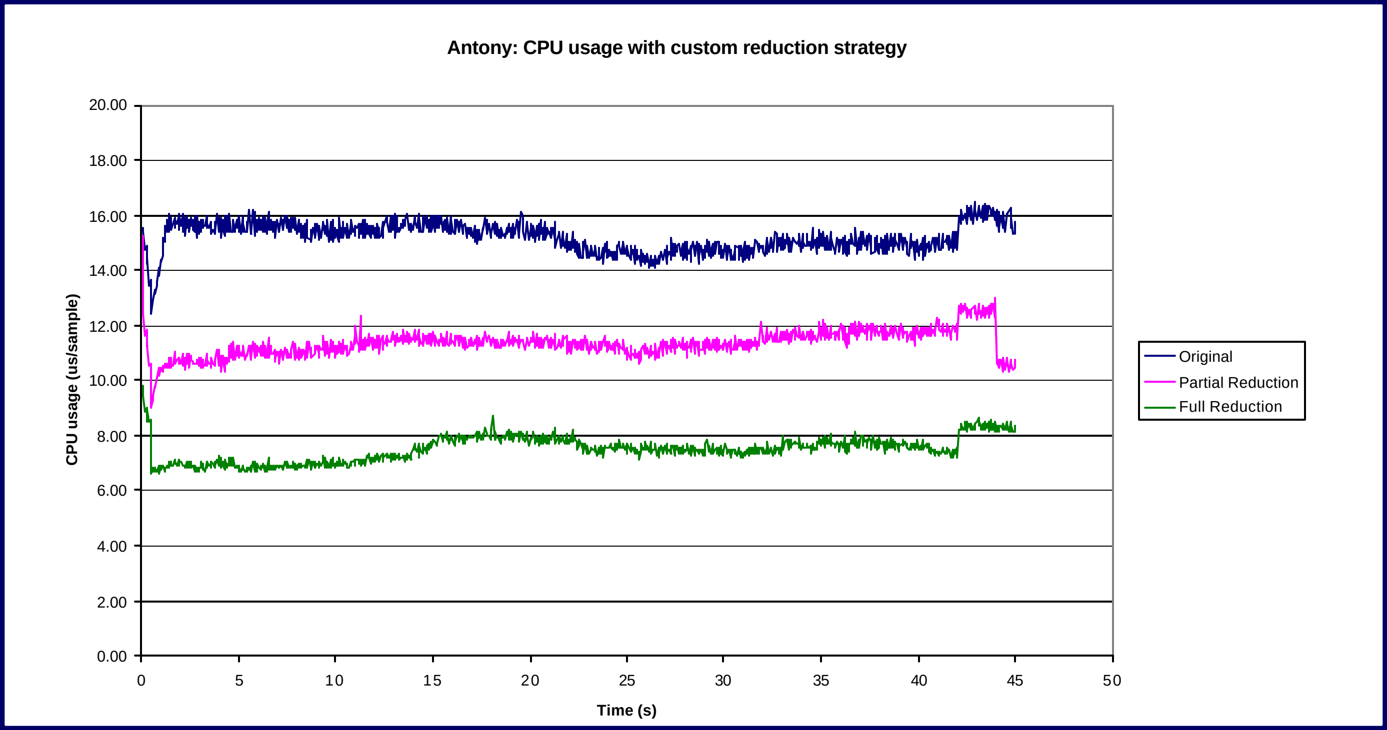
Mean CPU Time (µs/sample)

“Antony 2001”

- David Wessel, 1977
  - 4A Digital oscillator bank [DiGiugno, 1976]
- Algorithmically generated sinusoidal models
  - Random-frequency partials within moving frequency bands
  - Performer changes the frequency bands in real time
  - 3 voices with 200 partials each and independent band controls
- Little or no computation was saved using sinusoidal-model reduction strategy
- Custom reduction strategy was developed
  - Number of partials proportional to bandwidth

Results: *Antony*

Original C.1 Reduction C.2 Reduction C.3



Mean CPU Time (µs/sample)

Time (s)

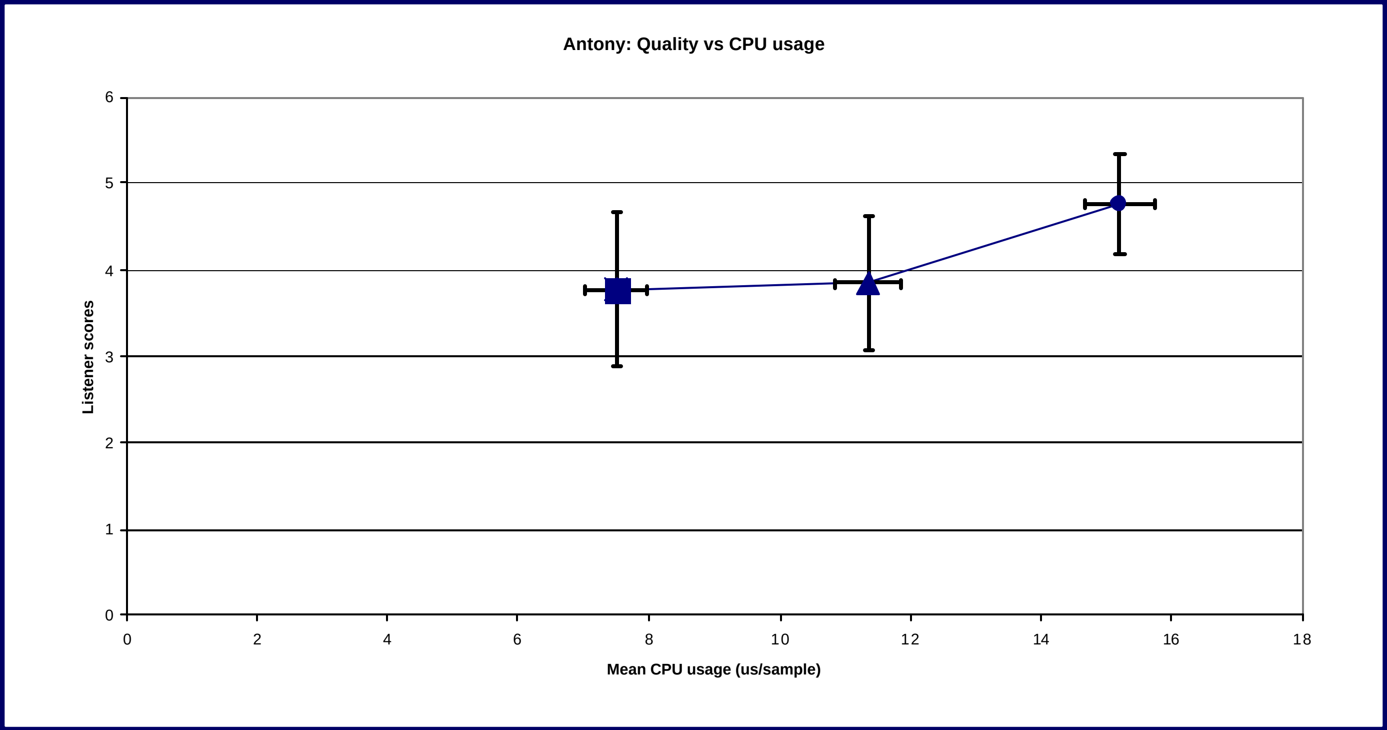
Results: *Antony*

Original C.1

Reduction C.2

Reduction C.3

Listener Score



Mean CPU Time (µs/sample)

Conclusions

- QoS failures can be averted dynamically and gracefully by targeted reductions in the computation used by synthesis algorithms

However…

- Care must be taken in choosing the right reduction strategy for a particular model.

Conclusions

- Best results when additional knowledge about models is available.
  - Algorithmically generated models
  - Resonance models

Future Research Directions

- Develop additional reduction strategies
  - E.g., strategy for vocal models
- Automatic selection of best reduction strategy
  - Machine learning (neural nets, graphical models)
- Other applications
  - Granular synthesis
  - Pitch detection
  - Video processing

Acknowledgements

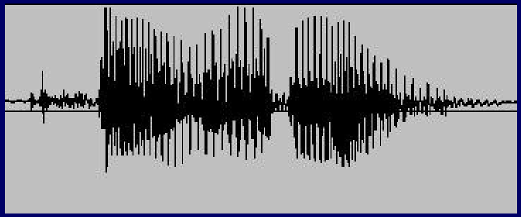
- Dissertation Committee
  - Lawrence A. Rowe, Co-Chair
  - David Wessel, Co-Chair
  - John Wawrzynek
  - Ervin Hafter
- Research Colleagues
  - Adrian Freed
  - Matthew Wright
  - Richard Andrews
- Musical Credits
  - David Wessel
  - Ronald Bruce Smith
  - Timothy Madden
  - Tsering Wangmo
  - Leah Fritz
- Funding
  - NSF Graduate Research Fellowship Program
  - Gibson Music, Inc

Finis

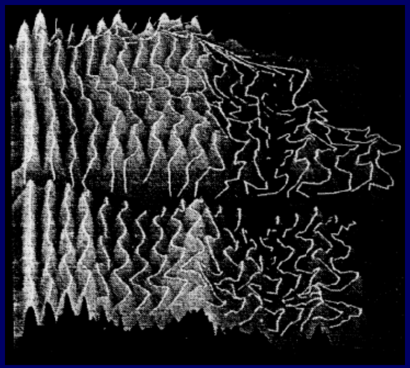
Models from Analysis

- Convert samples for frequency spectra
- Select peaks in spectra

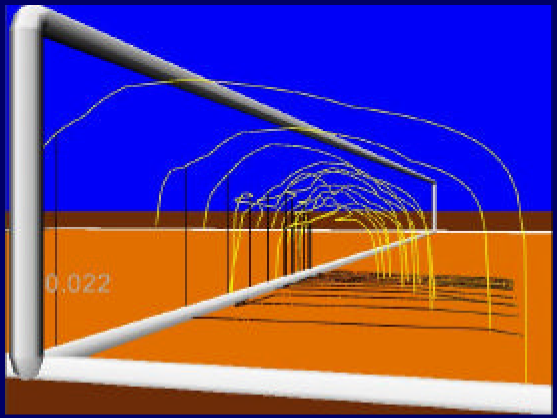
Sampled Waveform



Frequency Spectrum



Sinusoidal Model



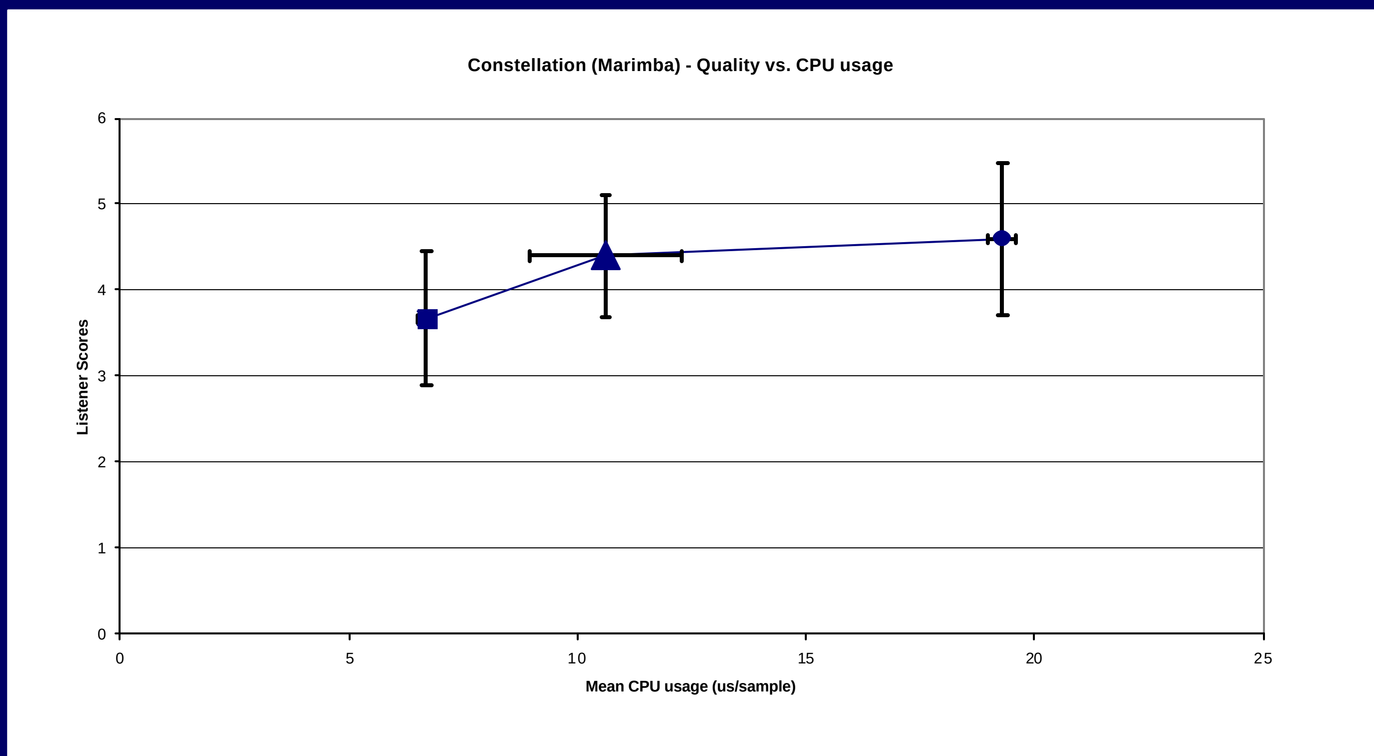
Results: *Constellation* (Marimba)

Original 

Reduction 

Reduction 

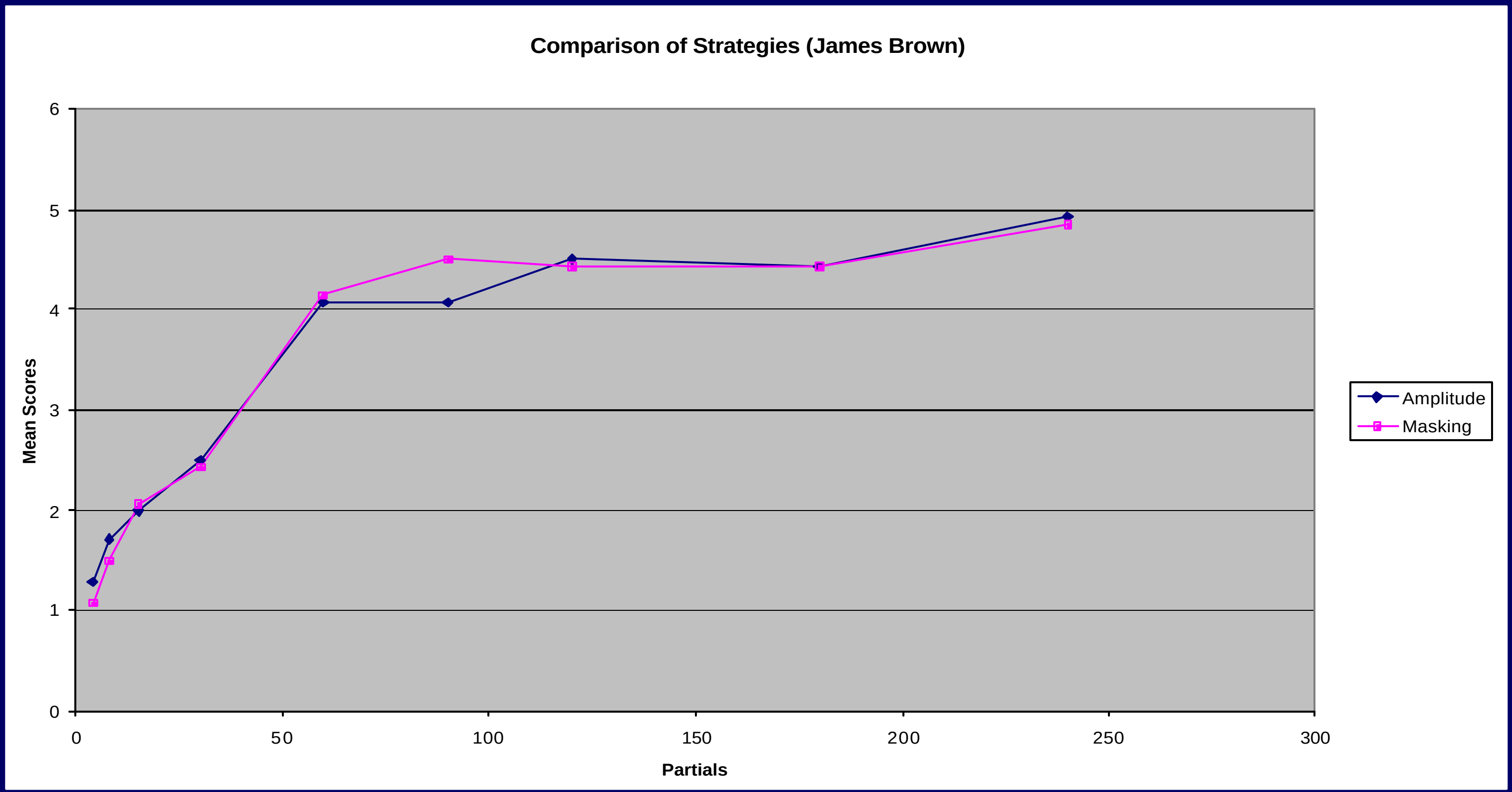
Listener Score



Mean CPU Time (µs/sample)

Sinusoidal model of James Brown and “The Original J.B.’s” (1970)

Original 240 120 60 30 15



Listener Score

Partials