Processamento de áudio com custos computacionais adaptáveis

Thilo Koch
Grupo de pesquisas em Computação Musical
IME - USP

4 de maio de 2015

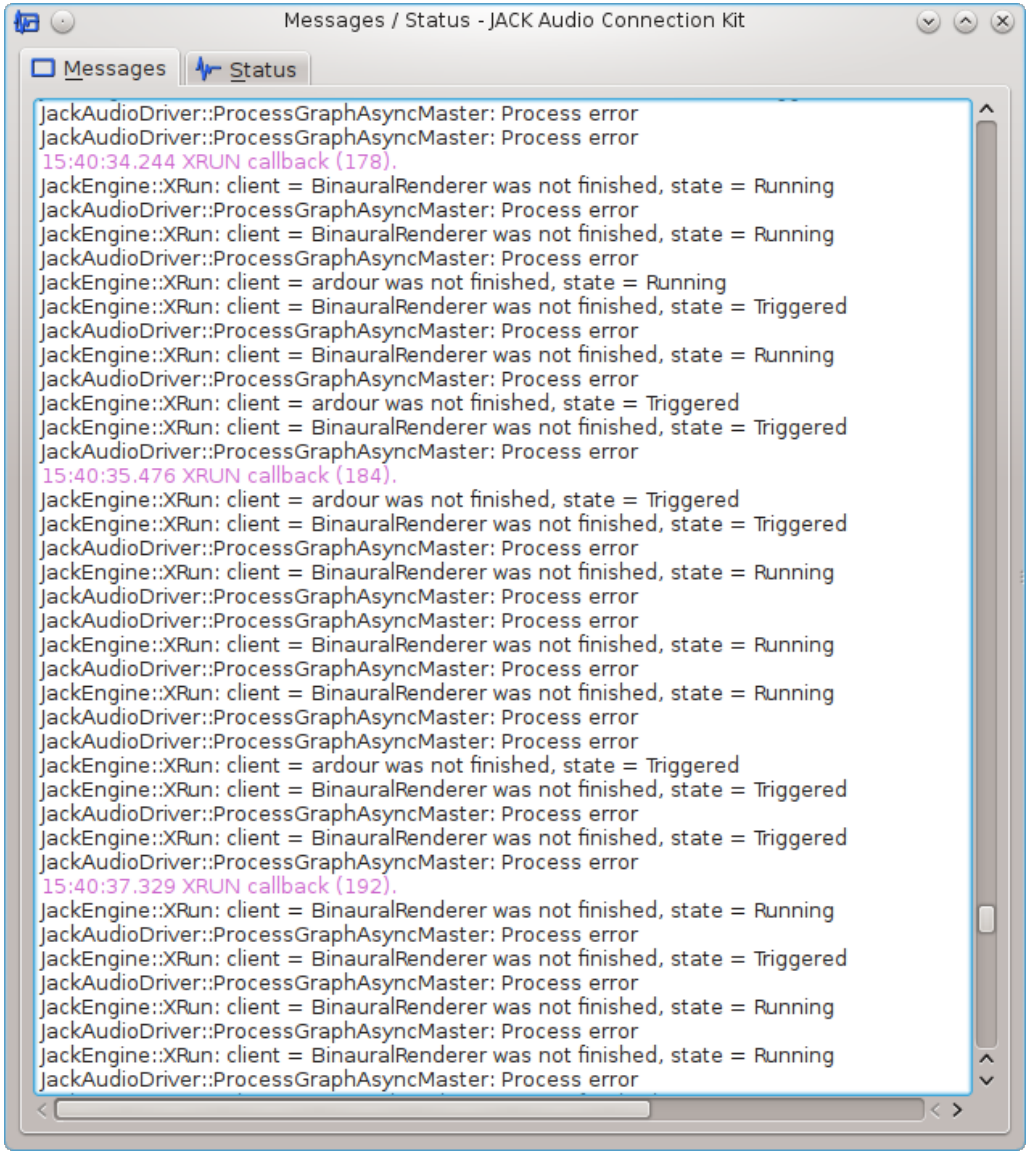
Prólogo

O que será apresentado no seminário?

- Ideias para/da minha tese;
- pesquisa em várias direções seguindo meu interesse inicial;
- tentativa de estruturar e explicar o que se pretende (esboço);
- pesquisa até agora: tem mais largura que profundidade;
- interesse por perguntas, comentários e sugestões.

Motivação

Situação de sobrecarga em processamento de áudio ao vivo.



Casos de uso - Caso **dinâmico**

Dado

Um sistema computacional que executa um programa que auraliza uma cena sonora espacializada (ou uma estação de trabalho de áudio digital).

Descrição do Problema

- A síntese em execução esgota toda capacidade computacional,
- o usuário adiciona mais um elemento de síntese *ao vivo*,
- por causa dos custos computacionais adicionais o sistema entra em uma situação de sobrecarga,

$\rightarrow$ a síntese falha (cliques, graves distorções, interrupções) devido à violação da condição de tempo real.

Casos de uso - Caso **dinâmico**

Expectativa

- O sistema deve **dinamicamente** degradar a qualidade da síntese para reduzir os custos computacionais (*trade-off*).

$\rightarrow$ o sistema consegue computar e reproduzir a cena com a qualidade reduzida sem falhar;

- o sistema deve aumentar a qualidade da síntese quando recursos computacionais são liberados.

Casos de uso - Caso **estático**

Dado

Um sistema computacional que executa um programa que auraliza uma cena sonora espacializada (ou uma estação de trabalho de áudio digital).

Descrição do Problema

Possibilidade de sobrecarga na execução da síntese em função dos recursos limitados da plataforma.

Expectativa

- O sistema deve avaliar os recursos da plataforma **na inicialização do programa** e
- configurar os algoritmos visando a melhor qualidade possível.

$\rightarrow$ o sistema consegue computar e reproduzir a cena na qualidade reduzida sem falhar.

Programas e custos

- Um programa é um mapeamento de um conjunto de entradas para um conjunto de saídas.
- **Entrada**: fluxos de áudio e definição de taxa de amostras, de tamanho de amostras, dados de controle ...
- **Saída**: fluxos de áudio com uma certa taxa de amostras por segundo, tamanho ...
- **Mapeamento**: cadeia de (sub)processos, fluxo de dados em grafo
- obviamente temos custos (e em cadeias de filtros custos se adicionam)

Tempo real - Definições

Tempo de resposta

O tempo entre a disponibilidade dos dados de entrada e a realização da resposta esperada.

Sistema de tempo real

Um sistema que tem que satisfazer restrições de tempo de resposta ou ariscar consequências sérias.

Sistema de tempo real "macio" *(Soft Real-Time System)*

Um sistema cujo desempenho é degradado mas não destruído quando falha em atender as restrições.

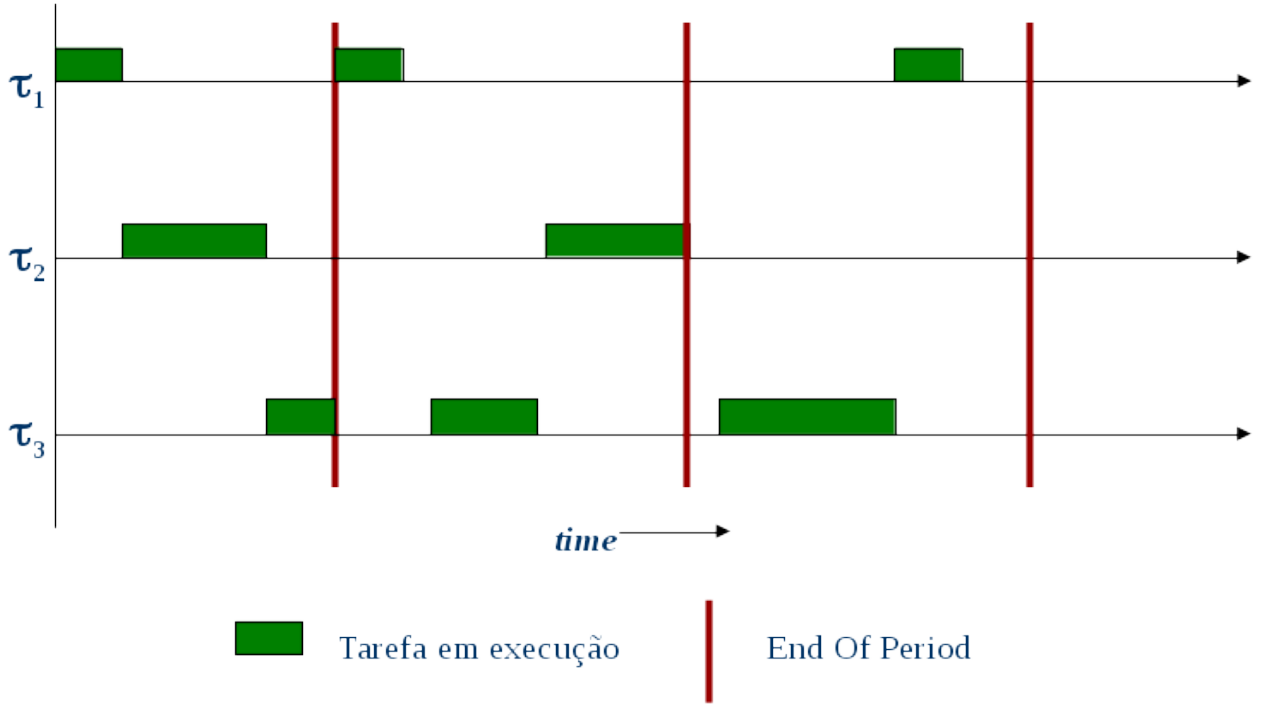
Utilização da CPU

Porcentagem de processamento não-ocioso da CPU.

- $U = \sum_{i=1}^{n} u_i = (\sum_{i=1}^{n} t_i)/t_{resposta} \leq 1$

Tempo real - Sistema de processamento de áudio

- Saída: 48 kHz, uma faixa de áudio $\rightarrow$ disponibilizar 1 amostra cada 20μs
- ou blocos de amostras (64 amostras $\sim$ 1.3 ms)
- Condição: todos subprocessos tem que terminar



Computação imprecisa / Aplicações flexíveis

Como lidar com situações U $>$ 1?

- Premissa: Um resultado imperfeito, porém no tempo oportuno é melhor que um resultado perfeito mas atrasado.
- Solução: Separar as tarefas em uma parte obrigatória e uma parte opcional ($t = t_{mandatory} + t_{optinal}$)
- Várias estrategias:
  - peneira - descartar dados (ou não executar partes da tarefa)
  - *milestone* - melhorar resultado monotonicamente
  - múltiplas versões - dependendo do tempo disponível uma versão da tarefa é escolhida

Metodologia

1. Desmultiplexar / desmembrar software de processamento de áudio e identificar elementos do processamento.
2. Analisar elementos e sondar potenciais para a flexibilização dos custos.
3. Analisar cadeias / grafos de processamento.
4. Desenvolver mecanismos de controle do processamento e gerenciamento de recursos.
5. Propor / implementar / avaliar protótipos.

Observação: Não procurar algoritmos mais eficientes (que seria otimização) mas *trade-offs*.

Sistemas de processamento de áudio

Quais sistemas vão ser analisados?

- Simulação e Auralização: modelagem física, espacialização, simulação de salas.
- *Digital Audio Workstation*: plug-in (filtros)
- Ambientes de programação de áudio: PureData, Gstreamer

Exemplo 1: Espacialização

- Espacialização: produção de um efeito aural tridimensional perceptível.
- Várias técnicas de processamento: *Ambisonics*, Síntese do campo sonoro, Síntese binaural.
- **Entrada**: fontes sonoras (faixas de som) com posição / movimentos no espaço
- **Saída**: som audível de uma até centenas de faixas de som

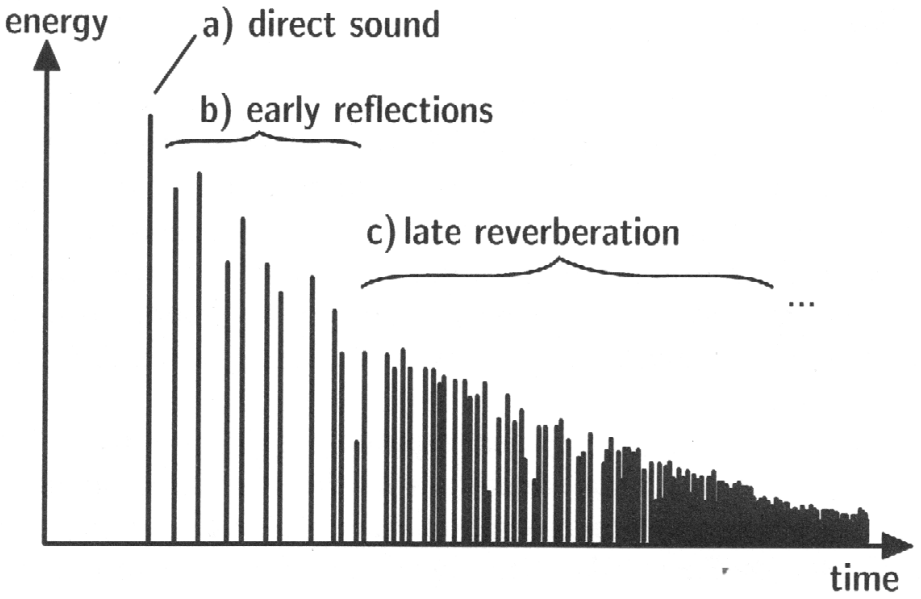
Espacialização

Método comparável com sistemas LOD (*Level of Detail*) na computação gráfica.

- Premissa: A resolução perceptiva para objetos cai com a distância.
- Objetos sonoros distantes podem ser também processados com menos detalhes.
- Objetos podem ser aglomerados em conjuntos que são processados como uma fonte só (cluster)
- Procedimento dinâmico com reavaliação periódica.

Modelagem física

Simulação de propriedades acústicas de salas: Resposta impulsiva



- Variação do tamanho da resposta impulsiva
- Variação de detalhes no modelo 3D da sala

Isso vale para outros tipos de modelagem de estruturas físicas mas sempre "cuidar" para não cortar detalhes importantes para a qualidade esperada.

Exemplo 2: Filtros

Filtro usado numa *Digital audio workstation* tipo plug-in;

- existem muitos(!) tipos de filtros e implementações;
- exemplo FIR: $y(n) = \sum_{i=0}^{M} a_i x(n-i)$;
- implementação com loop $\rightarrow$ O(NM).

Flexibilização / Mecanismo trade-off:

- Reduzir o número de coeficientes: aproximação (no domínio de frequências).
- Aproximar filtro com outro tipo de filtro mais "leve".
- Cadeias de filtros: variar taxa de amostragem; variar precisão da computação.

Controle e Gerenciamento de recursos

- Medição de recursos usados e conhecimento sobre estado dos elementos (plug-ins)
- Conhecimento sobre plug-ins: parâmetros de controle e sua influência nos custos e na qualidade.
- **Objetivo**: Otimização - maior qualidade com custos ainda computáveis
- Arquitetura
  - Camada adicional com conhecimento global?
  - Cada elemento decide autonomamente?
  - Mudança no mecanismo de escalonamento do sistema é sensata?
  - Muito depende do sistema de processamento de áudio.

Qualidade

Como avaliar qualidade do processamento de áudio? Percepção é subjetiva.

Opções:

- Semelhança matemática (diferença, minimizar soma dos quadrados de diferenças)
- Métodos de avaliação psico-acústicos (por exemplo: MPEG1 Layer3, *Spatial Audio Quality Inventory (SAQI)*)
- Produtor decide ("com botões"ou por configuração)

Para certas aplicações a ideia inteira não é desejável!

Ambientes de implementação e pesquisa

Quais sistemas / plataformas / ambientes estão (até agora) no foco?

- Plugins: LV2
- Sistema de espacialização: Sound Scape Renderer, Wonder
- Plataforma / framework de desenvolvimento: Gstreamer, Pure Data

Fim

Muito obrigado!

Perguntas? Comentários? Sugestões?

Contato: tiko@ime.usp.br