/INF01053 - Computação & Música (2020/1) - Tópicos Especiais em Computação XVI
Nomes: Izadora Dourado Berti e Letícia dos Santos

Tema: Processamento de Audio na GPU

1. Fundamentos

AUDIO DSP

O DSP de áudio desempenha um papel importante na indústria musical, com músicas, filmes e outras mídias audíveis sendo criadas e processadas inteiramente no mundo digital. O processamento é tradicionalmente realizado em um computador pessoal, por meio da unidade central de processamento (CPU) que é programada para aplicar o algoritmo de áudio a um fluxo de dados de entrada. Os computadores pessoais também contêm unidades de processamento gráfico avançado (GPU) que criam as imagens 2D e 3D em monitores de computador. Esses chips têm crescido em potência e chegaram a um ponto em que podem ser programados para computação alternativa de uso geral, por isso é possível inferir que o processador DSP dedicado se tornará obsoleto.

SOBRE GPU

As GPUs estão começando a atuar como mecanismos de computação paralelos de uso geral. GPUs modernas são, na verdade, GPGPUs: unidades de processamento gráfico de uso geral que realizam cálculos não especializados que normalmente seriam conduzidos pela CPU.

O início da utilização das GPUs para programação de aplicações de propósitos gerais impulsionou o desenvolvimento de facilidades na sua arquitetura e programação com a introdução da arquitetura CUDA (Computer Unified Device Architecture) apresentada pela NVIDIA, em 2006. A CUDA tem como objetivo facilitar a interface entre o programador e as aplicações GPU, sendo um modelo de programação e uma plataforma de computação paralela que pretende melhorar o aproveitamento do poder de processamento da GPU. O modelo de programação CUDA fornece aos programadores abstrações simples de organização hierárquica das threads, sincronização e memória permitindo a implementação adequada de programas para a GPU. A CUDA oferece suporte a linguagens de programação como C, C++, Fortran, OpenCL e DirectX. No início, o suporte OpenCL para GPUs NVIDIA não era tão eficiente em comparação com CUDA, mas não é mais o caso, Na verdade, ele não é

apenas para GPUs, OpenCL pode ser usado para executar programas em plataformas heterogêneas compreendendo CPUs, GPUs, DSPs e até FPGAs.

Pelo seu poder de paralelismo, GPUs vem sendo usada em várias áreas além do processamento gráfico de computadores pessoais, como mineração de moeda digital por exemplo. Por isso, é uma boa ideia usar para processamento de áudio em tempo real, já que são aplicações que requerem um processamento poderoso e rápido. Em vez de desenvolver chips DSP proprietários com uma vida útil limitada, fabricantes de áudio (e até de vídeo) poderiam desenvolver soluções de processamento de áudio em tempo real usando GPUs e economizar nos custos de desenvolvimento de hardware.

EXEMPLO DE PODER DE GPUs

A GPU G80 é massivamente paralela e internamente seu chip possui 128 SPs (16 SMs, cada um com 8 SPs), totalizando 500 gigaflops (FLOPS é um acrônimo que significa FLoating-point Operations). Cada SP tem uma unidade de multiplicação-adição (MAD) e uma unidade de multiplicação. Outras unidades de função especial executam funções de ponto flutuante tais como raiz quadrada (SQRT).

A arquitetura de uma GPU com suporte a CUDA pode rodar milhares de threads por aplicação. Uma boa aplicação roda tipicamente entre 5000 a 12000 threads simultaneamente. O chip G80 suporta até 768 threads por SM, que soma cerca de 12000 threads para este chip. Com 240 SPs, o GT200, mais recente, ultrapassa 1 teraflops e suporta 1024 threads por SM e até cerca de 30.000 threads para o chip. Comparando com multithreading simultâneo, uma CPU Intel típica suporta em média 2 ou 4 threads por núcleo [Kirk and Hwu 2010].

"Have a look at the Jetson Nano from NVIDIA for example. This small $99 USD(!) Developers Kit with 128 cores delivers an amazing 472 GFLOPS. It is being positioned as a platform for running modern AI algorithms with multiple neural networks in parallel. That computing power can be utilized for Audio DSP. After all, it is a GPGPU! In contrast, the high-end TMS320C6678 Multicore (8 cores) Fixed and Floating Point Digital Signal Processor from Texas Instruments can deliver 160 GFLOPS. Take note that while the TMS320C6678 is a high-end device, the Jetson is considered to be an entry-level offering by NVIDIA."

https://www.cycfi.com/2019/04/gpu-dsp-when-you-cant-have-enough-cores/

2. GPUs para processamento de áudio

TEORIA

Os aplicativos de processamento de áudio estão entre os mais intensivos em computação e frequentemente dependem de recursos DSP adicionais para desempenho em tempo real. No entanto, os DSPs de áudio programáveis geralmente estão disponíveis apenas para desenvolvedores de produtos. Placas de áudio profissionais com vários DSPs geralmente oferecem suporte a efeitos e produtos específicos, enquanto o hardware de “áudio de jogo”, por exemplo, ainda implementa apenas pipelines de função fixa que evoluem em um ritmo bastante lento. A ampla disponibilidade e o aumento do poder de processamento das GPUs podem oferecer uma solução alternativa. Os recursos da GPU, como instruções de multiplicação-acumulação ou unidades de execução múltiplas, são semelhantes aos da maioria dos DSPs. Além disso, os aplicativos de renderização de áudio 3D exigem um número significativo de cálculos geométricos, que se encaixam perfeitamente na GPU.

PESQUISAS NA ÁREA

O uso da GPU especificamente para processamento de áudio é um projeto de pesquisa recente que busca explorar as imensas capacidades de ponto flutuante de GPU para processamento de áudio, com foco na produção de música. Uma forma de demonstrar os benefícios da GPU no processamento de síntese e resultados de algoritmos comparando com o usual processamento baseado na CPU. Por exemplo, ao medir o desempenho da GPU contra o da CPU, Tsingos et. al. (Gallo e Tsingos, 2004; Moreland e Angel, 2003; Savioja ***et al.***, 2011) mostraram que para diversas aplicações é possível conseguir aceleração com fatores de 5 a 100 vezes. Ali são estudadas duas abordagens diferentes para implementações de aplicações DSP comuns: o mapeamento de problemas para a pipeline gráfica e a computação de propósito geral para GPU.

Os aplicativos de áudio que utilizam a GPU mostraram aumento de velocidade significativo em relação às versões da CPU. Uma Fast Fourier Transform mostrou ser quatro vezes mais rápida do que usar um CPU quad core de última geração. usaram uma GPU para realizar a síntese aditiva de um milhão de sinusóides. O CPU atingiu o pico de 439 sinusóides geradas em tempo real, enquanto o GPU atingiu o pico de 1395 sinusóides em tempo real. [9] simulou a síntese do campo de onda usando uma GPU nVidia GTX 285 contra uma CPU Intel Core i7-920. A GPU alcançou acelerações de três vezes e meia para tamanhos de quadro de 8192 amostras, tamanhos de quadro menores abaixo de 512 amostras favoreceram a CPU. Todos esses são sistemas que usam a GPU para aceleração de itens fora do escopo deste projeto, pois envolvem

síntese ou processamento no domínio da frequência. Todos, entretanto, mostram o desempenho potencial em um processador CUDA.

Com base nisso, GPUs podem ser usadas para processamento de áudio com desempenho semelhante ou superior em comparação com implementações de software otimizadas rodando em CPUs top-of-the-line. Foi demonstrado que as GPUs mais recentes superam as CPUs em uma série de outras tarefas, incluindo Fast Fourier Transform, uma ferramenta amplamente usada para processamento de áudio.

DIFICULDADES ENCONTRADAS PARA USO E IMPLEMENTAÇÃO

Apesar de todas as vantagens da GPU moderna comparada com um DSP dedicado, o principal problema a ser enfrentado é a latência e as taxas de transferência de dados com a entrada e saída de dados da CPU e da GPU. Mas 1) as GPUs modernas estão ganhando taxas de transferência de dados incríveis na faixa de GB/s e latências no µs e 2) existem técnicas de software que podem ajudar para mitigar o gargalo de transferência de dados.

Além disso, outras deficiências ainda impedem o uso eficiente de GPUs para aplicativos de processamento de áudio convencionais.

Devido às limitações nos modos de acesso às texturas e no tamanho das texturas, texturas 1D longas não podem ser indexadas facilmente. A filtragem de resposta ao impulso infinita (recursiva) não pode ser implementada de forma eficiente, pois os valores anteriores geralmente não estão disponíveis ao renderizar um determinado pixel em programas de fragmento. Incluir registradores persistentes para acumular resultados entre fragmentos pode resolver esse problema.

SÍNTESE DO SOM

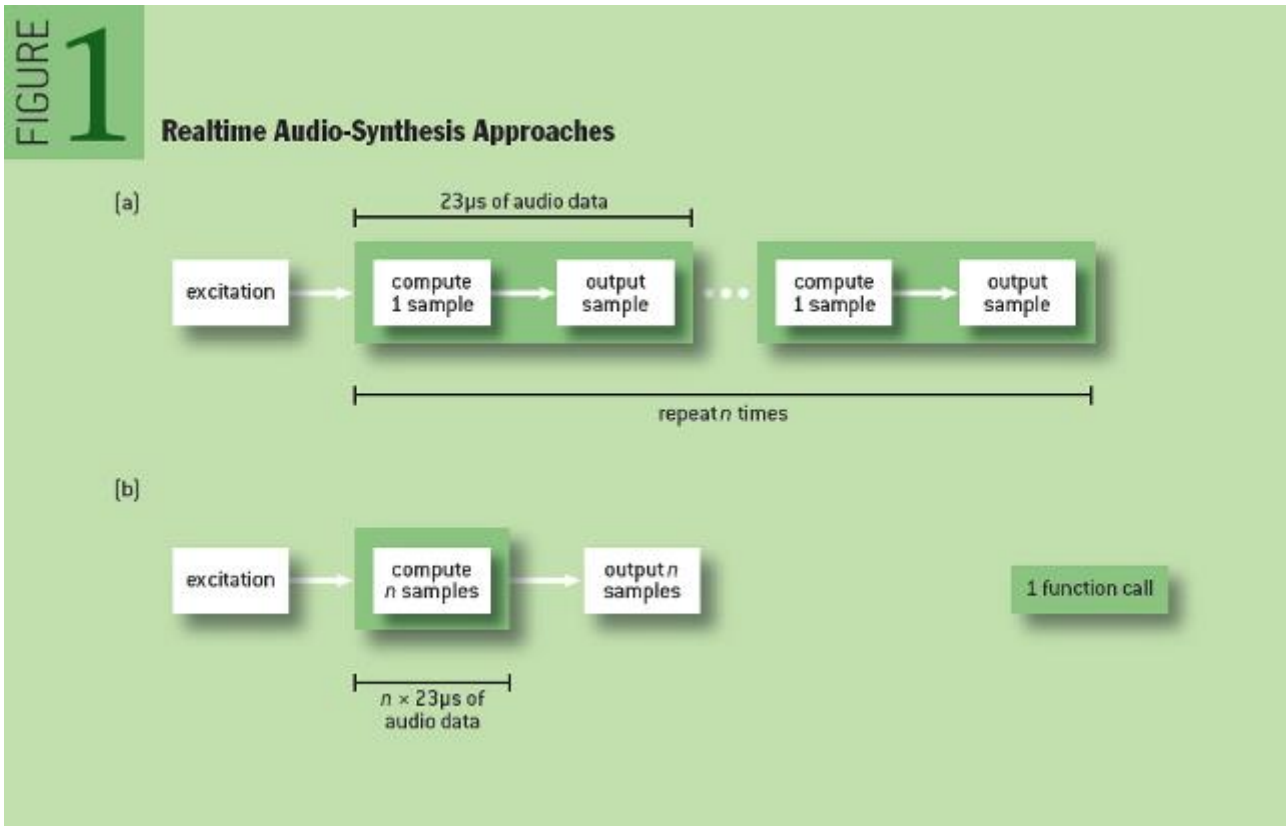
Existem várias abordagens para a síntese de som de modelagem física. Uma dessas abordagens, estudada extensivamente por Stefan Bilbao, 1 usa a aproximação da diferença finita para simular as vibrações de placas e membranas. A simulação de diferenças finitas produz sons realistas e dinâmicos (exemplos podem ser encontrados em http: //unixlab.sfsu.edu/~whsu/FDGPU). As simulações em tempo real baseadas em diferenças finitas de grandes modelos são normalmente muito intensivas em termos de computação para serem executadas em CPUs. A síntese de áudio em tempo real pode ser amplamente dividida em três etapas:

1. Excitação. Um evento de excitação sinaliza ao sintetizador que a geração de áudio em tempo real deve começar. Para acertar um prato virtual, por exemplo, um executante humano pode acertar uma tecla de um teclado, o que gera um evento de excitação.

2. Geração de amostra. Os dados de amostra de áudio são calculados para os sons desejados (por exemplo, a batida do prato).

3. Saída. Os dados gerados na etapa 2 são enviados ao software do sistema para reprodução pelo sistema.

A figura a seguir mostra duas abordagens para a síntese de áudio em tempo real: em (a), a abordagem “ingênua” calcula e produz uma única amostra por vez, enquanto em (b) a abordagem em buffer calcula várias amostras e as produz como um bloco.



3. Outros ambientes e trabalhos desenvolvidos

Dissertação de mestrado sobre Processamento de Áudio em tempo real em plataformas computacionais de alta disponibilidade e baixo custo

Numa metodologia que testou os seguintes seis algoritmos de processamento de áudio que requerem diferentes soluções para comparar performance de CPU e GPU, analisando e adaptando eles para as operações de GPU e CPU, analisando os resultados separadamente para cada algoritmo:

• Gain
• IIR 1st and 2nd order filters on an Allpass filter design
• FIR filters using the Window Design method • Biquad Filter (Direct Transform II)
• Amplitude and Ring Modulation of a Sine wave
• Dynamic Range compression

E os resultados mostram que a GPU teve, no geral, melhor performance com algoritmos que exigiam mais operações por amostra e quando vários canais estão envolvidos e o tamanho de quadro acima de 512 amostras é usado. Quanto mais altas essas variáveis, maior o desempenho relativo da GPU. As variações do coprocessador, em que a GPU e a CPU executam uma parte do algoritmo, eram prejudiciais em seu desempenho e, na maioria das situações, eram mais lentas do que o processamento da GPU ou da CPU independentemente. Todas as implementações alcançaram tempo real, exceto algumas execuções do filtro FIR na CPU. É interessante notar que contagens de threads maiores podem diminuir o desempenho. Isso inclui atenção cuidadosa aos recursos compartilhados, incluindo memória compartilhada e alocações SFU. Usar GPU para processamento de áudio em tempo real é possível e recomendado em ambientes de alta pressão de dados, desde que esses pontos cruciais sejam considerados.

SERVIDOR DE ÁUDIO COM CODIFICAÇÃO EM GPU

Este estudo sobre processamento paralelo de áudio em gpu proõe a implementação de um servido de áudio para melhorar sua realização e desviar das dificuldades com a GPU. http://www.inf.ufpr.br/bona/tg-luis.pdf

4. State of the art

DIGITAL AUDIO PROCESSING: ACUSTICA AUDIO – NEBULA 3
https://www.nvidia.com/content/GTC/documents/1011_GTC09.pdf

Nebula 3 é um plug-in VST de múltiplos efeitos capaz de replicar vários tipos de equipamentos de áudio caros, emulando o caráter e a personalidade do hardware analógico com rigorosa exatidão que se destaca de outros plugins, mas seu maior diferencial é que na versão pro (R$ 270,00), pode ser executado utilizando as mais recentes técnicas de processamento CUDA da NVIDIA, aliviando a CPU e trazendo as vantagens já citadas da gpu.

Suas tecnologias são divididas assim: Vectorial Engine - são os múltiplos efeitos simultâneos, não lineares, com variação de tempo e dependentes de nível - atualmente opera na CPU e o Kernel Engine – que processa grande número de kernels simultaneamente - é implementado inteiramente na GPU, o algoritmo

eficiente deixa resultados parciais para cada harmônico na GPU, apenas os resultados finais são transferidos da GPU.

NOSSA ANÁLISE CRÍTICA

Podemos perceber, baseado nos trabalhos apresentados que, apesar das dificuldades esperadas – a latência e taxas de transferência dos dados indo e voltando da CPU para GPU – e comprovadas como na dissertação de mestrado mencionada no tópico anterior, é inegável que a GPU proporciona um potencial muito elevado para processamento de áudio e deve ser mais estudada e usada nessa área, e esperamos poder aprender a tirar o melhor proveito dela na próxima etapa.

5. Referências

[7] Tsingos, N. (2009) Using programmable graphics hardware for acoustics and audio rendering. New York, Audio Engineering Society.

Processamento paralelo de audio em gpu:
http://www.inf.ufpr.br/bona/tg-luis.pdf

GPU sob o ponto de vista de arquiteturas paralelas, organização interna e utilização ao em sistemas de paralelismo massivo:
https://www.ic.unicamp.br/~cortes/mo601/trabalho_mo601/elisa_rodrigues_gpu/ra098329_relatorio.pdf

Audio processing on the gpu:
http://www.inf.ufrgs.br/gpuaudio/en/about/index.html

Have You Heard What a GPU Can Do? - A Revolution in Audio Processing:
https://www.nvidia.com/content/GTC/documents/1011_GTC09.pdf

Efficient 3D Audio Processing with the GPU:
https://www-sop.inria.fr/reves/personnel/Nicolas.Tsingos/publis/posterfinal.pdf

Why The Future Of Real-time Audio Processing Could Be In Graphics Cards:
https://www.pro-tools-expert.com/production-expert-1/2018/4/1/graphics-cards-why-the-future-of-sound-could-be-pictures

CUDA accelerated audio digital signal processing for real-time algorithms:
https://www.researchgate.net/publication/292854753_CUDA_accelerated_audio_digital_signal_processing_for_real-time_algorithms

Realtime GPU Audio
https://queue.acm.org/detail.cfm?id=2484010

https://www.ime.usp.br/~ajb/projeto/mestrado-ajb.pdf