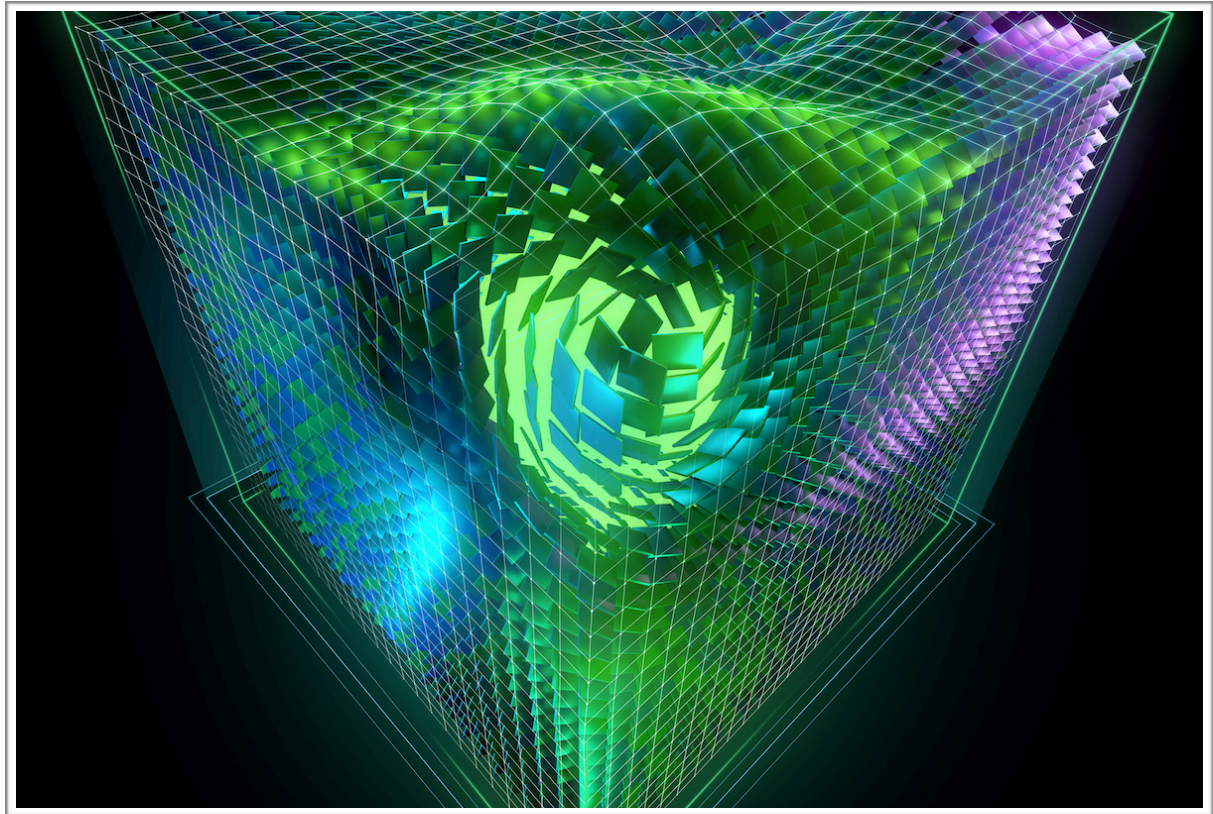


Relatório Computação em



Larga Escala

*Cálculo Computacional para Resolução de Determinantes pelo
Método de Gauss em CUDA*

Clony Abreu, Diogo

Junho 2019

Cálculo Computacional de Matrizes em CUDA

Utilizando Eliminação de Gauss com Pivot

Computação Paralela

Tradicionalmente, o software tem sido escrito para ser executado sequencialmente. Para resolver um problema, um algoritmo é construído e implementado como um fluxo serial de instruções. Tais instruções são então executadas por uma unidade central de processamento de um computador. Somente uma instrução pode ser executada por vez; após sua execução, a próxima então é executada.

Por outro lado, a computação paralela faz uso de múltiplos elementos de processamento simultaneamente para resolver um problema. Isso é possível ao quebrar um problema em partes independentes de forma que cada elemento de processamento pode executar sua parte do algoritmo simultaneamente com outros. Os elementos de processamento podem ser diversos e incluir recursos como um único computador com múltiplos processadores, diversos computadores em rede, hardware especializado ou qualquer combinação dos anteriores.

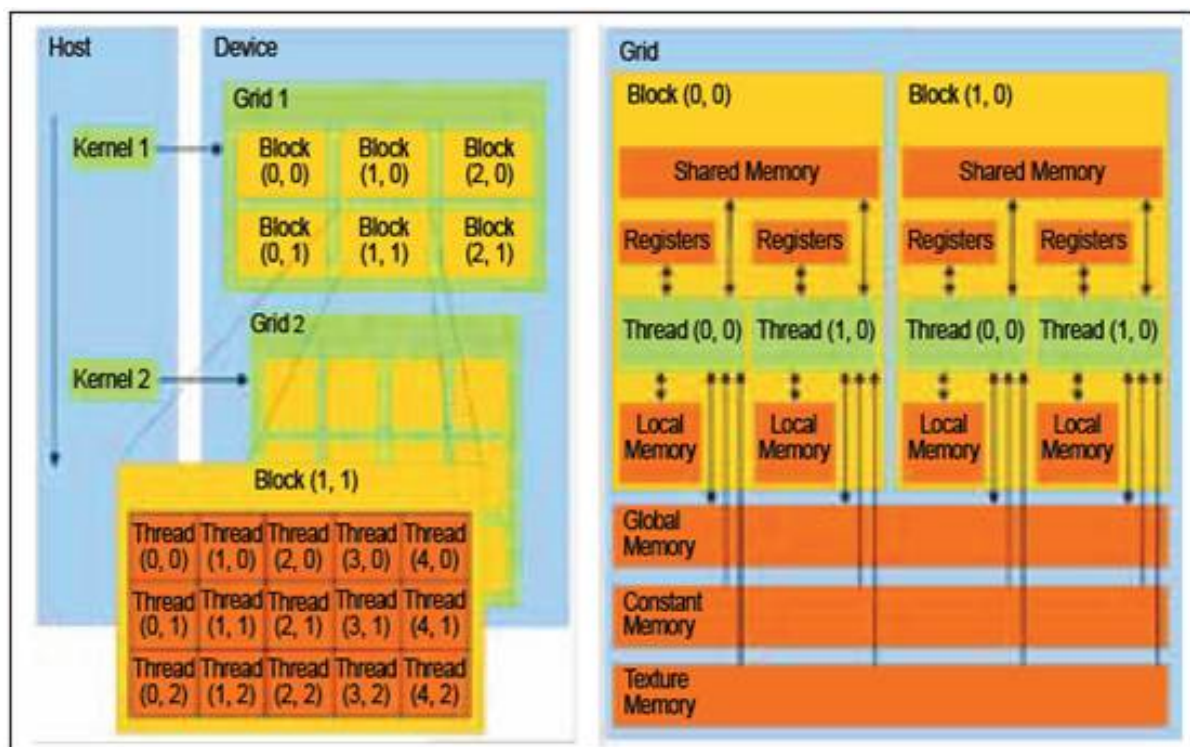


A computação paralela possui grande relevância na resolução de cálculos ou programas cuja complexidade computacional é considerada elevada. Um problema computacional é entendido como uma tarefa que é, em princípio, passível de ser resolvida por um computador (o que basicamente significa que o problema pode ser descrito por um conjunto de instruções matemáticas). Informalmente, um problema computacional consiste de instâncias do problema e soluções para essas instâncias do problema. A possibilidade de paralelizar o cálculo de resolução de problemas de complexidade computacional elevada, permite avanços no campo científico e tecnológico para a comunidade de pesquisadores. No contexto da complexidade computacional o ramo da matemática no estudo das matrizes traz contribuições importantes que são essenciais para as áreas da computação, engenharia civil, elétrica, mecânica, meteorologia, oceanografia e inúmeras outras.



O avanço da tecnologia reduziu os custos da computação paralela e, em conjunto com o desenvolvimento das pesquisas, ampliou os recursos disponíveis para a paralelização computacional. O desenvolvimento das unidades gráficas de processamento (GPU - Graphic Processing Unit), avançou especialmente na indústria de entretenimento e jogos eletrônicos. Cada vez mais as empresas desejam apresentar uma experiência imersiva aos utilizadores, oferecendo uma visualização gráfica que os aproximem da realidade. Com isso uma nova arquitetura de processamento gráfico foi necessária, o que potencializou a capacidade de processamento dessas unidades, quando comparadas com as unidades de processamento dos computadores (CPU). Além da performance gráfica de jogos, a indústria também disponibiliza soluções GPU para serem utilizadas em Computação Paralela. O caso aplicado ao estudo desse trabalho,

utiliza-se da API desenvolvida pela Nvidia e denominada de CUDA (Compute Unified Device Architecture). As atuais gerações de GPUs são basicamente processadores de fluxo multithread. Eles oferecem enormes quantidades de largura de banda e taxas de computação aritmética de ponto flutuante de precisão única. No processamento de fluxo, uma única função paralela de dados (kernel) é executada em um fluxo de dados, e isso é exatamente como o modelo de programação do cubo funciona. Um programa desenvolvido em Cuda é composto de duas partes: um código de host (CPU) que faz chamadas de kernel e um código de dispositivo (GPU) que realmente implementa o kernel. O código do host é conceitualmente um programa serial C, mas o código do dispositivo deve ser massivamente paralelo para aproveitar o poder da GPU.



Eliminação de Gauss

Eliminação de Gauss visa transformar um sistema de equações lineares em uma matriz triangular superior, a fim de resolver os elementos desconhecidos e derivar uma solução. Uma coluna dinâmica é usada para reduzir as linhas antes dele; depois da transformação, uma *back-substitution* é aplicada.

Portanto, para o cálculo da determinante será importante identificar os elementos da diagonal, aqui denominado de *pivot*, e percorrer a matriz realizando a operação de eliminação de Gauss, por iterações consecutivas de forma a reduzir os valores para a redução da complexidade computacional da operação.

