

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

LUCAS BARROS DE ASSIS

**Modelagem de ondas sísmicas através de
paralelismo de tarefas**

Monografia apresentada como requisito parcial
para a obtenção do grau de Bacharel em Ciência
da Computação

Orientador: Prof. Dr. Lucas Mello Schnorr

Porto Alegre
2021

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos André Bulhões

Vice-Reitora: Prof^a. Patricia Pranke

Pró-Reitora de Graduação: Prof^a. Cíntia Inês Boll

Diretora do Instituto de Informática: Prof^a. Carla Maria Dal Sasso Freitas

Coordenador do Curso de Ciência de Computação: Prof. Rodrigo Machado

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

RESUMO

A aplicação *Ondes3D* tem como objetivo realizar a simulação dos efeitos de uma onda sísmica. Apesar de contar com uma implementação paralela utilizando *OpenMP*, esse paralelismo só é obtido dentro de cada um de seus *macro-kernels*. O trabalho aqui apresentado estuda as alterações necessárias para dividir as estruturas utilizadas pelo simulador em ladrilhos, possibilitando posteriormente uma implementação em forma de tarefas utilizando a biblioteca *StarPU*. Essa divisão em tarefas permite um controle da granularidade dos processos paralelos através do tamanho dos ladrilhos utilizados, o que leva à uma otimização dos acessos em memória. Além disso, é possível que diferentes etapas do processo sejam executadas simultaneamente graças às especificações das dependências que o modelo de programação em tarefas inclui. Dessa forma, espera-se alcançar uma execução que aproveita arquiteturas *multi-core* de forma mais vantajosa que a versão original.

Palavras-chave: Programação paralela. programação baseada em tarefas. Ondes3D. StarPU.

Seismic waves modelling through task-based programming

ABSTRACT

The *Ondes3D* simulator aims to simulate the effects of a seismic wave. Even though it has a parallel implementation using *OpenMP*, this parallelism is only achieved inside each of its macro-kernels. The study presented in this document studies the modifications needed to split the structures used by *Ondes3D* in tiles, which allows a task-based implementation using the *StarPU* library. By splitting the code in tasks, it becomes possible to control the granularity of the parallel processes through the tile sizes, which enables a memory access optimization. Other than that, different steps of the computation can be executed simultaneously thanks to the dependency specifications from the task-based model. With these modifications, it may be possible to achieve an execution which exploits multi-core architectures even better than the original version.

Keywords: Parallel programming. Task-based programming. Ondes3D. StarPU.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

MPI Message Passing Interface

SUMÁRIO

1 INTRODUÇÃO	9
1.1 Contribuições.....	9
2 FUNDAMENTAÇÃO TEÓRICA	10
2.1 Ondes3D.....	10
2.2 Programação baseada em tarefas.....	10
2.2.1 StarPU	10
2.2.2 Matrizes <i>ladrilhadas</i>	10
3 PROJETO E IMPLEMENTAÇÃO	11
3.1 Estruturas de dados	11
3.2 Tarefaização dos <i>macro-kernels</i>	11
4 AVALIAÇÃO EXPERIMENTAL	12
4.1 Ambiente de testes.....	12
4.2 Resultados.....	12
5 CONCLUSÃO	13
REFERÊNCIAS.....	14

1 INTRODUÇÃO

Uma ferramenta importante para a mitigação dos riscos decorrentes de terremotos é a simulação da propagação de ondas sísmicas (DUPROS et al., 2010). A aplicação *Ondes3D* realiza essa simulação através do método de diferenças finitas, utilizando a biblioteca *OpenMP* para produzir paralelismo local e o protocolo *MPI* em contextos distribuídos. No entanto, o paralelismo local somente existe dentro de cada *macro-kernel*, isto é, não existe execução paralela entre diferentes etapas do cálculo. Uma possível melhoria pode ser obtida utilizando um modelo que permita um paralelismo ainda maior que o atual. A biblioteca *StarPU* é uma alternativa atual que implementa a programação baseada em tarefas para obter paralelismo. Dentro desse modelo, uma tarefa consiste em uma função cuja especificação inclui, além de seus parâmetros, os seus modos de acesso: somente leitura, somente escrita ou leitura e escrita. Essas tarefas são incluídas ao longo do código e, baseando-se na ordem de inserção e nos modos de acesso de cada uma delas, a biblioteca *StarPU* constrói um grafo acíclico direcionado representando as dependências entre essas tarefas. Durante a execução do programa, esse grafo é utilizado para escalonar as tarefas conforme a disponibilidade dos recursos computacionais. Implementando a aplicação *Ondes3D* sob a forma de tarefas, acredita-se ser possível alcançar um desempenho superior na execução de suas simulações. No entanto, o modelo de tarefas e a biblioteca *StarPU* exigem que mudanças sejam feitas nas estruturas de dados e no fluxo de código da aplicação. Este trabalho tem como objetivo estudar as alterações necessárias e analisar o nível de paralelismo alcançado pela implementação proposta.

1.1 Contribuições

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Ondes3D

2.2 Programação baseada em tarefas

2.2.1 StarPU

2.2.2 Matrizes *ladrihadas*

3 PROJETO E IMPLEMENTAÇÃO

3.1 Estruturas de dados

3.2 *Tarefaização dos macro-kernels*

4 AVALIAÇÃO EXPERIMENTAL

4.1 Ambiente de testes

4.2 Resultados

5 CONCLUSÃO

REFERÊNCIAS

DUPROS, F. et al. High-performance finite-element simulations of seismic wave propagation in three-dimensional nonlinear inelastic geological media. **Parallel Comput.**, v. 36, p. 308–325, 2010.