

Otimização de E/S Paralela em Simulações de Ondas Sísmicas

Natália G. Rampon¹, Jean L. Bez¹, Francieli Z. Boito², Philippe O. A. Navaux¹

¹Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Porto Alegre – RS – Brasil

²Departamento de Informática e Estatística
Universidade Federal de Santa Catarina (UFSC) – Florianópolis – SC – Brasil

{ngrampon, jlbez, navaux}@inf.ufrgs.br, francieli.boito@posgrad.ufsc.br

Resumo. *Este artigo descreve uma proposta de pesquisa que visa a melhoria de desempenho do sistema de E/S em aplicações de simulação de ondas sísmicas. Tais aplicações frequentemente produzem diversos arquivos pequenos como resultado, acessando assim diversas vezes os servidores de metadados. A otimização feita consiste em mapear tais arquivos virtualmente para um arquivo único. Para tal, a aplicação Ondes3D foi analisada no contexto do Grid5000.*

1. Introdução

Aplicações de computação de alto desempenho são geralmente executadas em clusters com diversas máquinas. Um sistema de arquivos paralelo (SAP) é responsável por prover uma estrutura compartilhada de armazenamento e garantir uma abstração do sistema de E/S para as aplicações. A entrada e saída paralela através desse sistema possui uma diferença marcante em velocidade se comparada com o processamento de dados, e tal diferença acaba por, muitas vezes, retardar as aplicações.

Esses sistemas tipicamente distribuem os arquivos entre diversas máquinas e possuem dois tipos de servidores, os servidores de dados e os de metadados. Os servidores de dados guardam os dados em si, enquanto os de metadados guardam informações sobre os dados, como tamanho e localização. Desta forma, para acessar um arquivo, os clientes precisam primeiro acessar os servidores de metadados. Tal ligação entre acessar dados e metadados, acaba por tornar o acesso a arquivos pequenos pior do que o acesso a arquivos grandes, isso porque, ao acessar um arquivo maior, temos mais informações a um custo menor de acesso a servidores de metadados [Ren et al. 2014]. Este fator acaba por tornar a escrita ou leitura de diversos arquivos pequenos um gargalo nas operações de E/S paralelas.

Analisamos, nessa pesquisa, a aplicação Ondes3D [Dupros et al. 2010], que utiliza discretamente o método numérico das diferenças finitas para resolver a equação diferencial da elastodinâmica. Na análise da aplicação, foi notado que ela produz diversos arquivos pequenos como resultado. Tal método de saída, como explicado, pode vir a aumentar seu tempo de execução devido aos diversos acessos aos servidores de metadados feitos para escrever em tais arquivos. Esse problema é ainda mais grave dado que a aplicação passa grande parte de seu tempo de execução em operações de E/S [Boito et al. 2017].

O objetivo dessa pesquisa é analisar o desempenho da aplicação Ondes3D quando diminuimos seu acesso a servidores de metadados através de uma diminuição do número de arquivos de saída. Foi utilizada para esse fim a biblioteca SIONlib [Frings et al. 2009],

que oferece uma abstração da escrita e leitura realizadas por vários processos em arquivos virtuais em um menor número de arquivos físicos.

O restante desse artigo se organiza do seguinte modo. As características da aplicação Ondes3D são discutidas na próxima seção. A metodologia experimental é mostrada na seção 3. Finalmente, a conclusão desse artigo é feita na seção 4.

2. Ondes3D

A aplicação Ondes3D é um simulador de ondas sísmicas. Ela representa as ondas sísmicas como um conjunto de equações elastodinâmicas, que são resolvidas pelo método das diferenças finitas. É recebido como entrada um modelo geológico de uma região, que representa o solo e o domínio tridimensional, e é produzida como saída uma série de dados sobre as condições do terreno nos pontos de interesse através dos passos da simulação.

Dois tipos de arquivos de saída são gerados pela aplicação: arquivos de estação e arquivos de superfície. Os arquivos de superfície contêm o estado de cada um dos planos xy, xz, e yz; e os de estação contêm o valor de nove variáveis através dos passos da simulação nas coordenadas em que as estações estão localizadas. A frequência com que os resultados são escritos nesses arquivos é um dos parâmetros de entrada do programa, podendo ser um valor para os de estação e outro para os de superfície. Os arquivos de estação são o enfoque dessa pesquisa, pois são gerados em maior número. Na simulação avaliada, são gerados 3 arquivos de superfície para cada fase de saída, sendo o arquivo do plano xy de 2.3MB, o do plano xz de 68KB, e o do plano yz 48KB; enquanto são gerados 650 arquivos de estação de 141KB cada.

A escrita dos resultados da simulação em tais arquivos era feita de forma muito frequente, com tamanhos de dados pequenos, a maioria de apenas 4 bytes, o que ocasionava uma lentidão na operação. Tal problema foi solucionado usando-se um buffer para acumular tais pedidos de escrita [Boito et al. 2017]. Usamos um buffer de 64KB em nossa implementação, pois foi a configuração de melhor desempenho observada para os arquivos de estação, com uma melhora de até 25.6% no desempenho das operações de I/O.

3. Metodologia

A fim de melhorar o desempenho de E/S da aplicação Ondes3D, é proposta a diminuição do número de arquivos gerados. Para tal propósito, iremos utilizar a biblioteca SIONlib, versão 1.7.0, que realiza um mapeamento de diversos arquivos virtuais em um número predefinido de arquivos físicos. Esta biblioteca foi escolhida, pois ela é não-invasiva e possui suporte para MPI e OpenMP, que são usados na aplicação Ondes3D.

Para realizar o mapeamento, a SIONlib possui uma rotina própria para abrir o arquivo em modo paralelo coletivo “`sion_paropen_mpi`” e outra para fechar “`sion_parclose_mpi`”, mas as rotinas de escrita e leitura tem a mesma sintaxe que suas respectivas funções padrões de E/S da linguagem C. É importante ressaltar que, enquanto as operações de abertura e fechamento do arquivo são coletivas, as leituras e escritas são feitas assincronamente. A biblioteca é que coordena as leituras e escritas no arquivo físico real, não precisando de cuidados especiais pelo usuário e nem pelo programa. Na Figura 1 pode ser observado como os diversos arquivos vistos pela aplicação são mapeados pela biblioteca para um único arquivo físico.

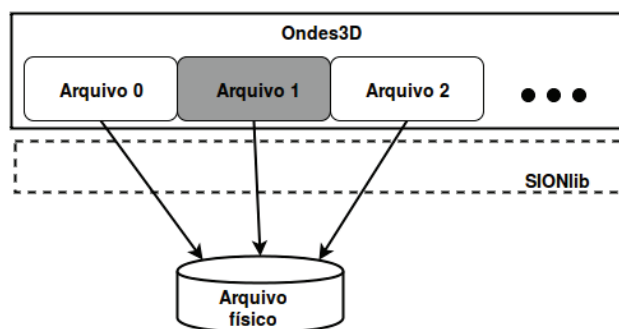


Figura 1. Funcionamento da biblioteca SIONlib

Os experimentos serão realizados no ambiente Grid5000, quatro máquinas do cluster Adonis serão usadas como servidores (tanto de dados quanto de metadados) com o sistema de arquivos paralelos PVFS para armazenamento e 32 máquinas do cluster Edel serão usadas como clientes. Cada nodo dos clusters Edel e Adonis possui dois processadores 4-core Intel Xeon E5520 2.27 GHz e 24GB de RAM. Um disco rígido SATA de 250GB é usado para armazenamento em cada servidor. A conexão entre os nodos é dada através de uma rede Ethernet de 1Gbps. Os nodos rodam sistema Debian 7, kernel versão 3.2, e usam Open MPI versão 1.6.5. PVFS versão 1.6.3 será utilizado como sistema de arquivos paralelos.

Os testes serão executados múltiplas vezes em ordem aleatória, para evitar efeitos inesperados de parâmetros não considerados e para garantir que as diferentes execuções sejam independentes. Isso permitirá uma análise estatística adequada dos resultados obtidos. Serão comparados o tempo de execução total da aplicação entre a versão padrão, a versão com o adicionamento do buffer de 64KB e a versão produzindo menos arquivos de saída (também munida do buffer), assim como o tempo que essas passaram exclusivamente na parte de E/S. A última comparação é onde se esperam maiores diferenças de resultados, por ser o enfoque dessa proposta de pesquisa.

Nos testes, será simulado um modelo tridimensional da Costa Azul francesa de tamanho 100km x 70km x 26km, com um espaçamento de grade de 200 metros. As simulações terão 1000 passos.

4. Conclusão e Trabalhos Futuros

Nesse artigo, propôs-se a diminuição do número de arquivos de saída gerados pela aplicação de simulação de ondas sísmicas Ondes3D. Esta otimização foi proposta para diminuir o gargalo no acesso aos servidores de metadados que a escrita de diversos arquivos pequenos origina. Para tal feito, propomos o uso da biblioteca SIONlib para passar os múltiplos arquivos usados originalmente em arquivos virtuais da biblioteca, que são então mapeados para um único arquivo no sistema de arquivos.

Como o presente artigo disserta sobre um trabalho em andamento, como trabalho futuro a execução dos experimentos e análise dos resultados serão feitas para aferir os benefícios da otimização. Além disso, as estratégias de otimização desenvolvidas no Ondes3D serão aplicadas a outras aplicações científicas que apresentam padrão de acesso semelhante.

Referências

- Boito, F. Z., Bez, J. L., Dupros, F., Aochi, H., Dantas, M., and Navaux, P. (2017). High performance i/o for seismic wave propagation simulation. In *25th International Euro-micro Conference on Parallel, Distributed and Network-Based Processing (PDP)*. IEEE.
- Dupros, F., Martin, F. D., Foerster, E., Komatitsch, D., and Romand, J. (2010). High-performance finite-element simulations of seismic wave propagation in three-dimensional nonlinear inelastic geological media. *Parallel Computing - Parallel Matrix Algorithms and Applications*, 36(5–5):308–325.
- Frings, W., Wolf, F., and Petkov, V. (2009). Scalable massively parallel i/o to task-local files. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, pages 1–11. ACM.
- Ren, K., Zheng, Q., Patil, S., and Gibson, G. (2014). IndexFS: Scaling File System Metadata Performance with Stateless Caching and Bulk Insertion. In *SC14: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 237–248.