

# Análise de Dados Observacionais sobre a Camada de Ozônio: uma Abordagem Usando MPI e OpenMP

Bruno M. Muenchen<sup>1</sup>, Taís T. Siqueira<sup>1</sup>, Lucas Nesi<sup>2</sup>  
Guilherme P. Koslovski<sup>2</sup>, Andrea S. Charão<sup>1</sup>  
Damaris K. Pinheiro<sup>1</sup>, Luiz Angelo Steffene<sup>3</sup>

<sup>1</sup> Universidade Federal de Santa Maria (UFSM) – Brasil

<sup>2</sup> Universidade do Estado de Santa Catarina (UDESC) – Brasil

<sup>3</sup> Université de Reims Champagne-Ardenne – França

**Resumo.** *Este trabalho implementa a análise de um conjunto de dados sobre a espessura da camada de ozônio terrestre, usando as ferramentas de programação paralela OpenMP e MPI, visando estudar a eficiência do processamento desses dados em servidores dedicados multicore e organizados em cluster. Os resultados obtidos mostram que a implementação com OpenMP consegue atingir speedup satisfatório, o que não ocorre com a versão em MPI.*

## 1. Introdução

Dados obtidos em simulações e observações são a matéria-prima de um novo paradigma da exploração científica [Hey et al. 2009]. Neste paradigma, grandes quantidades de dados passam por um processo de captura, curadoria e análise que, ao final, pode gerar novas descobertas. No que diz respeito às ciências atmosféricas, existem extensos repositórios com dados coletados durante décadas, por equipamentos distribuídos pelo globo ou embarcados em satélites, prestando-se a análises de vários aspectos da atmosfera terrestre.

Em um repositório mantido pela NASA (sigla em inglês de *National Aeronautics and Space Administration* – Administração Nacional da Aeronáutica e Espaço), do governo dos Estados Unidos, são disponibilizados dados observacionais de monitoramento da camada de ozônio usando instrumentos a bordo de satélites. Parte desses dados foi utilizada em trabalhos de pesquisa que investigaram efeitos do buraco de ozônio Antártico sobre o sul do Brasil [Pinheiro et al. 2011, Peres 2013]. Estes trabalhos motivam a realização de uma análise mais ampla dos dados existentes, usando ferramentas de computação de alto desempenho.

Um trabalho anterior explorou um *middleware* de computação distribuída e pervasiva para analisar este conjunto de dados [Steffene et al. 2016]. Tal trabalho é voltado para infraestruturas computacionais cujos recursos são heterogêneos e podem ser alocados sob demanda. Outro tipo de infraestrutura, que também pode beneficiar a análise dos dados em questão, consiste em servidores dedicados, com processadores *multicore* e possivelmente organizados em um *cluster*. O presente trabalho tem seu foco neste tipo de infraestrutura, apresentando implementações paralelas da análise dos dados sobre a camada do ozônio, usando as ferramentas MPI e OpenMP.

O restante deste texto está organizado como segue: na Seção 2 apresentam-se maiores informações sobre o conjunto de dados e sobre os cálculos necessários para a

análise; na Seção 3, descreve-se as estratégias de paralelização empregadas e, na Seção 4, apresentam-se os experimentos realizados e discutem-se os resultados sobre o desempenho obtido.

## 2. Caracterização dos Dados e da Análise

Os dados utilizados neste trabalho são coletados pelo satélite Nimbus-7, lançado pela NASA em outubro de 1978. Como descrito por [Mcpeters et al. 1998], os arquivos estão divididos em cabeçalho, nas três primeiras linhas, e em zonas de latitude, nas demais linhas. Cada zona de latitude contém os valores de longitude em grupos de três caracteres. No cabeçalho também é especificado o número de zonas de latitude e o número de longitudes em cada zona, bem como a cada quantos graus é feita a leitura.

Cada grupo de caracteres das longitudes representa a leitura feita pelo satélite em *Dobson Units* (DU). Uma DU representa o número de moléculas de ozônio necessárias para criar uma camada de ozônio puro de 0,01 mm de espessura a uma temperatura 0°C e pressão atmosférica de 1 atm. Sobre a superfície da terra, a espessura média da camada de ozônio é de 300 DU, ou uma camada com 3mm de espessura.

No repositório sobre monitoramento de ozônio mantido pela NASA<sup>1</sup>, tem-se arquivos de monitoramento diário desde 2005. Cada ano de monitoramento pelo satélite em questão representa mais de 1GB de dados brutos a serem analisados. Algumas métricas importantes para análise são a média, o desvio padrão, o valor mínimo e o valor máximo das leituras, em um dado período e em uma determinada região.

## 3. Implementações com MPI e OpenMP

O algoritmo implementado parte da leitura de um arquivo de entrada contendo  $n$  coordenadas as quais serão usadas para efetuar os cálculos da média das leituras, desvio padrão, mínima e máxima. Para os cálculos, o algoritmo sequencial faz a leitura de um arquivo de dados do Nimbus-7 e efetua cálculos para todas as coordenadas antes de seguir para o próximo arquivo.

Tanto para a versão OpenMP quanto MPI, a divisão do trabalho foi feita da mesma forma: cada *thread* recebe uma parte do conjunto de coordenadas e realiza os cálculos de forma idêntica ao algoritmo sequencial, ou seja, lê um arquivo de dados e faz os cálculos para as coordenadas as quais recebeu como parâmetro. A diferença entre as duas versões paralelas é apenas a necessidade de comunicação entre *threads* na versão MPI para o envio das coordenadas, devido ao fato de a versão OpenMP utilizar memória compartilhada e a versão MPI utilizar memória distribuída. Na versão MPI ainda há a necessidade de que todas as máquinas do *cluster* possuam uma cópia dos arquivos, pois não foi utilizado NFS.

## 4. Experimentos e Resultados

Os testes foram realizados em dois ambientes computacionais diferentes. O primeiro deles é uma máquina com 2 processadores Intel Xeon E5620 @ 2.40GHz de 4 núcleos e 12GB de memória total. O sistema operacional é Linux Debian 8.5 (Kernel 3.2.0), o compilador utilizado foi o GCC 4.9.2, com suporte a OpenMP, e usou-se Open MPI 1.6.5. O segundo

<sup>1</sup>Disponível em: <ftp://toms.gsfc.nasa.gov/pub/omi/data/ozone/>

ambiente conta com 14 máquinas com Intel Core i7-4770 @ 3.40GHz de 4 núcleos, 16GB de memória por nó e rede de 100Mbps, sendo usado apenas para testes com a versão MPI.

Para execução dos testes no primeiro ambiente, foram selecionados 260 arquivos, que compreendem o intervalo de tempo entre o início de janeiro e o final de setembro de 2016. Considerou-se uma área com 49 coordenadas diferentes.

Observando os dados apresentados nas Tabelas 1 e 2, é possível notar o aumento considerável no tempo de execução da solução desenvolvida com MPI. É possível justificar esse aumento devido à necessidade de transferência das mensagens que carregam as coordenadas com as quais cada *thread* deve trabalhar.

Já na solução desenvolvida em OpenMP, que não necessita essa transferência pois todas as *threads* compartilham o mesmo espaço de memória, o *speedup* se mostrou próximo do ideal quando utilizadas 2 ou 4 *threads*. Como o número de coordenadas é pequeno, o custo de *overhead* acabou ultrapassando o ganho em tempo quando utilizado um número maior de *threads*.

**Tabela 1. Tempo médio de execução (s) - Ambiente 1**

Threads	OpenMP	MPI
Sequencial	4,638	4,638
2	2,517	6,989
4	1,455	5,693
8	1,266	4,359
16	2,460	4,578

**Tabela 2. Speedup em relação ao sequencial - Ambiente 1**

Threads	OpenMP	MPI
Sequencial	1	1
2	1,842	0,664
4	3,188	0,815
8	3,663	1,064
16	1,885	1,013

Nas Tabelas 3 e 4, apresenta-se os resultados referentes as execuções no segundo ambiente. Como mencionado anteriormente, neste ambiente foi executada apenas a versão MPI do algoritmo. Os testes foram executados usando 260 e 3920 arquivos.

Para ambos os testes, o *speedup* mais próximo do ideal apresentou-se na execução com 7 *threads*, ou seja, quando dobramos o número de *threads* em relação à configuração usada como base para o cálculo do *speedup*, a execução é aproximadamente duas vezes mais rápida.

**Tabela 3. Tempo médio e speedup (260 arquivos) - Ambiente 2**

Threads	Tempo (s)	Speedup
3	2,027	1
5	1,533	1,322
7	0,860	2,354
9	1,087	1,865
11	1,112	1,822

**Tabela 4. Tempo médio e speedup (3920 arquivos) - Ambiente 2**

Threads	Tempo (s)	Speedup
3	29,597	1
7	12,334	2,399
9	15,898	1,861
13	22,719	1,302

## 5. Considerações Finais

Este artigo apresentou uma abordagem usando MPI e OpenMP para análise de dados da camada de ozônio. Os resultados obtidos com os algoritmos desenvolvidos foram

satisfatórios e mostram que as tecnologias testadas podem ser utilizadas para a análise de uma grande massa de dados. Como trabalhos futuros pretende-se testar os algoritmos em sistemas computacionais com poder computacional ainda maior e utilizando um conjunto de dados ainda maior.

## Referências

- Hey, T., Tansley, S., e Tolle, K., editors (2009). *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research, Redmond, Washington.
- Mcpeters, R. D., Bhartia, P. K., Krueger, A. J., Herman, J. R., Wellemeyer, C. G., Seftor, C. J., Jaross, G., Torres, O., Moy, L., Labow, G., Byerly, W., Taylor, S. L., Swissler, T., e Cebula, R. P. (1998). Earth Probe Total Ozone Mapping Spectrometer (TOMS) Data Products User's Guide.
- Peres, L. V. (2013). Efeito secundário do buraco de ozônio antártico sobre o sul do Brasil. Master's thesis.
- Pinheiro, D., Leme, N., Peres, L., e Kall, E. (2011). Influence of the antarctic ozone hole over south of brazil in 2008 and 2009. Technical report.
- Steffenel, L. A., Pinheiro, M. K., Pinheiro, D. K., e Perez, L. V. (2016). Using a pervasive computing environment to identify secondary effects of the antarctic ozone hole. *Procedia Computer Science*, 83:1007 – 1012.