

OPENMP APLICADO AO CÁLCULO NUMÉRICO EM PROBLEMAS NA FÍSICA

VINICIUS FONSECA HERNANDES¹; CARLOS ALBERTO VAZ DE MORAIS JÚNIOR³

Universidade Federal de Pelotas – viniciusfhernandes@gmail.com

Universidade Federal de Pelotas – carlosavjr@gmail.com

1. INTRODUÇÃO

O avanço tecnológico das últimas décadas possibilitou a resolução de problemas físicos cada vez mais complexos, através de computadores com uma grande quantidade de processadores e memórias. Dado o elevado custo na aquisição e alocação dessas máquinas, deve-se na medida do possível, utilizar mecanismos que extraiam o máximo de eficiência do poder computacional a disposição. Entre as atividades que exigem um custo computacional elevado e que podem se beneficiar da otimização está o cálculo numérico, o qual consiste na resolução de problemas através de métodos aproximativos.

Neste trabalho foi abordado como aumentar a eficiência da resolução de problemas físicos os quais utilizam recursos de cálculo numérico. A solução proposta foi a utilização da programação em paralelo, que consiste na utilização de múltiplos núcleos de um processador para execução de um programa, dividindo-o em partes a serem executadas simultaneamente pelos núcleos. O paralelismo foi dado utilizando OpenMP (Open Multi-Processing), um conjunto de diretrizes e rotinas que permitem a paralelização de processos de memória compartilhada, compatível com Fortran, C e C++. O que difere processos de memória compartilhada de processos de memória distribuída é que nesses uma memória comum pode ser acessada por vários processadores, enquanto naqueles cada processador tem sua própria memória HERMANNS, M. et al. (2002). O problema específico utilizado foi o cálculo numérico das equações dos parâmetros de ordem obtidos a partir do modelo vidro de spin Sherrington-Kirkpatrick KIRKPATRICK, S.; SHERRINGTON, D. et al. (1977).

Os objetivos desse trabalho consistem em aferir os benefícios da implementação de paralelismo com OpenMP, verificar se foi obtido um ganho computacional significativo e demonstrar a facilidade de conversão de códigos sequenciais em paralelos com OpenMP.

2. METODOLOGIA

Para a realização deste trabalho foi feita a implementação da ferramenta OpenMP para o problema específico de cálculo numérico das equações dos parâmetros de ordem obtidos a partir do modelo vidro de spin Sherrington-Kirkpatrick.

Primeiramente foi realizado um estudo da linguagem de programação Fortran para poder escrever o algoritmo necessário para resolução do problema. Em um primeiro momento foi implementado um método iterativo e o método de integração trapezoidal, para a resolução das equações:

$$m = \int Dz \tanh(\beta H(z))$$

e

$$q = \int Dz \tanh^2(\beta H(z)) ,$$

sendo $Dz = dz \exp(-z^2/2) / \sqrt{2\pi}$ e $H(z) = J\sqrt{q}z + J_0 m + h$. Depois foi implementado o código final utilizando o método iterativo como uma sub-rotina e os métodos de integração trapezoidal como funções a serem chamadas pelo programa principal (veja fluxograma na Figura 1). O programa final foi transformado de sequencial em paralelo, utilizando a ferramenta OpenMP; ou seja, o programa quando sequencial era executado por um só núcleo do processador e após paralelizado, através da diretriz !\$OMP PARALLEL SECTIONS, o código é dividido por meio de sub-diretrizes !\$OMP SECTION, onde cada sub-diretriz é executada por um núcleo do processador.

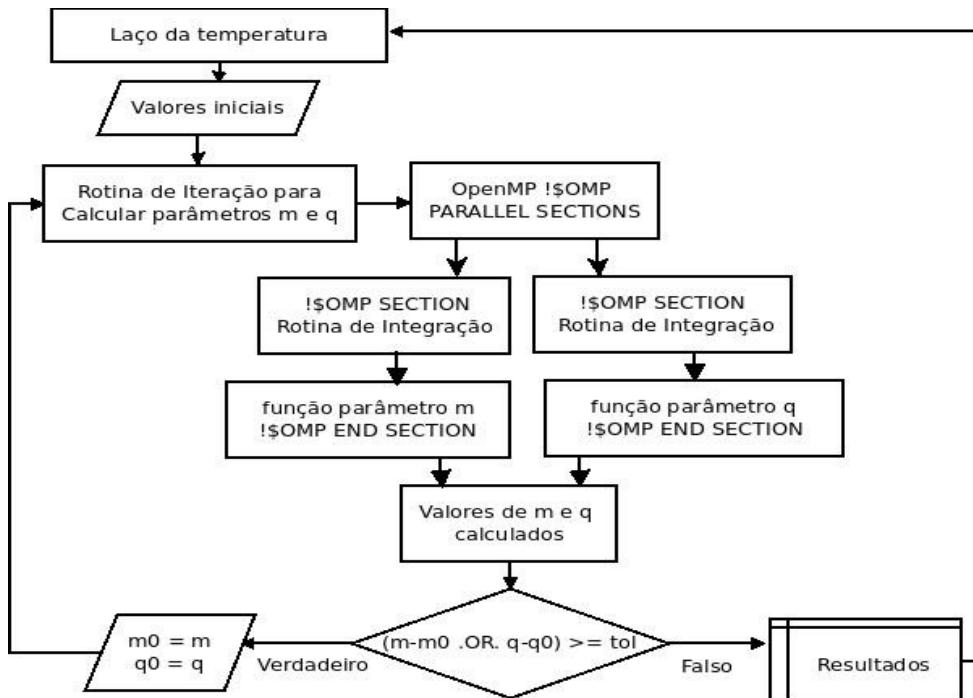


Figura 1: Fluxograma para cálculo dos parâmetros de ordem

Em seguida foram comparadas as velocidades necessárias para a execução do código por um computador com processador intel de dois núcleos. Ambos os códigos foram compilados no terminal em ambiente Linux, utilizando o compilador gfortran, depois executados com o comando de terminal “time ./a.out” para determinar a duração da execução.

3. RESULTADOS E DISCUSSÃO

O programa compilado com “gfortran -fopenmp”, executado usando o paralelismo, foi até 50% mais rápido do que o programa compilado sem “-fopenmp”, que o faz ser executado sequencialmente. Mais precisamente o tempo necessário para a execução sequencial do programa foi de 8,34 segundos, e quando executado em paralelo 4,23 segundos.

Pode-se aferir a facilidade na implementação do paralelismo com OpenMP no código já pronto, enquanto para tal, foram introduzidas somente seis linhas no arquivo a ser rodado.

4. CONCLUSÕES

Os resultados obtidos demonstram os benefícios da implementação do paralelismo em problemas físicos já prontos, principalmente utilizando OpenMP, que se mostrou acessível até mesmo para aqueles que não conheciam programação em paralelo. A diferença no tempo de execução pode aumentar a eficiência do poder computacional, obtendo os resultados desejados mais rapidamente.

Para dar continuidade ao trabalho já produzido pode-se aferir a utilidade de outras diretriz do OpenMP no mesmo problema, ou ainda testar o paralelismo em outras problemas já existentes.

5. REFERÊNCIAS BIBLIOGRÁFICAS

HERMANN, M. **Parallel Programming in Fortran 95 using OpenMP**, School of Aeronautics Engineering, Departamento de Motopropulsión y Termofluidodinámica. Universidad Politécnica de Madrid, 2002.

KIRKPATRICK, S.; SHERRINGTON, D. Infinite-ranged models of spin-glasses. **Physical Review B**, The American Physical Society, v. 17, p. 4384, 1977.