

Programação Paralela:

MPI

Silvio Romero de Araújo Júnior
Departamento de Engenharia Elétrica
Centro Universitário da FEI
São Bernardo do Campo, Brasil
silvioromerojr@yahoo.com.br

Reinaldo A. C. Bianchi
Departamento de Engenharia Elétrica
Centro Universitário da FEI
São Bernardo do Campo, Brasil
rbianchi@fei.edu.br

Resumo —. Este artigo apresenta uma melhoria no desempenho de Monte Carlo para o cálculo de área de um objeto cuja função é uma integral multidimensional utilizando o Message Passing Interface (MPI). O MPI é uma biblioteca padrão para troca de mensagens, amplamente utilizado em computação paralela.

Palavras chave – *Método de Monte Carlo; processamento paralelo; memória compartilhada; MPI*

I. INTRODUÇÃO

O MPI é um padrão utilizado em aplicações onde se necessita que haja uma troca de informações entre processos, que se executam em arquiteturas de memória distribuída. Normalmente usadas em computação de alta-performance e programação paralela. O MPI possui um conjunto de APIs de alto nível que fornece rotinas para gerenciamento de processos e operações de comunicação ponto-a-ponto e coletiva [1].

A. Objetivo

Utilizar a linguagem de programação C para implementar o método de Monte Carlo para o cálculo do volume de um objeto por meio de programação paralela baseada no MPI.

B. Motivação

O MPI é uma das estruturas de programação mais comuns usadas para implementar computação distribuída [2], tópico relevante devido à necessidade de otimização de recursos computacionais. Simplifica as tarefas de distribuição de trabalho e coordenação em um programa C, C++ ou Fortran multi-encadeado. Assim, avaliar seu funcionamento, compreender de forma sintetizada a implementação computacional de tal recurso, faz-se necessário.

C. Metodologia

Após pesquisa bibliográfica com base em livros e artigos abordando os conceitos aplicados ao relatório, aplicou-se o MPI para criar um programa para cálculo de volume de um sólido baseando-se no método de Monte Carlo.

II. REVISÃO DA LITERATURA

Calcular a integral de algumas funções pode ser uma tarefa complexa, dada a natureza do comportamento de determinadas funções. Uma abordagem aceitável para resolver isso é selecionar pontos aleatoriamente e calcular a média dos valores nesses pontos.

Esta é uma técnica estatística, por isso é importante fornecer um número grande de amostras para aumentar a precisão. Quanto mais pontos tivermos, mais próximos chegamos do valor real.

Esse processo demanda um custo computacional caro, cresce à medida que mais amostras são geradas. Uma vez que ele pode ser dividido em várias tarefas para ser executado simultaneamente, podemos melhorar o desempenho e acelerar a solução. Utilizando o paradigma de computação paralela, que implica no controle de duas ou mais operações que são executadas simultaneamente e cada uma delas implica seguir uma série de instruções.

III. IMPLEMENTAÇÃO PROPOSTA

Na implementação do código proposto (enviado como anexo) no presente relatório utilizou-se os seguintes recursos de Hardware:

- MacBook Pro (Retina, 15-inch, Early 2013);
- Processador: 2,4 GHz Intel Core i7;
- Memória: 8 GB 1600 MHz DDR3.

Com relação aos softwares utilizados, citam-se:

- Sistema operacional UBUNTU 18.04
- GCC 7.4

IV. EXPERIMENTOS E RESULTADOS

Para a validação do programa implementado, utilizou-se o mesmo para cálculo do volume do objeto apresentado no relatório da Aula 06 – Integração Numérica (O Método de Monte Carlo), variando-se o número de iterações conforme Tabela 1. Comparando os resultados simulados com e sem a biblioteca do MPI.

Tabela 1 – Resultados dos Testes

Iterações	Com o MPI		Sem o MPI	
	Valor Calculado	T(s)	Valor Calculado	T(s)
100000	22,0883	3,878	22,0315	12,980
1000000	22,1014	5,002	22,0913	129,712
10000000	22,0883	7,332	22,0902	219,081

A. Resultados

Pode-se notar comparando os resultados, que com o paralelismo o tempo de execução do programa decresce de forma significativa, provando a utilidade do MPI como ferramenta de computação distribuída.

V. CONSIDERAÇÕES FINAIS

O principal objetivo deste relatório foi examinar o possível ganho da utilização da programação paralela por meio da computação distribuída com a biblioteca do MPI. Usando este framework de código aberto, percebeu-se que ele aumenta consideravelmente a performance de aplicações que exigem muito processamento.

REFERÊNCIAS

- [1] Oliveira, A. T. “Uma Plataforma de Rede Definida por Software para Ambientes de Computação Paralela e Distribuída”. Dissertação de Mestrado, UFJF, 2019.
- [2] Pacheco, P. “An Introduction to Parallel Programming”, Morgan Kaufmann, Burlington, 2011.