

Relatório de Análise de Desempenho de Integração Numérica Usando Pthreads e OpenMP

1. Introdução

Este relatório apresenta a solução para o problema de integração numérica utilizando a regra do trapézio, paralelizada com as bibliotecas Pthreads e OpenMP. O objetivo é calcular a integral da função ($f(x) = \sqrt{10000 - x^2}$) no intervalo de 0 a 100, variando o número de threads e os intervalos de discretização para analisar o impacto na precisão e no tempo de execução.

2. Descrição do Problema

A integração numérica é uma técnica utilizada para calcular a área sob uma curva quando a integração analítica não é possível. Neste exercício, utilizamos a regra do trapézio para aproximar a integral da função ($f(x) = \sqrt{10000 - x^2}$) no intervalo de 0 a 100. A regra do trapézio divide o intervalo de integração em subintervalos menores e soma a área dos trapézios formados sob a curva.

3. Metodologia

Para resolver o problema, utilizamos duas abordagens de paralelização: Pthreads e OpenMP. Em ambas as abordagens, dividimos o trabalho entre múltiplas threads para acelerar o cálculo. A regra do trapézio é aplicada em cada thread, e os resultados parciais são somados para obter a integral final.

4. Resultados

Os resultados das integrações para diferentes números de threads e intervalos de discretização são apresentados nas tabelas abaixo.

Resultados com Pthreads

Intervalo de Discretização	1 Thread	2 Threads	4 Threads	8 Threads
0.000001	7853.981634	7853.981634	7853.981634	7853.981634
Tempo (s)	0.615000	0.328000	0.214000	0.135000
0.000010	7853.981634	7853.981633	7853.981633	7853.981633

Intervalo de Discretização	1 Thread	2 Threads	4 Threads	8 Threads
Tempo (s)	0.071000	0.039000	0.027000	0.016000
0.000100	7853.981631	7853.981631	7853.981631	7853.981631
Tempo (s)	0.010000	0.008000	0.008000	0.005000

Resultados com OpenMP

Intervalo de Discretização	1 Thread	2 Threads	4 Threads	8 Threads
0.000001	7853.981734	7853.981734	7853.981734	7853.981734
Tempo (s)	0.661000	0.393000	0.222000	0.141000
0.000010	7853.982634	7853.982634	7853.982634	7853.982634
Tempo (s)	0.073000	0.038000	0.026000	0.019000
0.000100	7853.991631	7853.991631	7853.991631	7853.991631
Tempo (s)	0.014000	0.006000	0.007000	0.005000

5. Análise dos Resultados

Precisão

A precisão da integral calculada foi consistente em ambas as implementações para diferentes intervalos de discretização. Conforme esperado, a precisão melhorou ligeiramente com a diminuição do intervalo de discretização. O valor esperado da integral é aproximadamente (7853.981634), e os resultados obtidos foram muito próximos a este valor, indicando uma alta precisão nas duas abordagens.

Tempo de Execução

Os tempos de execução diminuíram significativamente com o aumento do número de threads em ambas as implementações. No entanto, após um certo ponto, a redução no tempo de execução se tornou menos pronunciada devido à sobrecarga associada ao gerenciamento das threads.

- Para a discretização de (0.000001):
 - Em Pthreads, o tempo diminuiu de 0.615 segundos com 1 thread para 0.135 segundos com 8 threads.
 - Em OpenMP, o tempo diminuiu de 0.661 segundos com 1 thread para 0.141 segundos com 8 threads.

- Para a discretização de (0.00001):
 - Em Pthreads, o tempo diminuiu de 0.071 segundos com 1 thread para 0.016 segundos com 8 threads.
 - Em OpenMP, o tempo diminuiu de 0.073 segundos com 1 thread para 0.019 segundos com 8 threads.
- Para a discretização de (0.0001):
 - Em Pthreads, o tempo diminuiu de 0.010 segundos com 1 thread para 0.005 segundos com 8 threads.
 - Em OpenMP, o tempo diminuiu de 0.014 segundos com 1 thread para 0.005 segundos com 8 threads.

Comparação entre Pthreads e OpenMP

- **Precisão:** Ambos os métodos forneceram resultados muito próximos, com pequenas variações que podem ser atribuídas a diferenças internas nas bibliotecas e na maneira como a computação paralela é gerenciada.
- **Desempenho:** Ambas as implementações mostraram melhorias significativas no tempo de execução com o aumento do número de threads. No entanto, a sobrecarga de gerenciamento de threads impactou mais notavelmente o OpenMP em números maiores de threads comparado ao Pthreads.

6. Conclusão

As implementações de Pthreads e OpenMP demonstraram ser eficientes na paralelização do cálculo da integral. A precisão dos resultados foi consistente entre ambas as abordagens, e o tempo de execução diminuiu com o aumento do número de threads. A escolha do intervalo de discretização e do número de threads deve balancear a precisão desejada e o tempo de execução disponível.

7. Referências

- [Documentação do OpenMP](#)
- [Documentação do Pthreads](#)
- [Livro "Programming with POSIX Threads" de David R. Butenhof](#)