Implementando Algorítimos Integração Numérica com Paralelismo em C++

Danilo Sanchez Tuzita
(danilo_st@hotmail.com)

I. Resumo

Esse é um trabalho que tem como objetivo adaptar o algoritmo de Integração de Newton-Cotes, implementados no relatório 5 usando métodos de paralelismo para aumentar a performance.

II. Introdução

O algorítimo de integração numérica de Newton-Cotes pode ser implementado para ser processado em paralelo se dividirmos o intervalo de integração entre as *threads*.

III. Teoria

Para o entendimento desse trabalho é necessário conhecimentos básico de computação paralela.

IV. Proposta e Implementação

Nesse trabalho foi implementado três métodos de integração: O Método Retangular, A Regra Trapezoidal e A Regra de Simpson. Para uma explicação melhor detalhada vide relatório 5.

Os três métodos citados, para calcular-se a integral de certa função, divide o intervalo que se quer integrar, tira-se amostras nesse intervalo e estima-se uma forma aproximada da função nesse intervalo, como demonstra as figuras 1, 2 e 3. Considerando-se isso, podemos dividir esse processamento atribuindo a cada *thread* parte do intervalo total de integração.

Portanto para uma integração com intervalo de $[a, b]$, que será dividido em n quadrantes para t *threads*. A *thread* $n \bmod t$ calculará o intervalo $[a + i * s, a + (i + 1) * s]$ onde i é o número da iteração que começa em 0 e termina em n, e s é o tamanho dos quadrantes, ou seja, $\frac{b-a}{n}$.

V. Resultados

Para testar os métodos propostos, foi usado três funções, sendo essas: e^x, $\sqrt{1-x^2}$, e $e^{(-x^2)}$. Foi calculado a integral no intervalo $[0, 1]$ para cada uma das funções para cada método, com $n = 10^7$ e número de *threads* variado, para examinar a performance do algorítimo. Para os testes foi utilizado uma máquina com um processador *i5-6500 3.2GHz, 4 núcleos*.

As figuras 4, 5 e 6 demonstram como o algorítimo performa para cada uma das funções.

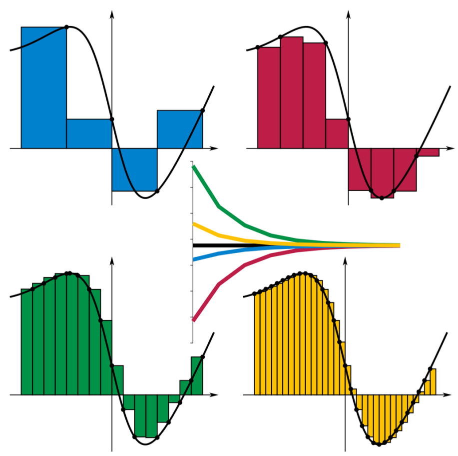

Fig. 1. Representação do Método Retangular. Fonte: Wikipédia

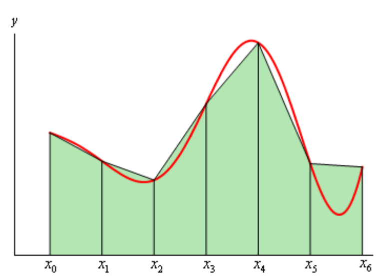

Fig. 2. Representação da Regra Trapezoidal. Fonte: http://tutorial.math.lamar.edu/Classes/CalcII/ApproximatingDefIntegrals.aspx.

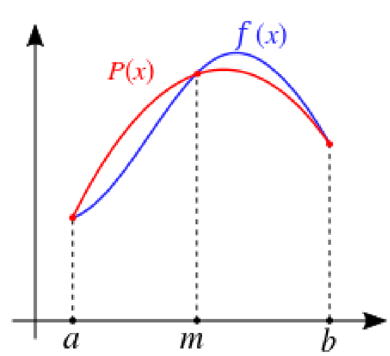

Fig. 3. Representação da Regra de Simpson. Fonte: Wikipédia

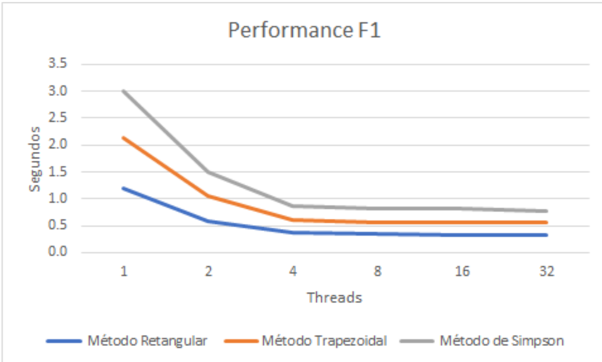

Fig. 4. Performance do cálculo da integral de e^x. Fonte: Autor

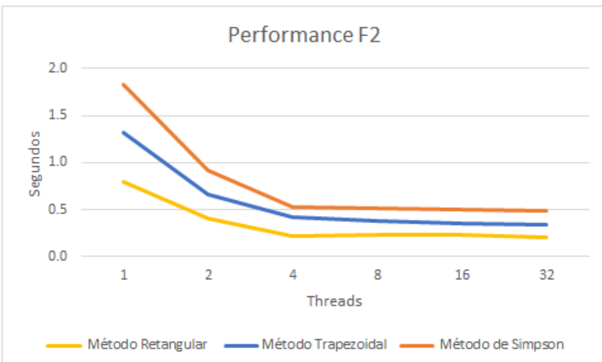

Fig. 5. Performance do cálculo da integral de $\sqrt{1-x^2}$. Fonte: Autor

VI. Conclusão

A utilização de métodos de integração numérica pode ser muito útil quando é difícil calcular a integral analiticamente ou quando não se tem uma função, apenas amostras dela em certos pontos.

Pode-se notar que quando os algorítimos são executados em paralelo o tempo é significantemente mais curto. Observa-se também que o tempo de processamento tende a ser duas vezes mais rápido quando também é duplicado a quantidade de *threads*, até o ponto onde a quantidade de *threads* excede a quantidade de núcleos de processamento como demonstra os resultados.

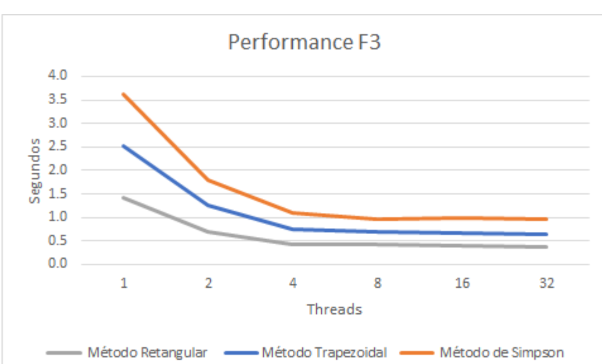

Fig. 6. Performance do cálculo da integral de $e^{(-x^2)}$. Fonte: Autor