Implementação e Análise do Algoritmo de Monte Carlo

Lucas Fontes Buzuti
Departamento de Engenharia Elétrica
Centro Universitário FEI
São Bernardo do Campo-SP, Brasil
lucas.buzuti@outlook.com

***Resumo*—Esse artigo tem uma finalidade acadêmica na implementação do algoritmo de Monte Carlo. O algoritmo foi testado a partir de duas funções adotadas, no qual dessas funções foram determinar suas respectivas áreas. Na implementação foi utilizada a programação orientada a objeto na linguagem *C++* em MPI (Message Passing Interface), tendo em foco a otimização e a velocidade na execução do algoritmo.**

***Index Terms*—algoritmo de monte carlo, otimização, programação orientada a objeto, C++, MPI**

I. Introdução

Esse artigo tem em seu objetivo a implementação do algoritmo de Monte Carlo. A implementação tem como alvo a utilização da linguagem *C++* em MPI. A linguagem *C++* é focada na otimização e na velocidade da execução de algoritmos e o MPI é um padrão para comunicação de dados em computação distribuída, além disso, é uma biblioteca de funções que oferece uma infraestrutura para essa tarefa.

As características de programação MPI são: um mesmo programa é iniciado em diversos computadores ou em núcleos diferentes, assim aumentando a eficiência na execução no código; em cada instância, uma parte da tarefa pode ser realizada; gerencia a criação e provém a maneira pela qual todas as instâncias se comunicam.

O algoritmo de Monte Carlo é um método numérico usado para encontrar soluções em problemas matemáticos, que pode ter diversas variáveis, que não podem ser facilmente solucionada.

II. Teoria

O algoritmo de Monte Carlo é um algoritmo randômico de uma classe de métodos estatísticos que se baseiam em amostragem aleatórias para obter resultados numéricos, que a entrada pode ser incorreta com um certa probabilidade. É usado em simulação física e computação estatística, pois consistem em funções complexas em que não é viável, ou é mesmo impossível, de obter uma solução analítica ou determinística [1].

Segundo [2], desde que a função em questão seja razoavelmente bem-comportado, pode ser estimada selecionando pontos aleatórios em um espaço N-dimensional. Assim, pode-se computar uma integral por Monte Carlo. Dado uma função multidimensional $f : X \to \mathcal{R}^M$ sobre o volume Ω, realiza-se a integral de f sendo:

$$I = \int_{\Omega} = f(\bar{x})d\bar{x}, \tag{1}$$

sendo que, pode-se calcular o volume Ω), um subconjunto do espaço $\mathcal{R}^M$ no qual deseja-se realizar a integração

$$V = \int_{\Omega} = d\bar{x}, \tag{2}$$

onde V é o volume de todo o espaço de integração. Coletando-se N pontos aleatórios uniformemente distribuídos em Ω, então têm-se que $\bar{x}_1 \ldots \bar{x}_n \in \Omega$. Portanto, pode-se usar a integração por Monte Carlo para estimar a função f sobre Ω, sendo assim, I pode ser aproximado por:

$$I = Q_n = V \left\langle f \right\rangle, \tag{3}$$

onde $\left\langle f \right\rangle$ é a soma das amostras da função: $\left\langle f \right\rangle = \frac{1}{N}\sum_{i=0}^{N-1} f(x_i)$. Pode-se estimar o erro sendo:

$$erro = V\sqrt{\frac{\left\langle f^2 \right\rangle - \left\langle f \right\rangle^2}{N}}, \tag{4}$$

onde $\left\langle f^2 \right\rangle$ é dado por: $\frac{1}{N}\sum_{i=0}^{N-1} f^2(x_i)$.

III. Proposta e Implementação

Esse artigo propõe utilizar a linguagem *C++* em MPI para a implementação[1] do algoritmo de Monte Carlo. Esta implementação foi feita em um computador com sistema operacional *Linux*, com o compilador *mpicxx* e para executar utilizou-se *mpiexec* com quatro processos a serem iniciados, a versão da linguagem utilizada foi a *C++11* e a versão do MPI foi a *mpich2*. Para desenvolver o algoritmo, foi utilizada a programação orientada a objeto (POO), onde visa a construção de classes e métodos.

Foi codificada uma classe denominada *MonteCarlo*. Para avaliar a implementação computou-se duas funções: $\int_0^1 \frac{4}{1+x^2}\,dx$ e $\int_0^1 \sqrt{x+\sqrt{x}}\,dx$.

[1] https://github.com/buzutilucas/scientific-programming/tree/master/Ex07

IV. Experimentos e Resultado

Para analisar o algoritmo, foi computado o erro das duas funções. As funções $\int_0^1 \frac{4}{1+x^2}\, dx$, $\int_0^1 \sqrt{x+\sqrt{x}}\, dx$ obtiveram os seguintes erros: 0.067 e 0.028 respectivamente com apenas 100 pontos aleatório. Ainda que, pode-se aumentar os números de pontos aleatório e diminuir o erro.

V. Trabalhos Correlatos

Método de monte Carlo está sendo utilizados em diversas áreas, tais como simulação física, estatística computacional e até em pesquisa para aprimoramento de modelos de Deep Learning [3].

VI. Conclusão

Nesse artigo pode-se implementar o algoritmo de Monte Carlo. Conclui-se que, quanto maior for a quantidade de pontos aleatório menor será o erro, e se for analisar o tempo de computação do programa, pode-se observar que: a computação teve um tempo em milisegundos, pois foi dividido os programas em quatro processo. Assim, deixando a computação do algoritmo mais eficiente.

Referências

[1] METROPOLIS, Nicholas et al. The beginning of the Monte Carlo method. Los Alamos Science, v. 15, n. 584, p. 125-130, 1987.

[2] BIANCHI, Reinaldo. Programação Científica 6a Aula. 2019 7 - 14 slides.

[3] Le, T.A., Igl, M., Rainforth, T., Jin, T., Wood, F. Auto-Encoding Sequential Monte Carlo. ICLR 2017.