

# **AVALIAÇÃO E COMPARAÇÃO DE DESEMPENHO DE ALGORITMOS NUMÉRICOS EM DISTINTOS AMBIENTES DE PROGRAMAÇÃO: C, LUA E MAPLE**

**Aluno: Ricardo Cavalcanti Marques**  
**Orientador: Therezinha Souza da Costa**

## **Introdução**

Freqüentemente na área da engenharia deve-se recorrer a métodos numéricos para solução de problemas. Nesses casos uma implementação computacionalmente eficaz e eficiente é de fundamental relevância. Além de obter uma resposta precisa, um bom algoritmo numérico deve nos fornecer uma resposta em um curto intervalo de tempo. Também podem ser levados em consideração alguns fatores secundários como clareza de sintaxe, facilidade de (re)compilação e portabilidade da linguagem e do executável gerado. Entretanto, para se satisfazer todas essas exigências, o ambiente de programação deve ser bem escolhido. Sobretudo quando se está trabalhando em uma aplicação crítica, o bom desempenho desses algoritmos é vital para o usuário final. É neste sentido que lançamos este projeto de pesquisa.

## **Objetivos**

O bom uso da aproximação numérica na solução de problemas matemáticos da engenharia é muito importante, mas não é muito estudado em nossos cursos. Com este projeto, esperamos incentivar mais alunos a buscarem novos caminhos nesta direção, procurando sempre considerar os aspectos computacionais mais importantes. Além disso, estamos levando mais alunos a conhecerem a nova linguagem de programação LUA, desenvolvida no Departamento de Informática da PUC-Rio, olhando-a do ponto de vista de sua utilização numérica. Outro aspecto importante é relativo ao estudo de parâmetros de avaliação do comportamento de software e de métodos de comparação de desempenho., o que abre caminhos para estudos em Engenharia de Software, além da Matemática da Computação.

## **Metodologia**

Neste projeto de pesquisa, fizemos implementações, em três linguagens de programação distintas, de quatro métodos numéricos para determinação de raízes de funções matemáticas e comparar os resultados dos testes. Para tornar a comparação mais interessante, escolhemos as linguagens C, Lua e Maple, que apresentam estruturas e recursos bem diferentes entre si. Nos métodos numéricos, optamos por aqueles eminentemente iterativos. Em particular, trabalhamos com o Método da Secante, o Método de Newton e suas variações.

Quanto às linguagens, C é uma linguagem compilada dos anos 70 que até hoje é muito utilizada para desenvolver aplicações de uso geral para janelas de prompt. Já Lua é uma linguagem leve (que ocupa pouco espaço em disco), portátil, interpretada (mas que pode ser compilada), dinamicamente tipificada, de sintaxe bem simples e rápida que vem ganhando muitos adeptos dentro e fora do Brasil. Finalmente Maple é um pacote de álgebra computacional criado nos anos 80 e que é muito usado no meio técnico-científico até os dias de hoje. Como linguagem Maple, é interpretado, dinamicamente tipificado, de sintaxe simples, porém pouco portátil e pesado.

Quanto aos métodos numéricos, o Método da Secante é o mais primitivo, recebendo dois valores iniciais de  $x$ , calculando a reta secante a certa função nesses valores de  $x$ , depois pegando-se o valor de  $x$  que intercepta o eixo das abscissas e repete-se o processo usando o  $x$  dessa interseção como novo primeiro valor inicial e colocando como novo segundo valor inicial o segundo valor inicial anterior. Já o Método de Newton, se baseia, em vez da reta

tangente a dois pontos, na reta secante à função em um determinado ponto, usando-se, então o operador diferencial. Por sua vez, o Método de Newton com Derivada Congelada vai usando a reta tangente à função em várias aproximações obtidas até que, a partir de um certo momento, passa a usar somente a reta tangente à função em uma certa aproximação. Por fim, o Método de Newton com Parâmetro de Correção leva em consideração a multiplicidade da raiz que pretende se achar, portanto, agilizando o Método de Newton quando usado em funções com raízes múltiplas.

Primeiro, precisávamos compreender bem os métodos numéricos e nos adaptarmos às linguagens selecionadas. Passada esta fase, pudemos construir nossos algoritmos e realizar os primeiros testes, apenas para se verificar se estão conceitualmente corretos. Assim sendo, estendemos nosso algoritmo para cronometrar os tempos de execução e gravar, de forma organizada, todos os cálculos do algoritmo em arquivos de log. Novamente, temos de testar a correção dessas operações. Nesta etapa podem ser feitas algumas avaliações de “conforto” da linguagem por parte do programador.

Com os algoritmos implementados, executamos várias vezes uma bateria de testes, composta por diversas funções, parâmetros e valores iniciais, cronometrando os tempos de cada execução. Assim, tomamos tempos médios de execução, os quais, usando amostragem estratificada, nos permitiram comparar o desempenho temporal das linguagens. Por outro lado, extraindo-se do log de testes informações como erros de aproximação e taxas de convergência dos métodos numéricos, pudemos comparar a qualidade dos cálculos numéricos nas três linguagens.

## Conclusões

Neste projeto de pesquisa fomos capazes de comprovar a eficiência de C e Lua sobre o Maple, tanto em aspectos de desempenho temporal como de qualidade numérica. Este fato decorre do Maple oferecer ferramentas e conforto em demasia ao usuário, o que o torna uma aplicação lenta e que não utiliza todo o potencial numérico de um computador, provavelmente, pecando na forma como representa números reais. Portanto, o ideal é não usar aplicações prontas, porém desenvolvê-las para um certo fim. Aliás, é isso que se espera que um engenheiro tenha de fazer freqüentemente, justificando a importância do ensino de linguagens de programação em cursos de Engenharia.

Como C é uma linguagem compilada, já era esperado que ela fosse bem rápida, porém para usuários inexperientes, C possui uma sintaxe mais “truncada”. Lua, entretanto, possui uma sintaxe muito fluida e é interpretada, o que também a torna ainda mais amigável ao usuário. Uma vez, que Lua se revelou quase tão eficiente quanto C, ela é ideal para o que propusemos nesse projeto de pesquisa: resolver problemas matemáticos comuns às engenharias.

## Referências

1 - RUGIERO, Márcia A. Gomes e LOPES, Vera Lúcia da Rocha, **Cálculo Numérico: Aspectos Teóricos e Computacionais**. McGraw-Hill, 1988.

2 – IERUSALIMSCHY, R. , DE FIGUEIREDO, L. H. e CELES, Waldemar, **Lua 5.1 Reference Manual** . Lua.org, August 2006

3 – Manual de Referência do Maple 9

4 – [www.lua.org](http://www.lua.org)