

# Linpack Benchmark

Isabela P. Loebel  
UNIOESTE  
Foz do Iguaçu, Brasil  
isabelaloebel@gmail.com

Kevin W. de Oliveira  
UNIOESTE  
Foz do Iguaçu, Brasil  
kevin337oficial@gmail.com

Gustavo J. Borges  
UNIOESTE  
Foz do Iguaçu, Brasil  
gustavo.juliano.borges@gmail.com

**Abstract** – The objective of this paper is the study of the benchmark tool called Linpack, as well as its history and the main topics that encompass its operation. Linpack benchmark is defined as a library of programs for solving computed linear algebra problems. Initially written in Fortran 66 in the 1970s by Jack Dongarra, it is based on another library named *Basic Linear Algebra Subprograms* (BLAS-1). Linpack encompasses three benchmarks, namely Linpack 100, Linpack 1000 and HPLinpack. Mainly used for performance measurements and ranking of supercomputers, it is the adopted package for *Top500* list metrics, assimilating the computer's architecture and the problem to be solved. The present work also brings a brief review of the disadvantages of the Linpack Benchmark.

**Resumo** – O objetivo desse trabalho é o estudo da ferramenta de benchmark chamada Linpack, assim como seu histórico e os tópicos principais que englobam seu funcionamento. Linpack benchmark é definido como como uma biblioteca de programas para resolução de problemas de álgebra linear computados. Escrito inicialmente em Fortran 66 na década de 1970, por Jack Dongarra, baseia-se em outra biblioteca nomeada *Basic Linear Algebra Subprograms* (Subprogramas de Álgebra Básica Linear, BLAS-1). O Linpack engloba três benchmarks, sendo elas o Linpack 100, o Linpack 1000 e o HPLinpack. Usado, principalmente, para medições

de desempenho e classificação de supercomputadores, é o pacote adotado para métricas da lista *Top500*, assimilando a arquitetura do computador e o problema a ser solucionado. O presente trabalho traz também uma breve revisão sobre as desvantagens do Linpack Benchmark.

**Palavras-chave** – Benchmark, pontos-flutuantes, desempenho, computadores.

## 1. INTRODUÇÃO

Supercomputador é um computador ou um conjunto de computadores interconectados projetados para executar processos que exigem uma grande quantidade de recursos computacionais [1].

Esses computadores são usados para calcular tarefas intensas e processamento de dados densos em diversas áreas, como meteorologia, prospecção de petróleo, bioinformática e mineração de dados.

A necessidade de medição do nível de desempenho desses supercomputadores fez-se notada para obter-se melhores avaliações dentro do âmbito computacional. Historicamente, essa é uma questão levantada de forma ampla, não só para supercomputadores, mas computadores em geral: quão rápida é a máquina?

Em demanda para obtenção da resposta dessa pergunta trivial, surgiu os benchmarks, que é o ato de executar um programa de computador, um conjunto de programas ou outras operações, a fim de avaliar o desempenho relativo de um

objeto, normalmente executando uma série de testes padrão e ensaios nele [2].

Com a evolução das arquiteturas de computadores, ficou mais desafiador comparar a performance de diferentes sistemas de computação somente olhando suas especificações. Por isso, testes foram desenvolvidos para serem feitos em diferentes sistemas, permitindo que esses resultados possam ser comparados entre as diferentes arquiteturas [2].

Ramificando, existem duas classes de benchmark, em aplicativos e sintético. Os benchmarks são desenvolvidos para usar um determinado tipo de comportamento em um componente ou sistema. Benchmarks "sintéticos" fazem isso através de programas especialmente criados para impor o tipo de comportamento desejado no componente. Em oposição, benchmarks em aplicativos executam programas mais realísticos no sistema. [2].

O Linpack benchmark, como alguns especialistas relatam, surgiu por acidente. Inicialmente, foi lançado apenas como um apêndice do manual de usuário do Linpak. Tinha como intenção apenas apresentar ao usuário uma ideia de quanto tempo levaria para resolver um problema, mas devido à sua notoriedade, houve então, maior interesse pelo aprofundamento do tópico, que seria então, desenvolvido como uma funcionalidade à parte do pacote Linpack.

## 2. HISTÓRIA

O projeto Linpack é definido como uma biblioteca de programas para resolução de problemas de álgebra linear computados [4][5]. Conceituado como um pacote de softwares matemáticos, o Linpack baseia-se na biblioteca BLAS (*Basic Linear Algebra Subprograms*) [6][7] e foi escrito em Fortran 66, desenvolvida por Jack Dongarra, Jim Bunch, Cleve Moler e Gilbert Stewart,

com o propósito de usá-la em supercomputadores na década de 1970.

Posteriormente, o pacote Linpack acabou sofrendo uma substituição pelo Lapack, que trazia como diferencial sua origem projetada para funcionamento eficiente em supercomputadores de memória compartilhada, trazendo uma visão de arquitetura moderna [8]. Contudo, o uso do Linpack ainda é significativo na área, visto que o pacote resolve de forma mais óbvia os determinantes do que o seu sucessor, Lapack [10][11].

Como já mencionado, o Linpack Benchmark, surgido do projeto Linpack, foi listado inicialmente como parte do manual de usuário do Linpack, publicado em 1979, relatado por alguns como um “surgimento por acidente”. A intenção era dar aos usuários do pacote original uma ideia de quanto tempo levaria para resolver certos problemas de matriz, e avaliar a capacidade de processamento de computadores comerciais [3].

Para suprir o mercado de paralelismo, a implementação paralela do Linpack benchmark foi batizada de HPL (*High Performance Linpack*), e é usada para comparar e classificar supercomputadores para a lista TOP500. Essa é uma lista de estatísticas de sistemas de propósito geral que são de uso comum para aplicativos de ponta e são classificados por seu desempenho no benchmark Linpack.

A lista é atualizada duas vezes ao ano, em junho e em dezembro, desde 1993 [3], e os autores da lista são explicitamente críticos quanto à avaliação, visto que asseguram que sistemas de propósito geral, nesse contexto, referem-se à sistema de computador que deve poder ser usado para resolver uma série de problemas científicos. Qualquer sistema projetado especificamente para resolver o problema de benchmark Linpack ou que tenha como objetivo principal uma alta

classificação no TOP500 é desclassificado [6].

### 3. CARACTERÍSTICAS

O Linpack como benchmark traz como objetivo a medição da taxa de execução de um ponto flutuante de um computador, onde é determinado pela execução de um programa que resolva um sistema denso de equações lineares, extrapolando os resultados de desempenho obtidos por vinte e três computadores diferentes resolvendo um problema de matriz de tamanho cem.

O tamanho da matriz foi escolhido devido às limitações de memória e CPU (*Central Processing Unit*) da época, construída de forma que dez mil entradas de ponto flutuante entre -1 e 1 são geradas aleatoriamente para preencher uma matriz densa geral e então, a decomposição LU (*lower-upper*), que fatora uma matriz como o produto de uma matriz triangular inferior e uma matriz triangular superior, com pivoteamento parcial é usada para o tempo.

Visando a atualização do problema proposto, novas versões com problemas adicionais foram dispostas, como matrizes de ordem trezentos e mil, incluindo restrições, a fim de permitir novas hipóteses de otimização à medida que as arquiteturas de hardwares começaram a implementar operações matriz-vetor e matriz-matriz.

Antes de citar os benchmarks, é necessário o entendimento do termo Mflop/s, que é uma taxa de execução, isto é, milhões de operações de ponto flutuante por segundo. Sempre que esse termo for usado, ele se referirá a operações de ponto flutuante de 64 bits e as operações serão adição ou multiplicação. Gflop/s refere-se a bilhões de operações de ponto flutuante por segundo e Tflop/s refere-se a trilhões

de operações de ponto flutuante por segundo.

Ao longo das fases do Linpack, foram arquitetados três benchmarks, sendo eles: Linpack 100, Linpack 1000 e por último, o HPLinpack. Todos eles devem ter sua precisão de acordo com a fórmula 1.

$$\|Ax-b\|_A \times n\epsilon \leq O(1)$$

Fórmula 1 – Fórmula de precisão.

Onde  $\epsilon$  é a precisão da máquina,  $n$  é o tamanho do problema,  $\|\cdot\|$  é a norma da matriz e  $O(1)$  corresponde à notação big-O.

Para cada sistema de computador, as seguintes quantidades são relatadas:

- $R_{\max}$ : o desempenho em GFLOPS para o maior problema executado em uma máquina.
- $N_{\max}$ : o tamanho do maior problema executado em uma máquina.
- $N_{1/2}$ : o tamanho em que metade da taxa de execução  $R_{\max}$  é alcançada.
- $R_{\text{pico}}$ : o GFLOPS de desempenho de pico teórico para a máquina.

Nota-se que, o pico teórico não se baseia em um desempenho real de uma execução de referência, mas em um cálculo manual para determinar a taxa de pico teórica de execução de operações de ponto flutuante para a máquina. Representa um limite superior no desempenho. Ou seja, o fabricante garante que os programas não ultrapassarão essa taxa [11].

O desempenho máximo teórico é determinado pela contagem do número de adições e multiplicações de ponto flutuante (com precisão total) que podem ser concluídas durante um período, geralmente o tempo de ciclo da máquina. Por exemplo, um Intel Itanium 2 a 1,5

GHz pode completar 4 operações de ponto flutuante por ciclo ou um desempenho máximo teórico de 6 GFlop/s [11].

Também é relevante ressaltar que, resolver um sistema de equações requer  $O(n^3)$  operações de ponto flutuante, mais especificamente, com a fórmula 2, adições e multiplicações de ponto flutuante.

$$2/3n^3 + 2n^2$$

Fórmula 2 – Fórmula de operações.

Tendo esse embasamento, pontuamos a seguir as especificidades dos benchmarks anteriormente citados.

### 3.1. LINPACK 100

Também chamado de benchmark Linpack Fortran. Em síntese, muito semelhante ao benchmark publicado originalmente em 1979. A solução é obtida por eliminação gaussiana com pivoteamento parcial, onde  $n$  é 100 na fórmula 2, e a ordem da matriz densa  $A$  que define o problema.

Para executar o programa, você precisará fornecer uma função de temporização chamada `SECOND`, que deve relatar o tempo de CPU decorrido. As regras básicas para executar este benchmark são que você não pode fazer alterações no código Fortran, nem mesmo nos comentários. Somente a otimização do compilador pode ser usada para melhorar o desempenho.

Em específico, esse benchmark mede o desempenho de duas rotinas da coleção de software Linpack. Essas rotinas são DGEFA e DGESL (essas são versões de precisão dupla, já SGEFA e SGESL são suas contrapartes de precisão simples). O DGEFA executa a decomposição LU com pivoteamento parcial e o DGESL usa essa decomposição para resolver o sistema de equações lineares dado.

A maior parte do tempo é passado na DGEFA. Uma vez que a matriz foi

decomposta, DGESL é usado para encontrar a solução. Esse processo requer operações de ponto flutuante  $O(n^2)$ , em oposição às operações de ponto flutuante  $O(n^3)$  de DGEFA.

Em suma, as regras básicas para esse benchmark são:

- O programa deve ser executado como está, sem alterações no código-fonte;
- Somente o compilador pode executar otimizações, isso em tempo de compilação;
- Usuário fornece uma função temporizada chamada `SECOND`, que retorna o tempo de execução da CPU para o processo.

Seu tamanho e a falta de flexibilidade de software não permitem que a maioria dos computadores modernos atinjam seus limites de desempenho. No entanto, ainda pode ser útil prever desempenhos em código escrito pelo usuário numericamente intensivo usando otimização de compilador [9].

### 3.2 LINPACK 1000

Traz consigo o desafio de desempenho mais próximo do limite da máquina, pois além de oferecer um tamanho de problema maior, uma matriz de ordem 1000, são possíveis alterações no algoritmo. Buscando maior flexibilidade, as regras básicas para executar este benchmark são um pouco mais relaxadas, pois você pode especificar qualquer solução de equação linear que desejar, implementada em qualquer idioma.

Um requisito é que seu método deve calcular uma solução e a solução deve retornar um resultado com a precisão prescrita, que não pode ser reduzida e o número de operações sempre será considerado como na fórmula 2, com  $n$  sendo 1000 [9].

Sumarizando, suas regras básicas são:

- Permissão de uma substituição completa do usuário da fatoração LU e das etapas do solucionador;
- A sequência de chamada deve ser a mesma das rotinas originais;
- O tamanho do problema deve ser de ordem 1000;
- Satisfação da solução condizente com a fórmula 1.

### 3.3. HPLINPACK

Visando o fato que, os benchmarks anteriores não são adequados para testar computadores paralelos, esse benchmark foi introduzido voltado à essa amostra específica. Chamado de *Highly Parallel Computing Benchmark*, é usado como referência para o relatório TOP500 [12].

Este benchmark tenta medir o melhor desempenho de uma máquina na resolução de um sistema de equações. O tamanho do problema e o software podem ser escolhidos para produzir o melhor desempenho, sendo assim, o tamanho  $n$  do problema pode ser feito tão grande quanto for necessário para otimizar os resultados de desempenho da máquina.

Novamente, a fórmula 2 será considerada como contagem da operação, independente do algoritmo utilizado, mas traz a restrição do uso do algoritmo de Strassen, já que distorce a taxa de execução real [12].

Condensando assim, as regras básicas:

- Permissão para uma substituição completa do usuário da fatoração LU e das etapas do solucionador;
- Satisfação da solução condizente com a fórmula 1;
- Não há restrições para o tamanho do problema.

## 4. CRÍTICAS

O Linpack benchmark trata-se de um benchmark padrão, podendo esse ser definido como um conjunto predefinido de problemas ou tarefas projetadas para testar o desempenho de um sistema ou programa sob certas condições.

Sendo assim, os benchmarks podem ser classificados em duas categorias: sintéticos e baseados em aplicativos. Benchmarks sintéticos são problemas artificiais que visam enfatizar um aspecto específico de um sistema, como largura de banda de memória ou overhead de comunicação. Os benchmarks baseados em aplicativos são problemas do mundo real que refletem a carga de trabalho típica de um sistema, como processamento de imagem ou simulação científica [14].

Quanto à desvantagem do Linpack benchmark, o principal ponto de acusação é a funcionalidade de sincronização e comunicação, sendo assim, um sistema paralelo com bom desempenho no Linpack pode não funcionar bem em um problema diferente que exija mais comunicação ou sincronização. Em segundo lugar, eles podem não explicar a variabilidade e imprevisibilidade de sistemas ou programas paralelos. Sistemas ou programas paralelos podem se comportar de maneira diferente dependendo do número, tipo e configuração de processadores, máquinas ou redes [11][14].

Atrai-se o sucesso do Linpack benchmark ao fator de escalabilidade do HPLinpack, ao fato de gerar um único número, tornando os resultados facilmente comparáveis e à extensa base de dados históricos que têm associado.

No entanto, logo após seu lançamento, o benchmark Linpack foi criticado por fornecer níveis de desempenho "geralmente impossíveis de serem obtidos por todos, exceto por

alguns poucos programadores que otimizam tediosamente seu código para aquela máquina e aquela máquina sozinha", porque ele apenas testa a resolução de sistemas lineares densos, que não são representativos de todas as operações normalmente realizadas em computação científica [13].

Jack Dongarra, principal autor do benchmark Linpack, afirmou que, embora eles enfatizem apenas a velocidade de "pico" da CPU e o número de CPUs, não é dada ênfase suficiente à largura de banda local e à rede, criando-se assim, a possibilidade de procura expansão do escopo do relatório de benchmark, visto que é de suma importância incluir mais características de desempenho e assinaturas para um determinado sistema [13].

Segundo Jack Dongarra, espera-se que o tempo de execução necessário para obter bons resultados de desempenho com o HPLinpack aumente efetivamente ao decorrer dos anos futuros. Em uma conferência realizada em 2010, o mesmo afirmou que espera tempos de execução de 2,5 dias em alguns anos [13].

## 5. REFERÊNCIAS

- [1] DE ROSE, C.; Navaux, P.; **Arquiteturas Paralelas**. Editora Sagra-Luzzatto, 2003. (Série Livros Didáticos – Número 15)
- [2] WIKIPEDIA.; **Benchmark** (Computação). 2022. Disponível em: <[https://pt.wikipedia.org/wiki/Benchmark\\_\(computa%C3%A7%C3%A3o\)](https://pt.wikipedia.org/wiki/Benchmark_(computa%C3%A7%C3%A3o))>. Acesso em: 30 de set. de 2023.
- [3] TAMPELINI, L. G.; **A 39ª lista Top500**. Os computadores mais rápidos do mundo. 2012. Disponível em: <[https://www.ic.unicamp.br/~cortes/mo601/trabalho\\_mo601/leonardo\\_tampellini\\_top500/Trabalho%20Top500%20\(final\).pdf](https://www.ic.unicamp.br/~cortes/mo601/trabalho_mo601/leonardo_tampellini_top500/Trabalho%20Top500%20(final).pdf)>. Acesso em: 30 de set. de 2023.
- [4] WIKIPEDIA.; **LINPACK**. 2020. Disponível em: <<https://pt.wikipedia.org/wiki/LINPACK>>. Acesso em: 30 de set. de 2023.
- [5] DONGARRA, J. J.; Luszczek, P.; Petitet, A.; **The LINPACK Benchmark: Past, Present, and Future**. 2001. Disponível em: <<https://www.netlib.org/utk/people/JackDongarra/PAPERS/hpl.pdf>>. Acesso em: 30 de set. de 2023.
- [6] TOP500. **The Linpack Benchmark**. 2023. Disponível em: <<https://www.top500.org/project/linpack/>>. Acesso em: 30 de set. de 2023.
- [7] DONGARRA, J. J.; **Frequently Asked Questions on the Linpack Benchmark and Top500**. 2007. Disponível em: <[https://www.netlib.org/utk/people/JackDongarra/faq-linpack.html#\\_Toc27885709](https://www.netlib.org/utk/people/JackDongarra/faq-linpack.html#_Toc27885709)>. Acesso em: 01 de out. de 2023.
- [8] SKIENA, S.; **LAPACK and LINPACK -- Linear Algebra PACKages**. 2008. Disponível em: <<https://www3.cs.stonybrook.edu/~algorith/implement/lapack/implement.shtml>>. Acesso em: 01 de out. de 2023.
- [9] DONGARRA, J. J.; LUSZCZEK, P.; **LINPACK Benchmark**. Encyclopedia of Parallel Computing. Springer, Boston, MA. 2011. Disponível em: <[https://doi.org/10.1007/978-0-387-09766-4\\_155](https://doi.org/10.1007/978-0-387-09766-4_155)>. Acesso em: 01 de out. de 2023.
- [10] USP. **Desempenho Benchmark & Profiler**. 2017. Disponível em: <<https://edisci.plinas.usp.br/pluginfile.php/3638688/course/section/835248/Desempenho%20-%20P2.pdf>>. Acesso em: 01 de out. de 2023.

[11] THE ASTROLOGY PAGE. **O que é o benchmark linpack?**. 2023. Disponível em: <<https://pt.theastrologypage.com/linpack-benchmark>>. Acesso em: 01 de out. de 2023.

[12] NETLIB. **Linpack**. Disponível em: <<https://www.netlib.org/linpack/>>. Acesso em: 01 de out. de 2023.

[13] WIKIPEDIA. **LINPACK benchmarks**. 2023. Disponível em: <[https://en.wikipedia.org/wiki/LINPACK\\_benchmarks](https://en.wikipedia.org/wiki/LINPACK_benchmarks)>. Acesso em: 01 de out. de 2023.

[14] LINKEDIN. **What are the advantages and disadvantages of using standard benchmarks for parallel computing?**. 2023. Disponível em: <<https://www.linkedin.com/advice/0/what-advantages-disadvantages-using-standard>>. Acesso em: 01 de out. de 2023.