

Predição de Eficiência Energética em Aplicações de HPC

Isabella Muniz^{1,2}, Micaella Coelho¹, Alexandre Porto¹, Hiago Rocha¹,
Kary Ocaña¹, Carla Osthoff¹

¹ Laboratório Nacional de Computação Científica (LNCC)

²Faculdade de Educação Tecnológica do Rio de Janeiro (FAETERJ)
Petrópolis – RJ, Brasil

{isamuniz, micaella, xandao, mayk, karyann, osthoff}@lncc.br

Abstract. This work proposes the evaluation of a machine learning-based methodology to support the selection of execution configurations in High-Performance Computing (HPC) environments. Using the Extra Trees Regressor (ETR) model and execution data from the RAxML application, it was possible to estimate combinations of nodes and threads that minimize the Energy-Delay Product (EDP). The results showed a low predictive error (MAE = 0.05), demonstrating the feasibility of the approach for reducing energy consumption and optimizing supercomputer utilization.

Resumo. O presente trabalho propõe a avaliação de uma metodologia baseada em aprendizado de máquina para apoiar a seleção de configurações de execução em ambientes de Computação de Alto Desempenho (HPC). Com o uso do modelo Extra Trees Regressor (ETR) e dados de execução da aplicação RAxML, foi possível estimar combinações de número de nós e de threads que minimizam o Energy-Delay Product (EDP). Os resultados demonstraram baixo erro preditivo (MAE = 0,05), evidenciando a viabilidade da abordagem para reduzir o consumo energético e otimizar a utilização de supercomputadores.

1. Introdução

O consumo de energia em ambientes de Computação de Alto Desempenho (HPC) tem se consolidado como um dos grandes desafios da área, em razão tanto da crescente demanda por aplicações científicas de larga escala, utilizando recursos computacionais para execução, quanto da necessidade de manter infraestruturas cada vez mais sustentáveis. Estimativas recentes indicam que os *data centers* já respondem por mais de 1% de todo o consumo de eletricidade mundial, o que reforça a urgência em buscar estratégias que conciliem desempenho e eficiência energética. Sob essa perspectiva, surge a necessidade de desenvolver estratégias para maximizar a eficiência energética sem degradar significativamente o desempenho das aplicações.

Nesse contexto, decisões relativas à alocação de recursos computacionais, tais como a definição do número de nós ou a escolha do número de *threads* por nó, podem exercer um impacto considerável no desempenho e na eficiência energética das aplicações. Opções inadequadas podem resultar não apenas em tempos de execução mais longos, mas também em subutilização dos recursos disponíveis, aumento de gargalos de comunicação, saturação de memória ou de E/S, e, consequentemente, elevação desnecessária do consumo energético do sistema.¹

¹<https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks>

Diversos trabalhos na literatura têm buscado abordar esse desafio por meio de estratégias baseadas em Aprendizado de Máquina, explorando a correlação entre características de execução e métricas de consumo energético para otimizar decisões de alocação de recursos. Por exemplo, o estudo de [Lorenzon et al. 2025] aplica o algoritmo de *Random Forest* para determinar, de forma preditiva, a frequência mais adequada da GPU e o nó de execução mais eficiente no supercomputador Frontier. Apesar de seus méritos, essa abordagem apresenta limitações, pois concentra-se exclusivamente em GPUs e no uso de DVFS, não contemplando a seleção de configurações em ambientes baseados em CPUs nem aplicações que demandam análise de eficiência energética sob diferentes combinações de nós e *threads*.

Este estudo é uma extensão de trabalho anterior [Coelho et al. 2025], no qual foi proposta e validada uma metodologia baseada no modelo *Extra Trees Regressor* (ETR) para recomendação de configurações de execução em sistemas de HPC, utilizando validação *Leave-One-Group-Out* (LOGO) e a métrica *Energy-Delay Product* (EDP), conhecida na literatura por combinar consumo energético e tempo de execução, como critério de otimização. No presente trabalho, o foco recai sobre a avaliação detalhada do modelo preditivo, com ênfase na análise das importâncias das variáveis e do erro médio absoluto (MAE), de modo a investigar a precisão da abordagem. Os resultados mostraram que a abordagem alcança baixo erro preditivo, sendo capaz de selecionar configurações próximas ao ótimo sem necessidade de reexecuções experimentais, demonstrando sua viabilidade como suporte automatizado à tomada de decisão em supercomputadores.

2. Metodologia de Predição Avaliada

A metodologia de predição avaliada neste trabalho se baseia no modelo ETR [Porto et al. 2026], cuja função é aprender padrões de desempenho a partir de execuções anteriores da aplicação. A Figura 1 mostra o fluxo de execução da metodologia, o qual consiste em duas fases: *Construção do Predictor* e *Uso do Predictor*, descritas a seguir.

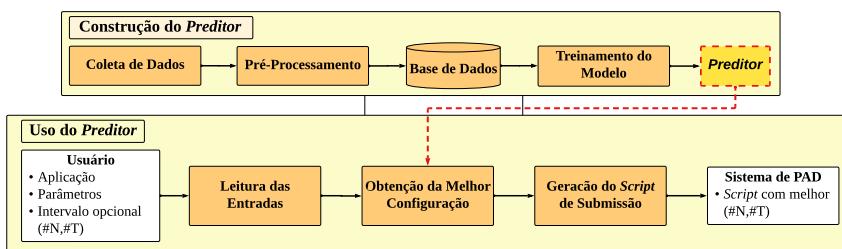


Figura 1. Fluxo Geral da Metodologia Proposta

Construção do Predictor. Nesta fase, é construído o modelo de ETR denominado *Predictor*, capaz de estimar o EDP de uma aplicação com base em execuções anteriores. Inicialmente, são coletados dados de execuções no ambiente HPC, variando os parâmetros da execução, e pré-processamos os resultados para remover ruídos identificados. Os dados são armazenados em um banco de dados e, em seguida, é treinado o *Predictor*.

Uso do Predictor. Após o treinamento, o *Predictor* é disponibilizado como um executável em *shell script*, permitindo que o usuário forneça o binário da aplicação, seus parâmetros de entrada e, opcionalmente, intervalos de valores para #N e #T. O modelo estima o EDP para cada combinação de configuração considerada e seleciona automaticamente a

que apresenta menor valor previsto, sem necessidade de executar a aplicação. Em seguida, gera um *script* de submissão completo, incluindo a configuração recomendada, parâmetros da aplicação e diretivas do gerenciador de filas (e.g., SLURM), pronto para execução no ambiente de HPC.

3. Metodologia Experimental

Ambiente computacional. O ambiente computacional consistiu em nós com 2 CPUs Intel Xeon Gold 6252 (48 núcleos no total), 384 GB de RAM, RHEL 8.8 (kernel 4.18) e interconexão InfiniBand, com gerenciamento de *jobs* pelo SLURM 17.02.

Aplicação avaliada e execuções. O estudo de caso utilizou a aplicação RAxML para inferência genética, variando oito tamanhos de arquivos de alinhamentos do vírus da Dengue e valores de *bootstrap* 100 e 1000. Foram testadas combinações de 1 e 5 nós com 2, 24 e 48 *threads*, realizando 5 execuções por configuração, totalizando 96 configurações e 480 execuções.

Medidas de desempenho. Como métricas de desempenho, foram considerados o tempo de execução e o consumo de energia. O tempo foi obtido em segundos por meio do comando *sacct* (SLURM), enquanto o consumo de energia foi coletado utilizando os contadores do *Running Average Power Limit* (RAPL), com tratamento para possíveis ocorrências de *overflow*. Com base nisso, EDP foi calculado como: $EDP(app) = Energia(app) \times Tempo(app)$, onde *Energia(app)* e *Tempo(app)* correspondem ao consumo de energia e ao tempo de execução medidos durante a execução da aplicação *app*.

Avaliação do modelo. O modelo foi avaliado com validação cruzada LOGO, em que grupos correspondentes a combinações de *bootstrap* e tamanho das entradas foram usados alternadamente como teste e treinamento. As réplicas de cada configuração foram agregadas pela mediana do EDP para reduzir ruído, repetindo o processo até que todos os grupos fossem testados. Para a avaliação, foi considerada a média dos erros absolutos (MAE) de todos os *folds*. Além disso, avaliamos a importância de cada variável do modelo (número de nós, número de *threads*, *bootstrap* e tamanho das entradas) para identificar quais têm maior influência na acurácia do modelo.

4. Resultados

A análise do desempenho em termos de erro absoluto mostrou que, considerando as 96 configurações avaliadas, o MAE global foi 0,05 e a mediana dos erros foi 0,00, evidenciando que, na maior parte dos casos, as previsões de EDP ficaram muito próximas dos valores observados. A dispersão dos resultados, refletida pelo desvio-padrão de 0,12, indica alguma variabilidade entre as configurações. Em termos de quartis, 25% foram 0,00, enquanto em 75% ficaram abaixo de 0,03, reforçando que a maior parte das previsões apresentou erro baixo. Os valores mínimo e máximo variaram de 0,00 a 0,74, mostrando que, embora raramente, algumas combinações de entrada resultaram em erros mais elevados, os resultados indicam bom ajuste preditivo, com poucos *outliers*.

A Figura 2 apresenta os valores de importância (eixo x) atribuídos a cada uma das quatro variáveis de entrada (eixo y). Observa-se que algumas variáveis exercem influência substancialmente maior sobre a predição do EDP. O Tamanho da entrada, embora apresente certa variação entre as iterações, mantém-se como a variável de maior impacto,

sugerindo que decisões baseadas nesse parâmetro podem ser mais eficazes na escolha de configurações energeticamente eficientes. Em comparação, o número de nós (*NNodes*) e o parâmetro *bootstrap* exibem importâncias intermediárias e relativamente próximas entre si, enquanto o número de *threads* (*Thread*) se destaca pela menor contribuição média, indicando efeito mais limitado sobre a previsão do EDP.

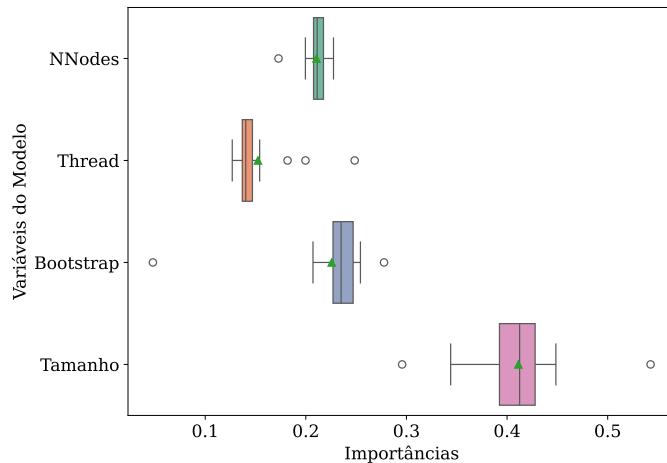


Figura 2. Importância das Variáveis de Entrada

5. Conclusão

O presente trabalho avaliou uma metodologia para predição da eficiência energética de execuções em sistemas de HPC. Utilizando a aplicação RAxML como estudo de caso, os resultados indicaram que a abordagem é capaz de fornecer previsões com baixo erro, obtendo um MAE global de 0,05 em diferentes cenários de entrada. A análise de importância das variáveis revelou que o tamanho da entrada exerce papel determinante na acurácia das estimativas de EDP. Como trabalhos futuros, pretende-se ampliar a avaliação do modelo com *benchmarks* representativos de HPC e um conjunto mais abrangente de configurações de alocação de recursos.

Referências

- Coelho, M., Porto, A., de A. Rocha, H. M. G., Muniz, I., Cardoso, D. O., Ocaña, K., Lorenzon, A., Navaux, P. O. A., and Osthoff, C. (2025). Modelagem preditiva de energy-delay product para otimização de submissões em supercomputadores. In *Anais do Simpósio em Sistemas Computacionais de Alto Desempenho (SSCAD)*, Brasil. Sociedade Brasileira de Computação (SBC).
- Lorenzon, A. F., Beck, A. C. S., Navaux, P. O., and Messer, B. (2025). Energy-efficient gpu allocation and frequency management in exascale computing systems. In *ISC High Performance 2025 Research Paper Proceedings (40th International Conference)*, pages 1–11. Prometheus GmbH.
- Porto, A. H., Coelho, M., Rocha, H. M., Osthoff, C., Ocaña, K., and Cardoso, D. O. (2026). Assuming the best: Towards a reliable protocol for resource usage prediction for high-performance computing based on machine learning. *Future Generation Computer Systems*, 175:108070.