

AMEC: Modelo para Análise de Aplicações com Microsserviços Baseado no Monitoramento do Consumo Energético

Lucas Oliveira da Silva¹, Rodrigo da Rosa Righi¹

¹Universidade do Vale do Rio dos Sinos (UNISINOS)
São Leopoldo – RS – Brazil

lucasooliveiranh@gmail.com, rrrighi@unisinos.br

Resumo. *A computação em nuvem está presente no cotidiano por meio de câmeras, sensores, sistemas de gestão e aplicativos móveis. A arquitetura de microsserviços, amplamente utilizada na nuvem, destaca-se pela escalabilidade, flexibilidade e identificação de problemas. Contudo, o aumento do poder computacional gera elevado consumo energético, exigindo estudos para maior eficiência. Este trabalho apresenta o modelo AMEC, que analisa gargalos de desempenho com foco no consumo energético em microsserviços. O modelo monitora em tempo real CPU, memória, rede e disco, gerando um laudo energético detalhado. Resultados iniciais indicam sua eficácia comparada ao estado da arte.*

1. Introdução

A Computação em nuvem sustenta diversos sistemas, como lojas virtuais, segurança e saúde, além de aplicações como reconhecimento de fala, processamento de imagens e LLMs. Essas soluções permitem acesso a informações de qualquer lugar e a qualquer momento, sendo fundamentais para a gestão de grandes volumes de dados. Estima-se que até 2025, o número de usuários globais de redes sociais ultrapasse 4,41 bilhões [Mahesar et al. 2024], o que aumenta a demanda por processamento de dados com baixa latência. Isso, por sua vez, contribui para o crescimento no consumo de energia, com a computação em nuvem representando atualmente 1,5% da eletricidade global e estimando-se que os data centers consumam até 13% até 2030 [Biçici 2024].

Diante dos custos e desafios relacionados ao alto consumo de energia, a busca por otimizações nos modelos de computação em nuvem é essencial. A arquitetura de microsserviços oferece benefícios como modularidade e escalabilidade, mas também traz desafios, como o aumento do consumo energético, impactando o meio ambiente e elevando os custos operacionais [Araújo et al. 2024]. Este trabalho apresenta o modelo AMEC, que visa analisar gargalos de desempenho a partir do consumo energético em microsserviços. O seu diferencial é o foco do consumo energético do dispositivo utilizado, uma área pouco explorada na literatura. O monitoramento em tempo real de recursos como CPU, memória e disco é complementado por uma análise detalhada do impacto no consumo de energia, resultando em um laudo energético com diagnósticos e planos de ação para otimização.

2. Trabalhos Relacionados

A literatura foi revisada a partir de pesquisas em plataformas como IEEE, Springer Link, ACM Digital Library e ScienceDirect, utilizando os termos 'Microservices', 'Cloud Com-

puting’ e ‘Energy Consumption’, o que resultou nos trabalhos selecionados. Os critérios de seleção incluíram artigos em inglês, publicados entre 2019 e 2024, e que apresentassem detalhes sobre a arquitetura dos sistemas, como métricas, técnicas de coleta e algoritmos utilizados. Os estudos escolhidos incluem, entre outros, os de [Shafi et al. 2024], [Wang et al. 2020], [Saboor et al. 2021] e [Turin et al. 2023].

A análise dos trabalhos selecionados revela uma variedade de abordagens voltadas à otimização do uso de recursos e à redução do consumo de energia. Alguns estudos, como [Shafi et al. 2024], são focados em recursos específicos, como CPU e memória, enquanto outros, como [Saboor et al. 2021] e [Turin et al. 2023], abordam a gestão dinâmica de microsserviços em VMs ou a distribuição baseada em rankings. Esses trabalhos destacam a oportunidade de otimizar os custos de microsserviços por meio do monitoramento do consumo energético.

3. AMEC

O modelo AMEC é composto por três blocos principais: o Gerenciador de Consumo Energético, o Gerenciador de Métricas e o Diagnóstico Energético. O primeiro bloco é responsável por monitorar o consumo de energia da aplicação, medindo a potência em tempo real. O segundo bloco coleta dados sobre a utilização de recursos, como CPU, memória, disco e rede, essenciais para entender a relação entre o consumo energético e os recursos computacionais utilizados. O Diagnóstico Energético, por sua vez, realiza a análise desses dados, identifica gargalos e gera um laudo detalhado com recomendações de otimização.

O AMEC permite diferentes abordagens de análise, como a comparação entre microsserviços ou a análise individual de um microsserviço específico. Na análise comparativa, os microsserviços executados em diferentes máquinas são avaliados com base no consumo energético, enquanto na análise individual, as funções ou métodos que mais consomem energia ao longo do tempo são identificados. Com essa arquitetura, o número de microsserviços e medidores pode variar conforme o cenário. O usuário, que pode ser um cliente final, desenvolvedor ou gerente de infraestrutura, interage com a aplicação e, após o processamento pelo AMEC, recebe um laudo energético detalhado. Embora seja possível executar mais de um microsserviço por computador, o AMEC recomenda a utilização de uma abordagem de múltiplos sensores, onde cada máquina executa apenas um microsserviço. Essa configuração facilita a associação direta do consumo de energia ao microsserviço monitorado, reduzindo interferências de outros processos. Cada microsserviço é, então, dedicado a um sensor específico, o que garante uma análise mais precisa e confiável.

O Gerenciador de Consumo Energético mede através de um sensor, em tempo real, a potência em Watts, tempo de operação e custo da energia, calculando o impacto real na aplicação. Os dados coletados são armazenados em um dataset XLSX para análises posteriores. O Gerenciador de Métricas do AMEC monitora CPU, memória, disco e rede, correlacionando esses dados ao consumo energético. O uso de CPU é registrado em porcentagem (0–100%) e armazenado em datasets individuais de cada microsserviço, servindo de base para o Diagnóstico Energético. O Diagnóstico Energético consolida essas informações, calcula a média das métricas e identifica picos de consumo. A clusterização destaca os principais gargalos da aplicação, oferecendo um panorama detalhado do uso

energético e sugerindo otimizações. O laudo gerado fornece um diagnóstico sobre os recursos consumidos por cada microsserviço.

Uma característica importante do modelo AMEC é a possibilidade de realizar múltiplas execuções da aplicação para observar seu comportamento sob diferentes condições. O modelo monitora a aplicação pelo menos cinco vezes para gerar comparações entre os cenários, permitindo a análise do consumo energético máximo e mínimo. Com base nessa análise comparativa, o AMEC gera recomendações precisas para a otimização do consumo de energia, identificando padrões e sugerindo melhorias. Essas recomendações são refletidas no laudo energético, que apresenta um diagnóstico detalhado, além de sugestões de ações para melhorar a eficiência do sistema.

O laudo energético gerado pelo AMEC contém várias seções, incluindo Contexto, Escopo, Metodologia, Métricas e Dados monitorados, Diagnóstico, Recomendações e Plano de Ação. O componente Contexto fornece um resumo sobre a aplicação analisada e o cenário em que o AMEC foi aplicado. O Escopo e Metodologia descrevem os serviços analisados, o tempo de observação e o período de coleta de dados. As seções de Métricas e Dados detalham o consumo de energia de cada microsserviço, bem como o uso de CPU, memória, disco e rede. O Diagnóstico analisa as métricas e destaca os pontos críticos de consumo e as possíveis anomalias. Por fim, o componente Recomendações e Plano de Ação sugere melhorias para otimizar o consumo de energia, seja ajustando o código ou alterando a arquitetura dos microsserviços. A estrutura do laudo energético permite adaptações para ser ajustada conforme o contexto da aplicação e a complexidade da arquitetura.

4. Metodologia

Para que o modelo seja testado e avaliado, é necessário que a aplicação em questão esteja configurada com um arquivo Dockerfile e utilizar a biblioteca psutil para registrar as métricas de uso de CPU, memória, disco e rede. Esse processo permite que o AMEC monitore a aplicação de maneira eficiente, para que as medidas registradas nos sensores possam ser estudadas com as demais métricas coletadas. O modelo foi validado em uma aplicação de processamento de imagens. Inicialmente, com a geração de figuras de menor complexidade com aumento progressivo de esforço. O principal objetivo dos testes foi avaliar o crescimento do consumo ao passo que as imagens geradas exigissem mais processamento do dispositivo.

5. Resultados

Os resultados obtidos com o modelo AMEC foram analisados em diferentes cenários de teste, utilizando duas máquinas conectadas aos sensores FE-0083 para medir o consumo energético. A comunicação ocorreu via rede local, sem recursos externos, e as máquinas estavam configuradas apenas com VS Code e Docker para minimizar interferências. Durante os testes, o sensor FE-0083, ligado entre a rede elétrica e cada máquina, registrou um consumo médio de 7W em stand by, variando entre 6W e 8W. Em um dos testes mais complexos, o microsserviço responsável pela geração de uma imagem de fractal de Mandelbrot em alta resolução apresentou um aumento significativo no consumo energético, com picos de até 20W. O tempo de processamento subiu de 230s para 720s, elevando o uso da CPU e o consumo de energia. Por outro lado, o microsserviço menos exigido teve

variações menores no consumo. Em cenários de geração de imagens simples, a relação entre uso da CPU e consumo energético foi fraca, pois os microsserviços não foram suficientemente estressados para gerar uma tendência clara. No entanto, em cenários mais exigentes, como a geração de fractais, observou-se uma correlação forte e positiva entre o uso da CPU e o consumo de energia, indicando que o aumento do processamento intensifica proporcionalmente o consumo energético.

6. Discussão

Os resultados reforçam a importância do modelo AMEC para a análise e otimização do consumo energético em microsserviços. O AMEC oferece uma visão abrangente sobre o comportamento energético das aplicações, permitindo tanto uma análise macro, que identifica os microsserviços mais consumidores de energia, quanto uma análise micro, que foca em funções ou métodos específicos dentro de um serviço. Além disso, o aumento do consumo observado entre cenários simples e complexos destaca a necessidade de estratégias de gestão de energia em sistemas que utilizam microsserviços.

7. Conclusão

Os resultados obtidos são promissores e indicam que o AMEC pode ser uma ferramenta valiosa para reduzir o consumo energético de aplicações. Para futuras pesquisas, a implementação de técnicas de inteligência artificial poderia melhorar o modelo, permitindo previsões e recomendações automatizadas no laudo energético. A integração de novos sensores e o uso de dados em nuvem poderiam otimizar a clusterização dos dados coletados, aprimorando ainda mais as recomendações geradas pelo AMEC.

Referências

- Araújo, G., Barbosa, V., Lima, L. N., Sabino, A., Brito, C., Fé, I., Rego, P., Choi, E., Min, D., Nguyen, T. A., and Silva, F. A. (2024). Energy consumption in microservices architectures: A systematic literature review. *IEEE Access*, 12:186710–186729.
- Biçici, E. (2024). A cloud monitor to reduce energy consumption with constrained optimization of server loads. *IEEE Access*, 12:25265–25277.
- Mahesar, A. R., Li, X., and Sajnani, D. K. (2024). Efficient microservices offloading for cost optimization in diverse mec cloud networks. *Journal of Big Data*, 11(1).
- Saboor, A., Mahmood, A. K., Omar, A. H., Hassan, M. F., Shah, S. N. M., and Ahmadian, A. (2021). Enabling rank-based distribution of microservices among containers for green cloud computing environment. *Peer-to-Peer Networking and Applications*, 15(1):77–91.
- Shafi, N., Abdullah, M., Iqbal, W., Erradi, A., and Bukhari, F. (2024). Cdascaler: a cost-effective dynamic autoscaling approach for containerized microservices. *Cluster Computing*.
- Turin, G., Borgarelli, A., Donetti, S., Damiani, F., Johnsen, E. B., and Tapia Tarifa, S. L. (2023). Predicting resource consumption of kubernetes container systems using resource models. *Journal of Systems and Software*, 203:111750.
- Wang, R., Ying, S., Li, M., and Jia, S. (2020). Hsacma: a hierarchical scalable adaptive cloud monitoring architecture. *Software Quality Journal*, 28(3):1379–1410.