Predição de Custo de Execução de FaaS em Provedor Público de Nuvem por meio do Framework Orama

Edson de Souza Sales[1], **Leonardo Rebouças de Carvalho**[1], **Aleteia Araujo**[1]

[1]Departamento de Ciência da Computação – Universidade de Brasília (UnB)
Caixa Postal 70910-000 – Brasília – DF – Brasil

edsondesouzasales@gmail.com, leouesb@gmail.com, aleteia@unb.br

Abstract. *Function-as-a-Service (FaaS) is a cloud computing paradigm that has gained notoriety for offering automatic scalability and simplicity in application execution. However, FaaS brings unpredictable execution costs, making financial planning difficult for applications with variable loads. This paper presents a tool, integrated with the Orama framework, that estimates FaaS execution costs across AWS provider. The results demonstrate that the proposed approach is capable of predicting costs with an average accuracy of 99.23%.*

Resumo. *Function-as-a-Service (FaaS) é um paradigma de computação em nuvem que tem se destacado por oferecer escalabilidade automática e simplicidade na execução de aplicações. Contudo, o uso de FaaS traz uma imprevisibilidade dos custos de execução, dificultando o planejamento financeiro em aplicações com cargas variáveis. Este trabalho apresenta uma ferramenta, integrada ao framework Orama, que estima os custos de execução de FaaS no provedor AWS. Os resultados mostram que a abordagem proposta é capaz de prever os custos com uma acurácia média de 99,23%.*

1. Introdução

A computação em nuvem tem revolucionado a forma como as aplicações estão sendo desenvolvidas. Estima-se que os gastos globais dos usuários finais com serviços de nuvem pública totalizarão 723,4 bilhões de dólares em 2025, superando os 595,7 bilhões de dólares de 2024 [Gartner, Inc. 2024]. Esses dados evidenciam a importância cada vez maior dessa abordagem nos sistemas atuais.

Entre os diversos modelos de serviço em nuvem, o Function-as-a-Service (FaaS) [Motta. et al. 2022] tem-se destacado por permitir a execução de funções isoladas em resposta a eventos, abstraindo o gerenciamento de uma infraestrutura. O mercado global de FaaS foi avaliado em 15,02 bilhões de dólares em 2024, com expectativa de crescimento a uma taxa composta anual (Compound Annual Growth Rate - CAGR) de 27,8% entre 2025 e 2030 [Grand View Research 2024].

Apesar das vantagens, o uso de FaaS traz imprevisibilidade nos custos de execução, especialmente em aplicações com alta variabilidade de chamadas e tempos de execução, o que dificulta o planejamento orçamentário. Isso ocorre porque as funções são ativadas por eventos e escalam automaticamente conforme a necessidade, sendo cobradas com base no número de execuções e no tempo de processamento utilizado, o que torna o modelo ideal para aplicações orientadas a eventos e com cargas de trabalho intermitentes.

Dessa forma, a imprevisibilidade dos custos dificulta que usuários e desenvolvedores façam as escolhas mais eficientes e econômicas entre os provedores para execução de cargas de trabalho FaaS [Hassan et al. 2021]. Esse problema pode ser explicado pelo uso de parâmetros não determinísticos, como o tempo de execução de uma função ou a quantidade de invocações, na composição do custo.

Diante do exposto, este trabalho tem como objetivo propor, por meio de Aprendizado de Máquina, uma ferramenta capaz de estimar os custos de execução de requisições FaaS em diferentes provedores. A proposta visa apoiar decisões de desenvolvedores e arquitetos de software quanto à escolha mais eficiente e econômica de provedores para execução de cargas de trabalho FaaS. A proposta se caracteriza pela capacidade de predição sem a necessidade de executar previamente as funções no ambiente em nuvem por meio do *framework* Orama [de Carvalho and Araujo 2023]. Os testes apresentados neste artigo foram todos na plataforma AWS, embora o Orama permita a execução nos provedores Google Cloud, Azure e Alibaba.

Para apresentar a solução proposta, o restante deste artigo está organizado em seis seções. A Seção 2 faz uma análise com alguns trabalhos relacionados. Em seguida, a Seção 3 apresenta os conceitos-chave utilizados no desenvolvimento da solução proposta. A Seção 4 descreve o *framework* Orama e seus objetivos. A Seção 5 detalha o funcionamento da solução proposta. A Seção 6 apresenta a metodologia usada nos testes realizados, e os resultados iniciais alcançados. Por fim, a Seção 7 sumariza os resultados e destaca trabalhos futuros.

2. Trabalhos Relacionados

Outros estudos foram feitos com o objetivo de criar um modelo de predição de custos e desempenho de funções no ambiente FaaS. Em geral, essas abordagens buscam prever métricas como tempo de execução ou custo final a partir de características da função, do ambiente de execução e de dados coletados previamente.

Os trabalhos identificados nesta pesquisa propõe métodos de predição que dependem de um processo prévio de perfilamento das funções. Esse processo envolve a execução das funções por métodos como *microbenchmarks* ou através da coleta de dados em produção para construir os modelos de previsão. O trabalho [Eismann et al. 2020] propõe um modelo para prever o custo de *serverless workflows* com base em dados prévios coletados no Google Cloud Functions. De forma semelhante, o artigo [Cordingly et al. 2020] utiliza o Serverless Application Analytics Framework (SAAF) para gerar modelos de desempenho e custo a partir de execuções prévias no AWS e na IBM Cloud. As soluções [Lin et al. 2023] e [Lin and Khazaei 2021] exploram estratégias de otimização de desempenho e custo a partir do perfilamento de funções no AWS.

Embora atinjam alta acurácia, essas abordagens são limitadas pela necessidade de perfilamento, oneroso em contextos com múltiplas funções, atualizações frequentes no código ou ausência de histórico de execução.

Uma comparação entre os trabalhos relacionados é apresentada na Tabela 1. A coluna Técnica de Predição indica o método utilizado para estimar o custo das funções. A coluna Suporte de *Framework* informa se há uma ferramenta desenvolvida para aplicar o modelo. A quarta coluna apresenta a acurácia média obtida por cada solução. Por fim,

a última coluna indica se o modelo depende de dados obtidos por execução prévia da função.

A solução proposta neste trabalho permite estimar os custos de execução de funções sem a necessidade de uma fase prévia de perfilamento, uma vez que o *framework* Orama estima o tempo de execução de uma função utilizando apenas seu código. Além disso, fornece simultaneamente predições para quatro provedores de FaaS. Essas características conferem maior simplicidade e aplicabilidade prática, principalmente, em cenários de planejamento financeiro e estimativa antecipada de custos.

Trabalho	**Técnica de Predição**	**Suporte de *Framework***	**Acurácia**	**Perfilamento**
[Eismann et al. 2020]	*Mixture Density Networks*	Não	96,1%	Sim
[Cordingly et al. 2020]	Regressões Múltiplas	SAAF	96,51%	Sim
[Lin et al. 2023]	*Conditional Stochastic Petri Net*	Não	97,78%	Sim
[Lin and Khazaei 2021]	Modelagem Analítica com Grafos	Não	99,97%	Sim
Este trabalho	Aprendizado de Máquina	Orama	99,23%	Não

Tabela 1. Comparação entre trabalhos relacionados.

3. Contextualização

A computação em nuvem é organizada em modelos de serviço com diferentes níveis de abstração, sendo os mais comuns Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) e Software-as-a-Service (SaaS). No IaaS, o usuário provisiona recursos como armazenamento, rede e processamento, com controle sobre sistemas operacionais e aplicações, mas sem gerenciar a infraestrutura física. No PaaS, o foco está na implantação de aplicações desenvolvidas com ferramentas do provedor. Já no SaaS, o consumidor acessa aplicações prontas, geralmente via navegador [Mell and Grance 2011].

Function-as-a-Service (FaaS) é um modelo com nível de abstração entre PaaS e SaaS, que permite a execução de funções em resposta a eventos [Schleier-Smith et al. 2021]. As suas principais características são a abstração de gerenciamento de infraestrutura, escalabilidade automática e modelo de cobrança baseado apenas nos recursos efetivamente utilizados durante a execução [de Carvalho et al. 2024].

Apesar desses benefícios, a flexibilidade e a execução sob demanda presentes no FaaS podem tornar a precificação imprevisível. Logo, a predição de tempo e de custo de execução desempenham um papel fundamental ao permitir que desenvolvedores e arquitetos de sistemas antecipem o comportamento de suas funções em diferentes cenários de nuvem. Ao estimar tempos de resposta e custo, antes da implantação, é possível otimizar a escolha do provedor com base em requisitos específicos de latência, custo e escalabilidade. Escolhas inadequadas de provedores ou configurações subótimas podem gerar

gastos excessivos, tornando imprescindível o uso de ferramentas que automatizem essa análise e auxiliem na tomada de decisões mais eficientes e econômicas. Nesse cenário, modelos de predição tornam-se úteis para estimar previamente esses custos.

Um modelo de predição é uma função que, com base em dados que relacionam um conjunto de características de um objeto a um resultado, é capaz de prever a saída correspondente para novas entradas [Kelleher et al. 2015]. No contexto de FaaS, destaca-se o *framework* Orama, que oferece uma ferramenta de predição do tempo de execução de funções nos principais provedores do mercado. Essa ferramenta extrai métricas Halstead [Khan and Nadeem 2023] a partir do código-fonte fornecido e estima o tempo de execução da função perante um nível de concorrência informado. Essa predição é feita para os provedores que o *framework* suporta, tais como AWS Lambda, Google Cloud Platform, Microsoft Azure e Alibaba Compute Functions.

4. Framework Orama

O *Orama* é um *framework* de código aberto[1] projetado para automatizar a execução de *benchmarks* em ambientes de *Function-as-a-Service* (FaaS). O seu principal objetivo é oferecer uma ferramenta eficiente, reprodutível e flexível para a avaliação do desempenho e do comportamento de funções *serverless* em diferentes provedores de nuvem [de Carvalho et al. 2024]. Ele permite que pesquisadores e engenheiros conduzam experimentos controlados de forma sistemática, reduzindo o esforço manual e o risco de inconsistências nos testes.

Um dos principais diferenciais do *Orama* é a sua capacidade de orquestrar todo o ciclo de vida de um experimento de *benchmarking*. O *framework* disponibiliza casos de uso predefinidos, tais como: uma calculadora simples, APIs com salvamento em *Object Storage* e em banco de dados gerenciados pela nuvem. Esses casos de uso podem ser implantados automaticamente em múltiplos provedores [Carvalho et al. 2023]. A execução dessas funções segue parâmetros configuráveis definidos pelo usuário, incluindo o nível de concorrência, número de repetições, tempo de intervalo entre invocações, uso de requisições de *warm-up* e tipo de *trigger*. Essa flexibilidade permite explorar cenários variados de uso, avaliando a resiliência e o desempenho das plataformas FaaS em situações realistas.

Além da orquestração de *benchmarks*, o *Orama* incorpora uma ferramenta de predição de tempo de execução baseada em Aprendizado de Máquina. Essa ferramenta analisa características extraídas do código da função, como número de instruções, uso de bibliotecas, volume de dados de entrada e chamadas de sistema, e considera também a configuração de execução (por exemplo, grau de concorrência) para estimar o tempo de execução previsto em cada provedor. Essa estimativa é gerada sem a necessidade de execução real da função, oferecendo uma abordagem eficiente para planejamento de cargas e comparação antecipada entre diferentes ambientes de nuvem.

A ferramenta de predição foi desenvolvida a partir de experimentos sistemáticos conduzidos com o próprio *Orama*. Ela é especialmente útil em contextos nos quais o custo ou o tempo de execução real inviabilizariam testes exaustivos. Com isso, torna-se possível simular e antecipar comportamentos esperados em diversos cenários, contribuindo para a

[1]Repositório público disponível em: `https://github.com/unb-faas/orama`

tomada de decisões mais informadas sobre alocação de recursos, seleção de provedores e ajustes de desempenho. Essa ferramenta serve de base para o modelo de estimativa de custo apresentado na próxima seção.

Ao simplificar a avaliação de soluções *FaaS* sob diferentes condições operacionais e incorporar uma camada preditiva baseada em dados reais, o *Orama* se estabelece como uma ferramenta robusta para análise comparativa, suporte à decisão e pesquisa acadêmica em computação em nuvem.

5. Proposta

A solução proposta consiste em obter uma estimativa de custo de execução de FaaS nos provedores públicos de nuvem a partir da estimativa de tempo fornecida pelo *framework* Orama. Essa estimativa de custo é calculada aplicando o tempo previsto às políticas de precificação específicas de cada provedor, considerando como parâmetros de entrada o código da função e sua configuração de execução. Um diagrama apresentando a proposta deste trabalho é mostrado na Figura 1

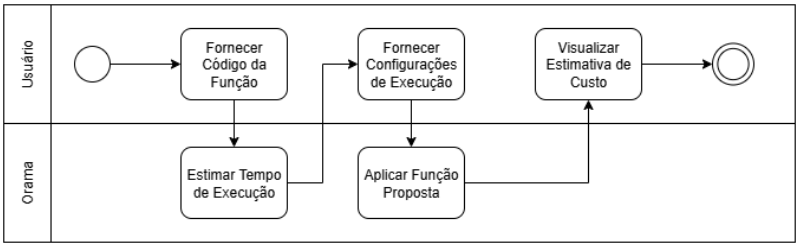

Figura 1. Diagrama da predição de custo via *framework* Orama.

Para isso, é necessário uma função que, a partir de um conjunto de parâmetros comuns entre os provedores, seja capaz de representar o custo final de execução. Assim sendo, foram considerados três componentes principais para o cálculo do custo: (i) custo por número de invocações, (ii) custo por gigabyte-segundo (GB·s) utilizado, (iii) custo por vCPU-segundo (vCPU·s). Esses componentes são mapeados para cada região de execução e, quando aplicável, para diferentes *tiers* de precificação. Os valores correspondentes devem ser obtidos a partir das políticas de precificação de cada provedor. Em conjunto, abrangem os principais fatores tarifados pelos provedores e permitem representar o custo associado a execução de funções.

Dessa forma, a função proposta recebe como entrada os seguintes parâmetros: tempo de execução da função, memória alocada, quantidade de vCPU alocada, provedor de FaaS, região de execução e/ou *tier* de precificação. A partir dessas entradas e dos valores de custo associados ao uso de cada recurso, a função retorna como saída o custo total estimado para a execução. O cálculo é realizado por meio da Equação 1.

$$\text{Custo Total} = N \cdot (C_{\text{invoc}} + C_{\text{mem}} \cdot M \cdot T + C_{\text{cpu}} \cdot U \cdot T) \tag{1}$$

Onde, N representa o número de invocações da função, M a memória alocada para a função (GB), T o tempo de execução estimado (s), e U a quantidade de CPU alocada para a função. A equação 1 também considera três componentes, em que C_{invoc}

representa o custo por invocação (USD), C_{mem} o custo por uso de memória ($\text{USD}/\text{GB} \cdot \text{s}$), e C_{cpu} o custo por uso de vCPU ($\text{USD}/\text{vCPU} \cdot \text{s}$). Esses três coeficientes variam conforme o provedor, a região geográfica e, quando aplicável, o *tier* de precificação.

Assim, utilizando o *framework* Orama para obter a estimativa de tempo de execução de uma função a partir do seu código e as configurações fornecidas pelo usuário, é possível reunir todos os parâmetros de entrada necessários para utilizarmos a função proposta da equação (1) e obter uma estimativa de custo de execução. Logo, a fim de validar a acurácia dessa estimativa, foi conduzido um experimento para comparar os valores previstos pelo modelo, com os custos reais observados em uma plataforma FaaS.

6. Resultados

O objetivo desta seção é apresentar os resultados iniciais obtidos a partir da solução proposta para estimar o custo na execução de FaaS. Para isso, foram comparados os valores estimados pelo *framework* Orama com os custos reais calculados a partir de execuções no serviço AWS Lambda. A Figura 2 ilustra o fluxo das etapas envolvidas nesse processo.

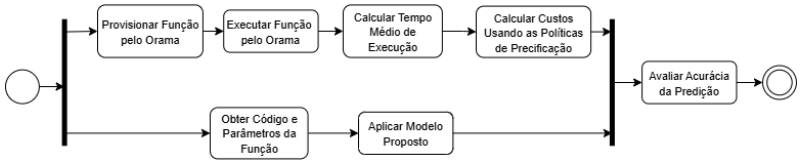

Figura 2. Diagrama da metodologia usada na avaliação da solução proposta.

A política de preços do AWS Lambda considera o número de invocações da função e o tempo de execução ponderado pela memória alocada (GB·s). Os custos desses componentes variam conforme a região e o nível de uso (*tier*) da conta. Essas regras foram aplicadas para o cálculo do custo real das execuções.

Todas as etapas da execução foram automatizadas pelo *framework* Orama, que é responsável por provisionar o ambiente de execução, definir os parâmetros da função e coletar os dados resultantes. Para este experimento, foram realizadas 1000 execuções com nenhum nível de concorrência da função Lambda Calc, escolhida por sua fácil reprodutibilidade. As execuções foram realizadas nas regiões *us-east-1*, *us-west-1*, *eu-central-1*, *ap-east-1* e *ap-southeast-2*, com 128 MB de memória alocada.

O tempo de execução de cada invocação foi obtido por meio dos registros gerados pelo *framework*. Em seguida, foi calculada a média desses tempos, que foi utilizada no cálculo do custo com base nas políticas de precificação do AWS Lambda, considerando também os parâmetros adotados pelo Orama no provisionamento da função e os valores de custo referentes ao *tier* aplicável às primeiras seis bilhões de requisições mensais de uma conta no AWS Lambda. Dessa forma, foi possível obter o custo real das execuções.

O custo obtido a partir da execução real da função foi comparado com o custo estimado pelo modelo proposto. Para essa comparação, foram utilizados o mesmo código e os mesmos parâmetros definidos pelo *benchmark* da função Lambda Calc, assegurando, assim, uma base consistente entre os dois valores. A fim de garantir maior precisão na comparação, os valores foram considerados com até oito dígitos após a vírgula, sem arredondamento na segunda casa decimal.

Assim sendo, a Tabela 2 apresenta os resultados iniciais obtidos. Nessa tabela, a função Lambda Calc foi executada 1000 vezes no AWS Lambda para cada uma das regiões listadas na primeira coluna. A segunda coluna indica o tempo médio dessas execuções na região correspondente. O custo real correspondente a 1000 execuções da função foi calculado com base na política de precificação do provedor e apresentado na terceira coluna. Os custos estimados pelo modelo integrado ao *framework* Orama estão dispostos na quarta coluna. Por fim, a quinta coluna apresenta a acurácia da estimativa, obtida por meio da comparação entre o custo real e o estimado.

Região	**Tempo Médio (ms)**	**Custo Real (USD)**	**Custo Estimado (USD)**	**Acurácia**
us-east-1	2,881	0,00020600	0,00020505	99,54%
us-west-1	2,713	0,00020566	0,00020505	99,70%
eu-central-1	2,618	0,00020545	0,00020505	99,81%
ap-east-1	2,678	0,00028767	0,00028000	97,33%
ap-southeast-2	2,650	0,00020552	0,00020505	99,77%

Tabela 2. Comparação entre custo real e estimado.

A acurácia média da estimativa foi de 99,23%. Esses resultados indicam que a abordagem proposta apresenta um bom potencial preditivo. A próxima seção discute trabalhos relacionados que também propõem modelos de predição de custo e desempenho no contexto FaaS.

7. Conclusão

Este trabalho apresentou uma solução para estimar os custos de execução de funções em ambientes FaaS sem a necessidade de execução prévia das funções. A proposta foi implementada como extensão do Orama, integrando a estimativa de tempo de execução a um modelo de cálculo de custos para diferentes provedores.

Ao eliminar a necessidade de perfilamento da função, a solução proposta amplia a aplicabilidade da predição de custos para fases iniciais do desenvolvimento de sistemas e facilita a comparação entre provedores. A metodologia experimental demonstrou que a ferramenta apresenta acurácia média de 99,23% na estimativa dos custos, validando sua viabilidade como apoio a decisões financeiras.

Como trabalhos futuros, propõe-se: (i) a utilização das APIs dos provedores para obtenção automatizada dos preços por recurso; (ii) a integração do cálculo de custo aos dados extraídos das execuções de *benchmark*; e (iii) a investigação de técnicas de aprendizado de máquina para aprimorar a acurácia da predição.

Referências

Carvalho, L., Kamienski, B., and Araujo, A. (2023). How faas with dbaas performs in different regions: an evaluation by the orama framework. In *Anais do XXIV Simpósio em Sistemas Computacionais de Alto Desempenho*, pages 241–252, Porto Alegre, RS, Brasil. SBC.

Cordingly, R., Shu, W., and Lloyd, W. J. (2020). Predicting performance and cost of serverless computing functions with saaf. In *2020 IEEE Intl Conf on DASC, Intl Conf on PiCom, Intl Conf on CBDCom, Intl Conf on CyberSciTech*, pages 640–649.

de Carvalho, L. R. and Araujo, A. (2023). Insights into the performance of function-as-a-service oriented environments using the orama framework. *SN Computer Science*, 4(3):305.

de Carvalho, L. R., Kamienski, B., and Araujo, A. (2024). Main faas providers behavior under high concurrency: An evaluation with orama framework distributed architecture. *SN Comput. Sci.*, 5(5).

Eismann, S., Grohmann, J., van Eyk, E., Herbst, N., and Kounev, S. (2020). Predicting the costs of serverless workflows. In *Proceedings of the ACM/SPEC International Conference on Performance Engineering*, ICPE '20, page 265–276, New York, NY, USA. Association for Computing Machinery.

Gartner, Inc. (2024). Gartner forecasts worldwide public cloud end-user spending to total $723 billion in 2025. Technical report, Gartner, Inc., London, U.K. Acessado em 17/07/2025.

Grand View Research (2024). Function as a service market size, industry report, 2030. `https://www.grandviewresearch.com/industry-analysis/function-as-a-service-market-report`. Forecast Period: 2025–2030.

Hassan, H. B., Barakat, S. A., and Sarhan, Q. I. (2021). Survey on serverless computing. *Journal of Cloud Computing*, 10(1):39.

Kelleher, J. D., Namee, B. M., and D'Arcy, A. (2015). *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*. The MIT Press.

Khan, B. and Nadeem, A. (2023). Evaluating the effectiveness of decomposed halstead metrics in software fault prediction. *PeerJ Computer Science*, 9:e1647.

Lin, C. and Khazaei, H. (2021). Modeling and optimization of performance and cost of serverless applications. *IEEE Transactions on Parallel and Distributed Systems*, 32(3):615–632.

Lin, C., Mahmoudi, N., Fan, C., and Khazaei, H. (2023). Fine-grained performance and cost modeling and optimization for faas applications. *IEEE Transactions on Parallel and Distributed Systems*, 34(1):180–194.

Mell, P. M. and Grance, T. (2011). Sp 800-145. the nist definition of cloud computing. Technical report, Gaithersburg, MD, USA.

Motta., M. A. D. C., Reboucas De Carvalho., L., Rosa., M. J. F., and Favacho De Araujo., A. P. (2022). Comparison of faas platform performance in private clouds. In *Proceedings of the 12th CLOSER,*, pages 109–120, Online. INSTICC, SciTePress.

Schleier-Smith, J., Sreekanti, V., Khandelwal, A., Carreira, J., Yadwadkar, N. J., Popa, R. A., Gonzalez, J. E., Stoica, I., and Patterson, D. A. (2021). What serverless computing is and should become: the next phase of cloud computing. *Commun. ACM*, 64(5):76–84.