

# Uma Estratégia para Realizar a Análise de Eficiência Energética em Serviços em Nuvens

Matheus Felipe Lima Domingos<sup>1</sup>, Gustavo Callou<sup>1</sup>, Ermeson Andrade<sup>1</sup>

<sup>1</sup>Departamento de Computação – Universidade Federal Rural de Pernambuco (UFRPE)  
52171-900 – Recife – PE – Brazil

**Abstract.** *Cloud computing services have been gaining popularity in the last decade. This growth raises concerns for providers about the cost and energy efficiency of these environments. It is also known that there is a relationship between the increased use of these system resources and their energy consumption. This work analyzes energy consumption in level of server usage in order to diagnose, in a testbed environment, whether it is possible to optimize this consumption based on energy efficiency. To accomplish this, energy consumption measurements were carried out based on stress tests, and the system's energy efficiency was verified. The results show that the server's energy efficiency varies depending on the load demanded from the energy source, with cases where energy consumption changed disproportionately with the server's CPU usage.*

**Resumo.** *Os serviços de computação em nuvem vêm ganhando popularidade desde a última década. Esse crescimento acaba por trazer preocupações aos provedores relacionadas ao custo e à eficiência energética desses ambientes. Sabe-se ainda que existe uma relação entre o aumento da utilização desses recursos do sistema e o seu consumo de energia. Este trabalho analisa o consumo de energia em relação ao nível de utilização de servidores a fim de diagnosticar, em um ambiente testbed, se é possível otimizar esse consumo em função da eficiência energética. Para isso, foram realizadas medições do consumo de energia, a partir de testes de estresse, e foi verificada a eficiência energética do sistema. Os resultados obtidos mostram que a eficiência energética do servidor varia em função da carga demandada à fonte de energia, com casos em que consumo de energia se alterou desproporcionalmente em relação à utilização de CPU do servidor.*

## 1. Introdução

Com o crescimento exponencial da Internet e do uso de computadores em geral, recursos como espaço de armazenamento e memória vêm se tornando cada vez mais valiosos, especialmente quando se trata de grandes empresas que precisam empregar esses recursos em seus negócios. Para atender a essa demanda, tecnologias surgiram para possibilitar a abstração de recursos computacionais por meio de virtualização, a fim de oferecer serviços computacionais através de um provedor. Vale destacar que nestes ambientes os clientes não precisam nem sequer se preocupar com a organização do espaço físico para acomodar as máquinas e nem com a manutenção das mesmas [Qian et al. 2009]. Dessa forma, a computação em nuvem vem como uma dessas tecnologias que facilitam o acesso a recursos computacionais configuráveis (como rede, aplicações, memória e armazenamento) de forma simples e conveniente [Cloud 2011].

Devido aos seus benefícios, os serviços de computação em nuvem são muito populares e altamente requisitados na atualidade. Serviços como o *Amazon Web Services* [aws] são extremamente utilizados, onde o *AWS* chega a ter centenas de milhares de negócios, com clientes espalhados entre mais de 190 países [Mathew and Varia 2014]. Dado o crescimento do uso de serviços em nuvem, a qualidade de tais serviços vem se tornando uma preocupação cada vez maior, principalmente devido à existência de diversos desafios relacionados à confiança e disponibilidade [Odun-Ayo et al. 2018]. Outro problema da área é a questão da eficiência energética, visto que para se disponibilizar esses serviços é necessário uma grande infraestrutura de vários data centers. Por exemplo, esses ambientes acabam por consumir uma energia significativa, estimada em cerca de 1,5% do consumo mundial. Sendo assim, existe uma demanda para se analisar e buscar melhorar o consumo energético de servidores e equipamentos utilizados para prover serviços na nuvem [Mastelic et al. 2014].

Os dispositivos principais para o funcionamento de serviços em nuvem são os servidores, já que são neles que os serviços são hospedados. Sendo assim, a maior parte do consumo de energia de um sistema em nuvem acaba por ser gerada pelos seus servidores. Portanto, para se buscar um aprimoramento da eficiência energética de um sistema em nuvem é de extrema importância a análise no nível de utilização e do consumo de energia dos servidores. Essa análise deve ser feita para que seja possível identificar os cenários de maior eficiência energética e, assim, ser possível reduzir tais consumos. Essa análise da eficiência energética de um servidor é crucial por múltiplas razões. Primeiramente, o alto consumo de energia dos servidores inevitavelmente traz custos maiores para as empresas que os utilizam. Uma estratégia para otimizar o consumo de energia dos servidores ajuda a diminuir esses custos. Além da economia, menos consumo de energia pode significar em uma redução da emissão de carbono. Esta redução acaba, assim, tendo efeitos para uma computação mais verde.

Este trabalho tem como objetivo a análise de um serviço de computação em nuvem com foco na melhora do consumo de energia a partir da identificação do nível de utilização da CPU com melhor eficiência energética. Para se atingir este objetivo, foi montado um sistema *testbed* em uma nuvem privada utilizando o *Apache Cloudstack* [apa]. Vale destacar que medições do consumo de energia buscando relacionar com o percentual de utilização do processador também se encontram como contribuições das atividades realizadas.

O artigo está estruturado conforme descrito a seguir. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 apresenta a metodologia utilizada para avaliar o consumo de energia e eficiência energética do sistema montado em laboratório. A Seção 4 apresenta detalhes sobre os experimentos e resultados atingidos. Por fim, a Seção 5 compreende a conclusão do trabalho e aponta possíveis trabalhos futuros para a pesquisa.

## **2. Trabalhos Relacionados**

Dado o crescente aumento da utilização de serviços providos por computação em nuvem, passou-se a encontrar alguns trabalhos na literatura que analisam o consumo de energia desses ambientes. Por exemplo, Lima et al. [de Lima et al. 2021] apresentaram uma abordagem de comparação entre dois sistemas de nuvem diferentes, o *OpenStack* e o *Apache Cloudstack*. Para isso, foi analisado e adotada uma estratégia que fez uso de

medições do desempenho e do consumo de energia de um serviço ofertado na nuvem. O serviço adotado como cenário de estudo foi o de um ambiente virtual de aprendizagem (AVA), onde foi utilizada uma regressão linear para se obter a relação entre o consumo de energia da máquina servidora com o nível de utilização da CPU. O resultado dessa relação foi aplicado como entrada para um modelo em rede de Petri estocástica (SPN). No entanto, não foi o foco dos autores analisar a eficiência energética da máquina servidora utilizada no experimento.

Berl et al. [Berl et al. 2010] identificaram a importância da eficiência energética e analisaram métodos para economia de energia em hardware computacional e infraestruturas de rede. O objetivo foi o de apontar quais os desafios que serão encontrados ao utilizar tais métodos a longo prazo em um ambiente de nuvem. No entanto, uma experimentação própria de um caso de eficiência energética não foi o foco da pesquisa.

Os autores em [Rahman et al. 2013] analisaram como economizar energia em ambientes de computação em nuvem *mobile*, revisando os problemas e soluções existentes propostas anteriormente na área de pesquisa. O trabalho conclui apontando os maiores obstáculos e necessidades ainda encontradas na área para trabalhos futuros. Uma análise sobre o conceito de energia eficiente não foi o foco da pesquisa.

Materwala et al. [Materwala and Ismail 2022] trouxeram uma abordagem para otimização do problema de agendamento de tarefas em ambiente de nuvem no intuito de minimizar o custo de energia e o tempo de execução. Para atingir este objetivo foi utilizado um algoritmo evolucionário multi-objetivo, e os resultados obtidos foram comparados com outras abordagens do estado da arte. Uma análise sobre a eficiência energética do sistema utilizado não foi o foco da pesquisa.

Katal et al. [Katal et al. 2023] exploraram extensivamente a questão de consumo de energia em ambientes em nuvem, trazendo um estudo sobre o impacto ambiental dos *data centers* e apresentando novas tecnologias que podem ser aplicadas em múltiplos níveis de software. O foco dessa pesquisa foi no desenvolvimento tecnológico a longo prazo, sem ser mencionado nada sobre uma proposição para análise de energia eficiente.

Os autores em [Venkateswaran et al. 2023] propuseram um modelo de arquitetura para serviços em nuvem com o objetivo de maximizar a eficiência energética do sistema. O trabalho buscou atingir este objetivo apresentando um modelo e o comparando a outros modelos de administração de recursos. Nesse estudo foram avaliados o consumo de energia e o nível de utilização do *host* físico. O trabalho não foca em estratégias para estressar o sistema de nuvem com o objetivo de obter dados para análise de eficiência energética.

Chaurasia et al. [Chaurasia et al. 2021] apresentaram um estudo em estratégias de redução de consumo de energia para serviços de nuvem. Essa estratégia buscou otimizar a utilização de recursos de um servidor a ponto de menos máquinas físicas serem necessárias para a infraestrutura do serviço. O trabalho categoriza e avalia os métodos existentes, discutindo suas características e limitações. A pesquisa cita uma proposição de estratégia própria como trabalho futuro.

A maioria dos trabalhos encontrados na literatura focam na redução do consumo de energia a partir da melhor alocação de servidores, por exemplo. Diferentemente, esse trabalho foca como principal contribuição a realização de experimentos em um ambiente *testbed* com o objetivo de se obter dados reais do consumo de energia, e relacionar com o

nível de utilização da CPU. Isso é feito tendo em vista que as fontes de energia de servidores possuem diferentes níveis de eficiência energética para cargas distintas do nível de utilização. Sendo assim, a partir da utilização de recursos da máquina física e do consumo de energia associado, busca-se identificar essa diferença na eficiência energética. Os dados obtidos do consumo são analisados a fim de se encontrar essa relação do aumento da utilização da CPU e o seu respectivo consumo de energia. Sendo assim, esse trabalho foca em tentar identificar os pontos de aumento e declínio do nível de eficiência energética.

### 3. Metodologia

A Figura 1 apresenta a metodologia adotada para avaliar o consumo de energia e a eficiência energética do serviço em nuvem analisado. A primeira etapa corresponde a um estudo detalhado sobre o problema a ser avaliado. Nessa etapa, também deve ser realizado um estudo sobre possíveis ferramentas que serão utilizadas para auxiliar na medição e coleta de dados das métricas de interesse.

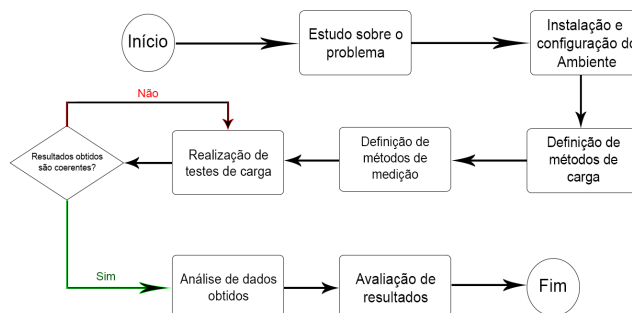


Figura 1. Metodologia.

Em seguida, a segunda etapa corresponde a instalação e configuração do ambiente de nuvem que será utilizado. Nesse trabalho em particular, foi escolhido o *Apache Cloudstack*[apa ] por ser *open source*, ter grande aceitação e utilização no mercado de nuvem privada *Cloudstack* [doc a]. Esse ambiente adota o sistema operacional(SO) *Centos 7* [cen ], que é um sistema baseado em Linux.

Uma vez já tendo o ambiente configurado e ativo, a próxima etapa consiste em definir as estratégias que serão utilizadas para estressar esse ambiente. O objetivo desse teste de estresse é conseguir aumentar o consumo de energia da máquina física a partir do aumento da carga aos serviços ofertados pelas VMs. Para isso, foram adotados dois métodos de geração de carga nas máquinas virtuais do ambiente em nuvem. O primeiro método considerou o framework *Locust*, que é *open-source*, para a realização de testes de carga em protocolos HTTP. O *Locust* permite a obtenção de informações sobre tempos de resposta, desempenho e porcentagem de falhas que podem ser vistos em tempo real e catalogados para análise posterior. Esse framework de testes de carga é montado a partir de um código em Python e permite que o usuário customize os parâmetros de entrada para os testes. Por exemplo, podem ser configurados o número desejado de usuários simultâneos, quantos usuários irão ser gerados por segundo, e o limite de usuários [loc ]. Assim, esta ferramenta foi utilizada para gerar requisições em um serviço web que está sendo hospedado em uma VM. Com a execução do *Locust*, mais recursos da máquina

virtual são utilizados, e isso faz com que mais recursos da máquina física servidora sejam utilizados. Esse incremento na utilização do servidor acaba por gerar mais consumo de energia. O outro método de geração de carga adotado foi a utilização do comando *stress*. Esse comando é uma ferramenta do Linux que consegue gerar carga em múltiplas partes do sistema (ex., processador, memória, e armazenamento), com o usuário podendo definir parâmetros específicos para cada tipo de comando [doc c]. Sendo assim, foi adotado o comando *stress* para estressar a CPU das VMs hospedadas no servidor em nuvem.

Em seguida, a quarta etapa compreende a definição da estratégia para medição do consumo de energia da máquina física que hospeda todo o ambiente. Para atingir esse objetivo, foi utilizado o dispositivo *Watts Up Meter*, um medidor de consumo que se conecta à tomada do dispositivo desejado e disponibiliza informações em tempo real sobre quantos watts de energia estão sendo consumidos. A ferramenta também registra tais informações em sua memória, para que seja possível extraí-las com um programa de computador suplementar que disponibiliza os dados guardados em forma de texto ou gráfico[Hirst et al. 2013].

Vale ressaltar que além do consumo de energia, nessa etapa também foi definido o método para medição e acompanhamento da utilização de CPU da máquina física. Nesse trabalho foi utilizada a biblioteca de monitoramento de sistema em Python *psutil* [doc b]. A partir de um *script* desenvolvido, foi possível obter as informações da utilização do processador da máquina física durante os testes de estresse. Foi possível, inclusive, ter conhecimento do percentual de uso de cada núcleo da CPU, incluindo o uso total.

Uma vez definidas as estratégias de carga e medição, a quinta etapa consiste na realização dos testes de estresse. Esses testes fazem uso das ferramentas de carga descritas anteriormente por um período de tempo determinado com o objetivo de se obter e armazenar os dados coletados pelo medidor de energia. Este processo é repetido várias vezes (ex., 30), com o objetivo de se poder ter uma maior confiabilidade nos dados coletados. Os parâmetros utilizados para os testes com o *Locust* foram os seguintes: 2000 usuários máximos com 20 usuários por segundo, 30000 usuários máximos com 500 usuários por segundo, 60000 usuários máximos com 1000 usuários por segundo, e 100000 usuários máximos com 2000 usuários por segundo. Enquanto nos testes com o comando *stress* o parâmetro utilizado foi o número de VMs que recebiam carga simultaneamente, variando de 1 até 6 VMs estressadas ao mesmo tempo.

Assim que concluído o processo de realização de testes, a etapa seguinte corresponde a avaliação dos resultados obtidos. Caso nenhuma anomalia seja observada, pode-se dar início à análise dos dados obtidos. Na sexta etapa, as informações obtidas são organizadas e analisadas a fim de se tirar uma conclusão sobre a eficiência energética da máquina avaliada. Para isso, comparam-se os resultados de testes diferentes com seus respectivos parâmetros. Após terminada esta etapa, é feita uma avaliação dos resultados finais e são geradas as conclusões sobre o ambiente em análise.

#### **4. Estudos de caso**

Esta seção apresenta dois estudos de caso que foram realizados para se obter os resultados das medições feitas.

## 4.1. Estudo de Caso I

Esse estudo teve como objetivo a utilização do Locust para a geração de carga em um web server configurado em uma VM no sistema *testbed* de nuvem em análise. A seção inicia com a descrição do ambiente adotado nesse estudo para, em seguida, descrever os resultados obtidos.

### 4.1.1. Ambiente em Nuvem

A Figura 2 mostra o ambiente de nuvem adotado para esse estudo. Esse ambiente utiliza a nuvem Apache Cloudstack em um servidor Dell PowerEdge T350. Esse servidor possui um processador da série Intel Xeon E-2300 com 8 núcleos e 16 *threads*, memória de 64GB e fonte de energia com capacidade de 700W. O serviço em nuvem hospeda uma máquina virtual com um serviço web server que consiste em apenas uma página HTML com uma mensagem em texto. Esse serviço recebe as requisições que são geradas a partir do Locust. Esta VM foi configurada com 512MB de memória RAM, 5GB de disco e utiliza 1 núcleo de CPU de 1GHz. O Locust gera a carga a partir de uma máquina física externa ao ambiente com o Cloudstack. A configuração dessa máquina tem o processador Intel i5 de 11ª geração e 16GB de RAM. O ambiente de teste também conta com um dispositivo de medição de energia, o *Watts up? Pro Meter*, onde o servidor PowerEdge está conectado diretamente. Além disso, o equipamento de medição armazena os dados do consumo de energia que podem ser coletados a partir de uma conexão USB.

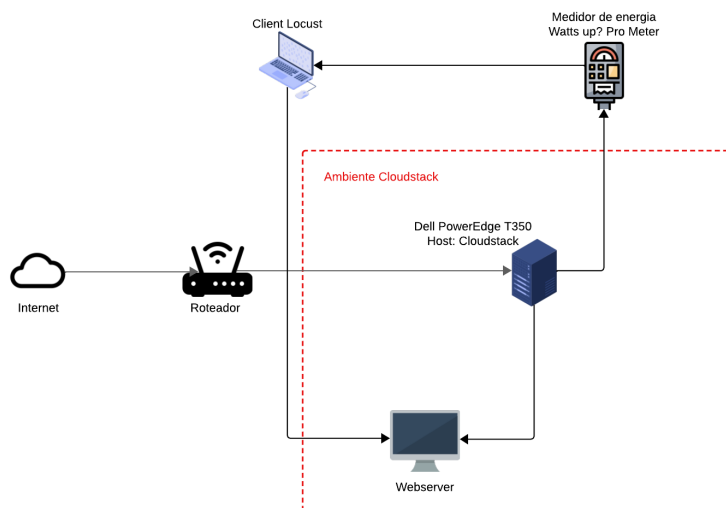


Figura 2. Ambiente do Estudo de Caso I

### 4.1.2. Resultados

Este estudo avalia os resultados obtidos pelos testes de estresse feitos com o framework Locust. Os parâmetros de teste principais para esse estudo são a quantidade máxima de usuários que podem existir simultaneamente para gerar carga no serviço web, e quantos desses usuários são gerados por segundo. É importante destacar que o Locust funciona de

forma com que o número de usuários vai sendo escalando com o tempo, até se chegar ao limite máximo definido.

Um outro parâmetro inicialmente considerado para este experimento foi o tipo de cliente HTTP que o Locust utiliza para fazer suas requisições. Existem no framework duas opções com desempenhos diferentes, chamadas *HttpUser* e *FastHttpUser*. No entanto, a opção com o pior desempenho (*HttpUser*) demonstrou que não era capaz de estressar o ambiente de maneira significativa, gerando pouquíssima alteração no consumo de energia. Logo, esta opção foi descartada e os testes de estresse foram feitos apenas utilizando o cliente HTTP de maior desempenho (*FastHttpUser*).

A Tabela 1 mostra os resultados obtidos sobre a variação de consumo de energia do ambiente sob diferentes cargas. Este valor corresponde à variação de consumo de energia obtido calculando a diferença do consumo no momento que o sistema se encontrava ocioso com o valor do consumo durante os testes de estresse. É possível observar que o consumo de energia alterou-se com a realização dos testes de estresse. Além disso, também é possível notar a variação do nível de utilização médio da CPU durante as diferentes cargas. Essa tabela também mostra a porcentagem de requisições que falharam em relação aos parâmetros de teste utilizados.

**Tabela 1. Resultados teste de estresse com Locust**

Nº Usuários	Usuários/seg	Var. Med. Energia	Uso CPU	Falhas
2000	20	4,64 W	1,84%	0%
30000	500	11,1 W	4,5%	1,6%
60000	1000	10,9 W	4,4%	2,8%
100000	2000	11,3 W	5%	1,75%

É importante ressaltar que o potencial máximo de carga do Locust depende da máquina onde o framework está sendo executado. Isso pode acabar limitando a capacidade de geração de estresse. Por exemplo, nas situações onde os parâmetros de número máximo de usuários e também de usuários gerados por segundo estão mais elevados, percebeu-se um aumento na porcentagem de falhas. Sendo assim, foi identificado que o Locust na máquina cliente geradora da carga estava sendo um gargalo no desempenho.

Durante os testes, o comportamento do Locust foi consistente, mantendo os mesmos níveis de requisições por segundo durante a maior parte do tempo. Em relação a utilização de CPU, foi observada a utilização de apenas 2 núcleos do processador da máquina para suportar o estresse aplicado. Destaca-se que mesmo subindo o nível máximo permitido pelo gerador da carga, obteve-se uma utilização desses núcleos que não chegava a ultrapassar a marca de 40%.

Dessa forma, o Locust no cliente acabou demonstrando ser o gargalo. As suas cargas acabaram por exigir demais da máquina cliente, se tornando um limitador em relação à capacidade de estressar os serviços web. Os resultados dos valores de consumo de energia e utilização de recursos obtidos foram baixos. A variação do consumo de energia chegou a atingir a marca de aproximadamente 11W, e o nível de utilização de CPU alcançou o valor de 5%. Sendo assim, foi realizado um segundo estudo com o objetivo de se conseguir aumentar o nível de utilização da CPU na máquina física do servidor.

## 4.2. Estudo de Caso II

Esse estudo teve como objetivo de estressar VMs no Cloudstack, a partir da utilização do comando *stress*, a fim de se identificar variações da eficiência energética da fonte do servidor. Essa seção inicia com uma descrição do ambiente levado em consideração para, posteriormente, serem descritos os resultados alcançados.

### 4.2.1. Ambiente em análise

A Figura 3 apresenta o ambiente de testes para esse estudo. Nesse ambiente, o servidor que hospeda o Cloudstack é um Dell com processador Intel i5 da 10ª geração, com 6 núcleos e 12 *threads*, 8 GB de memória e fonte de energia com capacidade de 200W. Este ambiente analisado possui 6 máquinas virtuais, e todas possuem as mesmas especificações de 512 MB de memória, 5 GB de disco e 1 núcleo de CPU. Similarmente, esse ambiente também faz uso do medidor de energia *Watts up? Pro Meter* que é conectado à tomada do servidor.

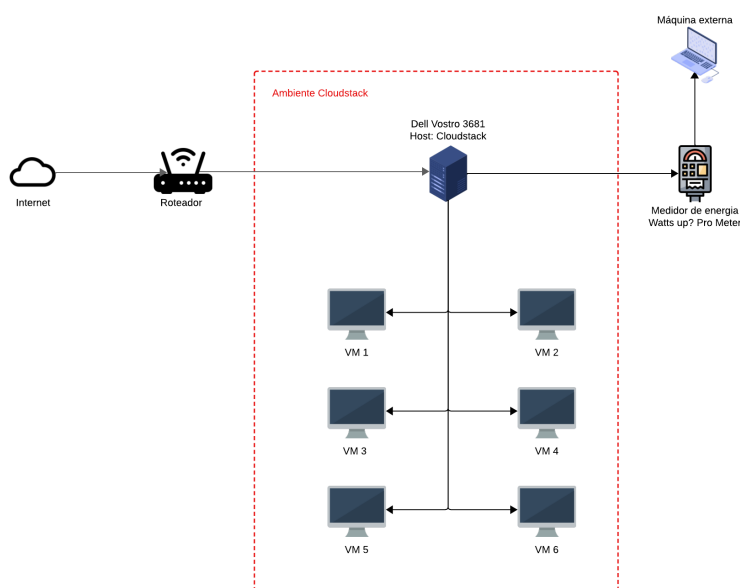


Figura 3. Ambiente do Estudo de Caso II

### 4.2.2. Resultados

Esse estudo adotou uma outra abordagem para testes de estresse no ambiente em análise utilizando o comando *stress*. Esse comando é responsável por realizar funções simples, porém efetivas para gerar carga em diferentes áreas de um sistema. Dessa forma, as VMs do ambiente em análise são estressadas executando esse comando. A medida que são levantadas e estressadas mais VMs, observou-se um aumento na utilização dos recursos físicos da máquina (ex., CPU) e também uma elevação no consumo de energia.

Esta abordagem proporcionou novos comportamentos nos padrões de consumo de energia e nos níveis de utilização de CPU da máquina servidora. Assumindo o nível

de utilização da CPU, os testes de estresse conseguiram ocupar totalmente um núcleo de processador para cada máquina virtual que estava executando o comando stress. Esse comportamento foi diferente da abordagem anterior que costumava ocupar, parcialmente, dois núcleos de processamento. Este comportamento fez com que o nível de utilização de CPU e de consumo de energia fosse extremamente consistente durante todos os testes.

A Tabela 2 contém os resultados obtidos dos testes para esse estudo com o comando stress. Similar a Tabela 1, temos os valores de variação de consumo de energia e utilização de CPU. Porém, este cenário considera o parâmetro de teste como sendo o número de VMs que estavam sendo estressadas simultaneamente no servidor.

**Tabela 2. Resultados teste de estresse com comando stress**

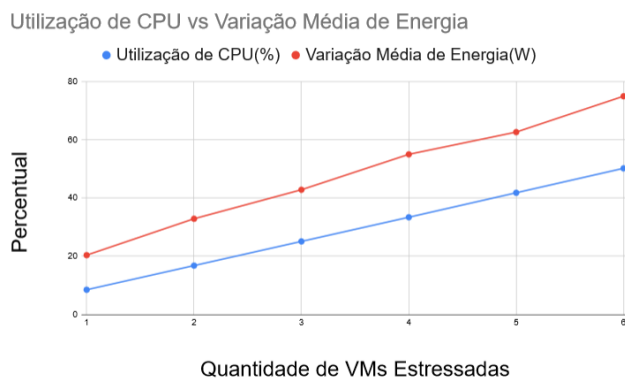
Nº VMs	Uso CPU	Var. Med. Energia	Dif. Var	Dif. Var(%)
1	8,5%	20,38 W	20,38 W	
2	16,8%	32,88 W	12,5 W	-38,66%
3	25,1%	42,84 W	9,96 W	-20,30%
4	33,4%	55 W	12,16 W	+22,08%
5	41,8%	62,64 W	7,64 W	-37,17%
6	50,2%	74,96 W	12,32 W	+61,25%

**Tabela 3. Resultados do comando stress comparado com capacidade total da fonte**

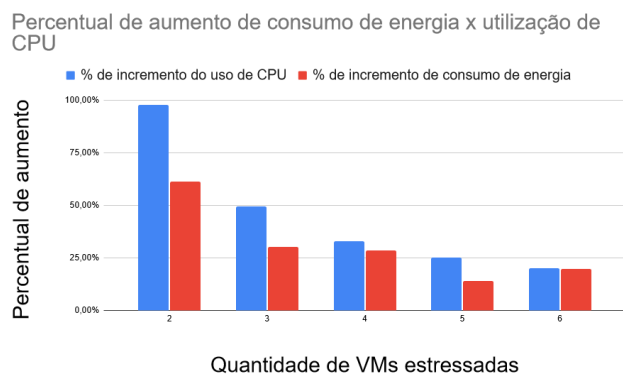
Nº VMs	Uso CPU	Consumo Med. Total	% consumo/cap. máx.
1	8,5%	37,98 W	19,09%
2	16,8%	50,52 W	25,34%
3	25,1%	60,4 W	30,32%
4	33,4%	72,6 W	36,4%
5	41,8%	80,28 W	40,22%
6	50,2%	92,62 W	46,38%

Além disso, há a comparação da diferença de variação de energia dos testes com um número de VMs com o teste anterior, tanto na forma do número exato quanto na porcentagem da diferença. A Tabela 3 mostra os dados do consumo médio total de energia (não só a variação), e a porcentagem correspondente do consumo total de energia com a capacidade máxima da fonte da máquina utilizada.

Observa-se que esta abordagem obteve números muito mais expressivos para ambas as métricas em análise do que o método usado anteriormente. Na Tabela 2, é possível ver que o nível de utilização de CPU se mantém de maneira quase proporcional ao número de máquinas virtuais recebendo as cargas. Sendo assim, para cada VM estressada, o uso de CPU total estava tendo um crescimento de aproximadamente 8,4%. No entanto, a métrica de variação de energia não segue o mesmo padrão. O consumo de energia da máquina avaliada aumenta conforme o nível de utilização de CPU aumenta. Porém, enquanto o uso de CPU aumenta de maneira relativamente proporcional e constante, a variação do consumo de energia não aumenta desta forma.



**Figura 4. Gráfico comparativo de Utilização de CPU x Consumo de energia**



**Figura 5. Gráfico comparativo de Utilização de CPU x Consumo de energia**

A Figura 4 ilustra essas duas métricas em análise. É possível observar os momentos de alta variação nos níveis de energia, e que são seguidas por variações expressivamente inferiores. Isso ocorre em contraste com a métrica do nível de utilização da CPU, que tem um crescimento consistente e previsível.

A Figura 5 apresenta uma comparação do percentual de aumento de cada métrica utilizada em relação ao teste anterior. Por exemplo, nos testes utilizando uma VM, a utilização de CPU foi de 8,5%, e nos testes utilizando duas VMs, a utilização de CPU aumentou para 16,8%. Isso corresponde a um aumento de 99,7%, representado pela primeira barra azul no gráfico. Este gráfico também demonstra o comportamento observado na Figura 4, mostrando que o incremento do nível de utilização da CPU a cada teste de estresse segue um padrão previsível. Por exemplo, os incrementos começam altos e vão diminuindo gradualmente. Enquanto que a variação do consumo de energia não segue o mesmo comportamento. Por exemplo, o aumento percentual na utilização de CPU entre os testes de quatro e cinco VMs foi de 25,15%, enquanto o aumento entre os testes de cinco e seis VMs foi de 20,1%. Comparando os mesmos pontos no aumento de consumo de energia, foi possível observar um aumento de 13,89% entre os testes com 4 e 5 VMs. Enquanto isso, foi obtido um aumento de 19,67%, um aumento percentual repentino, que demonstra a discrepância entre o aumento de uso de CPU e o aumento do consumo de energia. Estes resultados demonstram que a eficiência energética da máquina avaliada varia de acordo com o nível de utilização de seus recursos computacionais. Nas

observações obtidas, há momentos onde o aumento de uso não traz um aumento de consumo tão grande. Enquanto que em outros momentos, acabam ocorrendo variações significativas no consumo de energia. Isso acaba por denotar uma eficiência energética que varia de acordo com a carga a que a fonte de energia se encontra estressada.

## 5. Conclusão

Este trabalho apresentou uma abordagem para a avaliação de um serviço em nuvem em relação à sua eficiência energética. Essa avaliação é baseada em métricas de consumo de energia e utilização de recursos computacionais. A estratégia proposta fez uso de testes de estresse com parâmetros variáveis para obter os resultados das métricas em análise. Além disso, foram utilizadas duas ferramentas distintas para a realização de tais testes, o framework Locust e o comando de *stress*, no intuito de se comparar a viabilidade das duas estratégias. Os resultados demonstraram que o Locust pode exigir da máquina cliente, e assim, se tornar um gargalo. Os experimentos realizados mostraram a viabilidade do uso do comando *stress* para o estudo de caso com as métricas desejadas para a realização da avaliação. Os resultados obtidos demonstraram que a eficiência energética do sistema oscilou dependendo da carga aplicada. Foram detectados pontos onde se tinha uma eficiência energética superior a outras cargas demandadas. Como trabalho futuro, mais testes podem ser realizados para reforçar os resultados dos estudos apresentados neste trabalho. Por exemplo, utilizando máquinas com maior poder computacional, visto que os resultados nos estudos apresentados vieram majoritariamente de uma máquina com poder computacional de nível mediano. Outro possível avanço seria a proposição de modelos para a análise de consumo de energia, assim como ferramentas para dar suporte a tais modelos.

## Referências

- Amazon web services. <https://aws.amazon.com/pt/>. Accessed: 2023-08-01.
- Apache cloudstack. <https://cloudstack.apache.org/>. Accessed: 2023-08-01.
- Apache cloudstack documentation - quick installation guide. <https://docs.cloudstack.apache.org/en/4.17.0.0/quickinstallationguide/qig.html>. Accessed: 2024-02-23.
- Centos. <https://www.centos.org/>. Accessed: 2024-02-23.
- Locust documentation. <https://docs.locust.io/en/stable/index.html>. Accessed: 2024-02-15.
- Psutil documentation. <https://psutil.readthedocs.io/en/latest/>. Accessed: 2023-09-18.
- Stress command documentation. <https://linux.die.net/man/1/stress>. Accessed: 2024-02-05.
- Berl, A., Gelenbe, E., Di Girolamo, M., Giuliani, G., De Meer, H., Dang, M. Q., and Pentikousis, K. (2010). Energy-efficient cloud computing. *The computer journal*, 53(7):1045–1051.

- Chaurasia, N., Kumar, M., Chaudhry, R., and Verma, O. P. (2021). Comprehensive survey on energy-aware server consolidation techniques in cloud computing. *The Journal of Supercomputing*, 77:11682–11737.
- Cloud, H. (2011). The nist definition of cloud computing. *National Institute of Science and Technology, Special Publication*, 800(2011):145.
- de Lima, C. J., da Silva, A. V. G., de Andrade, E. C., and de Almeida Callou, G. R. (2021). Avaliação de desempenho e consumo de energia do ambiente moodle. *Research, Society and Development*, 10(5):e57010515191–e57010515191.
- Hirst, J. M., Miller, J. R., Kaplan, B. A., and Reed, D. D. (2013). Watts up? pro ac power meter for automated energy recording: A product review. *Behavior Analysis in Practice*, 6:82–95.
- Katal, A., Dahiya, S., and Choudhury, T. (2023). Energy efficiency in cloud computing data centers: a survey on software technologies. *Cluster Computing*, 26(3):1845–1875.
- Mastelic, T., Oleksiak, A., Claussen, H., Brandic, I., Pierson, J.-M., and Vasilakos, A. V. (2014). Cloud computing: Survey on energy efficiency. *Acm computing surveys (csur)*, 47(2):1–36.
- Materwala, H. and Ismail, L. (2022). Performance and energy-aware bi-objective tasks scheduling for cloud data centers. *Procedia Computer Science*, 197:238–246.
- Mathew, S. and Varia, J. (2014). Overview of amazon web services. *Amazon Whitepapers*, 105:1–22.
- Odun-Ayo, I., Ajayi, O., and Falade, A. (2018). Cloud computing and quality of service: Issues and developments. In *International Multi-Conference of Engineers and Computer Scientists*.
- Qian, L., Luo, Z., Du, Y., and Guo, L. (2009). Cloud computing: An overview. In *Cloud Computing: First International Conference, CloudCom 2009, Beijing, China, December 1-4, 2009. Proceedings 1*, pages 626–631. Springer.
- Rahman, M., Gao, J., and Tsai, W.-T. (2013). Energy saving in mobile cloud computing. In *2013 IEEE International Conference on Cloud Engineering (IC2E)*, pages 285–291. IEEE.
- Venkateswaran, N., Vidhya, K., Ayyannan, M., Chavan, S. M., Sekar, K., and Boopathi, S. (2023). A study on smart energy management framework using cloud computing. In *5G, Artificial Intelligence, and Next Generation Internet of Things: Digital Innovation for Green and Sustainable Economies*, pages 189–212. IGI Global.