

KaspaFog: uma abordagem na névoa para o gerenciamento de fontes e cargas de eletricidade de uma Microgrid com foco na redução energética

Eric B. Barros¹, Wesley O. Souza¹, Matheus T. M. Barbosa¹, Bruno G. Batista²
Bruno T. Kuehne², Dionisio Leite³, Maycon L. M. Peixoto¹.

¹ Departamento de Ciência da Computação – Universidade Federal da Bahia (UFBA)
Salvador – BA – Brasil

{eric.bernardes, wesleyoliveira, matheus.thiago, maycon.leone}@ufba.br

² Universidade Federal de Itajubá (UNIFEI) - Itajubá – MG – Brasil

{kuehne, guazzelli}@unifei.br

³ Universidade Federal do Mato Grosso do Sul (UFMS) - Ponta Porã – MS – Brasil

{dionisio.leite}@ufms.br

Abstract. *In Microgrids the energy is produced by combining renewable and non-renewable sources. Thus, it is essential to control non-renewable generation to avoid waste. This type of problem has been investigated by several researches, which use variations in the adjustment of the proportional–integral–derivative (PID) controller to avoid energy losses. However, none of the works employed a strategy to reduce the time that power generators are in balance. In this context, the paper presents KaspaFog, an approach that employs a data prediction strategy using the SARIMA model and a neural network with reinforcement learning to adjust the energy generation control. KaspaFog is an infrastructure in the fog supported by the cloud, due to the need for processing and fast response times. KaspaFog reaches 18% reduction in non-renewable compared to the Ziegler-Nichols adjustment.*

Resumo. *Em Microgrids a produção de energia é realizada combinando fontes renováveis e não renováveis de geração de energia. Desse modo, é fundamental o controle da geração não renovável para evitar o desperdício. Esse tipo de problema tem sido investigado por várias pesquisas, que empregam variações do ajuste do controlador Proporcional-Integral-Derivativo (PID) para evitar perdas de energia. Entretanto, nenhum dos trabalhos empregou uma estratégia para reduzir o tempo de equilíbrio entre os geradores de energia. Nesse contexto, este trabalho apresenta o KaspaFog, uma abordagem que emprega uma estratégia de previsão de dados utilizando o modelo SARIMA e uma rede neural com aprendizado por reforço para ajustar o controle da geração de energia. O KaspaFog é uma infraestrutura na névoa apoiada pela nuvem, devido à necessidade de processamento e tempos de respostas rápidos. Com o uso do KaspaFog, foi alcançada uma redução de 18% na produção de energia não renovável em comparação ao ajuste Ziegler-Nichols.*

1. Introdução

Sistemas de energia renovável (RES, do inglês *Renewable Energy Systems*) como sistemas fotovoltaicos e turbinas eólicas estão sendo amplamente instalados por motivações ecológicas, econômicas e políticas [Sinha and Chandel 2015, Hu et al. 2019, Hu et al. 2021]. Integrar RES à rede elétrica habilita a produção de energia por fontes heterogêneas. Redes elétricas com geração distribuída necessitam de uma maior capacidade de gerenciamento e coordenação [Hu et al. 2021].

As *Microgrids* oferecem mais capacidade de gerenciamento e coordenação em função da incorporação dos benefícios das tecnologias da informação e comunicação [Singh and Gore 2021]. Essas redes são de pequeno porte, podendo integrar diferentes tipos de fontes de energia [Meje et al. 2020]. Essas características, tornam possível a eletrificação de áreas remotas, implantação de recursos energéticos renováveis e descarbonização do fornecimento de energia [Papageorgiou et al. 2020]. Portanto, a utilização dessas redes pode trazer benefícios para a distribuição, consumo e produção de energia.

A combinação de várias fontes de energia aumenta a complexidade do gerenciamento da produção em *Microgrids*. A produção vinda das fontes renováveis varia com tempo, devido a fatores como o clima. Essas variações provocam grandes mudanças de tensão que precisam ser evitadas pois, é vital que a rede seja estável para a gestão e o planejamento de energia [Parise et al. 2017]. A energia oriunda de fontes não renováveis é utilizada para complementar a produção renovável e manter a estabilidade. Fontes não renováveis são mais caras devido ao gasto de combustível que é finito, ao contrário da renovável que é infinito. Portanto, é fundamental um controle da geração de energia para evitar o desperdício e obter um baixo custo operacional da produção [Parise et al. 2017].

A geração de energia pode ser controlada utilizando um controlador Proporcional-Integral-Derivativo (PID). Esse tipo de controle é utilizado quando é preciso manter uma determinada variável próxima dos valores especificados. Para manter o valor da variável próximo do valor especificado é realizado um ajuste. Para realizar esse ajuste, o PID precisa do tempo de acomodação, que é o período de tempo utilizado do início até o fim do ajuste. Quanto maior for o tempo de acomodação maior o desperdício de energia ou a probabilidade de problemas causados pela baixa tensão [Parise et al. 2017].

Visando melhorar o sistema de ajuste via PID, métodos de previsão de carga podem antecipar o ajuste por meio de dados históricos de consumo dos usuários. *Microgrids* possibilitam a coleta de grandes quantidades de dados de medidores inteligentes, unidades de medição fasorial, unidades terminais remotas, dispositivos eletrônicos inteligentes e sistemas de controle centralizado [Ekanayake 2020]. Os dados obtidos possuem granularidade e informações necessárias para melhorar o gerenciamento de ativos e operações [Dileep 2020]. Portanto, eles podem ser utilizados para tomar decisões inteligentes quanto ao consumo e produção na rede.

A tomada de decisão a partir dos dados exige poder de processamento e a aplicação de técnicas adequadas. A computação em névoa resolve os problemas de latência e Qualidade de Serviço transferindo parte do processamento da nuvem para a borda da rede [Chouikhi et al. 2019]. Diante do exposto, este trabalho apresenta o Kaspafog, uma abordagem na névoa para o gerenciamento de fontes e cargas de eletricidade

de uma *Microgrid* com foco na redução energética. Essa abordagem emprega predição de dados e redes neurais para ajustar de maneira antecipada a geração de energia convencional. Além disso, para atender a necessidade de processamento e tempo de respostas rápidas, sua infraestrutura utiliza a névoa apoiada pela nuvem. Desse modo, o objetivo de reduzir o desperdício de energia convencional foi alcançado, pois uma redução de perdas de energia foi obtida quando comparado com o controle PID tradicional.

O restante deste artigo está organizado como segue: a Seção 2 apresenta os trabalhos relacionados; a Seção 3 descreve os principais conceitos da solução proposta; a organização dos experimentos é apresentada na Seção 4; a análise dos resultados na Seção 5; e finalmente, as conclusões e os trabalhos futuros são apresentados na Seção 6.

2. Trabalhos Relacionados

Vários trabalhos consideram a utilização de técnicas nos ajustes do controle PID para se obter um melhor desempenho do sistema controlado [Parise et al. 2017] [Barros et al. 2019] [Saadatmand et al. 2021].

A pesquisa de [Hlava et al. 2016] aplica um controlador PID tradicional para manter o equilíbrio entre o consumo e a produção de energia, onde o controlador tem o objetivo de reduzir os picos de carga e para isso é utilizado o método de localização para calcular os parâmetros necessários para realizar o ajuste.

Objetivando manter a estabilidade de uma *Microgrid* com carga dinâmica, a pesquisa de [Parise et al. 2017] propôs um controlador PID/Fuzzy. Esse controlador foi comparado com o controlador PID tradicional em um cenário de mudanças bruscas de carga e ocorrência de falhas. Os experimentos foram realizados utilizando o simulador MATLAB/Simulink e o controlador *Fuzzy* apresentou melhor tempo de estabilização que o PID tradicional.

Na mesma linha dos trabalhos anteriores, a pesquisa de [Barros et al. 2019] foi a primeira a utilizar uma infraestrutura de comunicação de dados para realizar o processamento necessário para realizar o ajuste. Ela consiste na nuvem apoiada pela névoa para fornecer o poder computacional necessário para o ajuste e reduzir o tempo de resposta. Todavia, por utilizar o PID tradicional, não apresenta uma solução para reduzir o tempo de acomodação necessário para realizar o ajuste.

Em [Bharathi and Sasikumar 2021], é proposto um gerenciamento do fluxo de energia em um sistema ON-Grid baseado em painéis solares a fim de garantir a estabilização da rede elétrica. Para isso, foi proposto um algoritmo genético para controlar o processo de harmonização do sistema. Nas simulações realizadas no MATLAB/Simulink foi analisada a comparação de controladores PID baseados em PI, *Fuzzy* e *Genetic Algorithm* (GA). O PID baseado em GA gera o melhor desempenho, produz menos oscilação e tem menos tempo de estabilização em torno de 0,1 segundos em comparação ao PI, o que aponta uma tendência da utilização de algoritmos de otimização juntamente com o controlador PID.

Em [Yin and Lu 2021], os autores estabelecem um modelo para fontes de energia flexíveis usado para mitigar o controle de tensão descoordenado através de um algoritmo *Expandable Deep Width Learning* (EDWL). O algoritmo proposto foi simulado numericamente para realizar o controle preditivo em sistemas IEEE de 118, 300, 1354 e 2383

barramentos e obteve uma performance 0,47% menor que o PID convencional e 0,06% menor que *Deep Neural Networks* (DNNs).

Mais recentemente, a pesquisa de [Saadatmand et al. 2021] propôs o WMFOPID, um controlador proporcional, integral, derivativo e de ordem fracionária baseado em medição de área ampla. Esse controlador é apresentado como um auxiliar para outros controladores de geradores síncronos e visa o ajuste coordenado dos estabilizadores dos sistemas de energia utilizando o algoritmo de otimização por enxame de partículas.

Diante das pesquisas apresentadas, o presente trabalho inova com o desenvolvimento de um novo PID que emprega a predição de dados e redes neurais para reduzir o tempo de acomodação. Uma visão geral dos trabalhos mencionados nesta seção é apresentada na Tabela 1.

Tabela 1. Comparação dos Trabalhos Relacionados

Autores	Algoritmo	Objetivo	Nuvem	Névoa
[Hlava et al. 2016]	PID tradicional	Equilíbrio entre consumo e produção	Não	Não
[Parise et al. 2017]	Baseado em Fuzzy	Estabilidade com carga dinâmica e falha	Não	Não
[Barros et al. 2019]	PID tradicional	Utilização do PID na névoa.	Sim	Sim
[Saadatmand et al. 2021]	Otimização por enxame de partículas	Otimização do ajuste.	Não	Não
[Bharathi and Sasikumar 2021]	Algoritmo Genético	Otimização do ajuste.	Não	Não
[Yin and Lu 2021]	EDWL	Otimização do ajuste.	Não	Não
KaspaFog	<i>Deep Q-Learning</i> , Ziegler Nichols e SARIMA	Otimização do ajuste com e previsão de dados.	Sim	Sim

Conforme apresentado na Tabela 1, o estado da arte converge para sistemas que utilizam técnicas que vão além do PID tradicional. Com isso, o presente trabalho traz significativas contribuições na área de PID que utiliza conhecimento passado para tomada de decisão. O emprego da infraestrutura proposta por [Barros et al. 2019] para realizar o processamento necessário do ajuste e reduzir o tempo de resposta é outra contribuição importante, pois mostra que é possível se valer de sistemas com alto poder de processamento para a atividade de acomodação do PID.

3. KaspaFog

O KaspaFog utiliza técnicas de previsão de dados e ajustes do controle de tensão em *Microgrids*. Essas técnicas são combinadas no desenvolvimento de um novo ajuste para o gerenciamento da geração distribuída. A geração é realizada combinando fontes renováveis (painéis fotovoltaicos, turbinas eólicas) e não renováveis de energia (geradores de energias convencionais). Essas fontes atendem as demandas dos consumidores (residências, prédios comerciais e indústria) através do sistema de distribuição, sendo que a combinação dessas fontes ajuda a reduzir o custo geral da produção de eletricidade [Batoool et al. 2017]. O gerenciamento da produção de energia é realizado para evitar o desperdício e assim diminuir o custo operacional da produção. Esse gerenciamento é realizado por um controlador PID responsável por manter o equilíbrio da rede elétrica.

3.1. PID no controle na geração de energia

O PID é amplamente utilizado quando é necessário manter uma determinada variável próxima a valores especificados dentro de um determinado sistema. A aproximação da variável com o valor desejado pode ser obtida através do cálculo da taxa de erro proporcional, integral e derivada, como apresentado na Equação 1:

$$u(t) = Kp * e(t) + Ki * \int e(t)dt + Kd * \frac{de(t)}{dt} \quad (1)$$

Onde, $e(t)$ é a taxa de erro do valor desejado (*set point*) com o valor obtido (variável de processo) e Kp , Ki e Kd são os ganhos proporcional, integral e derivados, respectivamente.

O PID precisa ser ajustado sempre que ocorre uma variação na produção de energia renovável. Caso o ajuste não seja adequado, vai ocorrer desperdício de energia. Esse desperdício ocorre devido à lentidão que o sistema de controle leva para equilibrar a produção e consumo de energia. Para evitar o desperdício, o ajuste deve ser realizado rapidamente e de maneira correta. Os ajustes realizados corretamente com baixo tempo de resposta têm grande efeito na velocidade e estabilidade do sistema de controle [Mahdavian and Wattanapongsakorn 2013]. Portanto, métodos adequados para encontrar os parâmetros Kp , Ki e Kd adequados dentro do processo controlado precisam ser utilizados. Uma das formas de realizar o ajuste encontrada na literatura é o método de Ziegler-Nichols.

3.2. Autoajuste usando Ziegler-Nichols

O método Ziegler Nichols é um dos mais utilizado para encontrar os ganhos proporcional, integral e derivado de um controle PID. Esse método zera os ganhos integral e derivado, ajustando apenas o ganho proporcional até que a amplitude da saída do controlador fique constante. Em seguida, os valores de Kp , ganho crítico Ku e período crítico Tu são gravados, e os valores para Ki são testados para diminuir o erro estacionário. Após o teste de valores para Ki , o Kd pode ser utilizado para aumentar a velocidade de convergência para o valor desejado [Minchala-Avila et al. 2016]. A Equação 2 apresenta o método.

$$u(t) = 0,6 * Ku * e(t) + \frac{Tu}{2} * \int e(t)dt + \frac{Tu}{4} * \frac{de(t)}{dt} \quad (2)$$

Na Equação 2, o ganho proporcional Kp será $0,6Ku$ e os valores de Ki e Kd são obtidos seguindo a proporção de declínio de $1/4$ do Tu , ou seja, $Kp = 0,6 * Ku$, $Ki = \frac{Tu}{2}$ e $Kd = \frac{Tu}{4}$. Essa abordagem através de amostragens do sistema torna possível encontrar os melhores valores para os ganhos de um controlador PID. Entretanto, os cálculos podem levar a um estado de instabilidade principalmente em casos em que há variações do *set point*. Essa instabilidade atrasa a busca do limite estável do sistema pelo método, tornando-o menos eficiente.

3.3. Autoajuste usando Ziegler Nichols e aprendizado por reforço

O Kaspafog implementa um algoritmo para o ajuste do PID que combina o autoajuste Ziegler Nichols com uma técnica de aprendizado de máquina por reforço. Esse algoritmo

observa o tipo de amostragem obtida para melhorar os ganhos (K_p , K_i e K_d). As amostras são obtidas do consumo e da produção renovável e são utilizadas como entrada. A tomada de decisão do algoritmo é baseada na relação entre as entradas. O pseudocódigo do algoritmo proposto está representado no Algoritmo 1

Como apresentado no Algoritmo 1, a partir da relação entre as entradas é verificado a quantidade de energia convencional que precisa ser utilizada para complementar a produção de energia renovável. Após a verificação, o PID entra em ação ajustando o valor de K_p através da técnica de encontrar a primeira oscilação. Quando a oscilação é encontrada o valor de K_p é salvo e é iniciado a rede neural com cinco entradas e três saídas. Dessas cinco entradas três são os valores de K_p , K_i e K_d e duas são os ajustes que foram feitos nelas. As três saídas indicam o próximo ganho que será modificado.

Algoritmo 1: Treinando controle PID

```

1 Inicialize amostragem
2 para cada hora do dia faça
3   Obter amostra de cada hora
4    $K_i$ ,  $K_d$ , ajuste, ultima_recompensa  $\leftarrow$  0
5   repita
6      $K_p \leftarrow$  ajustarP(ajuste, kp)
7     PID.execute( $K_p$ ,  $K_i$ ,  $K_d$ , amostra)
8   até (não encontrar oscilação)
9   ultima_ação  $\leftarrow$  [ $K_p$ ,  $K_i$ ,  $K_d$ , ajuste, ajuste]
10  repita
11    ação  $\leftarrow$  agentPID.update(ultima_recompensa, ultima_acao)
12    se ação = 0 então
13       $K_p \leftarrow$  ajustarP(ajuste, ultima_recompensaP)
14    senão
15      se ação = 1 então
16         $K_i \leftarrow$  ajustarI(ajuste, ultima_recompensaI)
17      senão
18        se ação = 2 então
19           $K_d \leftarrow$  ajustarD(ajuste, ultima_recompensaD)
20    controle  $\leftarrow$  PID.execute( $K_p$ ,  $K_i$ ,  $K_d$ , amostra)
21    ultima_ação  $\leftarrow$  [ $K_p$ ,  $K_i$ ,  $K_d$ , ajuste, -ajuste]
22    se controle diminuiu perda então
23      ultima_recompensa  $\leftarrow$  positiva
24    senão
25      ultima_recompensa  $\leftarrow$  negativa
26  até (controle  $\geq$  102% and controle  $\leq$  98%)
27 save.model(agentPID)
28 sendtofog(agentPID, melhores_resultados)

```

O algoritmo de ajuste sempre tenta encontrar um valor onde as perdas do controle em relação ao *set point* fiquem próximas a zero. Como nem sempre é possível obter essa precisão, foi definido valores aproximados que fiquem entre 102% e 98% do *set point*. A rede neural faz alterações nos valores dos ganhos K_p , K_i e K_d e o algoritmo verifica através da amostra se o *PID* está mais próximo ou mais distante do *set point*. Caso a rede neural escolha um ganho que diminua a perda, ela ganha uma recompensa positiva, caso

contrário uma negativa. Essa estratégia possibilita que os ganhos do PID estejam sempre adequados ao sistema de energia e assim evitar perdas desnecessárias.

3.3.1. Previsões do consumo e da geração de energia

A previsão da produção de energia é uma das aplicações mais importantes de redes elétricas inteligentes [Ozer et al. 2021]. O *SARIMA* foi selecionado para melhorar a qualidade das amostras e auxiliar no treinamento da rede neural no ajuste do PID. Essa técnica de autorregressão para a análise de séries temporais pode ser utilizada para tentar prever valores futuros. Essa previsão é realizada tentando encontrar padrões em dados históricos com o objetivo de prever possíveis dados futuros. A capacidade de previsão do *SARIMA* pode ser utilizada para prever tendências de consumo e produção de energia. Essas tendências podem ser utilizadas para treinar uma rede neural através do aprendizado por reforço, melhorando a acurácia e o tempo de resposta das solicitações de ajuste do PID. Na Figura 1, são observadas as linhas real e previsão. É possível notar que a linha de previsão acompanha a tendência do comportamento real do sistema, reforçando, assim, o seu uso pelo Kaspafog.

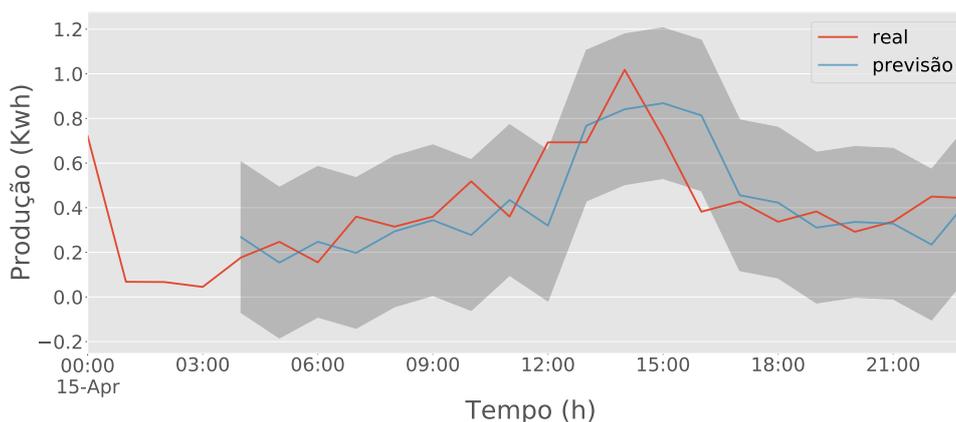


Figura 1. Comparação entre previsão e produção real usando *SARIMA*.

3.4. Nuvem no auxílio do ajuste

Para realizar a previsão através do *SARIMA* e encontrar os ajustes adequados para o PID é necessário capacidade de processamento. A computação em nuvem oferece capacidade de processamento e de receber os dados gerados pelos dispositivos inteligentes que compõem a rede elétrica em toda a sua extensão. Essa capacidade é oferecida por meio de serviços, o que auxilia na redução de custos e otimização dos recursos utilizados.

Todos os serviços da nuvem são oferecidos pelo seu conjunto de servidores, estes geralmente estão distantes do local onde os dados estão sendo coletados. Devido as longas distâncias, a transmissão de dados está mais sujeita a atrasos e gargalos que podem influenciar no tempo de resposta. Aplicações sensíveis a esses atrasos podem ser prejudicadas com problemas de latência e largura de banda [Garg et al. 2021]. Os atrasos também podem ocorrer devido a gargalos gerados pela grande quantidade de dados recebidas pelas aplicações. Como a capacidade dos canais de transmissão são limitadas as aplicações competem pelo meio para o envio dos dados. A Figura 2a ilustra um gargalo

no envio dos dados das aplicações da *Microgrid* para a nuvem. Como apresentado na Figura 2a, na parte inferior é possível observar as aplicações dos consumidores disputando o meio de transmissão para enviar dados a parte superior que representa a nuvem.

3.5. Névoa no auxílio do ajuste

Para evitar os problemas relacionados a utilização na nuvem novas arquiteturas estão sendo propostas. Uma delas é a utilização da computação em névoa que possibilita que todo o processamento ou parte dele seja realizado na borda da rede. O processamento é realizado pelo conjunto de nós da névoa que estão próximos das aplicações, possibilitando tempos de resposta mais rápidos e menor possibilidade de gargalos. Para processar todos os dados na névoa seria necessário uma infraestrutura robusta, o que poderia tornar a solução muito dispendiosa. Para reduzir os custos a nuvem pode ser combinada com a névoa. Essa configuração é vantajosa, pois torna possível aproveitar a grande capacidade de processamento da nuvem juntamente com os tempos de resposta mais rápidos da névoa. A combinação da nuvem com a névoa é ideal para aplicações que precisam da robustez da computação na nuvem e da localidade da computação na névoa [Abbasi et al. 2021]. Essa abordagem está ilustrada na Figura 2b, onde na parte inferior é possível observar os consumidores e envio de dados para a névoa. No centro da figura os nós da névoa são representados, sendo eles responsáveis por parte do processamento e com menor tempo de resposta devido a sua proximidade da borda. Na parte superior é representada a nuvem que apresenta grande capacidade de processamento, mas com tempos de resposta inferiores a névoa.

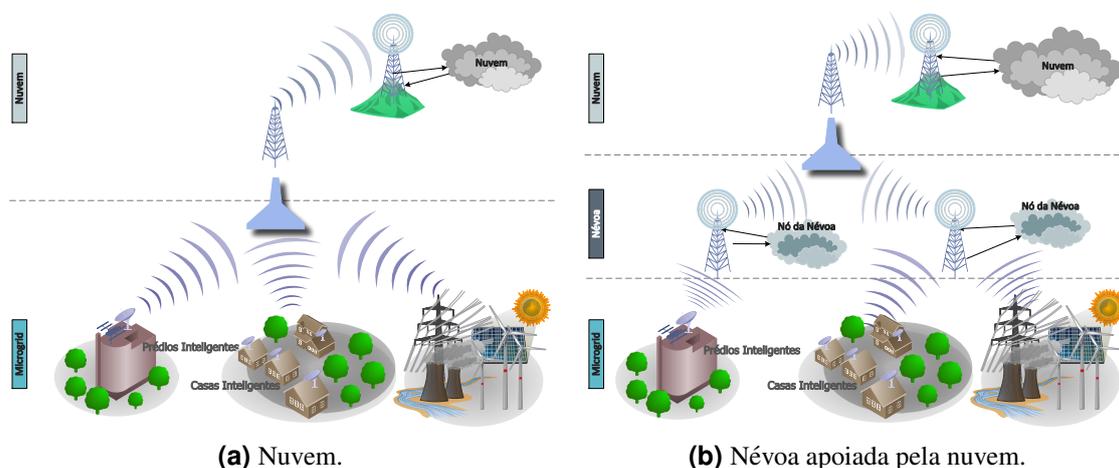


Figura 2. Utilização da nuvem e da névoa apoiada pela nuvem no processamento de dados em uma *Microgrid*.

3.6. Arquitetura proposta

O Kaspafog se aproveita dos benefícios da nuvem e da névoa, sendo que um fluxo do seu funcionamento é ilustrado na Figura 3. Na parte superior da Figura está a nuvem, que é responsável pelo recebimento dos dados e por parte do processamento. Esses dados são agrupados e armazenados diariamente para serem utilizados pelo *SARIMA* com o objetivo de fazer previsões tanto para o consumo quanto para a produção de energia. Com as previsões obtidas pelo *SARIMA* é possível identificar qual a tendência da utilização

de energia convencional e executar o Algoritmo 1 para realizar o treinamento da rede neural. Ao terminar o processamento do treinamento da rede neural, a nuvem envia para a névoa o modelo treinado e os dados que apresentam maior frequência de utilização. Utilizando essas informações a névoa está preparada para responder as solicitações vindas da *Microgrid*.

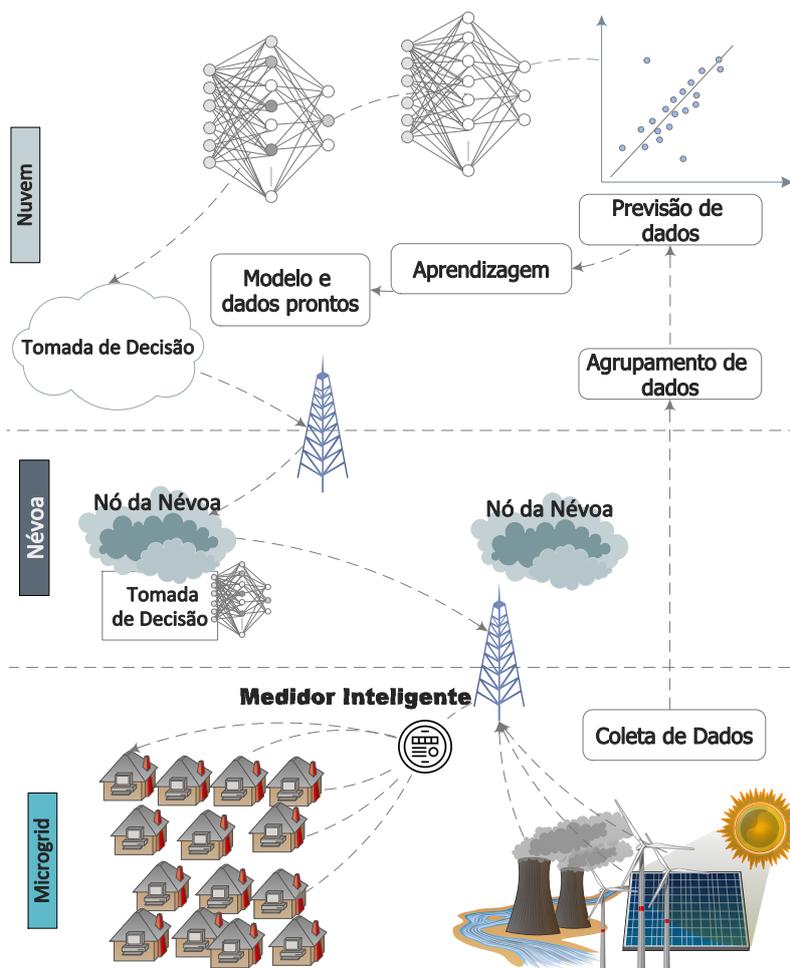


Figura 3. Arquitetura proposta utilizando a névoa como extensão da nuvem.

A *Microgrid* solicita ajuste ao Kaspafog sempre que houver variações na utilização da energia convencional e as perdas de energia aumentarem. O ajuste é influenciado pela infraestrutura e a capacidade de tomada de decisão da solução proposta. Portanto, esses fatores influenciam no desempenho e cada um deles foi analisado nos experimentos.

4. Organização dos experimentos

Para avaliar o desempenho da proposta foi utilizado um experimento fatorial completo, como demonstrado em [Bukh 1992]. No experimento fatorial completo são definidos dois níveis para cada fator k . Essa abordagem tem por objetivo medir o tempo de resposta e perdas de energia usando todas as combinações entre os níveis dos fatores. Cada um dos quatro fatores considerados (A, B, C e D) possuem dois níveis, obtendo assim uma

combinação de 2^4 . O experimento fatorial utilizado no presente trabalho pode ser visto na Tabela 2. Para efetuar os experimentos o simulador *Gridlab-D* (4.2v) foi usado para gerar os dados de consumo e produção de energia, e o simulador Omnetpp (4.6v) para simular a rede de dados.

Tabela 2. Fatores e níveis

Fatores	Níveis	
Algoritmos (A)	<i>KaspaFog</i>	<i>ZN Normal</i>
Infraestrutura (B)	Nuvem	Névoa
Tamanho-Cache (C)	2	8
Tráfego médio (D)	50MB	450MB

O cenário experimental foi definido com uma *Microgrid*, um gerador de energia convencional e uma infraestrutura para o processamento do ajuste e dos dados dos consumidores. A *Microgrid* é isolada possuindo 20 casas com capacidade de consumo de até 500 kWh/mês e geradores de energia eólica e solar com capacidade de geração de 500kWh/mês. Para complementar a produção, um gerador de energia convencional com capacidade de suprir toda a demanda de energia foi adicionado.

A comunicação da *Microgrid* e demais elementos pela infraestrutura de rede é limitada por enlaces com capacidade de 10 Gbit/s para tráfego de dados. A infraestrutura de processamento é composta pela nuvem e pela névoa. Para avaliar o desempenho da abordagem proposta quanto ao tempo de resposta, a combinação da névoa com a nuvem será comparada apenas com a nuvem. No caso da combinação, a névoa é posicionada na borda da rede e utiliza o modelo treinado e os dados da nuvem para fazer o ajuste do PID. Caso contrário, a nuvem treina o modelo com os dados e executa um algoritmo para o ajuste.

5. Análise dos Resultados

Para realizar os ajustes do PID, os algoritmos Ziegler Nichols convencional e o *KaspaFog* são comparados. Nos testes, todos os fatores e níveis descritos na Tabela 2 são considerados para a análise de sua influência nas variáveis de resposta. A Figura 4a mostra uma comparação entre a regulagem dos dois algoritmos. A regulagem feita pelo controlador é realizada tentando aproximar os valores da tensão do *setpoint* desejado. O *setpoint* é a diferença entre a produção renovável e o consumo, sendo que as correções da tensão são feitas na produção convencional de energia. Assim, a produção convencional está sempre sendo regulada de acordo com a relação das variações de energia renovável com o consumo.

A Figura 4b, mostra que o principal fator de influência nas duas variáveis de resposta (tempo de solicitação e desperdício de energia) foi o fator algoritmo com 37,63% e 74,96%, respectivamente. A infraestrutura obteve 21,70% de influência nos tempos de resposta das solicitações e 10,87% nas perdas de energia. Essa influência é explicada pela proximidade da névoa com o local de processamento. Essa proximidade evita que as requisições atrasem para receber os ajustes devido ao congestionamento da rede. Evitando o congestionamento, as requisições ganham em média 8 segundos na busca por novos ajustes quando a rede está saturada. Para acelerar as respostas da névoa foi disponibilizado um cache para armazenar os dados mais acessados. Essa modificação obteve

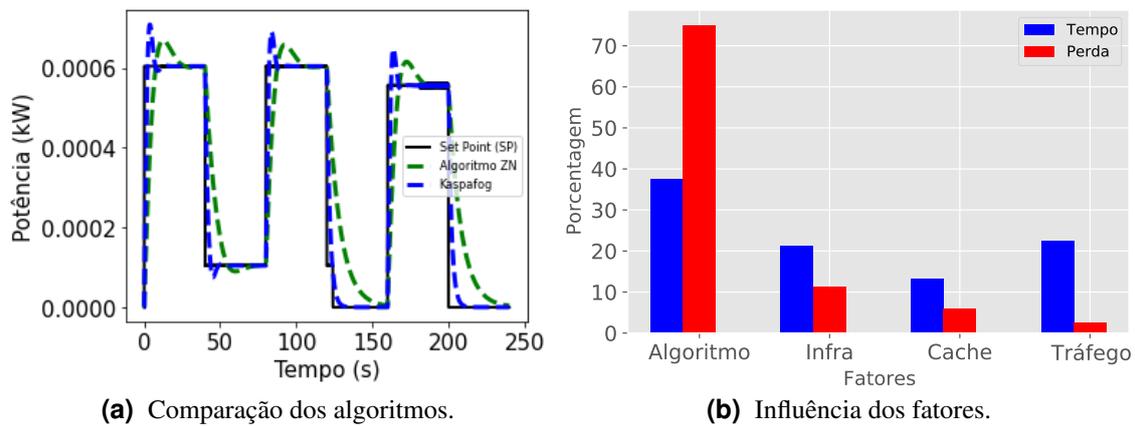


Figura 4. Comparação dos algoritmos e Fatores de influência.

13,10% de influência no tempo de resposta das solicitações e 7,93% nas perdas de energia. Esses resultados são obtidos devido a previsão dos dados e ao tamanho do cache que ajudam a evitar o *cache-miss*.

A perda de energia é influenciada pela variação de energia e pelo tempo necessário para encontrar a solução mais adequada. Um comparativo dessas perdas de energia usando os algoritmos propostos e a utilização da névoa e da nuvem pode ser visto nas Figuras 5 e 6. Na Figura 5, é mostrada a comparação entre os algoritmos utilizados na névoa, sendo consideradas duas visões dos experimentos. A primeira visão mostra o ambiente com o tráfego médio de 50 MB/s, sendo o Kaspafog o algoritmo com as menores perdas de energia utilizando caches de 2 ou 8 unidades. A segunda visão mostra o ambiente com o tráfego médio de 450 MB/s, nessa condição, embora as perdas de energia aumentem considerando abordagens com 2 unidades de cache, o Kaspafog consegue manter as perdas similares ao tráfego de 50 MB/s, quando utiliza 8 unidades de cache.

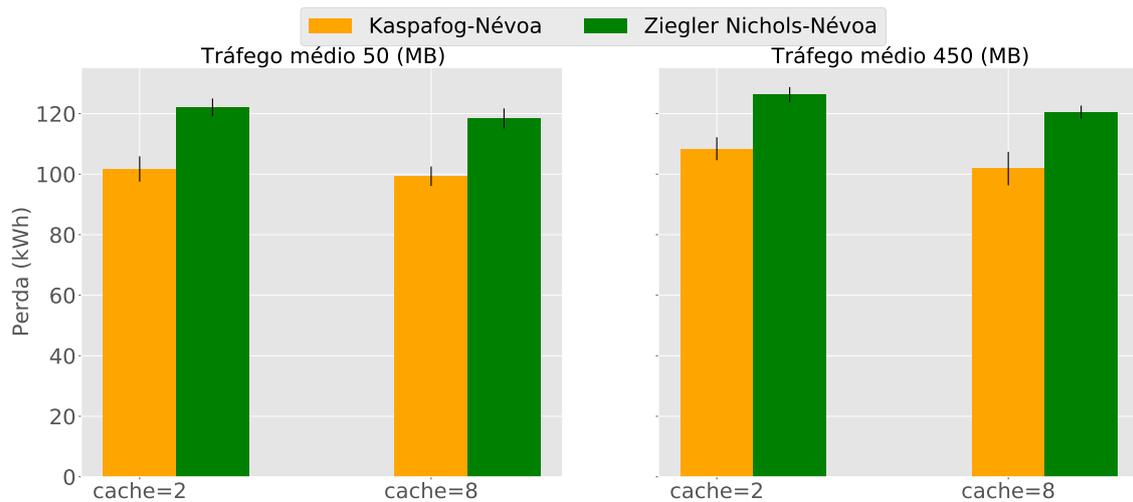


Figura 5. Desperdício de energia na névoa.

Na Figura 6, os experimentos foram realizados considerando uma infraestrutura de computação em nuvem, onde novamente foram analisadas duas visões, uma com tráfego médio de 50 MB/s e outra com tráfego médio de 450 MB/s. Observando as duas visões

é possível perceber que o Kaspafog manteve o desperdício menor em ambos os casos. Contudo, houve um aumento geral do desperdício de energia. Esse aumento se deve principalmente ao algoritmo Ziegler Nichols, que por não usar um cache com dados previstos demora mais tempo para achar os ajustes, principalmente no início da execução dos experimentos quando o cache ainda está vazio.

Assim como esperado, notou-se que quando a rede de dados está congestionada os tempos de respostas e o desperdício aumentam. Esse gargalo da rede ocorre quando cada casa começa a trafegar em média 450 MB/s. Contudo, a utilização do cache e da névoa reduzem a influência desse fator devido as suas características, permitindo que os tempos de repostas das solicitações sejam menores, mesmo com o tráfego alto. Esse fator influenciou os tempos de resposta em 22,65% e o desperdício de energia em 3,05% . A baixa influência no desperdício de energia se deve a ajuda da névoa e do cache, que entregam rapidamente ajustes ótimos diminuindo desperdícios de energia obtidos devido ao congestionamento da rede.

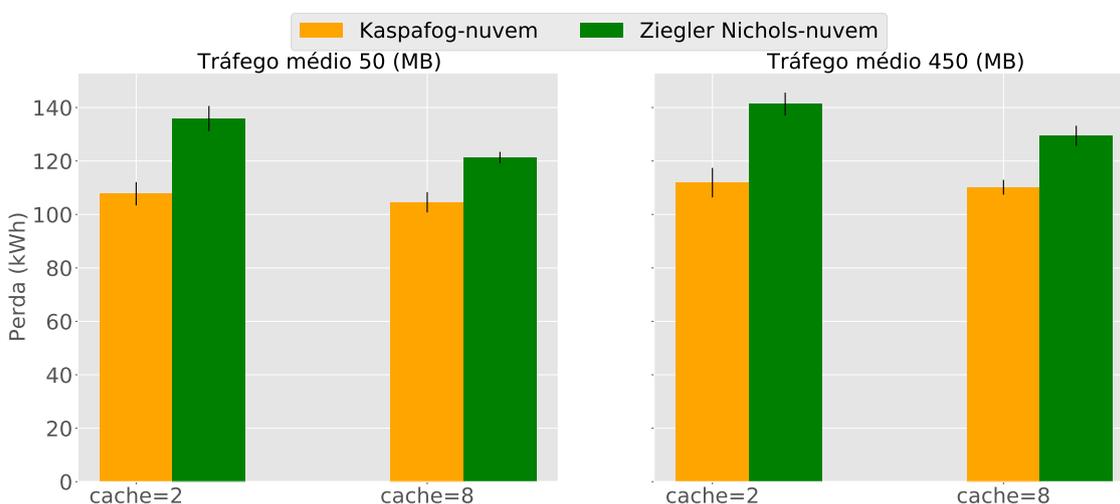


Figura 6. Desperdício de energia na nuvem.

O Kaspafog obteve o desempenho apresentado devido a composição de sua arquitetura. A capacidade de processamento e armazenamento da nuvem permitiu através de dados históricos realizar a predição de dados com o *SARIMA* e o treinamento da rede neural. O treinamento da rede neural com os dados previstos resultou em respostas mais rápidas para as solicitações de ajuste. O cache na névoa reduziu o tempo de transmissão do ajuste pela infraestrutura, reduzindo ainda mais o tempo de resposta das solicitações.

6. Conclusões e Trabalhos Futuros

Este trabalho utilizou uma infraestrutura de computação em névoa auxiliada pela nuvem para propor um novo algoritmo com o objetivo de reduzir o desperdício de energia. O algoritmo combina aprendizado por reforço *Deep Q-Learning* com o método Ziegler Nichols para obter melhores ajustes para o controle PID no equilíbrio da potência de uma *Microgrid*. O aprendizado por reforço é aplicado na rede neural treinada com dados obtidos do modelo estatístico *SARIMA*.

A abordagem proposta obteve uma redução do desperdício de energia em aproximadamente 18% em relação ao método Ziegler Nichols convencional. Essa redução

foi possível devido a combinações dos fatores propostos na arquitetura e a utilização do cache na névoa. O cache reduziu o tempo de resposta das solicitações evitando o congestionamento de dados. O algoritmo proposto também reduz tempo de resposta devido ao treinamento antecipado e ao refinamento dos parâmetros. O treinamento antecipado acelera a tomada de decisão possibilitando responder as solicitações mais rapidamente. Essas respostas são mais acuradas devido ao refinamento dos parâmetros que aceleram a convergência da variável de processo ao valor desejado (*set point*). Em trabalhos futuros, pretende-se empregar técnicas de aprendizado supervisionado a fim de encontrar soluções mais rapidamente para diminuir o desperdício de energia.

7. Agradecimentos

Os autores agradecem à FAPESB, FAPEMIG, CAPES, CNPq pelo apoio financeiro.

Referências

- Abbasi, M., Mohammadi-Pasand, E., and Khosravi, M. (2021). Intelligent workload allocation in iot–fog–cloud architecture towards mobile edge computing. *Computer Communications*, 169:71–80.
- Barros, E. B. C., Batista, B. G., Kuehne, B. T., Peixoto, M. L. M., et al. (2019). Fog computing model to orchestrate the consumption and production of energy in microgrids. *Sensors*, 19(11):2642.
- Batool, M., Islam, S. M., and Shahnia, F. (2017). Power transaction management amongst coupled microgrids in remote areas. In *2017 IEEE Innovative Smart Grid Technologies - Asia (ISGT-Asia)*, pages 1–6.
- Bharathi, K. and Sasikumar, M. (2021). Power flow control based on bidirectional converter for hybrid power generation system using microcontroller. *Microprocessors and Microsystems*, 82:103950.
- Bukh, P. N. D. (1992). The art of computer systems performance analysis, techniques for experimental design, measurement, simulation and modeling.
- Chouikhi, S., Merghem-Boulaia, L., and Esseghir, M. (2019). A fog computing architecture for energy demand scheduling in smart grid. In *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pages 1815–1821. IEEE.
- Dileep, G. (2020). A survey on smart grid technologies and applications. *Renewable Energy*, 146:2589–2625.
- Ekanayake, E. (2020). Use of data analytics in microgrids: A survey of techniques. In *2020 3rd International Conference on Power and Energy Applications (ICPEA)*, pages 103–107. IEEE.
- Garg, S., Kaur, K., Kaddoum, G., and Guo, S. (2021). Sdn-nfv-aided edge-cloud interplay for 5g-envisioned energy internet ecosystem. *IEEE Network*, 35(1):356–364.
- Hlava, J., Zemtsov, N., and Frantsuzova, G. (2016). Application of pid controller based on the localization method for ancillary service provision. In *2016 International Siberian Conference on Control and Communications (SIBCON)*, pages 1–6. IEEE.

- Hu, J., Li, Z., Zhu, J., and Guerrero, J. M. (2019). Voltage stabilization: A critical step toward high photovoltaic penetration. *IEEE Industrial Electronics Magazine*, 13(2):17–30.
- Hu, J., Shan, Y., Guerrero, J. M., Ioinovici, A., Chan, K. W., and Rodriguez, J. (2021). Model predictive control of microgrids—an overview. *Renewable and Sustainable Energy Reviews*, 136:110422.
- Mahdavian, M. and Wattanapongsakorn, N. (2013). Pid controller tuning and optimizing for greenhouse lighting application considering real-time pricing in the smart grid. In *2013 International Computer Science and Engineering Conference (ICSEC)*, pages 85–90. IEEE.
- Meje, K. C., Bokopane, L., and Kusakana, K. (2020). Microgrids control strategies: A survey of available literature. In *2020 International Conference on Smart Grid and Clean Energy Technologies (ICSGCE)*, pages 167–173. IEEE.
- Minchala-Avila, L. I., Palacio-Baus, K., Ortiz, J. P., Valladolid, J. D., and Ortega, J. (2016). Comparison of the performance and energy consumption index of model-based controllers. In *2016 IEEE Ecuador Technical Chapters Meeting (ETCM)*, pages 1–6. IEEE.
- Ozer, I., Efe, S. B., and Ozbay, H. (2021). A combined deep learning application for short term load forecasting. *Alexandria Engineering Journal*, 60(4):3807–3818.
- Papageorgiou, A., Ashok, A., Farzad, T. H., and Sundberg, C. (2020). Climate change impact of integrating a solar microgrid system into the swedish electricity grid. *Applied Energy*, 268:114981.
- Parise, G., Martirano, L., Kermani, M., and Kermani, M. (2017). Designing a power control strategy in a microgrid using pid/fuzzy controller based on battery energy storage. In *2017 IEEE International Conference on Environment and Electrical Engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*, pages 1–5. IEEE.
- Saadatmand, M., Mozafari, B., Gharehpetian, G. B., and Soleymani, S. (2021). Optimal coordinated tuning of power system stabilizers and wide-area measurement-based fractional-order pid controller of large-scale pv farms for lfo damping in smart grids. *International Transactions on Electrical Energy Systems*, 31(2):e12612.
- Singh, B. P. and Gore, M. M. (2021). Softmicrogrid: a software assisted microgrid for optimal prosumer satisfaction. *Technology and Economics of Smart Grids and Sustainable Energy*, 6(1):1–18.
- Sinha, S. and Chandel, S. (2015). Review of recent trends in optimization techniques for solar photovoltaic–wind based hybrid energy systems. *Renewable and Sustainable Energy Reviews*, 50:755–769.
- Yin, L. and Lu, Y. (2021). Expandable deep width learning for voltage control of three-state energy model based smart grids containing flexible energy sources. *Energy*, page 120437.