

Utilizando *Fog* para a Análise do Fluxo de Carga Dinâmico nas *Smart grids*

Eric Bernardes¹, Bruno T. Kuehne², Bruno G. Batista²,
Dionisio M. L. Filho³, Maycon L. M. Peixoto¹

¹ Universidade Federal da Bahia (UFBA)
Salvador – BA – Brasil

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

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

bernardes.eric@gmail.com, maycon.leone@ufba.br

Abstract. *With the scarcity of natural resources it is becoming increasingly difficult to meet the growing demand for energy. The current distribution models are unidirectional and only transmit power from the generators to the consumers, not considering alternative energy sources. Thus, with the emergence of smart grids, it is possible to create bidirectional distribution models, where alternative energy generators present in the consumers can transmit through the distribution network. Yet, to avoid an overload in the system it is necessary to calculate the power flow in real time. In this paper, fog computer was used to perform real-time power flow calculations, reducing the logical distance of information between distribution centers and consumers. Thus, the performance evaluation of power flow algorithms that can best be processed in the fog was made, and the results obtained showed that it is possible to realize the smart grid based on the fog computer and thus create intelligent electrical networks that react to the environment.*

Resumo. *Com a escassez de recursos naturais, atender a demanda de energia está se tornando cada vez mais difícil. Os atuais modelos de distribuição são unidirecionais e transmitem apenas energia dos geradores para os consumidores, não considerando fontes alternativas de energia. Assim, com o surgimento de redes inteligentes, é possível criar modelos de distribuição bidirecionais, onde geradores de energia alternativos presentes nos consumidores podem transmitir energia através da rede de distribuição. No entanto, para evitar uma sobrecarga no sistema, é necessário calcular o fluxo de energia em tempo real. Neste trabalho, foi utilizado um servidor fog para realizar cálculos de fluxo de energia em tempo real, reduzindo a distância lógica das informações entre os centros de distribuição e os consumidores. Assim, foi feita uma avaliação de desempenho dos algoritmos de fluxo de potência que melhor podem ser processados na Fog. Os resultados obtidos mostraram que é possível criar uma smart grid baseada auxiliada pela Fog e assim criar redes smart grids que reagem ao ambiente.*

1. Introdução

As redes elétricas atuais seguem um modelo convencional, onde as geradoras de energia transmitem unidirecionalmente a potência gerada para os destinos (residência, comércio ou

indústria). Essa transmissão é feita em várias etapas, passando por subestações, distribuidores e linhas de transmissão, o que torna esse tipo de transmissão ineficiente devido à perda da alta voltagem, principalmente nas linhas de transmissão [Jdeed 2016]. Além disso, a quantidade de energia consumida aumentou 2,5% por ano nos últimos 20 anos e devido a escassez de recursos energéticos no futuro não será possível atender toda essa demanda [Gungor et al. 2011].

Esse modelo de rede de distribuição transporta energia de várias centrais elétricas a um grande número de consumidores, mas não considera recursos energéticos alternativos. Além disso, não permite que o consumidor monitore ou controle o seu consumo de energia de forma inteligente [Okay and Ozdemir 2016]. Sendo assim, um novo modelo de distribuição de energia vem sendo proposto, permitindo a transmissão de energia de forma bidirecional (aproveitando energias renováveis geradas também pelos consumidores) e troca de informações para a administração das redes de distribuição, criando um modelo de distribuição inteligente: *smart grids*.

Logo, tentar prever o comportamento das redes de distribuição através da troca de dados realizada durante a transmissão de potência é uma área de pesquisa importante nas *smart grids* e que pode ser feita através do uso de simulações em tempo real, mas para isso, é necessário dispor também de uma infraestrutura de tecnologia da informação adequada. Atualmente, muitos modelos de *smart grids* utilizam como principal referência de armazenamento de dados *datacenters* localizados nas nuvens, que oferecem um monitoramento da geração e do consumo em tempo quase real da distribuição de energia [Okay and Ozdemir 2016].

Contudo, as *smart grids* vão além de monitoramento e controle de transmissão de potência, é necessário também alcançar o equilíbrio entre a geração de potência e o seu consumo. Esse equilíbrio atualmente é feito através do cálculo do fluxo de potência, feito em tempo de projeto e monitoramento das geradoras e distribuidoras. Porém, realizar um bom projeto do fluxo de potência depende do conhecimento sobre os efeitos de se adicionar novos geradores, aumentar a demanda e conectar novas linhas de transmissão ao sistema [Bokka 2010], tarefa que um simulador na nuvem pode realizar em tempo de operação da rede de distribuição. Entretanto, devido as várias camadas que são utilizadas na transmissão de potência, o envio e recebimento de dados para a nuvem pode acarretar em uma demora de atualização dos dados, na dificuldade de gerenciamento dos dispositivos *Internet of Things* (IoT) [Okay and Ozdemir 2016], devido as diferentes localizações, e a criação de sistemas mais complexos para processar dados de todas as camadas das *smart grid* ao mesmo tempo.

Logo, a distância entre a nuvem e as camadas das *smart grids* pode influenciar a atualização de dados, uma vez que a comunicação precisa ser feita pela internet. Assim, a *Fog* pode ser útil na realização de cálculos de fluxo de energia em tempo real, pois está localizada na borda de cada camada das *smart grids*. Contudo, os cálculos realizados na análise de fluxo de energia são complexos e os servidores da *Fog* não são tão poderosos quanto os servidores em nuvem. Assim, é necessário implementar os algoritmos de fluxo de energia considerando os dispositivos de transmissão (barramentos de geração e barramentos de carga) que o servidor na *Fog* está analisando.

Dependendo do número de geradores ou da carga consumida na rede elétrica, é aconselhável usar diferentes algoritmos de fluxo de carga, mas na literatura, não há indicação de qual técnica é mais apropriada para cada situação.

Este artigo, propõe o uso da arquitetura *Fog* como forma de reduzir a distância lógica entre a central de distribuição e os consumidores. Os dispositivos **IoT** na borda da rede têm mais eficácia e são menos custosos para lidar com as informações de fluxo de energia. Além disso, foi realizada a Avaliação de Desempenho dos métodos de Gauss-Seidel e Newton-Raphson considerando aspectos matemáticos como tolerância e aspectos físicos como quantidade de barramentos de carga e barramentos de geração, verificando a influência que estes têm nas variáveis de resposta, tensão e tempo de convergência. Esta avaliação de desempenho é usada para selecionar algoritmos de fluxo de energia apropriados. A avaliação de desempenho é uma técnica muito importante para o setor acadêmico e industrial, porque permite resultados confiáveis, economizando dinheiro e tempo.

Os resultados mostraram o *Newton-Raphson* como a melhor opção na maioria dos casos dos cenários analisados. Além disso, a maior contribuição deste artigo é fornecer uma *smart grid* baseada na *Fog* e, assim, criar redes elétricas inteligentes que são capazes de reagir rapidamente ao meio.

O restante deste artigo é organizado da seguinte forma: a Seção 2 apresenta os trabalhos relacionados e a relação dos mesmos com o trabalho proposto indicando o ponto onde a presente contribuição se encaixa, Seção 3 realiza um delineamento de como funcionam os algoritmos avaliados e na seção 4 como foi desenvolvida a avaliação, na Seção 5 são feitas as análises referentes aos resultados obtidos e na Seção 6 são apresentadas as conclusões bem como os trabalhos futuros recomendados.

2. Trabalhos relacionados

Sendo a próxima geração de rede elétrica, a *smart grid* usa distribuição de energia e informações para criar uma rede de distribuição de energia distribuída inteligente. Neste contexto, foi proposto a *Fog* como forma de melhorar o processamento e otimizar o fluxo de energia. Apesar da existência de vários trabalhos que exploram a infraestrutura *smart grid*, nenhum deles propôs o uso do paradigma *Fog* como forma de reduzir a distância lógica entre a central de distribuição e os consumidores.

[Singh and Bala 2016] fazem um estudo comparativo entre os algoritmos *Gauss-Seidel* e *Newton-Raphson* dentro da análise de fluxo de potência e considerando o sistema de barras IEEE 30, IEEE57 e IEEE 118, para determinar em qual momento um desses algoritmos pode ser melhor empregado. Foi verificado que utilizando *Gauss-Seidel* o número de iterações aumenta proporcionalmente de acordo com o número de barras enquanto o método *Newton-Raphson* permaneceu praticamente constante, mesmo com a variação do número de barras. Contudo, foi observado que para um sistema com um pequeno número de barras o método *Gauss-Seidel* é mais adequado.

[Rashmi 2016], [Vijayvargia et al. 2016], [Dharamjit 2012] and [Chatterjee and Mandal 2017] desenvolveram programas em MATLAB para calcular a tensão, o ângulo, a potência ativa e reativa de cada barramento dos sistemas IEEE9, 14, 30 e 57. Foram comparados os algoritmos *Gauss-Seidel*, *Newton-Raphson* e *Fast-Decoupled* e tiveram como objetivo verificar as mudanças da tolerância e do número de iterações. Como resultado foi verificado que o método *Newton-Raphson* converge mais rápido.

[Bokka 2010] desenvolveu uma técnica na análise do fluxo de carga que alterna entre os algoritmos tradicionais *Gauss-Seidel* e *Newton-Raphson* com o objetivo de diminuir o tempo de execução da convergência. Para testar esse novo algoritmo, [Bokka 2010] utilizou o sistema de barra IEEE 39 e comparou os resultados com os algoritmos tradicionais exe-

cutados isoladamente. Através dos resultados obtidos dessa comparação foi concluído que o método proposto foi mais rápido em tempo de convergência que os métodos tradicionais. Para pequenos sistemas o método *Gauss-Seidel* utilizou um menor número de iterações, mas para grandes sistemas o tempo de convergência é maior.

[Sedghi and Aliakbar-Golkar 2012] consideraram uma nova abordagem para os métodos de análise de fluxo de carga, onde relatam problemas nos algoritmos tradicionais devido a matriz de admitância ficar muito esparsa. Como consequência, foi proposto a utilização de uma rede neural difusa que é treinada pelos algoritmos tradicionais, como resultado o método teve maior confiabilidade de convergência e os neurônios difusos conseguiram superar as incertezas das entradas.

3. Smart grids e Análise do Fluxo de Potência

As *smart grids* foram desenvolvidas para economizar energia, reduzir custos de transmissão e aumentar a confiabilidade das redes elétricas. A melhor forma de suprir essas características é através do controle do consumo e da geração de energia [Seo et al. 2011]. Logo, é preciso coletar dados e interagir com todas as camadas da topologia das redes elétricas, obtendo comunicação em tempo real desde dos consumidores até as geradoras de energia, figura 1.

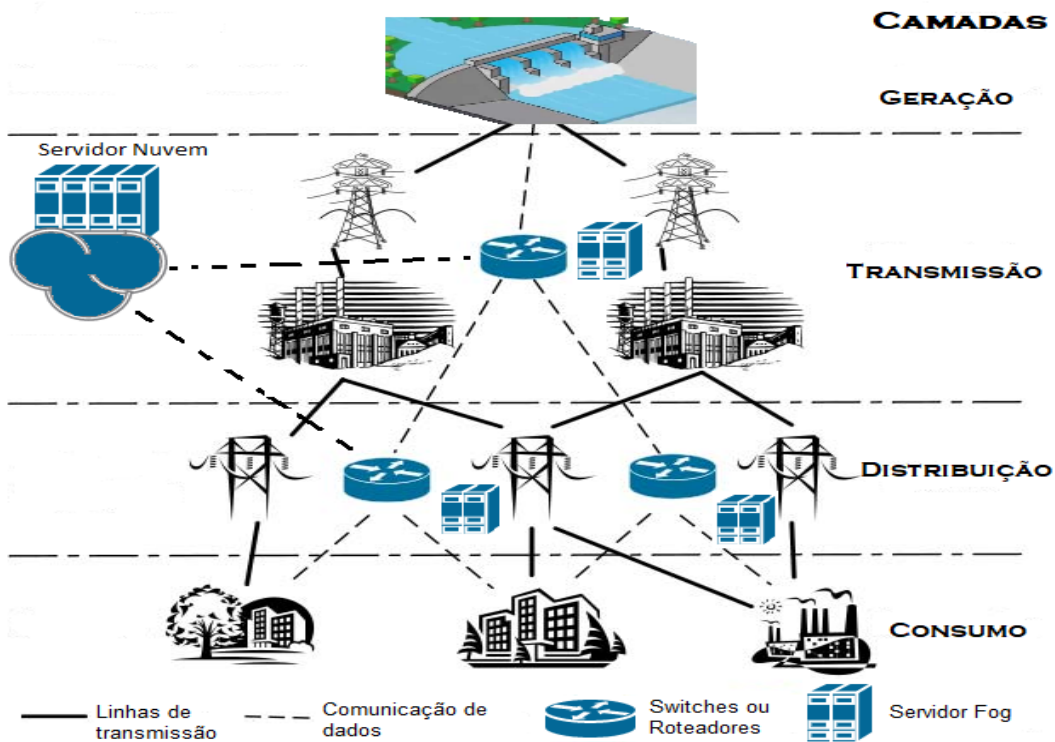


Figura 1. Topologia *smart grid* auxiliada por *Fog*

Fonte: Adaptado de [Seo et al. 2011]

A necessidade de comunicação em tempo real pode ser melhor suprida com o uso de servidores na borda da rede, que diminuem a latência melhorando o tempo de resposta das aplicações [Krishnan et al. 2015]. Assim, a utilização da *Fog* nas *smart grids* tornou-se fundamental para alcançar os benefícios que uma rede elétrica inteligente pode fornecer. No entanto, os servidores de borda não possuem a mesma capacidade de servidores localizados na nuvem, os algoritmos criados para a *Fog* precisam ser projetados considerando a economia de processamento, memória e armazenamento. Contudo, no universo das redes de transmissão, os métodos para projetar a capacidade de demanda e produção de energia são

complexos e necessitam de poder computacional para serem executados. Esses métodos, são estudados principalmente pela análise de fluxo de potência.

A análise de fluxo de potência consiste na determinação do estado da rede (magnitude das tensões, ângulos de fase), distribuição do fluxo e injeção de potência ativa e reativa em todas as barras dos sistema, essa análise permite o planejamento do crescimento e o monitoramento das melhores condições de operação das redes elétricas [Singh and Bala 2016].

Para obter essas informações são realizados cálculos através de um conjunto de equações e inequações algébricas, essas equações têm o objetivo de encontrar quatro variáveis V , θ , P e Q , onde V representa a magnitude da tensão, θ o ângulo da tensão, P a injeção de potência líquida e Q a injeção de potência reativa, mas para isso é feita uma modelagem estática do sistema, onde as variações das grandezas no tempo são lentas de modo que o efeito transitório desses sistemas podem ser desconsiderados. As equações básicas do fluxo de potência são obtidas impondo a conservação das potências ativas e reativas, onde a potência líquida injetada deve ser igual a soma das potências que fluem pelas barras da rede [Monticelli 1983].

$$P_k = \sum_{m \in \Omega_k} P_{km}(V_k, V_m, \theta_k, \theta_m) \quad (1)$$

$$Q_k + Q_k^{sh}(V_k) = \sum_{m \in \Omega_k} Q_{km}(V_k, V_m, \theta_k, \theta_m) \quad (2)$$

onde $k = 1, \dots, NB$, sendo NB o número de barras na rede. Ω_k representa conjunto de barras vizinhas a k . V_k, V_m as magnitudes das tensões das barras do ramo k - m . θ_k, θ_m os ângulos das tensões das barras do ramo k - m . P_{km} fluxo da potência ativa no ramo k - m . Q_{km} fluxo da potência reativa no ramo k - m . Q_k^{sh} componente de injeção de potência reativa devida ao elemento *shunt* da barra k ($Q_k^{sh} = b_k^{sh} V_k^2$), sendo b_k^{sh} a susceptância *shunt* ligada à barra k).

As variáveis representam as barras que compõem o sistema que é dividido em três tipo de barras: barra geradora (onde são dadas as variáveis P e V), barra de carga (onde são dadas as variáveis P e Q) e barra de referência (onde são dadas as variáveis V e θ), logo para cada barra é necessário encontrar as variáveis que faltam, utilizando os valores da barra anterior. Como no problema do fluxo de carga a variável θ é indeterminada é necessário a adoção de uma referência angular que é escolhida arbitrariamente na barra de referência [Saadat 2002]. Entretanto, ainda é necessário considerar o meio onde a potência é transmitida, para isso é preciso incluir nas fórmulas a resistência e aceitação da energia no material usado no transporte. Esse tipo de consideração é realizada através da impedância e admitância das linhas de transmissão [Saadat 2002].

Logo, para se calcular todas as variáveis envolvidas no projeto do fluxo de carga é necessário um modelo no qual o sistema de equações possa ser representado. O modelo mais utilizado é o matricial, onde através da matriz de admitância e do vetor de tensões é possível calcular o vetor de injeções de corrente para todas as barras.

$$I = Y.E \quad (3)$$

onde I é o vetor das injeções de corrente, cujos elementos são I_k , $k=1, \dots, NB$. Y é matriz de admitância. E o vetor de tensões cujos elementos são E_k , $k=1, \dots, NB$, onde $E_k = V_k e^{j\theta_k}$

Assim, sabendo arbitrariamente a variável θ da barra de referência e a corrente I injetada no sistema, pode-se encontrar a magnitude da tensão V e o ângulo θ de todas as barras. Com isso, é possível utilizar técnicas de resolução de sistemas de equações para determinar os valores de todas as variáveis das barras do sistema considerando as perdas do meio através da matriz de admitância e das fórmulas do fluxo de potência. As técnicas que são estudadas nesse artigo são *Gauss-Seidel* e *Newton-Raphson*

3.1. Método Gauss-Seidel

Gauss-Seidel é um método de resolução de sistema de equações que segue a forma $Ax = b$, onde A é uma matriz de tamanho $n \times n$ e x e b matrizes de tamanho $n \times 1$ como

$$\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

resolvendo a multiplicação de matriz usando x como referência

$$x_i = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j \right) \quad i = 1, \dots, n, \quad (4)$$

onde considerando as iterações $(m + 1)$ e utilizando os valores mais recentes de x tem-se o modelo *Gauss-Seidel*

$$x_i^{(m+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(m+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(m)} \right) \quad i = 1, \dots, n, \quad (5)$$

Feito a equivalência com o modelo de matriz de admitância $I = Y.E$ é encontrada a fórmula de *Gauss-Seidel* aplicada ao problema do fluxo de potência

$$E_k^{(m+1)} = \frac{1}{Y_{kk}} \left(\frac{S_k}{E_k^{(m)}} - \sum_{n=1}^{k-1} Y_{kn} E_n^{m+1} - \sum_{n=k+1}^{NB} Y_{kn} E_n^{(m)} \right) \quad (6)$$

sendo NB o número total de barras.

O algoritmo de *Gauss-Seidel* começa com a criação da matriz de admitância e depois a escolha da tolerância e dos valores iniciais para as variáveis da barra de referência. Nesse trabalho foi utilizado 1 para a tensão e 0 para o ângulo. Os valores iniciais são utilizados na equação (6) e esse processo só termina quando a diferença entre os novos valores e valores antigos forem menores que a tolerância especificada. Esse processo é executado para cada barra que existe no sistema até encontrar todas as variáveis, sendo que o valor achado anteriormente é utilizado para achar o próximo.

3.2. Método Newton-Raphson

Considerando, conforme definição, que são dados P e Q para as barras de carga, P e V para as barras geradoras e V e θ para a barra de referência é necessário calcular V e θ para as barras de carga, θ e Q para as barras geradoras e P e Q para a barra de referência. Uma vez resolvido esse problema, serão conhecidos o estado (V, θ) para todas as barras da rede, o que torna possível o cálculo das variáveis restantes [Monticelli 1983]. Sendo NPQ o número das barras de carga e NPV o número das barras geradoras, o problema do fluxo de carga pode ser decomposto em 2 (dois) subsistemas de equações.

O primeiro subsistema com $2NPQ + NPV$ equações e com número iguais de incógnitas, onde dados P e Q para barras de carga e P e V para as barras geradoras é preciso calcular V e Q para as barras de carga e θ para as barras geradoras, ou seja

$$P_k^{esp} - V_k \sum_{m \in \Omega_k} V_m (G_{km} \cos \theta_{km} + B_{km} \text{sen} \theta_{km}) = 0 \quad (7)$$

onde P_k^{esp} é a geração líquida de potência dada nas barras de carga e barras geradoras e V_k a tensão nas barras de origem do cálculo e V_m a tensão nas suas barras adjacentes

$$Q_k^{esp} - V_k \sum_{m \in \Omega_k} V_m (G_{km} \text{sen} \theta_{km} - B_{km} \cos \theta_{km}) = 0 \quad (8)$$

onde Q_k^{esp} é a injeção líquida de potência reativa dada nas barras de carga.

Após resolver o primeiro subsistema, serão conhecidos todos os valores V e θ de todas as barras e será necessário calcular P e Q para a barra de referência e Q para as barras geradoras, o que é um problema com NPV + 2 equações com o mesmo número de incógnitas, no qual todas as incógnitas vão aparecer de forma explícita, o que torna o processo de resolução trivial e podem ser encontradas através das equações 9 e 10.

$$P_k = V_k \sum_{m \in \Omega_k} V_m (G_{km} \cos \theta_{km} + B_{km} \text{sen} \theta_{km}) \quad (9)$$

$$Q_k^{esp} = V_k \sum_{m \in \Omega_k} V_m (G_{km} \text{sen} \theta_{km} - B_{km} \cos \theta_{km}) \quad (10)$$

Como o segundo subsistema de equações é facilmente calculado quando o primeiro é resolvido, o principal problema está resolução do primeiro subsistema, que precisa agrupar as incógnitas conforme equação 11.

$$x = \begin{pmatrix} \theta \\ \mathbf{V} \end{pmatrix} \quad (11)$$

onde θ é o vetor de ângulos de tensão das barras de carga e das barras geradoras com tamanho NPV + NPQ e \mathbf{V} é o vetor de tensões das barras de carga com tamanho NPQ.

As equações 7 e 8 podem ser reescritas em função de V e θ e ficam representadas respectivamente como 12 e 13

$$\Delta P_k = P_k^{esp} - P_k(V, \theta) = 0 \quad (12)$$

$$\Delta Q_k = Q_k^{esp} - Q_k(V, \theta) = 0 \quad (13)$$

Considerando a nova representação pode-se afirmar que

$$g(x) = \begin{pmatrix} \Delta P \\ \Delta Q \end{pmatrix} \quad (14)$$

e que

$$g(x) = 0 \quad (15)$$

Logo, é possível linearizar a função $g(x)$ em torno do ponto $(x, g(x))$ por meio da série de Taylor

$$g(x + \Delta x) \cong g(x) + J(x)\Delta x \quad (16)$$

onde $J(x)$ é a matriz Jacobiana dada pelas derivadas das equações de $g(x)$ que são definidas em 14, logo

$$J(x) = \begin{pmatrix} \frac{\delta(\Delta P)}{\delta\theta} & \frac{\delta(\Delta P)}{\delta V} \\ \frac{\delta(\Delta Q)}{\delta\theta} & \frac{\delta(\Delta Q)}{\delta V} \end{pmatrix} \quad (17)$$

Podendo encontrar Δx tal que

$$g(x) + J(x)\Delta x = 0 \quad (18)$$

sendo

$$\Delta x = \frac{-g(x)}{J(x)} \quad (19)$$

Concluindo que

$$\begin{pmatrix} \Delta P \\ \Delta Q \end{pmatrix} = \begin{pmatrix} \frac{\delta(\Delta P)}{\delta\theta} & \frac{\delta(\Delta P)}{\delta V} \\ \frac{\delta(\Delta Q)}{\delta\theta} & \frac{\delta(\Delta Q)}{\delta V} \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta V \end{pmatrix} \quad (20)$$

O algoritmo *Newton-Raphson* começa com a escolha das tensões, ângulos de tensões da barra de referência e da tolerância que será adotada, em seguida calcula-se $P(V, \theta)$ para as barras de carga e barras geradoras e $Q(V, \theta)$ para as barras de carga e verificando se a diferença entre o valor encontrado e o valor anterior é menor que a tolerância especificada. Sendo menor, o processo convergiu e a melhor solução para (V, θ) foi encontrada, caso contrário calcula-se a matriz Jacobiana e atualizam-se os valores achados. Esse processo se repete até que seja encontrado $P(V, \theta)$ e $Q(V, \theta)$ que diminuído do seu valor anterior seja menor que a tolerância.

3.3. Utilização da Fog na Análise do Fluxo de Potência

Um dos benefícios da análise do fluxo de potência é poder realizar o monitoramento da rede de distribuição, podendo prever as sobrecargas no sistema e ajudar na alteração da topologia de distribuição ajudando a incluir tanto barras geradoras como barra de carga [Singh and Bala 2016].

Atualmente, essa análise é estática e necessita de grande intervenção humana, onde as geradoras e distribuidoras de energia precisam estar em constante comunicação para manter o equilíbrio da rede. Com o surgimento das *SmartCities* e *smart grids*, a tendência da vida moderna é automatizar tarefas e buscar a utilização de recursos renováveis.

Para isso, esse trabalho propôs uma arquitetura em cima da topologia das *smart grids* atuais que utilizam a *Fog* para monitorar e interagir com as redes de distribuição. Essa arquitetura, na prática utiliza os dados obtidos através da avaliação desempenho para sugerir a quantidade de barras que cada equipamento localizado na borda das *smart grids* podem monitorar e interagir. Essa limitação de quantidade de barras se deve ao fato dos algoritmos atuais não comportarem um número muito grande de barras na sua execução, o que diante da proposta da *Fog* é uma boa situação, uma vez que para enviar os dados para a nuvem as *smart grids* precisam de um *gateway* que ficam em um raio de no máximo 15 Km, o que faz uma *Fog* para cada barra de carga no raio de 15 km. Dentro dessa distância pode ser considerada a existência de 1 barra geradora para 15 barras de carga, considerando as subestações como barras geradoras e os disjuntores como barra de carga.

Como o número de barras de carga pode variar conforme o crescimento das cidades e considerando que os equipamentos na borda da rede de dados é limitado, este trabalho apresenta um algoritmo que mistura os 2 algoritmos estudados e que pode ser usado na *Fog* para calcular a voltagem das barras de carga e através de simulações detectar possíveis sobrecargas no sistema.

```
1 dadosconsumo ← recebedadossmartgrid();
2 if host.processamento > 50% then
3   if dadosconsumo.quantbarras < 10 then
4     | valorbarras ← gaussseidel (dadosconsumo)
5   else
6     | valorbarras ← newtonraphson (dadosconsumo)
7   end
8 else
9   if dadosconsumo.quantbarras < 15 then
10    | valorbarras ← gaussseidel (dadosconsumo)
11   else
12    | valorbarras ← newtonraphson (dadosconsumo)
13   end
14 end
```

Algoritmo 1: Fluxo de Carga - Utilização da *Fog* para Fluxo de Carga

4. Avaliação de desempenho

O desempenho é um critério fundamental na concepção, aquisição e utilização de sistemas informatizados. O objetivo dos engenheiros, cientistas, analistas e usuários de sistemas informatizados é obter o maior desempenho para um determinado custo [Jain 1991]. Assim, dentro dos sistemas de distribuição de energia que estão se tornando cada vez mais informatizados é necessário realizar uma avaliação de desempenho para entender os principais fatores que influenciam nas variações de voltagem e tensões e como os sistemas informatizados podem ajudar a evitar esses problemas com o menor custo possível.

4.1. Planejamento dos experimentos

Para avaliar o desempenho dos algoritmos de fluxo de carga nas redes de distribuição, foi utilizado o programa MATLAB para realizar experimentos simulados no sistema de 39 barras adaptado do IEEE 39, Figura 2. Foram considerados 4 (quatro) fatores com 2 (dois)

níveis cada, onde os fatores escolhidos foram: algoritmos, tolerância, barras de carga e barras geradoras.

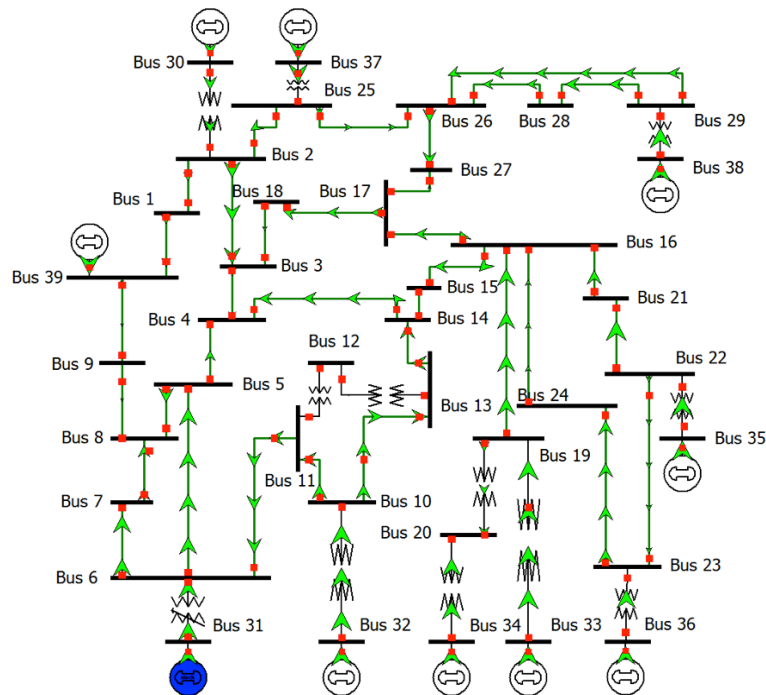


Figura 2. Modelo IEEE 39

Fonte: <http://icseg.iti.illinois.edu/ieee-39-bus-system/>

Foi utilizado o modelo fatorial 2^k , apresentado por [Jain 1991], no qual foram definidos os níveis como na tabela 1

Tabela 1. Fatores e níveis

Fator	Níveis	
Algoritmo	<i>Gauss-Seidel</i>	<i>Newton-Raphson</i>
Tolerância	0,0001	0,1
Barras de Carga	1	10
Barras Geradoras	1	29

O objetivo desses experimentos é extrair o quanto os fatores relacionados influenciam nas variáveis de resposta, que são a acurácia da voltagem e o tempo de convergência. Para isso, combinações foram feitas entre os níveis especificados conforme orienta [Jain 1991], onde um modelo com arranjo 2^4 é realizado. Os fatores e níveis do projeto de experimentos foram definidos através de valores mais prováveis e utilizados na literatura, permitindo avaliar a completude do efeito que essas variáveis provocam ao ambiente.

Nesse experimento as barras geradoras do sistema irão transmitir voltagem nominal 1 pu. Combinações onde só existem 11 barras, foram executadas considerando as barras mais próximas e assim pode-se desconsiderar a influência que a distância entre as barras poderia exercer sobre os resultados, pois quanto maior a distância maior a resistência que a potência irá receber. Além disso, o objetivo desse trabalho é oferecer um algoritmo que possa ser usado na *Fog* o que torna possível considerar as barras mais próximas, uma vez que a *Fog* só irá dividir virtualmente o sistema de transmissão em subsistemas. Para garantir a dinâmica

dos experimentos, o consumo de potência nas barras de cargas foram gerados aleatoriamente para cada experimento.

Foram executadas 10 replicações para cada experimento que tiveram pouca variação, assim foi considerado que a quantidade de replicações foi suficiente para a análise dos dados. O intervalo de confiança de 95% obtidos através da tabela *T-Student* foi utilizado para cada experimento. Assim pode ser adicionado aos gráficos de barras o intervalo da dispersão dos dados com probabilidade igual ao índice adotado. A partir dos valores obtidos para cada variável de resposta foram calculadas as suas variações totais, além da influência de cada um dos fatores em seus valores, como descrito em [Jain 1991].

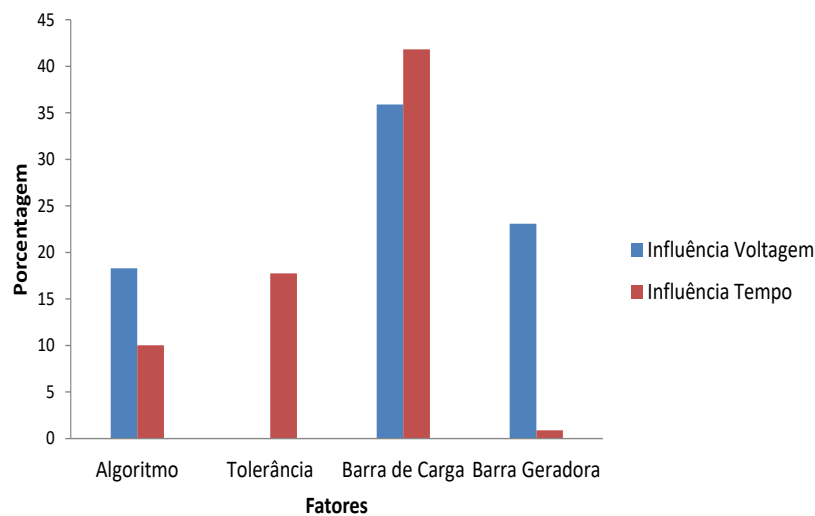


Figura 3. Influência dos fatores

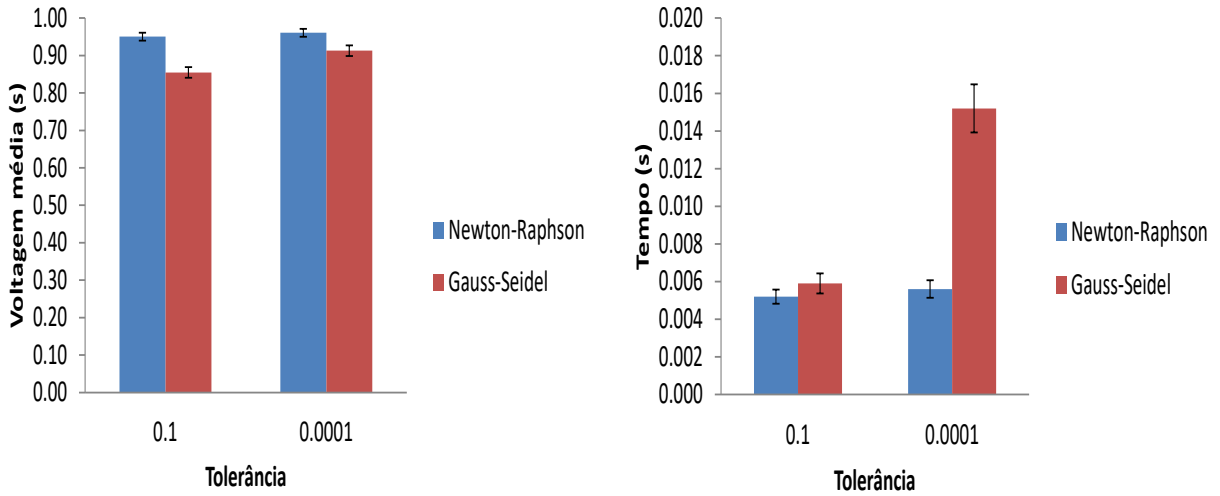
Como pode ser visualizado na Figura 3, os fatores algoritmo (18,29%), barra de carga (35,90%) e barra geradora (23,07%) são os fatores com maior porcentagem quando considerado a influência na tensão. Isso ocorre porque quanto maior a quantidade de barras pior será o desempenho do algoritmo *Gauss-Seidel*. Foi possível perceber também que a quantidade de barras geradoras influenciam na tensão quando estão em 30% do total de barras de carga considerando que o algoritmo usado seja *Newton-Raphson* para quantidades grandes barras ou *Gauss-Seidel* para sistemas com menos barras.

Além disso, a Figura 3 apresenta a influência dos fatores no tempo de convergência dos algoritmos. Essa influência se dá mais forte nos fatores barra de carga (41,83%) e tolerância (17,75%). Esse comportamento ocorre porque quanto maior a quantidade de barras de carga maior será a matriz de admitância (*Gauss-Seidel*) ou jacobiana (*Newton-Raphson*) e o processamento de matriz grandes consomem maior tempo computacional. Além disso, o fator tolerância também contribui para o tempo de processamento uma vez que quanto maior a tolerância menos cálculos precisam ser efetuados.

4.2. Discussão e Análise dos Resultados

Nos experimentos realizados com a maior quantidade de barras, 29 barras de carga e 10 barras de geração, a média da tensão nominal do algoritmo *Newton-Raphson* ficou dentro dos 95% sugeridos pelo módulo 8 da Agência Nacional de Energia Elétrica (ANEEL), mesmo para uma tolerância com menor precisão, contudo o algoritmo *Gauss-Seidel* não conseguiu manter a tensão dentro dessa especificação para nenhuma tolerância testada, no entanto

para a menor tolerância o valor ficou próximo do limite de aceitação, mas o tempo de processamento cresceu mais da metade do tempo que o método *Newton-Raphson*, pois a tolerância exigia maior precisão, como pode se visto na figura 4(a) e figura 4(b).

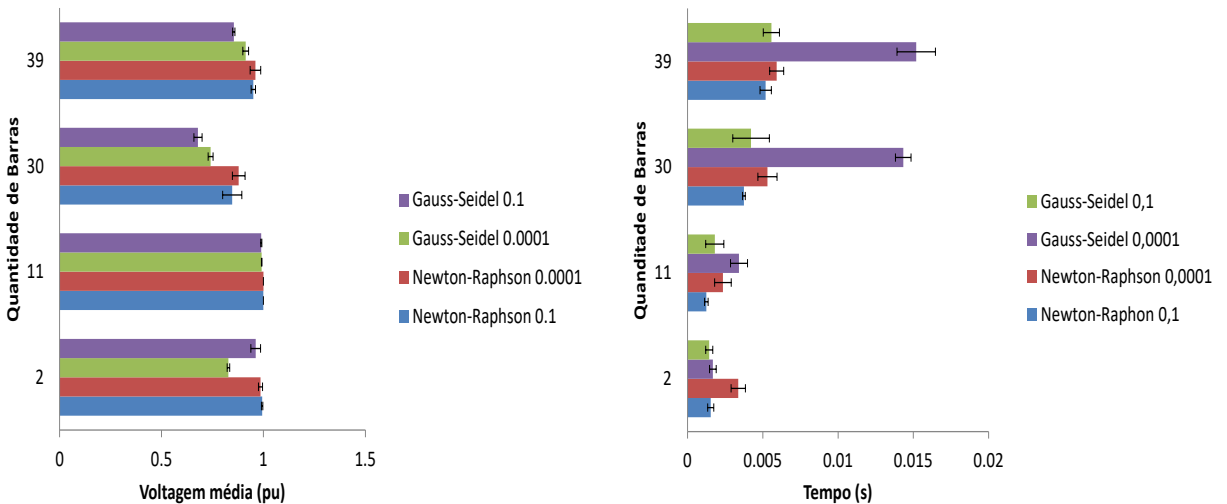


(a) 29 barras de carga e 10 barras geradoras - Voltagem (b) 29 barras de carga e 10 barras geradoras - Tempo

Figura 4. Comparação algoritmos

É importante destacar que a voltagem não caiu muito, devido ao fato de que a quantidade de barras geradoras está próximo de 30% do número de barras de carga do sistema, considerando que todas as barras geradoras entregam voltagem nominal 1 pu.

Para os casos onde existem muitas barras de carga, mas poucas barras geradoras a situação não se repete, e nenhum algoritmo consegue alcançar os 95% de qualidade da voltagem, o valor máximo alcançado foi 0,8785 de voltagem no algoritmo *Newton-Raphson* para uma tolerância de 0,0001.



(a) Acurácia da voltagem por quantidade de barras (b) Tempo de convergência por quantidade de barras

Figura 5. Efeitos causados pelas barras

O tipo de relação entre voltagem e quantidade de barras pode ser melhor observado na figura 5(a), onde foi realizada uma comparação entre a quantidade de barras e a acurácia

da voltagem. Quando é considerado a menor quantidade de barras, 2 (duas) sendo uma de carga e outra geradora, o único fator que implica na voltagem é o consumo que sai das barras de carga, a maioria dos testes feitos ficaram dentro da margem esperada. Quando é considerado 11 barras, apenas uma é de carga, sendo 10 barras geradoras o que faz a voltagem independente de qualquer outro fator ficar muito acima dos 95%. Contudo quando se aumenta para 30 barras, apenas uma é geradora sendo todas as outras barras de carga, nesse caso a voltagem tende a cair principalmente no algoritmo *Gauss-Seidel*. Quando a quantidade de barras geradoras volta a aumentar o sistema com 29 barras de carga e 10 barras geradoras a voltagem cresce tornando-se aceitável para o algoritmo *Newton-Raphson*.

A figura 5(b) representa o tempo de convergência por quantidade de barras, sendo fácil perceber que quando a quantidade de barras e a precisão da tolerância aumenta, o algoritmo *Gauss-Seidel* tende a demorar mais para convergir. O algoritmo *Newton-Raphson* teve poucas variações mesmo com o aumento da quantidade de barras, sendo que mostrou a característica de diminuir o tempo de convergência quando a quantidade de barras geradoras aumenta, esse comportamento pode ser explicado devido ao fato que para as barras geradoras na primeira, e mais complexa, etapa da resolução do algoritmo *Newton-Raphson* só precisar encontrar uma variável para as todas as equações, a θ conforme explicado na seção (3.2). Para as barras de carga é necessário encontrar 2 variáveis para todas as equações, a V e θ .

5. Conclusão

Este artigo apresenta o uso do paradigma da computação na *Fog* como uma extensão da computação na nuvem. Essa solução propõe a redução da distância entre as centrais de distribuição e os consumidores, proporcionando baixa latência e computação em tempo real. Assim, a *Fog* armazena e processa os dados para uma tomada de decisão, independentemente da comunicação com a nuvem.

Além disso, foi utilizado de um modelo estatístico para realizar uma avaliação de desempenho dos métodos *Gauss-Seidel* e *Newton-Raphson*. Ao analisar os resultados obtidos para as variáveis de resposta adotadas, voltagem e tempo de convergência, foi possível verificar o comportamento das redes de distribuição. Através deste comportamento, um algoritmo foi desenvolvido para configurar a *Fog* com o objetivo de monitorar as *smart grids* e atuar sobre seus componentes com o objetivo de manter o equilíbrio da distribuição de carga. Assim, os algoritmos de fluxo de carga são executados de acordo com suas características e com a capacidade dos servidores *Fog*, possibilitando respostas mais rápidas ao meio através da escolha do melhor algoritmo a ser utilizado no momento. Isso ocorre porque a implementação do método *Gauss-Seidel* além de mais simples, consome menos memória e ele só é chamado em momentos em que seus benefícios podem ser utilizados, ou seja, baixa quantidade de barras para serem analisadas.

Com relação ao método *Newton-Raphson*, embora possa ser utilizado na grande maioria dos casos da análise do fluxo de carga, sua implementação consome muita memória e processamento devido a utilização da matriz jacobiana, sendo assim, em um ambiente *Fog*, onde os servidores podem ter baixa capacidade de memória e processamento, a utilização dessa técnica só deve ser requerida como último recurso.

6. Agradecimentos

Os autores agradecem à FAPESB, FAPEMIG, CAPES, CNPq e à parceria MCTIC-UFBA - (Ministério da Ciência, Tecnologia, Inovação e Comunicação do Brasil) pelo apoio financeiro através do PROPCI/PROPG/PROPESQ/UFBA-004/2016.

Referências

- Bokka, N. (2010). Comparison of power flow algorithms for inclusion in on-line power systems operation tools.
- Chatterjee, S. and Mandal, S. (2017). A novel comparison of gauss-seidel and newton-raphson methods for load flow analysis. In *Power and Embedded Drive Control (IC-PEDC), 2017 International Conference on*, pages 1–7. IEEE.
- Dharamjit, D. (2012). Load flow analysis on ieee 30 bus system. *International Journal of Scientific and Research Publications*.
- Gungor, V. C., Sahin, D., Kocak, T., Ergut, S., Buccella, C., Cecati, C., and Hancke, G. P. (2011). Smart grid technologies: Communication technologies and standards. *IEEE transactions on Industrial informatics*, 7(4):529–539.
- Jain, R. (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. Wiley Professional Computing. John Wiley & Sons Chichester.
- Jdeed, M. (2016). Comparison of the smart grid simulation tools rapsim and gridlab-d.
- Krishnan, Y. N., Bhagwat, C. N., and Utpat, A. P. (2015). Fog computing—network based cloud computing. In *Electronics and Communication Systems (ICECS), 2015 2nd International Conference on*, pages 250–251. IEEE.
- Monticelli, A. J. (1983). *Fluxo de Carga em Redes de Energia Elétrica*. Centro de Pesquisas de Energia Elétrica.
- Okay, F. Y. and Ozdemir, S. (2016). A fog computing based smart grid model. In *Networks, Computers and Communications (ISNCC), 2016 International Symposium on*, pages 1–6. IEEE.
- Rashmi, Amit Verma, B. S. R. A. V. B. S. (2016). Comparative analysis of load flow methods for different network system. *International Journal of Computing and Technology*.
- Saadat, H. (2002). *Power Systems Analysis*. McGraw-Hill Primis Custom.
- Sedghi, M. and Aliakbar-Golkar, M. (2012). Analysis and comparison of load flow methods for distribution networks considering distributed generation. *International Journal of Smart Electrical Engineering*, 1(1).
- Seo, D., Lee, H., and Perrig, A. (2011). Secure and efficient capability-based power management in the smart grid. In *Parallel and Distributed Processing with Applications Workshops (ISPAW), 2011 Ninth IEEE International Symposium on*, pages 119–126. IEEE.
- Singh, J. and Bala, R. (2016). A case study: Comparison of newton-raphson and gauss-seidel load flow solution techniques in distributed transmission and generation electricity networks. *International Journal of Advance Electrical and Electronics Engineering*, 5(1):2278–8948.
- Vijayvargia, A., Jain, S., Meena, S., Gupta, V., and Lalwanib, M. (2016). Comparison between different load flow methodologies by analyzing various bus systems. *International Journal of Electrical Engineering*.