

## Algoritmo distribuído para autorrecuperação de *smart grids* utilizando um sistema multiagente reativo

Italo R. Campos<sup>1</sup>, Filipe Saraiva<sup>1</sup>

<sup>1</sup>Instituto de Ciências Exatas e Naturais – Universidade Federal do Pará (UFPA)  
Caixa Postal 479 – 66.075-110 – Belém – PA – Brazil

italo.ramon.campos@gmail.com, saraiva@ufpa.br

**Abstract.** *This paper presents a distributed algorithm for smart grids self-healing in distribution level, implemented by a reactive multiagent system. Through the multiagent system, is possible to coordinate the switches of the power system, so that it adapts to the changes in the grid. To validate the proposed algorithm, is used a test model with 15 nodes, which one of them is a source. The results are obtained via computational simulation and shown in this paper.*

**Resumo.** *Este artigo apresenta um algoritmo distribuído para autorrecuperação de redes elétricas de distribuição do tipo smart grids, implementado por um sistema multiagente reativo. Por meio do sistema multiagente, é possível coordenar as chaves do sistema elétrico, de modo que ele se adapte às mudanças que ocorrem na rede. Para validar o algoritmo proposto, utiliza-se um modelo de testes com 15 nós, sendo um deles uma fonte. Os resultados da pesquisa são obtidos mediante simulação computacional e apresentados neste artigo.*

### 1. Introdução

Na área dos Sistemas Elétricos de Potência (SEP), *smart grid* é um conceito que recebe cada vez maior atenção, uma vez que traz importantes funcionalidades sobre os sistemas elétricos tradicionais, tais como gerenciamento remoto, comunicação de via dupla, segurança, controle em tempo real, acesso à informação em tempo real, inclusão em larga escala de fontes distribuídas de energia, entre outros [Saraiva 2015]. Esses aspectos dos *smart grids* configuram o setor elétrico como um relevante campo de estudo, especialmente quando se fala em técnicas da computação.

*Smart grid* é o termo utilizado para designar o emprego de tecnologias de informação e comunicação aos sistemas elétricos, compreendendo todos os seus subsistemas [Larik and Mustafa 2015]. O resultado dessa combinação é a entrega de uma gama de funcionalidades, como as descritas acima. Além disso, esses sistemas podem ser estratificados em termos de camadas, definidas em camada básica, de comunicação e de aplicação [Jia et al. 2011].

No setor de distribuição, elenca-se abordagens na esfera da computação que giram em torno de técnicas de inteligência artificial e computacional em alto nível. Mais especificamente, essas abordagens atuam sobre a camada de aplicação dos *smart grids*, através da qual objetivam desenvolver métodos para prover funcionalidades específicas ao nível de distribuição dos SEP. A subárea de distribuição é uma das mais importantes dos

SEP devido ser a conexão entre os consumidores e o setor de transmissão [Jia et al. 2011], além de serem expostos à grandes variações de demanda e de ambiente.

No que diz respeito ao nível de distribuição, os *smart grids* assumem um papel de referência para o desenvolvimento do futuro dos *grids*. A expansão dessa área figura como grande aliada do desenvolvimento social, uma vez que, de alguma forma, o desenvolvimento do setor elétrico está vinculado ao desenvolvimento da sociedade. Prover novas funcionalidades aos *grids* tradicionais significa aproximar o consumidor final das informações relativas à rede – como o consumo –, assim como melhorar os canais de comunicação entre as prestadoras de serviço e seus clientes. Além disso, os *smart grids* são apontados como “aquecedores” do mercado elétrico, tornando-o campo de interesse por organizações que atuam na área de compra e venda dinâmica de ativos, no caso, de energia elétrica [Siano 2014]; [Motamedi et al. 2012]; [Rahimi and Ipakchi 2010].

Uma das principais funcionalidades dos *smart grids* é a autorrecuperação, e existem muitos autores que realizam pesquisas para abordar o problema usando um vasto leque de métodos. De fato, muitos métodos computacionais podem ser aplicados para resolver problemas de *smart grids*, especialmente os métodos de aproximação, já que eles podem entregar uma solução aproximada em um tempo computacional satisfatório. Dessa forma, os métodos de inteligência artificial e computacional são largamente utilizados pela comunidade científica.

O trabalho de [Ferreira et al. 2013], por exemplo, modela o problema de autorrecuperação usando uma função objetivo de minimização, a qual é processada em um algoritmo genético. Em uma outra abordagem, [Mahdi and Genc 2019] utilizam um método de ilhamento de diferentes conjuntos de nós da rede elétrica para realizar a autorrecuperação. A escolha das ilhas é realizada por meio de diversos algoritmos, dentre eles o algoritmo de agrupamento *K-means* e um algoritmo de lógica *Fuzzy*. Por outro lado, utilizando abordagens distribuídas, os trabalhos de [Souza 2015], [Sharma et al. 2018] e [Wang et al. 2016] relatam como é possível atingir a autorrecuperação através da coordenação mútua entre agentes inteligentes no sistema.

No campo das abordagens distribuídas, encontra-se o uso da técnica de sistemas multiagente, que consiste na organização de múltiplos agentes inteligentes, envolvidos em um ambiente, os quais se coordenam para atingir seus objetivos individuais e/ou coletivos [Bellifemine et al. 2007]. [Saraiva 2015] utiliza sistemas multiagente para implementar e simular uma série de funcionalidades dos *smart grids*, a partir das quais é possível validar que sistemas multiagente são ferramentas interessantes para o uso na simulação e implementação de *smart grids*.

Neste artigo, é apresentado um algoritmo distribuído para realizar a autorrecuperação de *smart grids* no nível de distribuição. A metodologia deste trabalho inclui a utilização de um sistema multiagente para simular redes elétricas de distribuição, bem como as operações que elas realizam para que a autorrecuperação seja efetuada. Este trabalho é a validação e melhoria do algoritmo apresentado em uma publicação anterior [Campos and Saraiva 2018].

Por meio de simulação computacional, o trabalho objetiva validar o algoritmo de autorrecuperação, inserindo-o na literatura como uma alternativa para implementação dessa funcionalidade em *smart grids*. Para validá-lo, foi desenvolvido um modelo de

testes de rede elétrica de distribuição com 15 nós, utilizando como referência o modelo de testes da IEEE com 33 nós [Baran and Wu 1989]. Os resultados foram coletados mediante *logs* escritos pelos agentes do sistema e, posteriormente, analisados.

O artigo é estruturado da seguinte forma: a presente seção apresentou o contexto, o objetivo, a metodologia e breves conceitos relacionados à *smart grids* e autorrecuperação; a Seção 2 descreve o problema abordado pela pesquisa e esclarece detalhes sobre as restrições do sistema elétrico levadas em conta pela pesquisa; posteriormente, a Seção 3 detalha o algoritmo de autorrecuperação e o sistema multiagente modelado; a seguir a Seção 4 apresenta como se dá o processo de simulação, bem como os resultados obtidos, além das respectivas discussões; por fim, a Seção 5 finaliza o trabalho, mostrando as conclusões da pesquisa.

## 2. Descrição do problema

A autorrecuperação pode ser definida como o restabelecimento automático de energia elétrica às cargas que foram afetadas por uma interrupção no seu abastecimento [Campos and Saraiva 2018]. O termo “automático” está relacionado à realização ativa de operações pela própria rede elétrica, sem a intervenção de um operador humano.

Quando se fala em redes de distribuição, existe uma série de restrições que precisam ser obedecidas para que não haja problemas de operação. Uma dessas restrições é o *princípio da radialidade*, que diz que as estruturas de redes de distribuição não podem conter ciclos, uma vez que, se ocorridos, podem causar um colapso em grandes proporções em toda a estrutura do *grid*. Nesta etapa do trabalho considera-se apenas o princípio da radialidade durante a execução do algoritmo de autorrecuperação.

Em termos de definição, tem-se que o conjunto de linhas ativas de uma rede elétrica de distribuição pode ser descrito como um grafo  $G$ , onde todo passeio  $p$  em  $G$  é definido por

$$p = \{v_1, v_2, \dots, v_k, v_{k+1}\}, \forall v_i \neq v_{k+1}, \text{ tal que } k > 0. \quad (1)$$

É pertinente salientar que as linhas ativas de uma rede de distribuição compreendem no conjunto de todas as linhas que estão conduzindo corrente elétrica ao longo da rede em um dado momento, isto é, que possuem uma chave elétrica fechando o circuito. Uma linha que não esteja, em um determinado momento, fazendo esse trabalho, não é uma linha ativa.

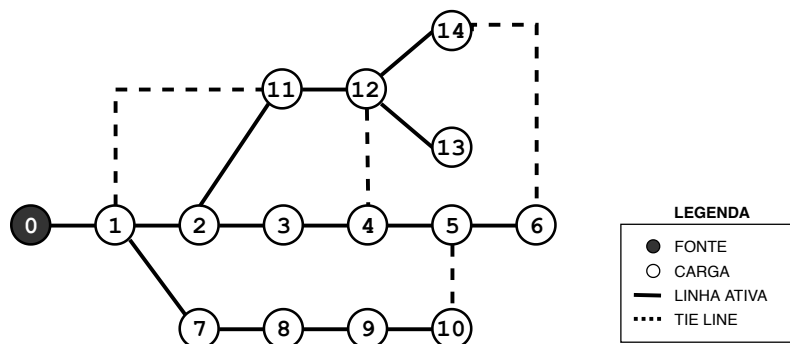


Figura 1. Topologia do modelo de testes de 15 nós.

Para realizar os estudos com o algoritmo desenvolvido por esta pesquisa, foi necessário utilizar um modelo de testes de rede elétrica de distribuição, para servir como suporte para as simulações e para as operações que o sistema multiagente realiza. Esta pesquisa, então, desenvolveu um modelo de testes com 15 nós, sendo um deles uma fonte, a partir da qual inicia o fluxo de corrente elétrica para os demais nós da rede. A topologia e os parâmetros utilizados no modelo foram aferidos com base no modelo de testes de 33 nós da IEEE, bem como nos valores de demanda, tensão, resistências das linhas, entre outros, definidos em [Baran and Wu 1989]. A Figura 1 traz a topologia do modelo de testes de 15 nós, ao passo que a Tabela 1 apresenta os os parâmetros do modelo de testes elaborado.

Br. No	Rc. Nd.	Sn. Nd.	Br. Parameters		Sn. Nd. Parameters	
			$r(\Omega)$	$x(\Omega)$	PL(W)	QL(var)
1	0	1	0.0922	0.047	100000	60000
2	1	2	0.493	0.2511	90000	40000
3	2	3	0.366	0.1864	120000	80000
4	3	4	0.3811	0.1941	60000	30000
5	4	5	0.819	0.707	60000	20000
6	5	6	0.1872	0.6188	200000	100000
7	1	7	0.164	0.1565	90000	40000
8	7	8	0.409	0.479	90000	40000
9	8	9	1.5042	1.3554	90000	40000
10	9	10	0.4095	0.4784	90000	50000
11	2	11	0.4512	0.3083	420000	200000
12	11	12	0.898	0.7091	420000	200000
13	12	13	0.896	0.7011	60000	25000
14	12	14	0.2842	0.1447	60000	25000
Tie lines						
15	1	11	1.5	1.5		
16	4	12	1	1		
17	5	10	0.5	0.5		
18	6	14	0.5	0.5		

**Tabela 1. Parâmetros de rede do modelo de teste de 15 nós.**

Observando a Figura 1, conta-se 14 linhas ativas e 4 *tie lines*. *Tie lines* (ou linhas de reserva), são linhas de distribuição presentes na rede com chave aberta, isto é, não conduzem corrente elétrica em determinado momento. O propósito dessas linhas é serem usadas como alternativas para religar nós que estejam afetados por uma eventual interrupção no abastecimento elétrico. Entende-se, portanto, que o objetivo do sistema multiagente é, depois de detectar uma falha no fornecimento de energia, reconfigurar a rede usando as *tie lines*, de forma que seja restabelecido o suprimento de energia elétrica aos nós afetados, obedecendo à restrição anteriormente descrita.

### 3. O sistema multiagente e o algoritmo de autorrecuperação

A presente abordagem utiliza-se de sistemas multiagente, que é uma técnica distribuída cujo objetivo é modelar um problema em termos de agentes independentes, de modo que eles operem em conjunto para alcançar objetivos específicos [Bellifemine et al. 2007].

Pelo fato de o problema abordado apresentar um “gatilho” – que é a detecção da falta de energia –, preferiu-se empregar um sistema multiagente do tipo reativo.

[Bellifemine et al. 2007] apontam sistemas multiagente desse tipo como mais apropriados para utilizar em casos onde se espera uma obtenção rápida de respostas. Para a implementação dos agentes, utilizou-se o *JADE Framework* [JADE 2017], que proporciona uma gama de ferramentas para comunicação de agentes, além de ser muito usado para abordagens com sistemas multiagente reativo.

Em termos de modelagem, foram desenvolvidos dois tipos de agentes para lidar com o problema em análise nessa pesquisa: o Agente de Carga (ou *Load Agent*) e o Agente de Chave (ou *Switch Agent*).

Os *Agentes de Carga* (AC) são os que possuem maior quantidade de atribuições no sistema. Dentre elas pode-se destacar:

- Verificação periódica dos sensores dos nós da rede elétrica, monitorando o ambiente e detectando eventuais faltas;
- Envio e manutenção de dados atualizados relacionados ao seu próprio estado e aos estados de agentes vizinhos, respectivamente;
- Conexão a novos AC, em tempo de execução, buscando atender ao dinamismo da rede elétrica;
- Eleição e candidatura à posição de agente ativo do conjunto de nós em falta;

Os *Agentes de Chave* (SW), por sua vez, são agentes com atribuições bem específicas. São elas:

- Conexão de um par de AC em tempo de execução;
- Execução de operações de abertura e fechamento de chaves elétricas, de acordo com as mensagens recebidas dos AC;

Para cada nó (ou carga) da rede elétrica, existe um AC alocado, ao passo que para cada linha, incluindo *tie lines*, existe um SW alocado. Portanto, no modelo utilizado são instanciados 15 AC, 14 SW para as linhas ativas e outros 4 SW para *tie lines*.

Há uma característica do sistema multiagente implementado importante de ser enfatizada. Para manter a abordagem descentralizada, os AC são instanciados sem conhecer o endereço dos demais agentes do sistema. Para conectá-los entre si, os SW são instanciados tendo como parâmetros os endereços de um par determinado de AC, mediando a conexão entre eles. Como consequência desse fato, os AC possuem uma visão limitada do sistema, conhecendo apenas os AC vizinhos a eles, com os quais mantêm comunicação. Essa característica foi nominada *visão mínima* do sistema.

### 3.1. O algoritmo distribuído

O algoritmo proposto por essa pesquisa possui quatro etapas básicas: isolamento, eleição de agente ativo, mapeamento e tomada de decisão. Cada uma dessas etapas é descrita nas próximas subseções. A partir deste ponto do trabalho, os termos “AC”, “nó” e “carga” serão usados indiscriminadamente, referindo-se a um determinado nó da rede elétrica que é controlado por um AC.

#### 3.1.1. Isolamento

Ao ser detectada a falta no sistema, os AC afetados por ela iniciam o processo de isolamento. O processo de isolamento nada mais é do que a atualização de dados mútua entre os AC que foram afetados pela falta, com o objetivo de identificar o local afetado.

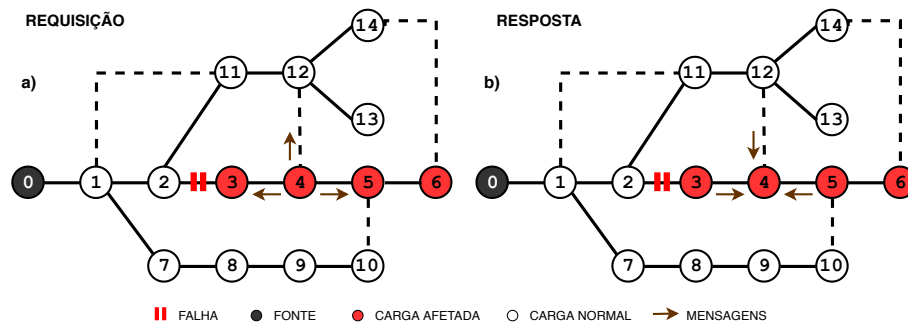


Figura 2. Passo de isolamento seguido pelo AC4.

A atualização dos nós é feita através da troca de mensagens. Os AC enviam mensagens a todos os seus vizinhos, buscando conhecer quais têm o fornecimento de energia normal e quais não. Essas mensagens seguem o protocolo requisição-resposta, isto é, o AC remetente faz uma requisição de atualização e o AC destinatário responde com uma mensagem, cujo conteúdo é um objeto com informações sobre o seu estado atual. Para uma melhor compreensão, será usado um exemplo com o modelo de testes, onde se tem uma falha na linha (2, 3). Nesse contexto, a área afetada é composta pelos nós 3, 4, 5 e 6. Concentrando-se apenas no nó 4 desse conjunto, teremos um comportamento como o descrito na Figura 2.

### 3.1.2. Eleição de agente ativo

Terminada a etapa de isolamento, é necessário escolher um dos AC para realizar a tomada de decisão em nome da área afetada. Esse processo se torna importante porque caso dois ou mais AC realizem a tomada de decisão, corre-se o risco de entrarem em conflito e haver decisões que violem a restrição de radialidade da rede. Sendo assim, os AC da área afetada participam de um processo de eleição, com o objetivo de escolher um deles para realizar o papel de agente ativo (AA). Ao fim do processo, os demais AC que não forem eleitos como AA, tornam-se agentes passivos (AP), aguardando as orientações do AA eleito e cooperando nos processos posteriores do algoritmo.

Somente AC externos podem se candidatar ao papel de AA. Um AC externo é um AC que tem pelo menos uma *tie line* que o liga a um outro AC que está com o abastecimento energético normal. Por consequência, AC que não sejam externos tornam-se automaticamente AP nesta etapa do algoritmo.

Em termos de sistema, a eleição é feita utilizando um espaço para publicação de serviços de agentes, disponibilizado pelo JADE na forma de *páginas amarelas*. As páginas amarelas são controladas por um agente específico da plataforma, denominado *Directory Facilitator* (DF), que é responsável por receber as mensagens de registro, cancelamento e consulta de serviços disponibilizados na plataforma. Os agentes candidatos, então, se registram no DF e aguardam um *limiar de tempo* para que todos os outros agentes candidatos tenham tempo de se registrar. Esse processo é necessário devido a visão mínima dos AC.

Para esta pesquisa, foi definido um limiar no valor de 370 ms, através do método de tentativa e erro. Esse valor foi aferido depois de uma série de testes com diversos

modelos de rede no ambiente de simulação e nunca foi extrapolado em nenhuma das simulações. Entretanto, sabe-se que em situações mais próximas à realidade, esse tempo pode variar, especialmente devido às variações da rede de comunicação.

Depois do período de espera, cada AC candidato consulta novamente o DF da plataforma em busca dos dados dos possíveis outros AC candidatos. Nesse momento, aplica-se o método de eleição escolhido por esta pesquisa, que consiste no AC candidato com maior ID<sup>1</sup>. O AC que tiver o maior ID dos registrados no DF, será o AA da área afetada. Os demais AC candidatos tornam-se AP nesta etapa do algoritmo.

No exemplo da Figura 2, os agentes candidatos são os AC 4, 5 e 6, deixando o AC3 como AP. Ao final do processo de eleição, os AC 4 e 5 serão AP e o AC6 será o AA da área afetada.

### 3.1.3. Mapeamento da área afetada

Uma vez eleito o AA da área afetada, é necessário que ele conheça os endereços dos AC afetados, dos SW envolvidos na falta, das *tie line* disponíveis, entre outras informações. Isso é necessário pelo fato de o AA conhecer, num primeiro momento, apenas as informações de seus AC vizinhos. Para lidar com o problema, realiza-se um mapeamento na área afetada, utilizando um método baseado em busca recursiva de grafos.

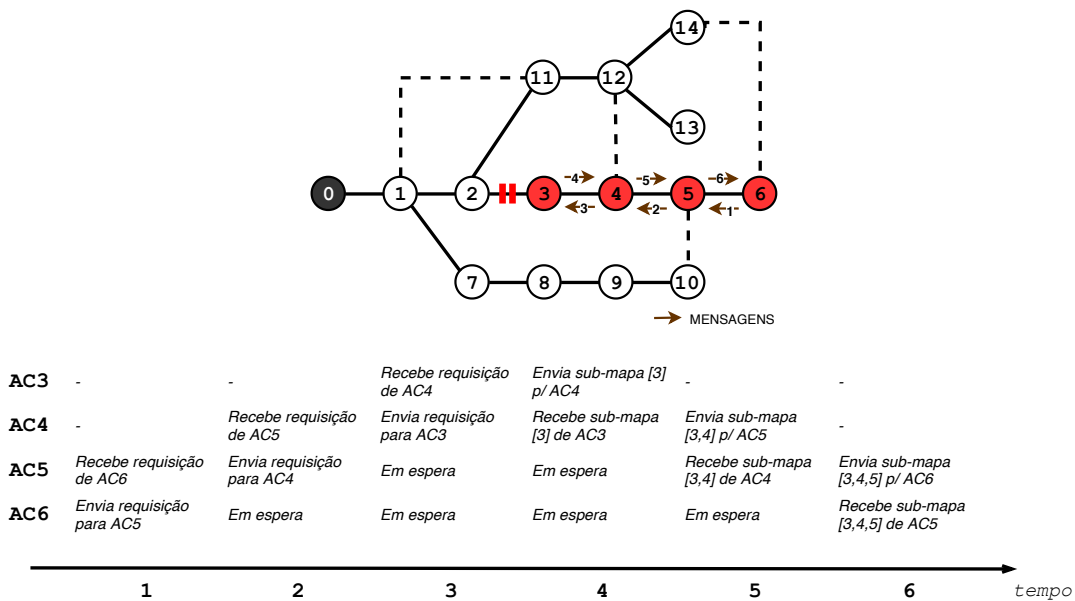


Figura 3. Passo de mapeamento realizado pelo agente ativo AC6.

Primeiramente, o AA envia uma requisição de mapeamento a todos os AC vizinhos afetados pela falta, esperando que eles respondam, por sua vez, com o sub-mapa correspondente a todos os seus vizinhos também afetados. Esse processo é repetido recursivamente em todos os vizinhos do AA até que chegue aos AC que não possuem vizinhos

<sup>1</sup>O ID de um AC é um número inteiro estabelecido para cada carga da rede elétrica, que é único entre todas as cargas do sistema

em falta, tempo quando todo o processo será revertido. Ao fim de todo o processo, o AA terá obtido o mapa com todos os endereços e informações da área afetada.

A etapa de mapeamento é ilustrada na Figura 3, que segue o exemplo tomado anteriormente. Observa-se que, no decorrer do tempo, há requisições recursivas ao longo da área afetada e espera dos AC que realizam a requisição, até que no tempo 4 as requisições começam a ser respondidas e o mapa vai sendo construído. É relevante notar que esse mapeamento é realizado através da troca coordenada de mensagens, que transportam requisições e estruturas complexas de dados da rede elétrica.

### 3.1.4. Tomada de decisão e religamento

Depois de mapeada a área afetada, o AA agora precisa realizar procedimentos para tomar decisão sobre como religar a área afetada à rede ativa, utilizando as *tie lines* disponíveis. Para isso, é utilizado o cálculo de fluxo de potência pelo do método de varredura de [Kersting 2012]. Esse cálculo permite obter informações cruciais sobre uma rede de distribuição a partir de um ponto por onde se inicia o fluxo de corrente elétrica.

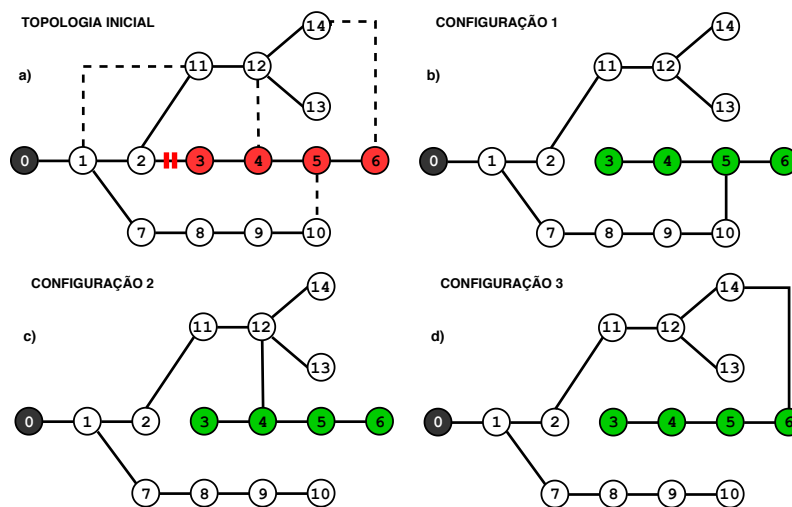


Figura 4. Configurações possíveis para a rede, para as quais o AA aplica o cálculo de fluxo de potência.

O critério de escolha de *tie lines* utilizado por esta pesquisa é o menor valor de perdas elétricas obtidas para determinada configuração de rede. Em outras palavras, entre as configurações possíveis para religar a área afetada, o AA deve escolher a configuração que resultar em menores perdas elétricas para o sistema.

Depois de realizar o cálculo do fluxo de potência para cada uma das configurações possíveis e obter seus respectivos valores de perdas elétricas, o AA escolhe a configuração mais interessante e comunica os SW responsáveis pelas chaves para que as operações de rede sejam executadas. Primeiramente o AA comunica ao SW da linha onde ocorreu a falha, solicitando a abertura da mesma e, posteriormente, solicita ao SW da *tie line* escolhida que feche o circuito, restabelecendo o suprimento de energia na área afetada.



#### 4. Simulações e resultados

As simulações foram realizadas da seguinte maneira: para representar o sensor de leitura das cargas da rede, utilizou-se o sistema de arquivos do Sistema Operacional, onde cada carga possui um arquivo único que representa o estado do fornecimento de energia. Em termos dos AC, a verificação é realizada através de um método que retorna um valor *booleano*, conforme denota o Algoritmo 1.

---

#### Algorithm 1 Função de verificação de sensor dos AC.

---

```

1: procedure LER_SENSOR(file)
2:   if file.exists() then
3:     return True                                     ▷ Fornecimento normal
4:   else
5:     return False                                   ▷ Fornecimento interrompido
6:   end if
7: end procedure

```

---

Para analisar as operações que os agentes realizaram na rede e verificar se a energia foi de fato restabelecida à área afetada, analisou-se a criação desses *arquivos de energia* pelos AC, bem como os *logs* escritos por AC e SW. O ambiente computacional utilizado nas simulações tem a configuração conforme segue: Sistema Operacional Linux (*Arch*), RAM de 10,7 GB e CPU Intel Core i5-2310 4 × 2.9 GHz, com instruções 64 bits.

#### 4.1. Resultados

Foram realizadas 10 simulações com o modelos de testes e os dados delas são apresentados na Tabela 2. Cada simulação parte da topologia inicial do modelo de testes, isto é, a simulação é iniciada, os agentes recuperam a rede, escrevem *logs* e, a seguir, são finalizados.

Simulação	Ch. aberta	Nós afetados	Ch. fechada	Tempo (ms)	Perdas (kW)
1	(2, 3)	4	(6, 14)	942	1,9493
2	(1, 7)	4	(5, 10)	862	1,8035
3	(11, 12)	3	(6, 14)	777	2,4197
4	(2, 11)	4	(6, 14)	769	9,6396
5	(1, 2)	9	(1, 11)	856	48,6029
6	(1, 2)	9	(1, 11)	802	48,6029
7	(2, 3)	4	(5, 10)	780	2,3636
8	(2, 3)	4	(5, 10)	858	5,5397
9	(1, 2)	9	(5, 10)	808	53,2739
10	(2, 11)	4	(4, 12)	852	19,6604

**Tabela 2. Resultados das simulações com o modelo de testes.**

Ressalta-se que, para as simulações de 7 a 10, foram utilizados valores de resistência e reatância diferentes dos definidos no modelo de testes original (vide Tabela 1). O objetivo com essas alterações é de verificar como o sistema multiagente se comporta quando submetido à mudanças na estrutura da rede elétrica. As alterações nos valores de resistência e as linhas que sofreram as alterações são especificadas na Tabela 3.

Em todas as simulações os agentes conseguiram executar corretamente as operações de reconfiguração de rede, realizando a recuperação dos nós afetados pela

falta. Pode-se observar que, nos dados da Tabela 2, obteve-se como maior tempo de autorrecuperação o valor de 942 ms, ao passo que para o menor tempo, o valor alcançado é de 769 ms. A média dos valores de tempo de autorrecuperação é calculado em 830,6 ms.

Simulação	Linha	Resistência original ( $\Omega$ )	Resistência modificada ( $\Omega$ )
7	(6,14)	0.5	2.0
8	{(6,14), (5,10)}	{0.5, 0.5}	{2.0, 1.5}
9	(1,11)	1.5	2.0
10	(6,14)	0.5	1.5

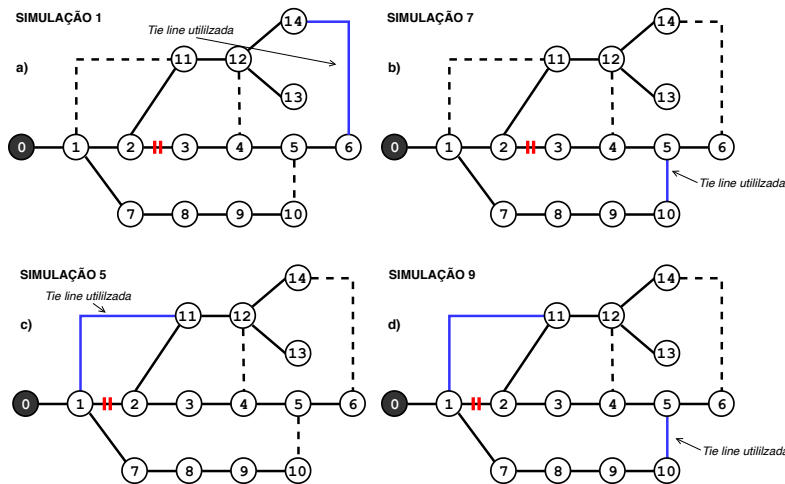
**Tabela 3. Alterações nos valores de resistências de algumas linhas para as simulações de 7 a 10.**

É possível observar ainda que, agrupando os dados da Tabela 2 pelo número de nós afetados, há a formação de três grupos: os grupos com 3, 4 e 9 nós afetados. A média de tempo de autorrecuperação para cada grupo é calculada em 777 ms, 843,8 ms e 822 ms, respectivamente.

#### 4.2. Simulações 1 e 7

Nas simulações 1 e 7, observa-se um comportamento do sistema multiagente interessante de ser enfatizado. O cálculo de perdas elétricas considera diversas variáveis da rede, entre elas a resistência e reatância das linhas de distribuição.

Observando os dados da simulação 1 na Tabela 2, percebe-se que o sistema multiagente detecta a falha na linha (2, 3) e utiliza a *tie line* (6, 14) para restaurar o fornecimento de energia elétrica à área afetada, como mostra a Figura 5a.



**Figura 5. Topologia da rede para as simulações 1, 5, 7 e 9.**

No entanto, na simulação 7, o valor de resistência é aumentado em 4 vezes na linha (6, 14). Em face disso, para o mesmo ponto de falha que na simulação 1, o sistema multiagente escolhe dessa vez a linha (5, 10) (Figura 5b). Essa mudança na escolha se justifica pelo fato de o valor de perdas elétricas ter aumentado em função do aumento da resistência na linha (6, 14), efetuado nessa simulação. Observa-se também que o valor de perdas elétricas para a simulação 7 foi maior do que o obtido na simulação 1, razão pela qual a linha (5, 10) não é escolhida na simulação 1.

### 4.3. Simulações 5 e 9

No caso das simulações 5 e 9, o processo é semelhante ao que ocorre nas simulações 1 e 7, respectivamente. Entretanto, nota-se que os valores de perdas elétricas são muito maiores para as simulações 5 e 9. Isso ocorre porque o número de nós envolvidos na falta, que é 9, é consideravelmente maior que nas simulações 1 e 7. Esse fator, assim como a resistência das linhas, impacta no cálculo de fluxo de potência e, conseqüentemente, na decisão dos AC do sistema.

## 5. Conclusões e trabalhos futuros

Este trabalho apresentou um algoritmo distribuído, implementado por um sistema multiagente reativo, que segue quatro etapas básicas para reconfigurar a rede e realizar a autorrecuperação de *smart grids* no nível de distribuição. Por meio de simulação computacional, foi possível validar o algoritmo proposto nesta pesquisa, bem como a sua implementação utilizando sistemas multiagente. A pesquisa também apresentou resultados de 10 simulações, realizadas em ambiente computacional, para um modelo de testes de rede de distribuição com 15 nós, sendo um deles uma fonte.

Uma característica importante do algoritmo implementado é que somente os nós envolvidos na falta, de fato, executam operações para a autorrecuperação da rede. Isso permite que o sistema consiga concentrar a carga de trabalho somente na região do *grid* onde ocorreu a falta, deixando os demais nós livres.

Uma segunda característica, é que o algoritmo realiza, somente duas operações de chaveamento. Isso porque a etapa de tomada de decisão é feita por meio do cálculo de fluxo de potência. Operações de chaveamento são custosas à rede, portanto é de grande interesse minimizá-las durante abordagens de reconfiguração da rede.

Como pontos sensíveis do sistema, é possível concluir que o algoritmo não lida com todo tipo de alteração que pode ocorrer na rede, como, por exemplo, nos casos em que a energia volta à área afetada antes do término do algoritmo. Além disso, nota-se que o método de eleição do agente ativo não é robusto contra atrasos que possam ocorrer na rede de comunicação, já que se baseia em um limiar de tempo determinado por tentativa e erro.

Atualmente, os autores estão trabalhando em melhorar o método de eleição de agentes ativos no sistema, baseando-o em uma outra técnica que não depende de um tempo de espera. Em trabalhos futuros, espera-se também ampliar as restrições consideradas pelo algoritmo para outras variáveis do sistema elétrico de distribuição, como queda de tensão, prioridade de cargas, análise de demanda, entre outras. Para resolver algumas dessas questões, pode-se usar técnicas como balanceamento de carga, corte seletivo de cargas e ilhamento com geração distribuída, métodos já utilizados em outros trabalhos e que podem ser incorporados ao algoritmo proposto.

## Referências

- Baran, M. E. and Wu, F. F. (1989). Network reconfiguration in distribution systems for loss reduction and load balancing. *IEEE Transaction on Power Delivery*, 4(2).
- Bellifemine, F., Caire, G., and Greenwood, D. (2007). *Developing multiagent systems with JADE*. Jhon Wiley & Sons, Liverpool.

- Campos, I. R. C. and Saraiva, F. (2018). Proposta de modelo de autorrecuperação de sistemas de distribuição de energia elétrica utilizando sistemas multiagente. In *12th Workshop-School on agents, environments and applications*, Fortaleza. 12th Workshop-School on agents, environments and applications.
- Ferreira, L. R., Siebert, L. C., Ayala, H., and Aoki, L. C. D. (2013). Solução do problema de self-healing para redes de distribuição radiais através de otimização via algoritmo genético. In *Simpósio Brasileiro de Automação Inteligente 2013, UFC*, Fortaleza. Simpósio Brasileiro de Automação Inteligente 2013.
- JADE (2017). Java agent development framework (version 4.5.0). <http://jade.tilab.com/>.
- Jia, D., Meng, X., and Song, X. (2011). Study on technology system of self-healing control in smart distribution grid. In *China Electric Power Research Institute*, Beijing. The International Conference on Advanced Power System Automation and Protection.
- Kersting, W. (2012). *Distribution System Modeling and Analysis*. CRC Press, Boca Raton.
- Larik, R. M. and Mustafa, M. W. (2015). Technologies used in smart grid to implement power distribution system. *TELKOMNIKA Indonesian journal of electrical engineering*, 16(2):232–237.
- Mahdi, M. and Genc, V. M. I. (2019). A real-time self-healing methodology using model- and measurement-based islanding algorithms. *IEEE Transactions on Smart Grid*, 10(2):1195–1204.
- Motamedi, A., Zareipour, H., and Rosehart, W. D. (2012). Electricity price and demand forecasting in smart grids. *IEEE Transactions on Smart Grid*, 3(2):664–674.
- Rahimi, F. and Ipakchi, A. (2010). Demand response as a market resource under the smart grid paradigm. *IEEE Transactions on Smart Grid*, 1(1):82–88.
- Saraiva, F. O. (2015). *Aplicações híbridas entre sistemas multiagentes e técnicas de inteligência artificial para redes inteligentes de distribuição de energia elétrica*. PhD thesis, Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos.
- Sharma, A., Srinivasan, D., and Trivedi, A. (2018). A decentralized multi-agent approach for service restoration in uncertain environment. *IEEE Transactions on Smart Grid*, 9(4):3394–3405.
- Siano, P. (2014). Demand response and smart grids—a survey. *Renewable and Sustainable Energy Reviews*, 30:461 – 478.
- Souza, F. A. (2015). Modelo baseado em sistema multiagente para autorrecuperação com corte seletivo de carga e ilhamento com geração distribuída para redes elétricas inteligentes. Dissertação, Departamento de Engenharia Elétrica, Universidade Federal do Paraná, Curitiba.
- Wang, Z., Chen, B., Wang, J., and Chen, C. (2016). Networked microgrids for self-healing power systems. *IEEE Transaction on Smart Grids*, 7(1):310–319.