

Reactive multi-agent system applied to self-healing in Smart Grids

Sistema multiagente reativo aplicado ao problema de autorrecuperação em *smart grids*

Italo Campos^{1*}, Filipe Saraiva¹

Resumo: This paper presents a decentralized algorithm for application in the smart grids self-healing problem, at the distribution level. The algorithm implementation is made using a reactive multi-agent system, which models the electrical grid in terms of autonomous agents which perform the algorithm operations in a distributed and parallel way. To validate this algorithm, two distribution network test models are used: a 15 bus model and a 33 bus model — standardized by IEEE. The results are obtained by means of computational simulation and shown in this paper, to each one of the network models. The results show that the proposed approach is able to recover all the nodes of the grid, within the simulation conditions. Moreover, it is seen that the multi-agent system directs the work load exactly to the failure point, preventing the involvement of the entire grid to the self-healing process.

Keywords: Decentralized algorithm — Multi-agent system — Self-healing — Smart grids

Resumo: Este artigo apresenta um algoritmo descentralizado para aplicação ao problema de autorrecuperação de redes elétricas inteligentes — *smart grids* — no setor de distribuição. Para implementação do algoritmo é utilizado um sistema multiagente reativo, o qual modela a rede elétrica em termos de agentes autônomos que executam as operações do algoritmo de forma distribuída e paralela. Para validação, são usados dois modelos de testes de rede elétrica de distribuição: um modelo de 15 barras e outro de 33 barras — modelo normalizado pela IEEE. Os resultados são obtidos por meio de simulação computacional e mostrados neste artigo, para cada um dos modelos de rede. Os resultados apontam que a abordagem proposta é capaz recuperar todos os nós da rede, nas condições das simulações. Além disso, verifica-se que o sistema multiagente direciona a carga de trabalho exatamente para o ponto onde a falha ocorre, evitando que toda a rede se envolva no processo de autorrecuperação.

Palavras-Chave: Algoritmo descentralizado — Sistemas multiagente — Autorrecuperação — Smart grids

¹Instituto de Ciências Exatas e Naturais, Universidade Federal do Pará, Brazil

*Corresponding author: italo.campos@icen.ufpa.br

DOI: <http://dx.doi.org/10.22456/2175-2745.97479> • Received: 18/10/2019 • Accepted: 05/03/2020

CC BY-NC-ND 4.0 - This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

1. Introdução

Na área dos Sistemas Elétricos de Potência (SEP), *smart grid* é um conceito que recebe cada vez mais atenção, uma vez que traz importantes funcionalidades sobre os sistemas elétricos tradicionais tais como gerenciamento remoto, comunicação de via dupla, resistência a ataques cibernéticos, controle em tempo real, acesso à informação em tempo real, autorrecuperação de falhas, inclusão em larga escala de fontes de energia distribuída, entre outros [1]. Esse aspecto dos *smart grids* configura 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 das tecnologias de informação e comunicação aos sistemas elétricos, compreendendo todos os seus subsistemas [2]. O

resultado dessa combinação é a entrega de uma gama de funcionalidades, além do aumento considerável nos índices de qualidade, confiabilidade e continuidade do serviço [1, 2]. Além disso, esses sistemas podem ser estratificados em termos de camadas, definidas em camada básica, de comunicação e de aplicação [3]. Essa estratificação mostra que os *smart grids* podem ser analisados de diferentes pontos de vista, como, por exemplo: desenvolvimento de *hardware* especializado – medidores inteligentes, dispositivos de proteção, sensores, etc.; protocolos de rede e comunicação; algoritmos de tomada de decisão; leitura, análise e entrega de dados, entre outros.

Existem pesquisas que se dedicam a estudar exclusivamente algum dos níveis dos *smart grids*. No nível de distribuição, verifica-se na literatura abordagens na esfera da Computação, as quais giram em torno de técnicas de inteligência arti-

ficial e computacional em alto nível. Mais especificamente, essas abordagens atuam sobre a camada de aplicação dos *smart grids*, por meio da qual implementam funcionalidades inovadoras ao setor de distribuição dos SEP. Este mesmo nível de distribuição é um dos mais importantes e críticos dos SEP, uma vez que provê a conexão entre os consumidores e o setor de transmissão [3], além de ser exposto a grandes variações de demanda e de ambiente.

No que diz respeito ao setor de distribuição, os *smart grids* assumem o papel de modelo de desenvolvimento para o 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 informações de consumo, estado da rede, etc. –, assim como melhorar os canais de comunicação entre as prestadoras de serviço e seus clientes. Além disso, os *smart grids* são apontados como grandes dinamizadores do mercado elétrico, tornando-o campo de interesse por organizações que atuam na área de compra e venda dinâmica de energia elétrica [4, 5, 6].

Em qualquer dos níveis dos *smart grids*, uma das funcionalidades mais importantes é a autorrecuperação e existem muitos autores que realizam pesquisas para abordar o problema usando um vasto leque de métodos. Autorrecuperação é a capacidade que uma rede elétrica tem de se reconfigurar, de forma a restaurar o fornecimento de energia elétrica a barras afetadas por uma eventual falha na rede [1].

De fato, muitos métodos computacionais podem ser aplicados para resolver os problemas de autorrecuperação e reconfiguração nos *smart grids*, podendo ser agrupados em duas classes de métodos: centralizados – em geral, métodos matemáticos, sistemas *fuzzy*, metaheurísticas, técnicas de otimização, etc. – e descentralizados – em especial, sistemas multiagente (SMA)[7]. Dessa forma, é notável como muitos métodos de inteligência artificial e computacional são largamente usados pela comunidade científica para lidar com esse problema de natureza combinatória [8].

Abordagens centralizadas requerem que a topologia da rede elétrica seja conhecida de antemão, tal como os endereços dos dispositivos envolvidos nas operações de rede. Para sistemas críticos, como o sistema elétrico, essas informações costumam ser mantidas pelos gerenciadores da rede (sistemas SCADA - *Supervisory Control and Data Acquisition*), porém, a tarefa de acompanhar as mudanças que ocorrem no *grid* pode ser desafiadora, assim como requer manter um poder computacional compatível com o porte da rede. Por outro lado, as abordagens distribuídas se baseiam na teoria dos Sistemas Distribuídos, definidos como sistemas que possuem componentes situados em diferentes computadores que, interligados por uma rede de comunicação, executam tarefas coordenadas por meio da troca de mensagens, de modo a aparentarem ser uma única máquina [9].

Sobre as pesquisas na área, o trabalho [10], por exemplo,

modela o problema de autorrecuperação usando uma função objetivo de minimização, a qual é processada em um algoritmo genético. Por meio dessa abordagem centralizada, os autores visam encontrar uma configuração de chaves elétricas que satisfaçam restrições do sistema, bem como permitam a autorrecuperação do mesmo. Em uma outra abordagem, também centralizada, [11] 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 diversas técnicas, entre eles o algoritmos de agrupamento *K-means* e algoritmos *Fuzzy*.

Por outro lado, utilizando abordagem descentralizada, o trabalho [7] propõe uma abordagem de autorrecuperação utilizando SMA, com dois tipos diferentes de agentes. A ideia é recuperar as cargas utilizando o maior número de *switches* seccionadores, de modo que a configuração final seja parecida com a configuração pré-falta. Se não é possível usar *switches* seccionadores, *tie switches* são utilizados em seu lugar. O trabalho afirma que a abordagem produz resultados com qualidade equivalente a um método centralizado de *Mixed Integer Programming*.

Em outra abordagem descentralizada, o estudo [8] também propõe uma estratégia de autorrecuperação baseada em SMA. O sistema proposto pelo trabalho possui três diferentes tipos de agentes trabalhando em diferentes níveis: agentes de carga, que são responsáveis basicamente por fazer requisições de recuperação, nos casos de falta; agentes de barra, que agem como intermediadores das requisições entre os agentes de carga e agentes geradores; agentes geradores, que são quem fazem a tomada de decisão, verificando se é possível atender às requisições dos agentes de carga respeitando as restrições da rede.

Além desses, existem uma série de outros trabalhos que utilizam abordagens descentralizadas para o problema de autorrecuperação, fazendo uso ou não da tecnologia de MAS, por exemplo, [12, 13, 14], entre muitos outros. Por questões de espaço, foram resumidas as principais características de [7, 8, 12, 13, 14] no quadro da Figura 1.

A coluna do quadro na Figura 1 “Classes de agentes” refere-se ao número de classes de agentes utilizadas pela abordagem, isto é, os tipos de agentes empregados. Já a coluna “Sistema de testes” indica qual sistema de teste foi utilizado pela abordagem.

Verifica-se que existem abordagens muito robustas para o problema de autorrecuperação, cada uma lidando com um conjunto de restrições e utilizando diferentes ideias. Obviamente, quanto mais rebuscado o método, maior será sua complexidade em *design*, implementação e execução, especialmente quando se fala em métodos centralizados de busca por uma solução ótima. Esse fato pode pôr em cheque a viabilidade de alguns métodos, principalmente se for considerado que o cenário de aplicação é um ambiente com recursos computacionais limitados. Nesse sentido, este trabalho visa contribuir para a literatura da área propondo um método de recuperar a rede elétrica por meio da localização e análise

Referência	Classes de agentes	Sistema de teste	Ideia principal	Destques
[7]	2	16 barras	Utilizar linhas com <i>switches</i> seccionadores no lugar de <i>tie switches</i> ; manter a configuração final próxima do estado pré-falta.	Compara os resultados com uma abordagem centralizada MIP.
[8]	3	5 barras	Agentes de carga negociam com agentes geradores para serres religados; decisão a cargo dos agentes geradores.	A abordagem considera a capacidade do gerador, capacidade das linhas, perfis de tensão e balanceamento de potência.
[12]	4	4 alimentadores, 2 subestações, 68 cargas, 5 geradores distribuídos	Utilizar geração distribuída para recuperar cargas em estado de falta, caso não haja possibilidade de recuperação dos nós por chaveamento; é utilizado ilhamento intencional de acordo com prioridade de cargas.	SMA reativo; ilhamento intencional; prioridade de cargas; lida com a detecção da falha; interação de JADE com MATLAB.
[13]	6	38 e 119 barras	Por meio de uma complexa série de passos, verificar o estado da rede durante a falta para, através de valores estimados, verificar quais condições são mais favoráveis para formar ilhas utilizando geração distribuídas e baterias.	Utiliza geração e armazenamento distribuído; considera fontes de energia renovável; considera um grande conjunto de restrições; utiliza múltiplas camadas de agentes para negociação; abordagem adaptada para ambientes incertos decorrentes do uso de energia renovável.
[14]	-	6 e 40 micro grids	Utiliza <i>micro grids</i> autônomos como cenário de estudo. Cada <i>grid</i> opera de maneira independente, mas ligados entre si por canal de comunicação. Nos casos de falha, utiliza geração distribuída e coaliso com outros <i>grids</i> para fazer a recuperação dos nós.	Não utiliza SMA; dá suporte a múltiplas falhas simultâneas; a descentralização das redes requisitaram menor custo computacional que métodos centralizados.

Figura 1. Trabalhos relacionados que utilizam abordagens descentralizadas.

das configurações do *grid* de maneira descentralizada e fazendo uso de poucos recursos. Utilizando a informação local de cada barra, o método mapeia a rede em busca de alguma configuração que possa ser usada para restaurar as barras em falta. Esse mapeamento pode ser usado para diversos fins. Nesta etapa do trabalho, utiliza-se o mapeamento para religar os nós em falta utilizando uma configuração que produz menos perdas elétricas.

Nos *smart grids*, existem as figuras da subestação com suas fontes, das linhas de distribuição com suas chaves seccionadoras (*switches*), das barras e cargas consumidoras, etc., cada uma utilizando equipamentos especializados interligados por rede de comunicação. Esses sistemas são considerados críticos, portanto, não são tolerantes a falhas, necessitando de uma resposta rápida quando elas ocorrem. Visando essas características, o método proposto foi desenhado para funcionar de maneira distribuída e com poucas etapas. Além disso, foi escolhida a tecnologia de SMA reativo para implementá-lo, uma vez que esta possui características compatíveis com esses sistemas.

Neste artigo, é apresentado um algoritmo descentralizado para realizar a autorrecuperação de *smart grids* no setor de distribuição. A metodologia deste trabalho inclui a utilização de um SMA reativo para simular modelos de redes elétricas de distribuição, bem como as operações necessárias para realizar a autorrecuperação. Por meio de simulação computacional, o trabalho objetiva validar o algoritmo de mapeamento e autorrecuperação, inserindo-o na literatura como uma alternativa para implementação dessa funcionalidade em *smart grids*. Até este ponto do trabalho, foram considerados apenas cenários com um único ponto de falha.

Para validação, foram utilizados dois modelos de rede elétrica de distribuição: um de 15 barras, desenvolvido nesta pesquisa para validação primária do algoritmo e do SMA implementado, utilizando como referência o modelo de testes da IEEE com 33 barras [15]; o próprio modelo de testes da IEEE de 33 barras [15]. Os resultados foram coletados

mediante verificação da configuração pós-falta da rede. Por meio dessa análise é possível aferir se a configuração pós-falta é compatível com alguma configuração de autorrecuperação, isto é, não utilizando a linha com falha e recuperando os nós. Para auxiliar esse processo, é obtido o grafo da rede pela análise de *logs* dos agentes, que são arquivos contendo as informações referentes aos objetos da rede.

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 SMA modelado, evidenciando os processos que ocorrem durante a autorrecuperação; 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 sobre a pesquisa.

2. Descrição do problema, restrições da rede e modelos de teste

O escopo desta pesquisa é delimitado pelo problema de autorrecuperação em redes do tipo *smart grids*, no setor de distribuição. A autorrecuperação é uma das mais importantes funcionalidades dos *smart grids* e pode ser definida como o restabelecimento automático da energia elétrica às cargas que são afetadas por uma interrupção no fornecimento [1]. O termo ‘automático’ está relacionado à realização ativa de operações pela própria rede elétrica, sem a intervenção de um operador humano.

Existe uma série de análises e considerações que precisam ser realizadas para que a funcionalidade de autorrecuperação seja, de fato, implementada, conferindo a ela um alto grau de complexidade. Gerenciamento de fontes distribuídas ao longo da rede, tomada de decisão, verificação do estado dos nós da rede, verificação de variação de tensão e demanda dos consumidores, são alguns exemplos de variáveis que podem ser levadas em consideração quando se pensa em uma abordagem para implementar a autorrecuperação de uma rede elétrica [16].

Quando se fala em redes de distribuição, existem diversas restrições que precisam ser obedecidas para que não haja problemas durante sua operação, evitando assim que ocorra um colapso geral em todo o sistema ao invés de operá-lo corretamente. Uma dessas restrições é o *princípio da radialidade*, que diz que a configuração das linhas ativas da rede de distribuição não pode conter ciclos, uma vez que, se ocorridos, podem causar um colapso imensurável em toda a estrutura do *grid*. Se divididas as restrições em níveis de severidade, é possível classificar a radialidade como participante em um dos mais altos níveis. A rigor, todas as restrições do sistema de distribuição são relevantes e devem ser satisfeitas para que seja possível prestar um serviço contínuo e com qualidade. Entretanto, esta etapa do trabalho considera apenas o princípio

da radialidade durante os processos realizadas pelo algoritmo de autorrecuperação.

Em termos de definição formal, 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

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

É pertinente salientar que as linhas ativas de uma rede de distribuição compreendem o conjunto de todas as linhas que estão conduzindo corrente elétrica ao longo da rede, 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.

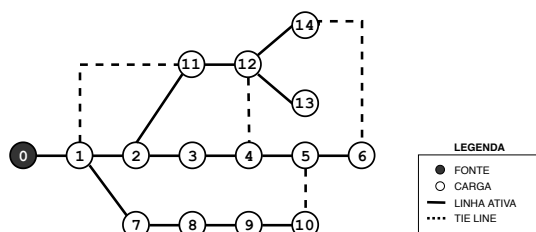


Figura 2. Topologia do modelo de testes de 15 barras.

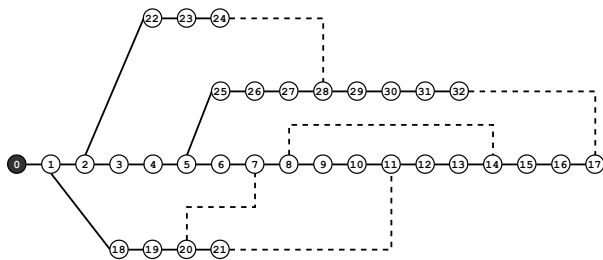


Figura 3. Topologia do modelo de testes de 33 barras.

Para efeito de validação do algoritmo e do SMA apresentados nesta pesquisa, foi necessário eleger modelos de testes de redes de distribuição – simplesmente modelos de testes – para que servissem de base para as operações de redes realizadas nas simulações com o SMA implementado. Em um primeiro momento, foi decidido trabalhar com um modelo de teste que dispusesse de um número de barras razoável e uma topologia adequada para averiguação do comportamento do SMA. Formulou-se então o modelo de testes de 15 barras, sendo um deles uma fonte, a partir da qual inicia o fluxo de corrente elétrica para as demais barras da rede. A topologia e as especificações de rede utilizadas no modelo foram aferidas com base no modelo de testes de 33 barras da IEEE, definidas em [15]. Em adição, utilizou-se também o próprio modelo de teste de 33 barras da IEEE, de modo que as simulações pudessem apresentar resultados em um modelo de testes já normalizado e frequentemente utilizado na literatura. As Figuras 2 e 3 trazem as topologias dos modelos de testes de 15 e 33 barras, respectivamente, ao passo que a Tabela 1 apresenta os detalhes técnicos do modelo de testes de 15 barras apenas.

Por questões de espaço, as especificações de rede do modelo de testes de 33 barras podem ser consultados em [15].

Br. No	Rc. Nd.	Sn. Nd.	Br. Parameters		Sn. Nd. Parameters	
			r(Ω)	x(Ω)	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 barras.

Observando a Figura 2, na topologia de 15 barras, conta-se 14 linhas ativas e 4 tie lines. Já na topologia de 33 barras, mostrada na Figura 3, conta-se 32 linhas ativas e 5 tie lines. Tie lines (ou linhas de reserva), são linhas de distribuição inseridas à rede com chave elétrica aberta, isto é, não conduzem corrente elétrica em determinado momento. O propósito dessas linhas é de serem usadas como alternativas para religar nós que estejam afetados por uma eventual interrupção no abastecimento elétrico. Define-se, portanto, que o objetivo deste trabalho é apresentar um SMA que, depois de detectar uma falha no fornecimento de energia, mapeia a rede de forma descentralizada em busca de uma configuração que use as tie lines disponíveis, de forma que seja restabelecido o suprimento de energia elétrica às barras afetadas, obedecendo à restrição de radialidade da rede. Vale ressaltar que o SMA desenvolvido até essa etapa da pesquisa lida com cenários com apenas um único ponto de falha.

3. O sistema multiagente e o algoritmo de autorrecuperação

A presente abordagem faz uso de SMA, que é uma técnica da área de Inteligência Artificial Distribuída [1] e consiste em modelar um problema em termos de agentes independentes, de modo que eles operem em conjunto e se coordenem mutuamente para alcançar objetivos específicos e/ou coletivos [17]. Entende-se por agente um sistema de computador situado em um ambiente, o qual é capaz de realizar ações autônomas no ambiente, objetivando cumprir seus objetivos particulares [18].

Os smart grids são sistemas críticos, com equipamentos especializados distribuídos ao longo da rede, cada um com poder computacional limitado e interligados entre si por uma

rede de comunicação. Ademais, o próprio problema estudado apresenta um gatilho – que é a detecção da falta de energia –, o qual pode disparar uma série de ações. Mentalizando essas características, optou-se por utilizar a tecnologia de SMA reativo, destacando-se as seguintes razões:

- SMA são ideais para operar em um ambiente distribuído [1];
- SMA são adaptáveis para funcionar em ambientes de poder computacional reduzido;
- SMA reativos são ativado por gatilhos;
- SMA reativos retornam respostas mais rápidas que outros tipos de arquitetura de SMA [17];
- A abordagem utilizada não requer o uso de *reasoning* dos agentes.

Além dessas características, os SMA utilizam processamento paralelo, entregando respostas mais rápidas sem a necessidade do conhecimento completo do sistema [19]. SMA também são melhores escaláveis que os sistemas centralizados e, dependendo da implementação, são capazes de entregar respostas com qualidade equivalente aos métodos centralizados [7, 20]. Em adição, a abordagem escolhida não inclui nenhum método de otimização centralizado para alcançar os resultados pretendidos, a própria coordenação dos agentes já proporciona as informações necessárias para que possam ser feitas tomadas de decisão sobre a configuração da rede. A busca pela reconfiguração ótima da rede está fora do escopo deste trabalho.

3.1 Modelagem do sistema multiagente

Em termos de modelagem, foram desenvolvidos dois tipos de agentes para lidar com o problema em evidência 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 e executam praticamente todas as etapas do algoritmo. Numa visão geral, os AC se comportam como descreve a Figura 4.

Com relação aos *Agentes de Chave* (SW), seus comportamentos são muito simples e podem se resumir em uma só atividade: são responsáveis pela abertura e fechamento das chaves elétricas, de acordo com a solicitação dos AC.

Em cada barra (ou carga, ou ainda nó) da rede elétrica, existe um AC alocado, ao passo que em cada linha, incluindo *tie lines*, existe um SW alocado. Portanto, nos modelos de testes utilizados, a alocação de agentes é feita da seguinte maneira: para o modelo de 15 barras, 15 AC, 14 SW para as linhas ativas e outros 4 SW para *tie lines*; para o modelo de 33 barras, 33 AC, 32 SW para linhas ativas e outros 5 SW para *tie lines*. Destaca-se que não há diferença na modelagem entre os nós comuns e o nó fonte. Isso permite a escalabilidade do sistema para modelos de teste maiores.

Há uma característica do SMA implementado importante de ser enfatizada. Para manter a abordagem descentralizada,

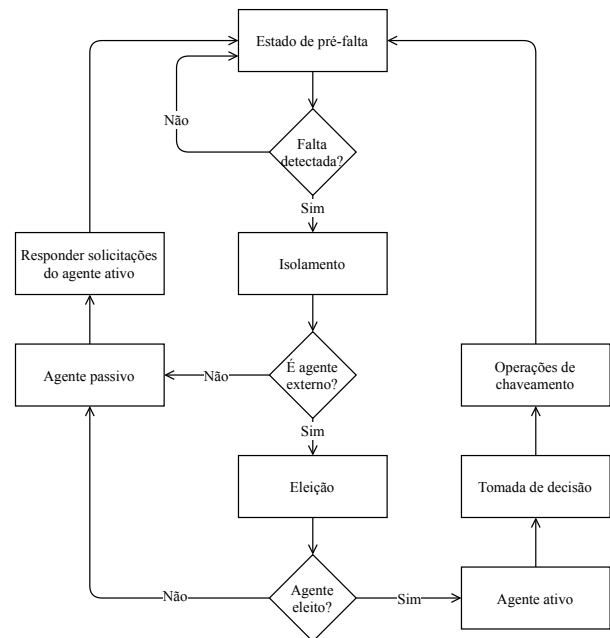


Figura 4. Esquema do comportamento dos Agentes de Carga.

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 mantém comunicação. Essa característica foi denominada como *visão mínima* do sistema.

A visão mínima promove um grande nível de independência entre os agentes residentes na rede, o que favorece características como escalabilidade e adaptabilidade do SMA às grandes variações da rede em tempo de operação. Por outro lado, aumenta-se o grau de complexidade da coordenação dos agentes de modo que torna-se difícil, por exemplo, coletar dados de agentes que estão muito afastados entre si. Entretanto, falhas ocorrem frequentemente em locais isolados do *grid*, o que significa que, na prática, a visão mínima torna-se vantajosa por manter informações pontuais apenas nos locais onde ela realmente é necessária.

3.2 O algoritmo

O algoritmo proposto por essa pesquisa possui quatro etapas básicas: isolamento, eleição de agente ativo, mapeamento e tomada de decisão. Todos eles podem ser visualizados de maneira geral na Figura 4. Cada uma dessas etapas é descrita com detalhes nas próximas subseções. A partir deste ponto do trabalho, os termos ‘AC’, ‘nó’, ‘barra’ e ‘carga’ serão usados indiscriminadamente, referindo-se a um determinada barra da rede elétrica que é controlada por um AC.

3.2.1 Isolamento

Os primeiros procedimentos do algoritmo a serem executados pelo SMA acontecem ao ser detectada uma falta na rede

elétrica, momento em que apenas 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 isolar o local afetado. Essa ação dos agentes corresponde ao processo em que é necessária a atualização das informações de estado da rede que, devido à falta, sofreu alterações. Nesse momento, cada agente afetado atualiza sua visão particular da rede.

A atualização dessas informações é realizada através da troca de mensagens entre os nós da área afetada. Essas mensagens seguem o protocolo requisição-resposta do modelo cliente-servidor, isto é, um AC faz uma requisição de atualização a um outro AC que esteja diretamente ligado a ele e este, por sua vez, responde com uma mensagem cujo conteúdo é um objeto contendo informações sobre o seu estado particular. A Figura 6 traz um diagrama de sequência que ilustra, de maneira geral, como ocorre a troca de mensagens entre os agentes na etapa de isolamento. Considere que os AC genéricos da Figura 6 estão conectados conforme representado pelo grafo da Figura 5.

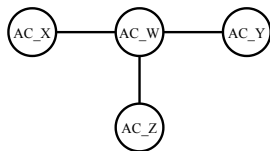


Figura 5. Esquema de conexão entre os AC da Figura 6

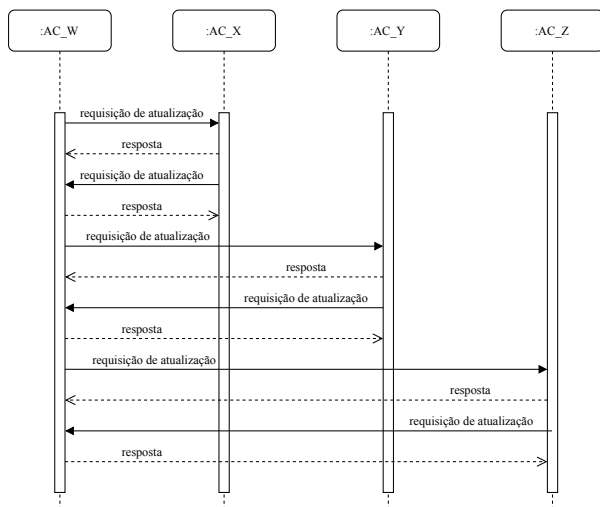


Figura 6. Diagrama de sequência da etapa de isolamento.

Note que, apesar do diagrama sequenciar as mensagens de requisição em uma certa ordem, na prática, isso não ocorre por que as essas mensagens de requisição são assíncronas e paralelas. Ao final do processo de isolamento, todos os AC envolvidos na falta estarão atualizados e capazes de detectar o ponto de falha.

3.2.2 Eleição de agente ativo

Terminada a etapa de isolamento, é necessário escolher um dos AC dentre o conjunto de nós afetados para realizar a

tomada de decisão em nome do restante dos agentes da área afetada. Esse processo se torna importante porque caso dois ou mais AC realizem a tomada de decisão, corre-se o risco de ambos entrarem em conflito e haver decisões que violem a restrição de radialidade da rede. Sendo assim, o processo de eleição visa fazer os agentes evitarem eventuais conflitos, escolhendo um dentre eles para realizar o papel de agente ativo (AA). Ao fim do processo, os demais AC não eleitos como AA, tornam-se agentes passivos (AP), aguardando as orientações do AA e cooperando nos processos posteriores do algoritmo. Agentes ativos e passivos são definidos abaixo:

- *Agente Ativo (AA)*: é um AC eleito para executar o cálculo de fluxo de potência na rede e tomar as decisões sobre a reconfiguração da mesma. Somente AC externos podem se candidatar para ser 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.
- *Agente Passivo (AP)*: é todo o AC não eleito para ser AA. Os AP ficam em estado de espera, aguardando as requisições de mapeamento providas do AA.

Se a área afetada não dispuser de agentes externos, não haverá candidatos para AA, o que culminará na não reparação da rede e o algoritmo é finalizado sem entregar uma resposta. Na prática, isso significa que aquela área afetada não dispõe de *tie lines* para realizar o processo de autorrecuperação, justificando o comportamento dos agentes.

Em termos de implementação, 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, sendo necessário um método mais rebuscado de se realizar o processo de eleição.

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 ID¹. O AC que tiver o ID igual

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

ao maior ID dos registrados no DF será o AA da área afetada. Os demais AC candidatos tornam-se AC passivos nesta etapa do algoritmo.

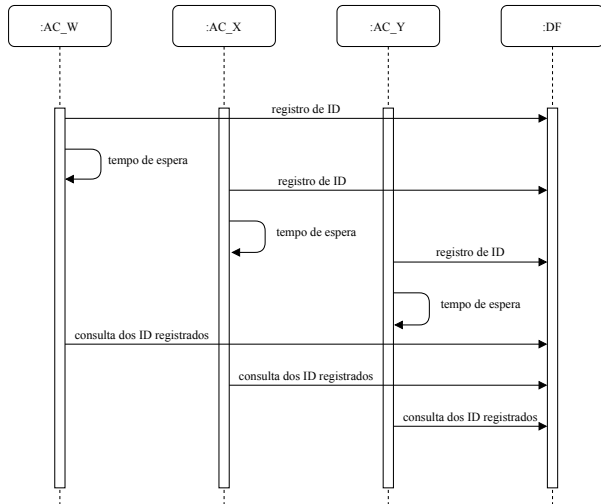


Figura 7. Diagrama de seqüência da etapa de eleição.

No diagrama da Figura 7, são utilizados agentes genéricos. Após o registro dos agentes no DF e o limiar de tempo, o agente que tiver maior ID será eleito como AA.

3.2.3 Mapeamento da área afetada

Uma vez eleito, é necessário que o AA conheça os endereços de todos os outros AC afetados, dos SW envolvidos na falta, do SW onde a falha ocorreu, dos SW das *tie line* disponíveis para autorrecuperação, entre outras informações. Isso é necessário por consequência da visão mínima dos agentes. Dessa maneira, um mapeamento na área afetada é realizado utilizando um algoritmo baseado em busca em profundidade de grafos. O algoritmo segue descrito abaixo.

Primeiramente, o AA envia uma requisição de mapeamento a todos os seus AC vizinhos afetados pela falta, esperando que eles respondam, por sua vez, com o submapa² correspondente à parte da rede onde residem. Esse processo é repetido recursivamente em todos os vizinhos do AA até que se chegue aos AC que não possuem vizinhos em falta, tempo quando será executado o processo inverso. Ao fim de todo o processo, o AA terá obtido o mapa com todos os endereços e informações da área afetada. A Figura 9 traz um diagrama de seqüência desse processo, considerando o agente genérico AC_W como AA. Além disso, considere que os agentes genéricos na Figura 9 estão conectados conforme o grafo da Figura 8.

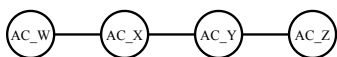


Figura 8. Esquema de conexão entre os AC da Figura 9

²O termo *submapa* designa uma estrutura de dados que contém informações parciais sobre a rede. É, na prática, uma parte do mapa da área afetada que corresponde à visão de cada agente que a compõe.

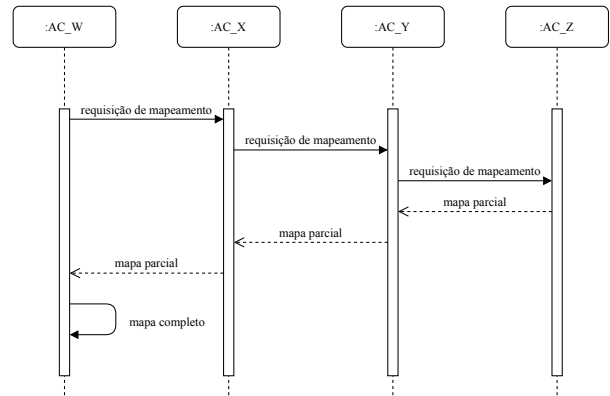
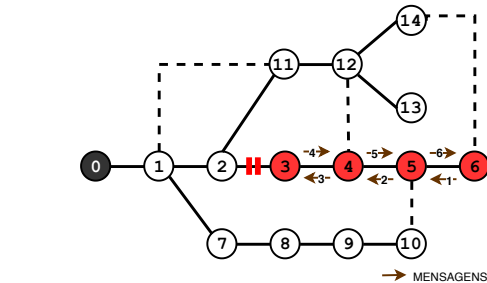


Figura 9. Diagrama de seqüência da etapa de mapeamento.

Fazendo um paralelo com o algoritmo de busca em profundidade de grafos, é necessário visitar todos os vértices do grafo utilizando a lógica pós-ordem, onde cada nó primeiramente faz a busca em seus adjacentes para, então, ser visitado. Nesse contexto, o grafo em questão é a área afetada, o nó de início é o AA e a ação de visita é a resposta, contendo o submapa da rede, de um AC ao seu solicitante.

Neste momento, será usado um exemplo de mapeamento em um cenário baseado no modelo de testes de 15 barras, com o objetivo de ilustrar como o processo de mapeamento ocorre. A Figura 10 traz este exemplo, detalhando no diagrama o estado dos AC da área afetada no decorrer do tempo. As setas no diagrama indicam mensagens que são trocadas entre os agentes durante o processo de mapeamento.



t. tempo

	AC6	AC5	AC4	AC3
1	Envia requisição para AC5	Recebe requisição de AC6	-	-
2	Em espera	Envia requisição para AC4	Recebe requisição de AC5	-
3	Em espera	Em espera	Envia requisição para AC3	Recebe requisição de AC4
4	Em espera	Em espera	Recebe submapa [3] de AC3	Envia submapa [3] p/ AC4
5	Em espera	Recebe submapa [3,4] de AC4	Envia submapa [3,4] p/ AC5	-
6	Recebe submapa [3,4,5] de AC5	Envia submapa [3,4,5] p/ AC6	-	-

Figura 10. Etapa de mapeamento realizado pelo agente ativo AC6.

Observa-se na Figura 10 que, no decorrer do tempo, há requisições longo da área afetada, partindo do AA e indo em direção às extremidades da área afetada. Também observa-

se a espera dos agentes que realizam a requisição até que, no tempo 4, estas começam a ser respondidas e o mapa vai sendo construído em sentido reverso ao inicial. As mensagens trocadas transportam requisições e estruturas complexas de dados que serão usadas para construir o mapa final pelo AA.

3.2.4 Tomada de decisão e religamento

Depois de mapear a área afetada, o AA agora precisa realizar procedimentos para tomar decisão sobre 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 [21]. Esse cálculo permite obter informações relevantes sobre uma rede de distribuição, como valores de tensões nas barras, correntes nas linhas e perdas elétricas, tudo isso a partir de um ponto por onde se inicia o fluxo de corrente elétrica. O cálculo de fluxo de potência é utilizado na área da Engenharia Elétrica tanto na fase de planejamento quanto na de operação de redes elétricas, servindo como importante ferramenta para manutenção do estado normal de operação de um *grid*, uma vez que seu resultado permite se conhecer o que está ocorrendo (ou pode ocorrer) na rede [22].

O critério de escolha de *tie lines* utilizado por esta pesquisa se baseia no 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.

O termo ‘configuração de rede’ remete à topologia que a rede pode assumir depois de algumas operações de chaveamento. Na prática, o número de configurações que uma rede pode assumir é uma questão combinatória, e existem muitos trabalhos que abordam esse problema com métodos de aproximação. Entretanto, para este trabalho, serão levadas em conta apenas configurações que resultem, no máximo, na abertura da linha onde foi detectada a falha e no fechamento de uma das *tie lines* disponíveis. Por essa ótica, percebe-se que o número de configurações possíveis para qualquer modelo de teste é igual ao número de *tie lines* disponíveis para uso pela área afetada.

Ao obter o mapa da rede, o AA executa o cálculo do fluxo de potência para cada nova configuração possível, a partir de cada um dos nós com energia mais próximos da área afetada, utilizando as *tie lines*. Depois de cada execução do cálculo, os valores obtidos são armazenados e usados em posteriores comparações para decidir qual a melhor configuração, de acordo com o critério estabelecido nesta pesquisa.

A Figura 11 continua o exemplo da Figura 10 e mostra as possíveis configurações que a rede pode assumir para este caso. O cálculo de fluxo de potência é executado a partir dos AC 10, 12 e 14, buscando, entre eles, a melhor configuração que a rede pode assumir para, então, recuperar a área afetada. Dentre as configurações mostradas na Figura 11, o AA escolherá a que resultar em menor perda elétrica, valor que é obtido apenas por meio da execução do cálculo de fluxo de potência.

Depois de realizar o cálculo do fluxo de potência, o AA

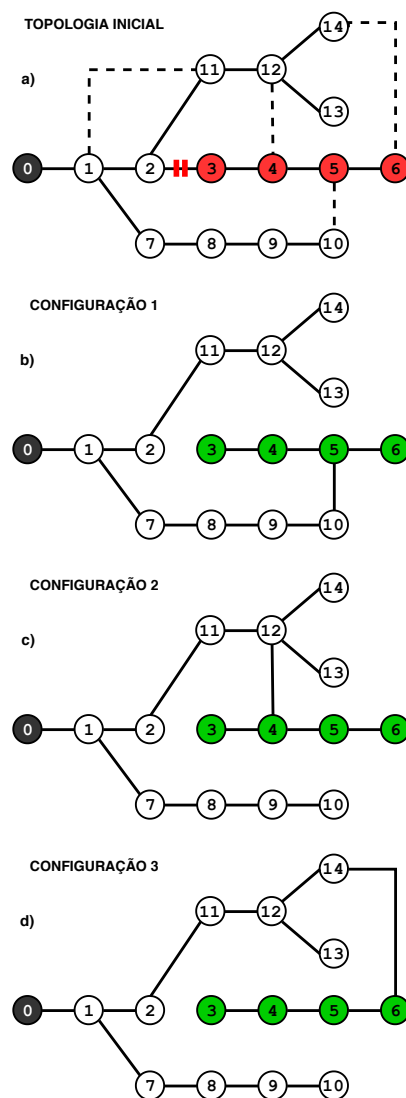


Figura 11. Configurações possíveis para o modelo de testes de 15 barras, para as quais o AA aplica o cálculo de fluxo de potência.

escolhe a configuração mais interessante e comunica os SW responsáveis pelas chaves para que as operações de chaveamento 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. Este procedimento é realizado obrigatoriamente nessa ordem, garantindo assim que a rede se manterá radial mesmo que a linha com falha seja posteriormente reparada ou que a energia retorne ao sistema durante o processo de autorrecuperação.

4. Simulações e resultados

Para a implementação dos agentes, utilizou-se o *JADE Framework* [23], que proporciona uma gama de ferramentas para comunicação de agentes, além de ser muito usado para

abordagens com SMA reativo. Já as simulações com o algoritmo e com o SMA foram realizadas conforme descrevem os próximos parágrafos.

Para representar o sensor de leitura das cargas, utilizou-se o sistema de arquivos do sistema operacional, onde cada carga possuiu um arquivo único que representa o estado do fornecimento de energia, aqui denominado *arquivo de energia*. Do ponto de vista dos AC, a verificação desses arquivos é realizada através de uma função que retorna um valor *booleano*, conforme denota o Algoritmo 1. Esse processo modela a verificação periódica que cada AC deve fazer na barra onde reside para monitorar faltas.

Algorithm 1 Função de verificação de sensor dos AC.

```

1: function 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 function

```

A partir do momento em que um agente detecta uma interrupção no fornecimento de energia, este inicia a execução do algoritmo, na etapa de isolamento.

Para simular as faltas no sistema, foram analisados os grafos correspondentes a cada topologia dos modelos de testes utilizados na simulações e, então, escolhiam-se os nós que seriam afetados por uma eventual falha em uma linha arbitrária. Uma vez escolhidos, os arquivos de energia correspondentes a esses nós eram excluídos simultaneamente e os agentes iniciavam a execução do algoritmo.

Para analisar os resultados das operações que os agentes realizaram na rede e verificar se a energia foi de fato restabelecida à área afetada, observou-se a criação de novos *arquivos de energia* para cada AC afetado, bem como os *logs* escritos pelos 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 gerais

Ao todo, foram realizadas 15 simulações, sendo 7 utilizando o modelos de testes de 15 barras (*15-bus*) e outras 8 com o modelo de testes de 33 barras (*33-bus*). A maior quantidade de simulações com o modelo de testes *33-bus* se justifica pelo fato de este ser um modelo maior e com mais opções interessantes de pontos de falha que o modelo *15-bus*. Os principais dados das simulações estão apresentados na Tabela 2.

Os termos ‘Sim.’, ‘Ch. aberta’, ‘NA’ e ‘Ch. fechada’, são abreviações para número da simulação, chave aberta, número de nós afetados e chave fechada, respectivamente.

Nesse momento, ressalta-se que, para as simulações 6, 7 e 13, foram utilizados valores de resistência e reatância diferen-

Sim.	Topologia	Ch. aberta	NA	Ch. fechada	Tempo (ms)	Perdas (kW)
1	15-bus	2,3	4	6,14	942	1,9493
2	15-bus	1,7	4	5,10	862	1,8035
3	15-bus	11,12	3	6,14	777	2,4197
4	15-bus	2,11	4	6,14	769	9,6396
5	15-bus	1,2	9	1,11	856	48,6029
6	15-bus	2,3	4	5,10	780	2,3636
7	15-bus	2,11	4	4,12	852	19,6604
8	33-bus	13,14	4	17,32	992	0,9122
9	33-bus	9,10	8	17,32	972	9,4530
10	33-bus	5,6	12	11,21	1071	43,6317
11	33-bus	2,3	15	24,28	1062	134,7714
12	33-bus	1,18	4	11,21	885	4,4140
13	33-bus	2,3	15	24,28	1188	263,0780
14	33-bus	5,25	7	24,28	939	12,4986
15	33-bus	2,22	16	17,32	568	107,6112

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

tes dos definidos no modelos de testes originais. O objetivo dessas alterações é de verificar como o SMA se comporta quando submetido à mudanças na estrutura da rede elétrica, verificando se a tomada de decisão deles é influenciada pela configuração do *grid*. As alterações nos valores de resistência e as linhas que sofreram as alterações são especificadas na Tabela 3. Além disso, destaca-se que as simulações 14 e 15 foram feitas em série, o que será explicado nas próximas subseções.

Sim.	Linha	Res. original (Ω)	Res. modificada (Ω)
6	(6,14)	0.5	2.0
7	(6,14)	0.5	1.5
13	(24,28)	0.5	2.0

Tabela 3. Mudanças nos valores de resistências de algumas linhas para as simulações 6, 7 e 13.

Em todas as simulações os agentes conseguiram executar corretamente as operações de reconfiguração de rede, realizando a recuperação dos nós afetados pela falta. Em nenhum dos casos houve a eleição de múltiplos agentes ativos, validando o algoritmo de eleição para este modelo, sob essas condições.

Em relação aos dados da Tabela 2, obteve-se como maior tempo de autorrecuperação o valor de 1188 ms, na simulação 13, ao passo que para o menor tempo, o valor alcançado é de 568 ms, na simulação 15. Entre os valores obtidos nas simulações com o modelo *15-bus*, o maior tempo de autorrecuperação é de 942 ms e o menor tempo é de 769 ms. Por sua vez, entre os valores obtidos nas simulações com o modelo *33-bus*, o maior tempo é de 1188 ms e o menor é de 568 ms. A média dos valores de tempo de autorrecuperação para as simulações com o modelo *15-bus* é calculado em 834 ms, ao passo que para as simulações com o modelo *33-bus* é calculado em 959,6 ms. A média de tempo considerando todas as simulações é calculada em 901 ms.

Faz-se importante destacar que os resultados de tempo foram obtidos a partir do ambiente de simulação modelado, o que significa que, em um ambiente real, o tempo de resposta seria invariavelmente diferente destes resultados, dado que muitos outros fatores podem influenciar no processo de autorrecuperação, como o fato de que os nós estão distantes

geograficamente uns dos outros e que os *delays* da rede de comunicação podem ser diferentes para cada caso.

4.2 Considerações gerais

Analisando os dados mostrados na Tabela 2, é possível notar pelo menos duas características importantes.

A primeira é que o tempo de autorrecuperação não varia em grande escala quando se compara os dois modelos de testes utilizados e, tampouco, em relação ao número de nós afetados. É claro que isso é dito comparando dois modelos que, relativamente, não são tão diferentes em tamanho e complexidade, o que indica que, para se aferir se o tempo de autorrecuperação está condicionado a algum desses fatores, de modo empírico, é necessário testar o SMA em modelos de testes muito maiores que estes. Porém, cabe-se dizer que, mesmo analisando o algoritmo descrito na Seção 3 de modo simplório, fica claro que o tempo de resposta do sistema dependerá muito mais do número de nós afetados que de qualquer outro fator.

Em segundo lugar, é notável como as perdas elétricas aumentam quando as resistências nas linhas também são aumentadas. Obviamente isso é um efeito elétrico claramente descrito pela Física, mas, olhando mais detalhadamente, pode-se notar que os valores de perdas elétricas aumentam porque o SMA não conta mais com *tie lines* interessantes para realizar a autorrecuperação, uma vez que os valores de resistência dessas linhas foram aumentados. Isso indica que o SMA realmente faz a tomada de decisão baseado-se no estado do sistema e não em alguma característica pré-programada do modelo de testes. Essa característica do SMA será descrita com mais detalhes nas próximas subseções.

A etapa de simulações também mostrou que o SMA modelado não consegue lidar corretamente com múltiplos pontos de falhas na rede. Depois de análise, esta limitação foi atribuída ao ponto centralizador utilizado pelo SMA, o agente *Directory Facilitator*, no caso. Essa afirmação é justificada pelo fato de este agente precisa manter o ID do AA até que todos os demais agentes terminem de consultá-lo durante o processo de eleição. Para que esta limitação seja resolvida, é necessário aplicar algum método de identificar o tipo de relação entre os pontos de falha, mas esse quesito está fora do escopo deste trabalho.

4.3 Simulações 1 e 6

Nas simulações 1 e 6, ambas no modelo *15-bus*, observa-se um comportamento do SMA interessante de ser notado. O cálculo de perdas elétricas considera diversas variáveis da rede elétrica, entre elas a resistência e reatância das linhas da rede.

Observando os dados da simulação 1 na Tabela 2, percebe-se que o SMA 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 12a. Para essa configuração, tem-se as perdas elétricas no valor de 1,9493 kW.

No entanto, na simulação 6, o valor de resistência é aumentado em 4 vezes na linha (6,14), conforme descreve a Tabela 3. Em face disso, para o mesmo ponto de falha que na simulação

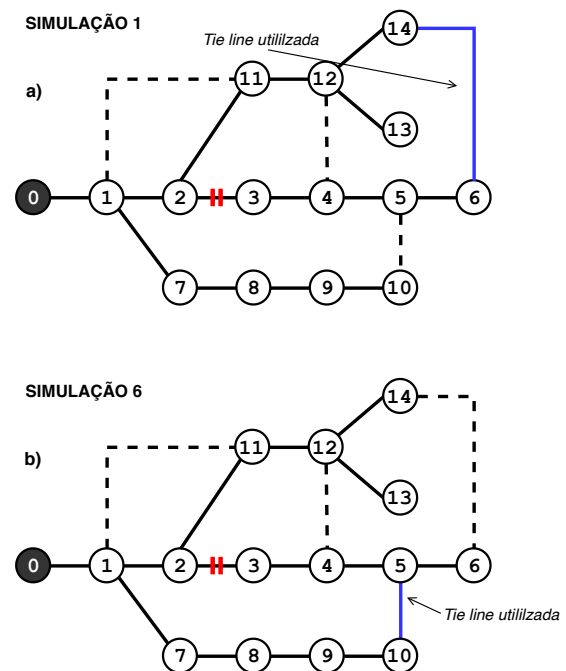


Figura 12. Topologia da rede *15-bus* durante as simulações 1 (a) e 6 (b).

1, o SMA escolhe, dessa vez, a linha (5,10). 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 6 foi maior do que o obtido na simulação 1, razão pela qual a linha (5,10) não é escolhida na simulação 1.

4.4 Simulações 11 e 13

No que diz respeito às simulações 11 e 13, ambas no modelo *33-bus*, os processos ocorreram de forma análoga. Primeiramente, na simulação 11, conforme mostra a Figura 13, o SMA detectou uma falha na linha (24,28) e utilizou a *tie line* (24,28) para realizar a autorrecuperação da rede na área afetada, com o valor de perdas elétricas sendo 134,7714 kW. Obviamente esse valor de perdas elétricas está relacionado ao número de cargas que foram restauradas na autorrecuperação, que é de 15.

Entretanto, na simulação 13, foi utilizado o mesmo ponto de corte na rede com uma alteração na resistência da linha (24,28), conforme descrito na Tabela 3. É possível observar que, ainda assim, na simulação 13, o SMA elege a linha (24,28) como a que apresenta menor valor de perdas elétricas e a utiliza novamente na autorrecuperação da área afetada, conforme mostra a mesma Figura 13. Esse fato, somado ao descrito na subseção anterior, evidencia a forma como o SMA faz a tomada de decisão sobre a *tie line* a ser utilizada na autorrecuperação. A tomada de decisão é feita em tempo de execução, utilizando parâmetros de estado da rede que estão sendo monitorados pelas visões particulares de cada agente em tempo real, efetuando, assim, o mapeamento da área afetada e

a tomada de decisão. Esse é um aspecto importante dos SMA e é usado a favor da abordagem apresentada neste artigo.

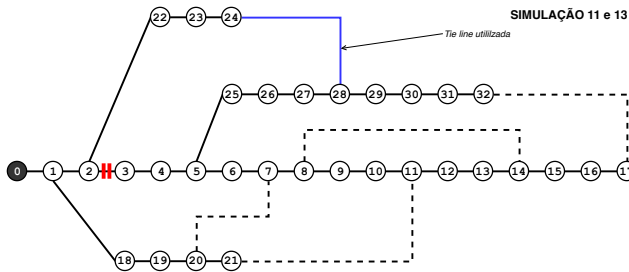


Figura 13. Topologia da rede 33-bus durante as simulações 11 e 13.

Em adição, é possível observar que, embora a mesma *tie line* (24,28) seja escolhida para recuperar a área afetada na simulação 13, os valores de perdas elétricas para essa simulação são mais altos que os apresentados na simulação 11.

4.5 Simulações 14 e 15

As simulações 14 e 15, ambas feitas sobre o modelo 33-bus, destacam uma outra característica observada durante as simulações realizadas com o SMA apresentado neste artigo. Essas duas simulações foram feitas em série. Em outras palavras, a partir da configuração inicial do modelo 33-bus iniciou-se a simulação 14 e, após o SMA ter terminado o processo de autorrecuperação, a configuração do modelo ficou: linha (5,25) aberta e a linha (24,28) fechada, fazendo parte da configuração ativa da rede, conforme mostra a Figura 14.

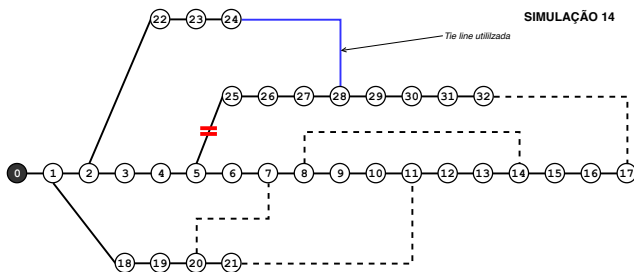


Figura 14. Topologia da rede 33-bus durante a simulação 14.

Sem finalizar o SMA, a linha (5,25) foi reposta ao modelo como uma *tie line* e, então, iniciou-se a simulação 15. Foi realizado o corte na linha (2,22) e o sistema iniciou novamente o algoritmo de autorrecuperação. Analisando as *tie lines* disponíveis, o sistema contou com as linhas (5,25) e (17,32) e elegeu a linha (17,32), com um valor de perdas elétricas em 107,6112 kW, conforme mostram a Tabela 2 e a Figura 15.

Essa simulação foi conduzida unicamente para verificar como o SMA iria lidar com uma alteração na rede depois de já haver ocorrido um processo de autorrecuperação. O resultado mostra que o sistema conseguiu acompanhar as mudanças que ocorreram na rede em tempo de execução, recuperando os nós de uma área afetada por uma nova falha na rede de

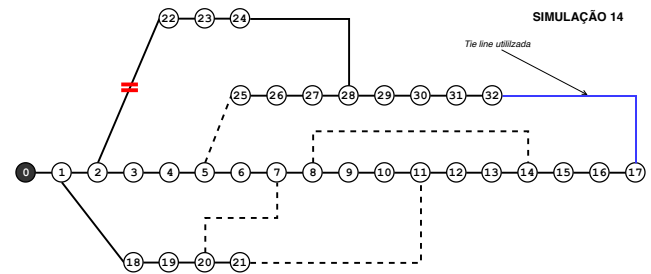


Figura 15. Topologia da rede 33-bus durante a simulação 15.

modo correto. Analisa-se que esse comportamento é natural ao SMA, já que podem fazer a leitura do ambiente onde estão imersos para, depois de troca coordenada de mensagens, tomarem decisões que caminhem rumo à satisfação de seus objetivos próprios e/ou coletivos. Assim, essa característica pode ser muito bem explorada em um ambiente altamente dinâmico, como em sistemas de distribuição.

Embora seja essa uma característica interessante, é significativo ressaltar que uma série de autorrecuperações sucessivas podem gerar configurações de redes inviáveis, isto é, que violem as restrições do *grid*, como níveis de tensão e sobrecarga de linhas, fatores que interferem na qualidade do serviço. Uma série de estudos podem ser aplicadas para verificar essas configurações, porém este esforço está fora do escopo deste trabalho.

5. Conclusões e trabalhos futuros

Este trabalho apresentou um algoritmo descentralizado, implementado por um SMA 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, bem como a sua implementação utilizando SMA. A pesquisa também apresentou resultados de 15 simulações, realizadas em ambiente computacional, para dois modelos de testes de rede de distribuição, um com 15 barras e outro com 33 barras, ambos com apenas uma fonte.

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

Na abordagem utilizada, nota-se que o SMA executa apenas duas operações de chaveamento. Isso foi possível porque a abordagem lida apenas com a restrição de radialidade da rede. Entretanto, a real contribuição dessa etapa do trabalho é permitir que a tomada de decisão seja feita de maneira virtual, por meio do cálculo de fluxo de potência. O agente ativo, em posse do mapeamento da rede, consegue simular os religamentos em todas as configurações possíveis e, depois de tomar uma decisão, realmente requisita as operações de chaveamento. Espera-se que, em abordagens futuras, este mapeamento auxilie a diminuir o número de chaveamentos nas abordagens de reconfiguração de redes. Esse tipo de operação

é custoso, portanto é de grande interesse sua minimização.

Conclui-se também que o SMA é determinante para que o algoritmo consiga acompanhar o dinamismo da rede. Quando a rede sofre alguma alteração, os agentes conseguem tomar diferentes decisões, dependendo do estado do ambiente em que estão imersos, no caso, a rede elétrica. Por isso é possível notar nas simulações realizadas as diferentes decisões do AA, em configurações diferentes da rede.

Como pontos sensíveis do sistema, nota-se que o método de eleição do agente ativo não é robusto contra atrasos que possam ocorrer na rede, já que se baseia em um limiar de tempo determinado por tentativa e erro. No entanto, os autores já trabalham num outro método de eleição de agentes ativos no sistema, baseando-o em uma outra técnica que não depende do tempo. Além deste, destaca-se que o método ainda lida com ponto de falha único, não podendo ser utilizado em ambientes com múltiplos pontos de falha.

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. Em adicional, espera-se estender a abordagem para lidar com cenários onde há múltiplos pontos de falha sem precisar aumentar em grande escala a complexidade da implementação da abordagem. 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.

Agradecimentos

Os autores agradecem ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo apoio financeiro ao desenvolvimento desta pesquisa. Para os autores, é claro o caráter fundamental que tem o incentivo financeiro de agências sérias de pesquisa para o desenvolvimento da ciência no Brasil.

Contribuição dos Autores

Italo Campos contribuiu para este trabalho com o planejamento e a execução do mapeamento da literatura, a estruturação da metodologia, a implementação do MAS, a execução do experimento e a análise dos resultados. Além disso, o pesquisador escreveu o texto do artigo e as correções. Filipe Saraiva contribuiu como orientador da pesquisa, além de atuar como revisor preliminar do texto. O pesquisador também atuou como especialista da área durante a avaliação da execução da metodologia e mapeamento da literatura.

References

[1] SARAIVA, F. d. O. *Aplicações híbridas entre sistemas multiagentes e técnicas de inteligência artificial para redes*

inteligentes de distribuição de energia elétrica. 154 p. Tese (Thesis) — Universidade de São Paulo, São Carlos, jun 2016. Disponível em: <http://www.teses.usp.br/teses/disponiveis/18/18154/tde-06062016-094659/>.

[2] LARIK, R. M.; MUSTAFA, M. W. Technologies Used in Smart Grid to Implement Power Distribution System. *TELKOMNIKA Indonesian Journal of Electrical Engineering*, v. 16, n. 2, p. 232–237, 2015. Disponível em: https://www.researchgate.net/publication/288416295_{_}Technologies_{_}Used_{_}in_{_}Smart_{_}Grid_{_}to_{_}Implement_{_}P.

[3] JIA, D.; MENG, X.; SONG, X. Study on technology system of self-healing control in smart distribution grid. In: *2011 International Conference on Advanced Power System Automation and Protection*. Beijing: IEEE, 2011. p. 26–30. Disponível em: <http://ieeexplore.ieee.org/document/6180379/>.

[4] SIANO, P. Demand response and smart grids—a survey. *Renewable and Sustainable Energy Reviews*, v. 30, p. 461–478, 2014. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1364032113007211>.

[5] MOTAMEDI, A.; ZAREIPOUR, H.; ROSEHART, W. D. Electricity Price and Demand Forecasting in Smart Grids. *IEEE Transactions on Smart Grid*, v. 3, n. 2, p. 664–674, jun 2012. Disponível em: <http://ieeexplore.ieee.org/document/6204245/>.

[6] RAHIMI, F.; IPAKCHI, A. Demand Response as a Market Resource Under the Smart Grid Paradigm. *IEEE Transactions on Smart Grid*, v. 1, n. 1, p. 82–88, jun 2010. Disponível em: <http://ieeexplore.ieee.org/document/5454394/>.

[7] NAGATA, T.; OKAMOTO, K. A decentralized distribution power system restoration by using multi-agent Approach. In: *2014 International Electrical Engineering Congress (iEECON)*. IEEE, 2014. p. 1–4. Disponível em: <http://ieeexplore.ieee.org/document/6925838/>.

[8] MISHRA, R. K.; SWARUP, K. S. Smart distribution network restoration using multi agent system. In: *2016 National Power Systems Conference (NPSC)*. Bhubaneswar, India: IEEE, 2016. p. 1–6. Disponível em: <http://ieeexplore.ieee.org/document/7858875/>.

[9] COULOURIS, G. *et al. Sistemas Distribuídos - 5ed: Conceitos e Projeto*. [S.l.]: Bookman Editora, 2013.

[10] FERREIRA, L. R. *et al.* 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: [s.n.], 2013. Disponível em: <http://www.sbai2013.ufc.br/pdfs/5997.pdf>.

[11] MAHDI, M.; GENÇ, V. M. I. A Real-Time Self-Healing Methodology Using Model- and Measurement-Based Islanding Algorithms. *IEEE Transactions on Smart Grid*, Institute of Electrical and Electronics Engineers Inc., v. 10, n. 2, p. 1195–1204, mar 2019. Disponível em: <https://ieeexplore.ieee.org/document/8060544/>.

- [12] Augusto de Souza, F. *et al.* A Multiagent Framework for Self-Healing Mechanisms Considering Priority-Based Load Shedding and Islanding with Distributed Generation in Smart Distribution Grids. *IEEE Latin America Transactions*, v. 15, n. 4, p. 632–638, apr 2017. Disponível em: <http://ieeexplore.ieee.org/document/7896348/>.
- [13] SHARMA, A.; SRINIVASAN, D.; TRIVEDI, A. A Decentralized Multi-Agent Approach for Service Restoration in Uncertain Environment. *IEEE Transactions on Smart Grid*, Institute of Electrical and Electronics Engineers Inc., v. 9, n. 4, p. 3394–3405, jul 2018. Disponível em: <https://ieeexplore.ieee.org/document/7752932/>.
- [14] NETWORKED Microgrids for Self-Healing Power Systems. *IEEE Transactions on Smart Grid*, Institute of Electrical and Electronics Engineers Inc., v. 7, n. 1, p. 310–319, jan 2016. Disponível em: <http://ieeexplore.ieee.org/document/7127033/>.
- [15] BARAN, M.; WU, F. Network reconfiguration in distribution systems for loss reduction and load balancing. *IEEE Transactions on Power Delivery*, v. 4, n. 2, p. 1401–1407, apr 1989. Disponível em: <http://ieeexplore.ieee.org/document/25627/>.
- [16] CAMPOS, I. *Aplicação de Sistemas Multiagentes ao Problema de Autorrecuperação em Sistemas Elétricos de Distribuição do tipo Smart Grid*. 96 p. Dissertação (TCC) — Faculdade de Computação, Universidade Federal do Pará, Belém, 2018.
- [17] BELLIFEMINE, F.; CAIRE, G.; GREENWOOD, D. *Developing multiagent systems with JADE*. Liverpool: Jhon Wiley & Sons, 2007.
- [18] WOOLDRIDGE, M. *An Introduction to MultiAgent Systems*. [S.l.]: Wiley, 2009.
- [19] REN, Y. *et al.* Agent-based restoration approach for reliability with load balancing on smart grids. *Applied Energy*, v. 249, p. 46–57, sep 2019. Disponível em: <https://linkinghub.elsevier.com/retrieve/pii/S0306261919307871>.
- [20] TORRES, B. S.; FERREIRA, L. R.; AOKI, A. R. Distributed Intelligent System for Self-Healing in Smart Grids. *IEEE Transactions on Power Delivery*, IEEE, v. 33, n. 5, p. 2394–2403, oct 2018. Disponível em: <https://ieeexplore.ieee.org/document/8375649/>.
- [21] KERSTING, W. *Distribution System Modeling and Analysis*. Boca Raton: CRC Press, 2012.
- [22] MEIER, A. von. *Electric Power Systems: A Conceptual Introduction*. [S.l.]: Wiley, 2006. (Wiley Survival Guides in Engineering and Science).
- [23] JADE. *Java Agent Development Framework (version 4.5.0)*. 2017. <http://jade.tilab.com/>.