# Resilience Assessment of Self-healing Systems with Redundancy

R. Ribeiro, M. Bessani, M. Fogliatto, and C. Maciel

Abstract—There is a need to analyse systems using complex networks modelling. In this study, we use this modelling to peform the resilience analysis of self-healing systems. These systems have the ability to change their topology to recover themselves from a failure or to adapt to new situations. Also, the resilience of a reference system from the REDS (REpository of Distribution Systems) repository and of some expansions of it with the addition of edges to the system using an Small-world topology method were analysed. An algorithm that simulates the reconfiguration of the system in case of failure was implemented. The analysis was performed for a variable number of edges' failures. The resilience investigation was done using the fraction of nodes still connected to an source after the system recovery over the total number of nodes in the system. A study of cost-benefit was also performed for one of the tested system expansions. The results show the benefits of adding redundancy to the system, which was expected, and show that the cost-benefit has a non-linear relation with the increase of redundancies in the system.

Index Terms-Resilience, Self-healing Systems, Complex Networks, Small-world.

## I. INTRODUÇÃO

N A ATUALIDADE, com o grande desenvolvimento tec-nológico que vem ocorrende nológico que vem ocorrendo, os sistemas modernos tem ficado cada vez mais complexos [1]. Uma abordagem para análise e projeto desse tipo de sistema é pelo uso de sistema de sistemas (SoS). SoS são sistemas que coordenam e conectam sistemas autônomos, formando um sistema maior que consegue completar tarefas que os sistemas individuais ou uma aglomeração não coordenada deles não conseguiria [2]. Considerando-se isto, observa-se a necessidade de análise utilizando-se de modelagem por redes complexas [3], pois este método modela conexões de forma genérica e diversa. Esta flexibilidade da modelagem de conexões faz com que este método seja capaz de integrar e unificar a relação entre estrutura e dinâmica [4]. Sendo assim, a análise por redes complexas tem sido aplicada para a modelagem de diversos sistemas do mundo real, como um sistema de transmissão epidêmica [5], computação em nuvem [6], o fluxo global de turismo [7], redes de distribuição de energia [3], em redes de comunicação ótica [8], entre outros [4].

Por sua importância prática, a robustez e resiliência de um sistema tem sido o foco de intensa pesquisa [1], sendo que a modelagem por redes complexas é comumente utilizada para se estudar a resiliência e robustez de sistemas complexos [3],

R. R. M. Ribeiro, M. S. S. Fogliatto and C. D. Maciel are with the Department of Electrical and Computing Engineering, São Carlos School of Engineering, University of São Paulo, São Carlos, SP, Brazil. site: http://www.sel.eesc.usp.br/lps/. e-mail: rafael.mendes.ribeiro@usp.br.

M. Bessani is with the Department of Electrical Engineering, Federal University of Minas Gerais, Belo Horizonte, MG, Brazil, mbessani@eee.ufmg.br.

[9]-[13]. Em alguns trabalhos ela é feita para se reduzir o impacto de tragédias [9] e em outros para avaliar ou reduzir a vulnerabilidade de sistemas elétricos à ataques e falhas [3], [10], [12], [13]. Sendo que, nesta segunda categoria alguns desses trabalhos estudam especificamente a resiliência de sistemas reconfiguráveis, definidos como sendo sistemas que têm a capacidade de alterar a sua topologia de forma automática ou não [3], garantindo que o sistema se recupere de falhas e/ou adapte-se a uma nova situação. Dentre esses trabalhos pode-se destacar o trabalho de Bessani et al. (2018), que utilizou um método multi-agente para reestabelecer as cargas após falhas em um sistema de distribuição de energia considerando diferentes níveis de capacidade nas fontes, e os de Ribeiro et al. (2018) e Quattrociocchi et al. (2014), que realizaram uma análise topológica da resiliência do sistema desconsiderando as cargas do sistema.

Considerando-se os trabalhos que estudam resiliência observa-se que a definição de resiliência é diferente em diversos trabalhos científicos [14]. Em [14] realizou-se um levantamento de diversos trabalhos e encontrou-se quatro definições principais para resiliência utilizadas na literatura. A primeira sendo como a capacidade do sistema de se recuperar de falhas, a segunda como sendo equivalente a robustez, a terceira como a capacidade de se adaptar a eventos que desafiam os limites do sistema e a quarta como a capacidade do sistema de continuar funcionando de forma correta mesmo com a expansão do sistema ao longo do tempo. A primeira definição é utilizada em estudos como o de [15], para o estudo da definição e medida da resiliência, de [16], que realiza estudos genéricos de resiliência, e o de [3], [10], [12] e [13], que fazem um estudo em áreas aplicadas, no caso na transmissão e distribuição de energia elétrica.

Apesar dos recentes trabalhos de [3], [13] e [17] que estudam a resiliência de sistemas considerando a sua reconfiguração, ainda é sugerido o estudo de resiliência para sistemas reais, como sugerido por [13]. Em [3] foi feito um estudo para este caso, mas utilizando uma abordagem de multiagentes diferente da abordagem utilizada e sugerida por [13]. Desta forma, neste trabalho foi feita a análise da resiliência de um sistema de distribuição de energia do repositório REDS (REpository of Distribution Systems) [18], com adição de arestas pelo método de Pequeno Mundo [19], utilizando-se da modelagem por redes complexas e considerando a resiliência definida como equivalente a robustez.

Na Seção II são descritos os softwares e hardware utilizados para as simulações e o procedimento da simulação. Na Seção III são apresentados os resultados obtidos e dados das simulações e na Seção IV são apresentadas as conclusões feitas a partir do exibido na Seção III.

# II. MATERIAIS E MÉTODOS

Neste estudo, realizou-se simulações para se analisar a resiliência do sistema do repositório **REDS** utilizando-se da representação por redes complexas. A simulação considerou falhas apenas em arestas, tanto as ativas quanto as de redundância. A adição de redundâncias ao sistema foi feita utilizando o método de *Pequeno Mundo*, permitindo-se assim testar a resiliência do sistema para diferentes níveis de redundância. Isto aplicado para um sistema do repositório **REDS** e expansões do mesmo, essas que serão explicadas em mais detalhes a seguir. Os subtópicos a seguir descrevem melhor as escolhas feitas e o procedimento da simulação.

#### A. Softwares Utilizados

A linguagem de programação **Julia** foi anunciada como um projeto de código aberto em fevereiro de 2012, sendo uma linguagem de programação dinâmica de alto nível para computação numérica [20]. A linguagem utiliza um compilador *just-in-time* extremamente eficiente que permite um desempenho que compete com a da linguagem C [20].

**Python** é uma linguagem de programação de alto nível, interpretada, de *script*, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte criada em 1991. Ela tem uma grande comunidade que oferece suporte. A escolha por sua utilização é devida ao fato de que ela é amplamente utilizada em aplicações científicas [21] e por ser *open-source*, logo não se precisando adquirir uma licença para a sua utilização.

A biblioteca **NetworkX** é uma biblioteca da linguagem de programação **Python** criada para habilitar e facilitar a criação, manipulação e estudo da estrutura, dinâmica, funcionamento de redes complexas [22]. Sua escolha se deve ao fato de ser *open-source* e contar com uma ampla comunidade que fornece documentação e suporte.

A biblioteca **Matplotlib** é uma biblioteca da linguagem de programação **Python** voltada para a criação de gráficos para o desenvolvimento de aplicações, *scripts* interativos e geração de imagens de qualidade para publicações para diversas interfaces de usuário e sistemas operacionais [23]. Ela foi escolhida por ser *open-source*, contar com uma ampla comunidade e por já ter funções implementadas na biblioteca **NetworkX** que a utilizam.

A biblioteca PyCall é uma biblioteca da linguagem de programação **Julia** que habilita a chamada direta e a interoperação completa com a linguagem de programação **Python** [24]. Assim, podendo-se utilizar funções da linguagem de programação **Python** na linguagem de programação **Julia**, incluindo a importação e utilização de bibliotecas como a **NetworkX** e **Matplotlib**. Logo, a bibliotecas **PyCall** foi escolhida para permitir a utilização das bibliotecas **NetworkX** e **Matplotlib** com a linguagem de programação **Julia** para o desenvolvimento do código.

# B. Descrição do Cluster

O cluster utilizado para a execução das simulações foi um Beowulf Cluster [25] composto por 13 máquinas, sendo uma máquina servidor e doze máquinas escravas. Todas as máquinas com **Julia** 0.6.0, **Python** 2.7.13, Matplolib 2.2.0, **NetworkX** 2.1, **Kernel** 4.6.6-300.fc24.x86 64 e *Ethernet Gigabit*. Doze das máquinas utilizam **Fedora** 24 *Server* atualizado em 16/08/2016 e duas máquinas utilizam **Fedora** 24 *Workstation* atualizado em 16/08/2016 e 17/08/2018. As máquinas contam com processadores i7-4770 CPU @ 3.49GHz com 8 núcleos e memória 8+8 GiB de RAM com velocidade de 1333MHz. O *cluster* tem um *switch* de 24 canais *Ethernet* 10 Gbit.

#### C. Repositório REDS

O repositório **REDS** (*REpository of Distribution Systems*) é um repositório de casos de teste aberto para comparação e obtenção de resultados de pesquisas em problemas como soluções para fluxo de energia, reconfiguração de redes, alocação de capacitores, balanceamento de carga, análise de contingência, entre outras [18].

#### D. Redes Complexas

A representação de sistemas por redes complexas utiliza-se de representações por grafo [4]. Um grafo é um par ordenado G = (V, E), formado por um conjunto não vazio de nós V e por um subconjunto E de pares não ordenados de V chamados de arestas [26]. Essa representação possibilita a modelagem de qualquer sistema que possa ser representado por interconexões entre seus elementos.

Sendo um grafo G = (V, E) com dois nós referenciados como fonte s e o dreno t, onde  $s \neq t$ , pode-se obter um caminho entre s e t utilizando-se do algoritmo de busca em largura (BFS) [27] a partir de s e t simultaneamente. Quando as duas buscas se encontram em um nó, o caminho entre t e s foi encontrado.

## E. Fração de Nós Conectados (FoS)

Para as simulações o valor utilizado para analisar a resiliência foi a fração de nós conectados a uma fonte após a recuperação da falha sobre o número total de nós consumidores (que não são fontes).

### F. Método Pequeno Mundo

O método *Pequeno Mundo* utilizado para adicionar arestas de redundância ao sistema é uma adaptação de um método de geração de um grafo de topologia *Pequeno Mundo* [19]. A topologia *Pequeno Mundo* é relacionada com a característica do grafo possuir alta eficiência, diâmetro e distância geodésica média pequena [19].

A adaptação do método de geração foi feita de forma que ao invés de se adicionar arestas em um grafo dos nós conectados ao seus vizinhos, esse grafo também pode ser chamado de anel, para criar um grafo de topologia *Pequeno Mundo*, se adiciona-se arestas pelo mesmo método anterior em um grafo que já tem conexões. Em específico a função que foi adaptada foi a *newman\_watts\_strogatz\_graph* da biblioteca **NetworkX**. Nela se tem um parâmetro *p* (probabilidade de se adicionar uma nova aresta para cada aresta) que controla o número de

arestas adicionadas ao sistema e que, por isso, foi adotado como sendo o valor r utilizado nesse trabalho para determinar o nível de redundância.

Desta forma, o método de *Pequeno Mundo* adiciona novas arestas de redundância ao grafo de forma de que para cada aresta (u, v) do grafo base seja sorteado um valor entre 0 e 1 e caso esse valor seja menor do que a probabilidade r de adição de redundância, uma nova aresta (u, w) é adicionada. Deve-se observar que na nova aresta (u, w), w é um nó do grafo escolhido aleatoriamente. Desta forma, adicionando novas arestas ao sistema da mesma forma que proposta em [28].

## G. Descrição da Simulação

Como esse estudo considera a reconfiguração do sistema na análise da resiliência, era necessário se ter um programa que simulasse a reconfiguração do sistema. Por isso, um programa que busca um caminho de conexão dos nós desconectados a fonte foi implementado. Para isso, o sistema foi considerado como sendo composto por três grafos, um grafo ativo do sistema ( $G_{ativo}$ ), um grafo de redundância ( $G_{redundância}$ ) e um grafo de arestas em falha ( $G_{falha}$ ), e, assim, a busca por um caminho de conexão da fonte aos nós em falha é feita no grafo que é a união do  $G_{ativo}$  e  $G_{redundância}$ .

Deve-se observar que o SoS é composto por redes de distribuição, onde cada rede possui diversos nós consumidores (carga) e nós fontes (geradores). Para simplificação, nesta análise o sistema foi considerado como tendo arestas de capacidade infinita, sem nenhum tipo de perdas e não direcionadas. Logo, a existência de um caminho entre um nó e uma fonte é suficiente para considerar que a sua demanda será suprida.

O procedimento da simulação é descrito a seguir:

- Leitura do sistema a partir do arquivo fornecido pelo repositório **REDS**;
- Adição de arestas de redundância para o nível de redundância especificado;
- 3) Simulação de uma falha em aresta pela transferência do elemento do  $G_{ativo}$  para o  $G_{falha}$ ;
- 4) Execução do algoritmo de reconfiguração, buscando reconectar os nós desconectados à fonte;
- Os processos 3 e 4 são executados até a desconexão total do sistema;
- Para toda a execução é medido um índice, que é a razão de nós conectados à fonte após a falha e a reconfiguração pelo número total de nós do sistema (FoS);

Na Figura 1 pode-se observar um exemplo gráfico do procedimento de falha e reconfiguração do sistema. Na Figura 2 pode-se observar a topologia do sistema *bus\_83\_11* do repositório **REDS** utilizado neste trabalho. Por se desejar analisar sistemas maiores, esta topologia foi triplicada e teve o número de nós fonte reduzidos, ligando-se os nós das replicações na fonte central, obtendo-se a topologia *bus\_276\_6* que pode ser observada na Figura 3. A topologia *bus\_276\_6* teve os seus nós fonte reduzidos novamente, deixando apenas uma fonte no sistema, obtendo-se o sistema *bus\_281\_1* que pode ser observada na Figura 4.



Fig. 1. Exemplo de falha e reconfiguração do sistema. Em (a) pode-se ver o sistema original e marcada pelo X a aresta selecionada para falha, a fonte é o nó quadrado de cor azul, os nós conectados a ela são da cor verde e representam nós carga, as linhas sólidas são conexões ativas e arestas tracejadas são conexões de redundância. Em (b) pode-se observar o estado do sistema após a falha com a aresta em falha e o nó desconectado do sistema após a reconfiguração, aonde o nó em falha volta a se conectar a fonte se conectando a aresta de redundância marcada pelo R.



Fig. 2. Topologia do sistema *bus\_83\_11* que foi utilizado, os nós consumidores podem ser vistos em verde, os nós fonte pode ser visto em quadrado na cor azul, conexões ativas são da cor amarela e as conexões de redundância são tracejadas na cor verde.

#### H. Análise de Custo-Benefício

Uma análise de custo-benefício foi feita para os resultados do sistema *bus\_281\_1*. Para isso, o custo da adição de uma redundância no sistema foi considerado como sendo proporcional a distância euclidiana. Assim sendo, tomamos que o custo de redundância de um sistema é dada por

$$Custo = \sum_{i} \sqrt{(Pos_{i1x} - Pos_{i2x})^2 + (Pos_{i1y} - Pos_{i2y})^2}$$

onde  $(Pos_{ijx}, Pos_{ijy})$  é a posição em duas dimensões dos nós que a aresta de redundância conecta e *i* itera sobre todas as arestas de redundância do sistema. Esse valor foi calculado para os níveis de redundância de 0.1, 0.3, 0.5, 0.7 e 0.9 do sistema *bus\_281\_1* e plotado com os seus respectivos valores de ganho de redundância em relação ao nível anterior, assim se obtendo uma curva de custo-benefício. Este valor é calculado pela diferença percentual para atingir o patamar de 50% do sistema conectado com o nível de redundância anterior.

#### III. RESULTADOS E DISCUSSÃO

Das simulações feitas se obteve figuras similares a Figura 5 obtida para o sistema *bus\_83\_11* que tem 83 nós consumidores e 11 nós fontes. Nela é possível ver a curva de resiliência,



Fig. 3. Topologia do sistema *bus\_276\_6* que foi utilizado, os nós consumidores podem ser vistos em verde, os nós fonte pode ser visto em quadrado na cor azul, conexões ativas são da cor amarela e as conexões de redundância são tracejadas na cor verde.

com a média marcada por ponto e desvio padrão marcado pelo sombreado para os diferentes níveis de redundância e a marcação visual dos níveis de 90%, 50% e 10% do sistema conectado. Todas as simulações foram repetidas 100 vezes para se obter as curvas dos resultados. O resumo de todos os resultados obtidos se encontram na Tabela I que mostra o número de falhas para o qual a média das simulações atingiu os patamares de 90%, 50% e 10% do sistema conectado e a diferença percentual entre níveis consecutivos de redundância no número de falhas para atingir o patamar de 50% do sistema conectado.

Em todos os resultados observa-se que há um aumento da resiliência para todos os aumentos de redundância do sistema por adição de arestas pelo método de *Pequeno Mundo*.

No geral das simulações se observa que o maior aumento de resiliência foi para os níveis de 0.3 e 0.5 de redundância, sendo que entre 0.0 e 0.1 e entre 0.5, 0.7 e 0.9 há pouca diferença em relação ao aumento de resiliência anterior. Este fato é reforçado pela medida da diferença percentual entre o número de falhas para o nível de 50% do sistema para níveis consecutivos de redundância que se encontra na Tabela I (Dif.) onde o menor valor é de 12.5% e o maior é de 22.22%.

Das simulações feitas também foi medido os tempos de execução, obtendo-se os dados da Tabela II, que pode ser vista a seguir. Da Tabela II é possível se observar que sempre há um aumento do tempo computacional com o aumento do nível de redundância, o que é esperado considerando que isto implica em um aumento no número de arestas, logo aumentando o número de falhas e reconfigurações a serem executadas. Além disso, pode-se observar que um aumento do número de nós do sistema também aumenta o tempo de



Fig. 4. Topologia do sistema *bus\_281\_1* que foi utilizado, os nós consumidores podem ser vistos em verde, os nós fonte pode ser visto em quadrado na cor azul, conexões ativas são da cor amarela e as conexões de redundância são tracejadas na cor verde.



Fig. 5. Resultado da simulação para o sistema de 83 nós consumidores e 11 nós fonte para falhas em arestas com adição de arestas pelo método de *Pequeno Mundo* para diversos níveis de redundância mostrados em diferentes cores. k é o número de falhas.

execução, como esperado. Também se percebe que o número de fontes do sistema é o fator que mais afetou o tempo de execução, sendo que um sistema pequeno com 11 fontes demorou aproximadamente 3.2 vezes mais para ser analisado do que um sistema 3 vezes maior que ele que contém somente uma fonte. Este aumento da demora provavelmente se deve a demora do algoritmo de reconexão ao trabalhar com diversas fontes.

Considerando-se a análise de custo-benefício do sistema *bus\_281\_1*, obteve-se a curva da Figura 6. Nela, é evidenciado

Sistema	r	90%	50%	10%	Dif. (%)
	0.0	3	15	55	-
bus_83_11	0.1	3	16	57	6.67
	0.3	3	18	66	12.50
	0.5	4	22	81	22.22
	0.7	4	25	-	13.64
	0.9	5	27	-	8.00
	0.0 6 41	155	-		
bus_276_6	0.1	7	45	164	9.76
	0.3	8	53	191	17.78
	0.5	10	62	223	16.98
	0.7	10	66	239	6.45
	0.9	11	74 266	266	12.12
	0.0 6 36 1	128	-		
bus_281_1	0.1	6	38	140	5.56
	0.3	7	44	159	15.79
	0.5	8	53	194	20.45
	0.7	9	58	205	9.43
	0.9	9	63	222	8.62

TABELA I Resultados Obtidos das Análises

TABELA II Tempos Computacionais das Simulações Feitas

Sistema	r	Tempos	Tempo total	
bus_83_11	0.0	13min 50s		
	0.1	13min 47s		
	0.3	14min 18s	1h 26min 34s	
	0.5	14min 35s		
	0.7	14min 56s		
	0.9	15min 4s		
bus_276_6	0.0	19min 57s		
	0.1	20min 29s		
	0.3	21min 18s	<b>2h</b> 11min <b>3</b> c	
	0.5	22min 31s	211 1111111 38	
	0.7	22min 54s		
	0.9	23min 51s		
bus_281_1	0.0	4min 0s		
	0.1	4min 5s		
	0.3	4min 22s	0h 27min 4s	
	0.5	4min 40s		
	0.7	4min 54s		
	0.9	5min 1s		

que o custo-benefício apresenta um máximo, indicando que existe um ponto ideal para investimento em redundância do sistema. Este ponto ideal para o sistema *bus\_281\_1* com adição de redundância pelo método de *Mundo Pequeno* é próximo ao nível de r = 0.5, logo merecendo uma investigação futura mais detalhada.

# IV. CONCLUSÃO

Considerando os resultados, conclui-se que o aumento da redundância é benéfico para o sistema, mas existem níveis de redundância que são mais críticos do que outros. Mais especificamente, observa-se que para adição de redundâncias



Fig. 6. Curva de custo-benefício para o sistema *bus\_281\_1*, onde o aumento do custo indica o aumento da redundância do sistema e Dif. é o mesmo definido na Tabela I.

no sistema pelo método de *Pequeno Mundo* os níveis de redundância críticos (que mais aumentam a resiliência do sistema em relação ao nível anterior) são 0.3 e 0.5. Logo sugerindo que, em caso de investimento no aumento de redundâncias em um sistema utilizando este método, o melhor custo-benefício será para os níveis de 0.3 a 0.5. Este melhor custo-benefício foi observado para o sistema *bus\_281\_1* como sendo próximo ao nível de 0.5, como foi sugerido pelos resultados anteriores.

Além disso, pela análise dos tempos computacionais das simulações feitas, é possível perceber que a análise pode ser feita em um tempo relativamente curto. Também percebe-se que esse tipo de análise pode ser feita para sistemas maiores, preferencialmente para sistemas com um número pequeno de fontes, tendo em vista que um pequeno aumento no número de fontes afeta muito mais o sistema do que um grande aumento no número de nós.

#### AGRADECIMENTOS

À Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) pelo apoio financeiro, processo nº 2017/21941-9, e ao apoio financeiro da CAPES – Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil

#### REFERÊNCIAS

- L. K. Gallos and N. H. Fefferman, "Simple and efficient self-healing strategy for damaged complex networks," *Physical Review E*, vol. 92, p. 052806, Nov. 2015.
- [2] B. Zhao, X. Wang, D. Lin, M. M. Calvin, J. C. Morgan, R. Qin, and C. Wang, "Energy management of multiple microgrids based on a system of systems architecture," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6410–6421, Nov 2018.
- [3] M. Bessani, R. R. M. Ribeiro, G. A. Pagani, M. Aiello, and C. D. Maciel, "Robustness of reconfigurable complex systems by a multi-agent simulation: Application on power distribution systems," in 2018 Annual IEEE International Systems Conference (SysCon), April 2018, pp. 1–6.
- [4] L. D. F. Costa, J. Oliveira, Osvaldo, G. Travieso, F. A. Rodrigues, P. Villas Boas, L. Antiqueira, M. P. Viana, and L. Correa Rocha, "Analyzing and modeling real-world phenomena with complex networks: a survey of applications," *Advances in Physics*, vol. 60, pp. 329–412, May 2011.

- [5] G. Zhu, G. Chen, and X. Fu, "Effects of active links on epidemic transmission over social networks," *Physica A: Statistical Mechanics* and its Applications, vol. 468, pp. 614 – 621, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378437116307518
- [6] G. Travieso, C. Ruggiero, O. Bruno, and L. da F. Costa, "A complex network approach to cloud computing," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2016, 04 2015.
- [7] S. Lozano and E. Gutiérrez, "A complex network analysis of global tourism flows," *International Journal of Tourism Research*, vol. 20, no. 5, pp. 588–604, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/jtr.2208
- [8] R. Choji Freitas, J. Ferreira Martins, C. Jose Albanez Bastos, H. Alves Pereira, D. Augusto Ribeiro Chaves, E. Santos Leitao, and R. Cesar Lira Silva, "Osnr-based restoration algorithm for optical network resilience to node failures," *IEEE Latin America Transactions*, vol. 10, no. 4, pp. 1893–1900, June 2012.
- [9] Disaster Resilience: A National Imperative. Washington, DC: The National Academies Press, 2012. [Online]. Available: https://www.nap.edu/catalog/13457/disaster-resilience-a-nationalimperative
- [10] M. Panteli and P. Mancarella, "The grid: Stronger, bigger, smarter?: Presenting a conceptual framework of power system resilience," *IEEE Power and Energy Magazine*, vol. 13, no. 3, pp. 58–66, May 2015.
- [11] I. Bachmann, P. Reyes, J. Bustos, and A. Silva, "Multidimensional network resilience analysis," *IEEE Latin America Transactions*, vol. 14, no. 6, pp. 2912–2914, June 2016.
- [12] Z. Bie, Y. Lin, G. Li, and F. Li, "Battling the extreme: A study on the power system resilience," *Proceedings of the IEEE*, vol. 105, no. 7, pp. 1253–1266, July 2017.
- [13] R. R. M. Ribeiro, M. Bessani, and M. D. Maciel, "Simulação e análise de resiliência em sistemas reconfiguráveis," in XXII Congresso Brasileiro de Automática, João Pessoa-PB, Brazil, 2018.
- [14] D. D. Woods, "Four concepts for resilience and the implications for the future of resilience engineering," *Reliability Engineering & System Safety*, vol. 141, pp. 5 – 9, 2015, special Issue on Resilience Engineering. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0951832015000848
- [15] S. Hosseini, K. Barker, and J. E. Ramirez-Marquez, "A review of definitions and measures of system resilience," *Reliability Engineering & System Safety*, vol. 145, pp. 47 – 61, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0951832015002483
- [16] Y. Zhang, R. Kang, R. Li, and C. Yang, "A comprehensive analysis method for system resilience considering plasticity," in 2016 International Conference on Industrial Engineering, Management Science and Application (ICIMSA), May 2016, pp. 1–4.
- [17] W. Quattrociocchi, G. Caldarelli, and A. Scala, "Self-healing networks: Redundancy and structure," *PLOS ONE*, vol. 9, no. 2, pp. 1–7, 02 2014. [Online]. Available: https://doi.org/10.1371/journal.pone.0087986
- [18] R. Kavasseri and C. Ababei, "REDS: REpository of Distribution Systems," Acessed on: 11 Nov. 2018. [Online]. Available: http://www.dejazzer.com/reds.html
- [19] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'smallworld'networks," *nature*, vol. 393, no. 6684, p. 440, 1998.
- [20] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A Fresh Approach to Numerical Computing," *ArXiv e-prints*, p. arXiv:1411.1607, Nov. 2014.
- [21] T. Oliphant, "Python for scientific computing," Computing in Science & Engineering, vol. 9, pp. 10–20, 06 2007.
- [22] A. Hagberg, P. Swart, and D. S Chult, "Exploring network structure, dynamics, and function using networkx," 01 2008.
- [23] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science Engineering*, vol. 9, no. 3, pp. 90–95, May 2007.
- [24] S. G. Johnson, "Calling Python functions from the Julia language," Acessed on: 22 Nov. 2018. [Online]. Available: https://github.com/JuliaPy/PyCall.jl
- [25] T. Sterling, *Beowulf cluster computing with Linux*. Cambridge: MIT Press, 2002.
- [26] A.-L. Barabási, R. Albert, and H. Jeong, "Mean field theory for scale-free random networks," *Physica A Statistical Mechanics and its Applications*, vol. 272, no. 1, pp. 173–187, Oct 1999.
- [27] C. Y. Lee, "An algorithm for path connections and its applications," *IRE Transactions on Electronic Computers*, vol. EC-10, no. 3, pp. 346–365, Sep. 1961.
- [28] M. Newman and D. Watts, "Renormalization group analysis of the small-world network model," *Physics Letters A*, vol. 263, no. 4, pp. 341 – 346, 1999. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0375960199007574



Rafael Rodrigues Mendes Ribeiro é graduado em Engenharia Elétrica pela Escola de Engenharia de São Carlos, da Universidade de São Paulo (EESC-USP). Atualmente é mestrando na área de Sistemas Dinâmicos, laboratório de Processamento de Sinais, na Escola de Engenharia de São Carlos, da Universidade de São Paulo (EESC-USP). Possui experiência na área de engenharia elétrica, com ênfase em resiliência de sistemas.



Michel Bessani possui graduação (2012), mestrado (2015) e doutorado (2018) em Engenharia Elétrica pela Escola de Engenharia de São Carlos, da Universidade de São Paulo (EESC-USP). Atualmente é Professor no Departamento de Engenharia Elétrica na Escola de Engenharia da Universidade Federal de Minas Gerais (DEE-UFMG). Suas áreas de interesse são: confiabilidade e resiliência de sistemas, modelagem estatística e inteligência computacional.



Matheus de Souza Sant'Anna Fogliatto possui graduação em Engenharia Elétrica pela Universidade Federal de Mato Grosso (2017/2). Atualmente é mestrando na área de Sistemas Dinâmicos, laboratório de Processamento de Sinais, na Escola de Engenharia de São Carlos, da Universidade de São Paulo (EESC-USP).



Carlos Dias Maciel possui graduação em Engenharia Eletrônica IME (1989) e doutorado em Engenharia Biomédica pela COPPE/UFRJ (2000). Atualmente é professor associado da USP (Universidade de São Paulo) e tem experiência na área de Engenharia Biomédica, com ênfase em processamento de sinais e instrumentação biomédica. Atua principalmente nos seguintes temas: processamento de sinais biomédicos, modelos probabilísticos (Redes Bayesianas Dinâmicas) e algoritmos computacionalmente intensos.