UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

**Uma abordagem tolerante a falhas para a previsão de
desastres naturais baseada em IoT e aprendizado de máquina**

Gustavo Antonio Furquim

Tese de Doutorado do Programa de Pós-Graduação em Ciências de
Computação e Matemática Computacional (PPG-CCMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Gustavo Antonio Furquim

Uma abordagem tolerante a falhas para a previsão de desastres naturais baseada em IoT e aprendizado de máquina

Tese apresentada ao Instituto de Ciências
Matemáticas e de Computação – ICMC-USP,
como parte dos requisitos para obtenção do título
de Doutor em Ciências – Ciências de Computação e
Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e
Matemática Computacional

Orientador: Prof. Dr. Jó Ueyama

USP – São Carlos
Outubro de 2017

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

F983a Furquim, Gustavo Antonio
Uma abordagem tolerante a falhas para a previsão
de desastres naturais baseada em IoT e aprendizado
de máquina / Gustavo Antonio Furquim; orientador Jô
Ueyama. -- São Carlos, 2017.
111 p.

Tese (Doutorado - Programa de Pós-Graduação em
Ciências de Computação e Matemática Computacional) --
Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo, 2017.

1. Previsão de Desastres Naturais. 2. Rede
de Sensores sem Fio. 3. Internet das Coisas. 4.
Aprendizado de Máquina. 5. Tolerância a Falha. I.
Ueyama, Jô, orient. II. Título.

Gustavo Antonio Furquim

**A tolerant-fault approach for natural disasters forecasts
based on IoT and machine learning**

Doctoral dissertation submitted to the Institute of
Mathematics and Computer Sciences – ICMC-USP, in
partial fulfillment of the requirements for the degree of
the Doctorate Program in Computer Science and
Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Prof. Dr. Jó Ueyama

**USP – São Carlos
October 2017**

*Para meu filho, Miguel Furquim Benincasa, você deu sentido e valor para todos os meus
erros e acertos anteriores ao seu nascimento.*

AGRADECIMENTOS

Ao meu Filho, Miguel Furquim Benincasa, por ter cedido seu tempo com o pai para que este trabalho fosse concluído. Você merece todo meu tempo, carinho e amor e, agora, vai tê-los de volta.

À Micheli de Oliveira Chicaroni, apesar de pouco tempo, que parece voar ao seu lado, você se tornou uma pessoa muito especial na minha vida. Obrigado por todo o suporte e compreensão. Você fez minha volta fazer sentido. Te amo!

Aos meus amigos e companheiros do Intermídia e LaSDPC, Geraldo, Mano, Heitor, Neto, Lucas, Felipe, Vinicius, Bruno, Humberto, Sandra, Raiza, Flor, Tiago, Marcio, Flávia, Rodrigo, entre outros. Pesquisar ao lado de vocês é bem mais interessante.

Special thanks to my coworkers and friends from ANTS laboratory at UOIT and from UOIT, Roozbeh, Saghar, Eduardo, Mauricio, Tommo, Thiago, Alex, Wellen, Caio, Mehran, Catherine, Lidya, ReZa, Eddy, Henry Julian, Aida, Hassan, Tasnim, Larry among others. The way you welcome and accept people is amazing. You created a beautiful and wonderful experience for a foreigner.

I want to thank UOIT from the opportunity given to me and for the great experience at a University abroad. Special thanks to my friends in Canada or the ones who returned to their countries, Bertram, Wenbo, Henry Chan, Jon, Edward (Tan Jun Qiu), Terry, Cindy, Benjamin, Marcus, Luiz Felipe, Alice, Angela, Gilberto, Juliene, Nathália, among others. I do not know what I would have done without all your support.

I also owe special thanks to Dr. Richard and Dr. Khalil by the help, attention, patience, and support given to me. Your teachings will follow me through life.

Aos funcionários da secretaria da Pós do ICMC. Vocês ficam bem na linha de frente entre professores e alunos, dando suporte para ambos os lados. Não imagino o tamanho do desafio que é estar em suas posições. Merecem todo o reconhecimento e respeito que, normalmente, são negligenciados.

Ao IFSP e seus funcionários do campus Araraquara (professores e administrativos). Muito obrigado pelo suporte financeiro, tempo desprendido e por todas as vezes que precisei de ajuda, a qual nunca me foi negada.

Ao Gustavo Pessin, por ter iniciado a ideia desta Tese e por, várias vezes, me dar condições de seguir os caminhos que eu queria para este projeto.

Ao Prof. Jó Ueyama, por sua orientação, contatos, oportunidades dadas a mim, pelas vezes em que me ouviu e por oferecer a oportunidade de aprender a lidar com novas e, muitas vezes, excelentes experiências.

À CAPES pelo suporte financeiro fornecido durante meu estágio no exterior pelo processo *nº* 3752/15-1.

“Não há necessidade de consultar um psicólogo para saber que quando você denigre o outro é porque você mesmo não consegue crescer e precisa que o outro seja rebaixado para você se sentir alguém.”
(Papa Francisco ([GARCIA, 2015](#)))

RESUMO

FURQUIM, G. A. **Uma abordagem tolerante a falhas para a previsão de desastres naturais baseada em IoT e aprendizado de máquina.** 2017. 111 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2017.

O aumento na quantidade e na intensidade de desastres naturais é um problema que está se agravando em todo o mundo. As consequências desses desastres são significativamente ampliadas quando ocorrem em regiões urbanas ou com atuação humana devido à perda de vidas e à quantidade de bens materiais afetados. O uso de redes de sensores sem fio para a coleta de dados e o uso de técnicas de aprendizado de máquina para a previsão de desastres naturais são opções viáveis, porém novas tendências tecnológicas têm se mostrado promissoras e podem agregar na tarefa de monitoramento de ambientes e na previsão de desastres naturais. Uma dessas tendências é adotar redes de sensores baseadas em IP e utilizar padrões emergentes para IoT. Nesse contexto, esta Tese propõe e analisa uma abordagem chamada SENDI (*System for dEtecting and forecasting Natural Disasters based on IoT*), um sistema tolerante a falhas baseado em IoT, WSN e AM para a detecção e a previsão de desastres naturais. O SENDI foi modelado empregando o ns-3 e validado utilizando dados coletados por uma WSN real instalada na cidade de São Carlos - Brasil, a qual realiza a coleta de dados de rios da região. Esse sistema também prevê a possibilidade de falhas na comunicação e a perda de nós durante a ocorrência de desastres, além de agregar inteligência aos nós para realizar a distribuição de dados e de previsões, mesmo nesses casos. Esta Tese também apresenta um estudo de caso sobre previsão de enchentes que utiliza a modelagem do sistema e os dados colhidos pela WSN. Os resultados dos experimentos mostram que o SENDI permite gerar alertas para a tomada de decisões em tempo hábil, realizando as previsões mesmo com falhas parciais no sistema, porém com acurácia variável dependendo do nível de degradação do mesmo.

Palavras-chave: Previsão de Desastres Naturais, Rede de Sensores sem Fio, Internet das Coisas, Aprendizado de Máquina, Tolerância a Falha.

ABSTRACT

FURQUIM, G. A. **A tolerant-fault approach for natural disasters forecasts based on IoT and machine learning**. 2017. 111 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2017.

Natural disasters have been increasing intensely all around the globe. The consequences of these disasters are significantly amplified when they occur in urban areas or places where there are human activities due to loss of lives and assets. The usage of Wireless Sensor Networks (WSN) for data collection and Machine Learning (ML) to create natural disasters forecast models are viable options. However, new technology trends have been showing promising results, which can aggregate to the tasks of environmental monitoring and natural disasters forecast. One of these new trends is to adopt IP based sensor networks and to use emergent *Internet of Things* (IoT) standards. In this context, this Thesis presents and analyzes an approach called SENDI (*System for dEtecting and forecasting Natural Disasters based on IoT*), a fault-tolerant system based on IoT, ML and WSN to detect and forecast natural disasters. SENDI was modelled using ns-3 and validated by means of real data collected by a WSN installed in São Carlos - Brazil, which collects the data of rivers around the region. This system also foresees the possibility of communication failures and loss of nodes during disasters, also adding intelligence to the nodes in order to perform the distribution of data and forecasts, even in such cases. This Thesis presents a case study about flash flooding forecast as well, which uses the system model and the data collected by the WSN. The results of the experiments show that SENDI allows to generate warnings in time to make decisions as such predictions can be foreseen even if partial failure of the system occurs. However, there is a variable accuracy, which depends on the system degradation.

Keywords: Natural Disasters Forecast, Wireless Sensor Network, Internet of Things, Machine Learning, Fault-tolerance.

LISTA DE ILUSTRAÇÕES

Figura 1 – Componentes de um nó sensor - Figura adaptada	32
Figura 2 – Exemplo de nó do sistema REDE.	35
Figura 3 – Posição dos sensores (pontos amarelos), pluviômetro (ponto azul claro - instalado juntamente com o roteador), estação base (ponto vermelho) e roteador (ponto azul claro) na cidade de São Carlos	36
Figura 4 – Imagem capturada pelo sensor 3.	37
Figura 5 – Relação entre redes de sensores e IoT.	39
Figura 6 – Exemplo de uma rede neural genérica.	44
Figura 7 – Exemplo de uma rede neural recorrente genérica.	45
Figura 8 – Exemplo de uma árvore de decisão.	46
Figura 9 – Exemplo de separação de 2 classes em R^2	47
Figura 10 – Exemplo de uma Rede Bayesiana.	48
Figura 11 – Arquitetura do sistemas SENDI vista como <i>tiers</i>	53
Figura 12 – Pilha de protocolos utilizada na simulação do sistema SENDI e imple- mentada em ns-3.	55
Figura 13 – Exemplo de instância RPL.	56
Figura 14 – Roteamento de um pacote até seu destino.	57
Figura 15 – Instalação de um novo nó e disseminação das informações necessárias para o protocolo de roteamento utilizado.	58
Figura 16 – Fluxograma de comportamento dos nós do <i>tier 2</i> - definição de <i>cloud</i> inativa.	59
Figura 17 – Fluxograma de comportamento dos nós do <i>tier 2</i> quando <i>cloud</i> está inacessível - distribuição dos dados e seleção do modelo de previsão.	59
Figura 18 – Fluxograma de comportamento dos nós do <i>tier 1</i>	60
Figura 19 – Modelo de dissipação de energia de rádio.	61
Figura 20 – Fluxograma representando os <i>rounds</i> de um nó sensor.	66
Figura 21 – Exemplo de leitura realizada pelos sensores da RSSF REDE.	67
Figura 22 – <i>Dataset</i> utilizado no estudo de caso sobre enchentes.	68
Figura 23 – Exemplo de dados coletados pelo sensor de pluviosidade - sistema REDE.	68
Figura 24 – Classes para a previsão.	74
Figura 25 – Exemplo de tamanho de janela (W) e tempos de previsão em uma série temporal fictícia de nível de água.	74
Figura 26 – Instâncias corretamente classificadas para $t + 1$	75

Figura 27 – Instâncias corretamente classificadas para $t + 2$	76
Figura 28 – Instâncias corretamente classificadas para $t + 3$	76
Figura 29 – AMI para a série temporal do nível do rio.	78
Figura 30 – % de vizinhos próximos falsos considerando.	79
Figura 31 – Nível esperado do rio e nível previsto para $\tau = 1$ com $m = 4$	80
Figura 32 – Nível esperado do rio e nível previsto para $\tau = 11$ com $m = 21$	80
Figura 33 – Nível do rio esperado e previsto durante a enchente.	84
Figura 34 – Representação da abordagem de previsão distribuída.	85
Figura 35 – Comparação gráfica entre as leituras reais e as previsões realizadas para o instante $t + 1$	87
Figura 36 – Comparação gráfica entre as leituras reais e as previsões realizadas para o instante $t + 2$	88
Figura 37 – Análise dos resultados obtidos pelo SENDI.	91
Figura 38 – Distribuição da pontuação segundo o critério de qualidade.	111

LISTA DE ALGORITMOS

Algoritmo 1 – Algoritmo de eleição de líder.	64
--	----

LISTA DE TABELAS

Tabela 1 – Tabela de comparação dos artigos relacionados.	72
Tabela 2 – Porcentagem de instâncias corretamente classificadas. Variações do tamanho da janela de dados (W) e do tempo de previsão do estado (E) do rio	75
Tabela 3 – Erro quadrático médio da MLP treinada com a série desdobrada . . .	79
Tabela 4 – Correlação e Coeficiente de correlação da MLP	79
Tabela 5 – Nível do rio e volume de chuva na enchente de 13/04/2014.	81
Tabela 6 – Nível do rio e volume de chuva MLP: MAE, RMSE e R^2	82
Tabela 7 – Correlação e Coeficiente de correlação da MLP	82
Tabela 8 – Nível do rio em abril de 2014 - MLP e RNR: MAE, RMSE e R^2 ($\tau = 1$ e $m = 4$).	83
Tabela 9 – Nível do rio e volume de chuva em abril de 2014 - RNN: MAE, RMSE and R^2	83
Tabela 10 – Performance das técnicas de AM para análise e uso na abordagem distribuída.	85
Tabela 11 – Resultado das técnicas de AM para a previsão distribuída em $t + 1$. . .	86
Tabela 12 – Resultado das técnicas de AM para a previsão distribuída em $t + 2$. . .	86
Tabela 13 – Parâmetros de simulação.	88
Tabela 14 – Número de nós ativos por <i>round</i>	89
Tabela 15 – Porcentagem de acertos e erros na previsão (alerta vermelho).	90
Tabela 16 – Porcentagem de acertos e erros na previsão (alerta amarelo).	90
Tabela 17 – Total de artigos retornado por cada base	110

LISTA DE ABREVIATURAS E SIGLAS

R^2	Coeficiente de Determinação
6LoWPAN	<i>IPv6 over Low power Wireless Personal Area Networks</i>
AM	Aprendizado de Máquina
AMI	Auto-Mutual Information technique
BRBES	<i>Belief Rule Based Expert System</i>
DAG	<i>Directed Acyclic Graph</i>
DfPL	Device-free Passive Localization
DODAG	<i>Destination Oriented DAGs</i>
EPC	<i>Electronic Product Code</i>
FNN	False Nearest Neighbors Method
GIS	sistemas de informação geográfica
GPS	sistemas de posicionamento global
IIS	<i>Integrated Information System</i>
IoT	<i>Internet of Things</i>
IP	<i>Internet Protocol</i>
ITU	<i>International Telecommunication Union</i>
LLNs	<i>Low-Power and Lossy Networks</i>
MAE	Mean Absolute Error
MCS	Mobile Crowd-Sensing
MiRe	<i>Middleware</i> , projetado e desenvolvido por Coelho (2014) e que provê confiabilidade nas informações em RFFS
MIT	<i>Massachusetts Institute of Technology</i>
ML	Machine Learning
NIBH	Núcleo Integrado de Bacias Hidrográficas
ns-3	<i>network simulator</i>
NTIC	Novas Tecnologias de Informação e Comunicação
REDE	<i>REde de sensores sem fio para Detectar Enchentes</i>
RFID	<i>Radio Frequency Identifier</i>
RMSE	Root Mean Square Error
RNR	Rede Neural Recursiva
RPL	<i>IPv6 Routing Protocol for Low-Power and Lossy Networks</i>

RSSF	Rede de Sensores Sem Fio
SaaS	<i>Software as a Service</i>
SENDI	<i>System for dEtecting and forecasting Natural Disasters based on IoT</i>
SNR	Signal-to-Noise Ratio
SOM	<i>Self-Organizing Map</i>
SVM	<i>Support Vector Machine</i>
TI	Tecnologia da Informação
WAN	<i>Wide Area Network</i>
WSN	Wireless Sensor Networks

SUMÁRIO

1	INTRODUÇÃO	25
1.1	Contextualização	25
1.2	Problema, Hipótese, Objetivo e Justificativa	27
1.3	Abordagem de Pesquisa	29
1.4	Organização da Tese	30
2	CONCEITOS DE RSSF E IOT	31
2.1	Conceitos de Redes de Sensores sem Fio	31
2.2	RSSF para a Detecção de Enchentes - REde de sensores sem fio para Detectar Enchentes (REDE)	34
2.3	Conceitos sobre Internet of Things	37
2.4	Considerações Finais	40
3	APRENDIZADO DE MÁQUINA PARA PREVISÃO DE ENCHEN- TES	41
3.1	Conceitos sobre Aprendizado de Máquina	42
3.1.1	<i>Redes Neurais Artificiais</i>	43
3.1.1.1	<i>Arquiteturas Recorrentes</i>	44
3.1.2	<i>Árvores de Decisão</i>	45
3.1.3	<i>Support Vector Machine - SVM</i>	46
3.1.4	<i>Rede Bayesiana</i>	47
3.2	Considerações Finais	48
4	TOLERÂNCIA A FALHAS EM PREVISÃO DE DESASTRES NA- TURAIS	51
4.1	Conceitos sobre Tolerância a Falhas	51
4.2	Arquitetura do Sistema SENDI	52
4.3	SENDI - Detalhes de Implementação	54
4.3.1	<i>Configuração da Pilha de Protocolos IoT nos Nós</i>	54
4.3.2	<i>Algoritmo de Roteamento</i>	55
4.3.3	<i>Comportamento dos Nós do Tier 2</i>	58
4.3.4	<i>Comportamento dos Nós do Tier 1</i>	60
4.3.5	<i>Protocolo de Eleição dos Nós Líderes</i>	61

4.3.6	<i>Abordagem de Tolerância a Falhas</i>	63
4.3.7	<i>Dados Utilizados no Estudo de Caso sobre Enchentes</i>	65
4.4	Trabalhos relacionados	69
4.5	Criação dos Modelos de Previsão, Análise e Resultados	73
4.5.1	<i>Combinando AM e RSSF para a Previsão de Enchentes</i>	73
4.5.2	<i>Análise de Dados Utilizando a Teoria do Caos</i>	75
4.5.3	<i>Previsões Utilizando Combinação de Diferentes Tipos de Sensores</i>	80
4.5.4	<i>Previsões Utilizando Combinação de Diferentes Sensores de Nível</i>	84
4.6	SENDI - Experimentos e Resultados	87
4.7	Considerações Finais	91
5	CONCLUSÃO	93
5.1	Síntese das Contribuições	93
5.2	Limitações e Trabalhos Futuros	94
5.3	Publicações e Trabalhos Desenvolvidos	95
REFERÊNCIAS		97
APÊNDICE A	DESCRIÇÃO DO PROCESSO DE REVISÃO SISTE- MÁTICA REALIZADO	107
A.1	Planejamento	107
A.2	Condução e Resultados	110

INTRODUÇÃO

1.1 Contextualização

Desastres naturais, como deslizamentos de terra, enchentes, inundações, incêndios, erupções vulcânicas e danos causados por esses eventos são problemas globais que acabam por ocasionar perdas financeiras e de vidas. Essa situação é agravada pelas mudanças das condições climáticas do planeta e é observada principalmente em regiões urbanas, como no estado de São Paulo, Brasil. Nessas regiões, a destruição do ecossistema é mais acentuada e acaba afetando o meio ambiente e modificando o clima local devido à poluição e à falta de planejamento. [Knox e Kundzewicz \(1997\)](#) afirmam que mesmo uma modesta mudança climática causa um impacto significativo na frequência e na magnitude de enchentes, por exemplo. Uma situação que ilustra essas mudanças é a criação de ilhas de calor, as quais acabam modificando o padrão de chuvas da região ([UEYAMA *et al.*, 2010](#); [MENDIONDO, 2008](#); [TJ](#); [SCOFIELD, 2002](#)).

[Arjun *et al.* \(2015\)](#) afirmam que o aquecimento global é a principal razão do aumento do número de desastres naturais e que o monitoramento do clima, através de sensores, é algo viável e necessário para realizar previsões e alertas em tempo hábil para a tomada de decisões sobre terremotos, deslizamentos de terras, enchentes, entre outros. No caso específico de enchentes, [Freitas e Ximenes \(2012\)](#) descrevem que todo ano cerca de 102 milhões de pessoas são afetadas por enchentes em todo o mundo e é esperado que esse número cresça nos próximos anos, pois as regiões que mais sofrem com enchentes são países em desenvolvimento e áreas urbanas. Essas são justamente as características da cidade de São Carlos - SP, Brasil, onde o clima tem mudado nos últimos anos, tornando o ambiente mais propício para desastres naturais, segundo [Barbosa e Vecchia \(2009\)](#), [Pozza \(2005\)](#).

[Schulz *et al.* \(2008\)](#) reafirmam o que foi dito por [Freitas e Ximenes \(2012\)](#) e acres-

centam que as técnicas atuais não diminuem significativamente os danos causados por enchentes. Oprea e Matei (2010) também confirmam esse fato e dizem que, na prática, muitos desastres naturais, como as enchentes, não podem ser evitados, porém seus efeitos podem ser mitigados e administrados, desde que previstas com antecedência.

Assim, redes de sensores sem fio são uma opção vantajosa para realizar o monitoramento de ambientes naturais, já que apresentam as seguintes características: (i) baixo custo, principalmente em relação à infraestrutura, (ii) acessibilidade em ambientes inóspitos, (iii) simples desenvolvimento e implantação, (iv) possibilidade de utilização de sensores de alta precisão, além da (v) adaptabilidade a mudanças ambientais, como afirmado por Seal *et al.* (2012). Especificamente na cidade de São Carlos - SP, Brasil, encontra-se instalada uma rede de sensores sem fio chamada *REde de sensores sem fio para Detectar Enchentes* (REDE), desenvolvida pelo Instituto de Ciências Matemáticas e de Computação (ICMC) - Universidade de São Paulo (USP) e descrita nos trabalhos de Ueyama *et al.* (2010) e Hughes *et al.* (2011), e essa faz parte do projeto e-NOE, que visa realizar o monitoramento de rios urbanos.

Hughes *et al.* (2011) afirmam que até o momento o sistema REDE só é capaz de detectar enchentes quando elas já iniciaram. O sistema, no entanto, deveria prever as enchentes e gerar alertas a tempo de possibilitar que a população tome medidas preventivas e evacue as áreas de risco. Isso pode ajudar a reduzir vários problemas decorrentes de enchentes.

Entretanto, convém salientar que a alta heterogeneidade nas tecnologias utilizadas por Rede de Sensores Sem Fio (RSSF) já em uso, acaba por dificultar o desenvolvimento e a integração de RFFS a novas tecnologias e a RFFS já existentes, devido à quantidade de soluções proprietárias e não-proprietárias. Uma das possíveis soluções para esse impasse é seguir a tendência atual e adotar redes de sensores baseadas em IP e utilizar padrões emergentes como o 6LoWPAN/IPv6 (MAINETTI; PATRONO; VILEI, 2011). A utilização desses padrões viabiliza a comunicação entre RFFS e a Internet e possibilita enxergar os sensores como *smart objects*, o que interliga o mundo físico a sistemas online (ZELENKAUSKAITE *et al.*, 2012).

Em um cenário de previsão e de gerenciamento de desastres naturais, os dados coletados pelos sensores podem ser analisados a partir de outras informações obtidas via Internet, como imagens e previsões via satélite ou valores de variáveis aos quais os sensores não teriam acesso. Além disso, um cenário no qual os sensores estão conectados à Internet viabiliza o uso de tecnologia como *cloud computing* e redes sociais para realizar previsões e gerar alertas online sobre os ambientes monitorados.

No contexto deste projeto, a visão de IoT permite estender o sistema REDE e trazer o conceito de acessibilidade de dispositivos ou de serviços remotos, que auxiliam a distribuir conhecimento, tomar decisões ou atuar no mundo real, através do sensoramento

e da geração de informações. Logo, os nós de uma RSSF (como o sistema REDE) podem ser considerados *smart objects*, desde que preencham alguns requisitos como a coleta de dados, a utilização desses dados via Internet e a inteligência coletiva entre todos os nós e não somente sobre a situação e os dados de um único nó. Dessa forma, podemos dizer que o gerenciamento e a previsão de desastres naturais são cenários em que a integração com o mundo real acontece de forma espontânea e o emprego de RSSF para essa finalidade se torna um caminho natural (MAINETTI; PATRONO; VILEI, 2011; ZELENSKAUSKAITE *et al.*, 2012).

O próximo passo para a evolução do sistema REDE é que ele possa prever enchentes e ser estendido para outros tipos de desastre naturais. Essas previsões devem ser realizadas mesmo se houver uma falha em parte do sistema, visto que o sistema REDE ambiciona ambientes inóspitos e necessita, portanto, de mecanismos de tolerância a falhas. Outra grande possibilidade de avanço é a integração do sistema REDE a tecnologias emergentes, como os padrões vindos do IoT, já que um dos principais objetivos deste trabalho é explorar as áreas acima citadas.

Portanto, este projeto traz 3 contribuições principais: (i) a utilização e a avaliação de novos padrões e tecnologias emergentes provenientes da IoT juntamente com RSSF para a coleta de dados em ambientes naturais e a sua distribuição, (ii) a utilização dos dados coletados para a previsão de desastres naturais a partir de técnicas de Aprendizado de Máquina (AM), com um estudo de caso sobre as características de rios e de chuvas na cidade de São Carlos – SP, Brasil e, (iii) a proposta de uma arquitetura tolerante a falhas baseada em IoT e em AM para o sistema REDE, a qual compreenderia os modelos de previsão para a detecção de desastres naturais de forma autônoma, mesmo quando parte do sistema estivesse comprometida, e a geração de alertas para a população local.

Nas seções seguintes, encontram-se a contextualização e o problema apontado, além da hipótese, do objetivo e da justificativa do trabalho (Seção 1.2).

1.2 Problema, Hipótese, Objetivo e Justificativa

O **problema** principal desta Tese relaciona-se aos danos causados por desastres naturais, principalmente em áreas urbanas, como na cidade de São Carlos - SP, Brasil, e à falta de mecanismos que possam prever esses desastres e mitigar os danos causados. É necessário utilizar dados coletados por sistemas similares ao REDE para prever os desastres e gerar alertas para a população de regiões de risco. Além disso, novas tecnologias emergentes, como padrões e tecnologias voltadas à IoT, podem ser utilizadas juntamente com RSSF para facilitar a aquisição, a distribuição e o processamento dos dados coletados e aumentar a acurácia das previsões. Uma vez que sistemas de previsão de desastres naturais normalmente se encontram em áreas de risco e devem funcionar principalmente

durante a ocorrência de um desastre, mecanismos de tolerância a falhas são altamente desejados. Sendo assim, a implementação de mecanismos de tolerância a falhas nesses sistemas é outro ponto chave deste trabalho.

Muitos trabalhos têm sido realizados nessa área, porém ainda é necessário aumentar a acurácia das previsões, principalmente em situações extremas nas quais parte do sistema é comprometido. Essa necessidade é ainda mais aparente quando o foco é a combinação de IoT, RSSF, técnicas de AM e mecanismos de tolerância a falhas a fim de realizar tais previsões.

Sendo assim, o **objetivo** principal deste trabalho é apresentar uma abordagem que melhore a acurácia de previsões de desastres naturais e a confiabilidade e a disponibilidade de tais previsões. A fim de prever tais desastres, são utilizadas técnicas de AM, cujos modelos gerados são embutidos nos nós do sistema, o que possibilita que esses nós sejam autônomos ao realizar as previsões e que se reorganizem, mesmo em situações em que existam falhas na comunicação ou no *hardware* do sistema. Para tanto, os objetivos específicos desta Tese são: (i) avaliar a utilização de técnicas de AM, sobre dados coletados por RSSF, para prever o comportamento de ambientes naturais, como rios; (ii) analisar a combinação de sensores, posicionados em regiões geograficamente diferentes, e de seus dados, para aumentar a acurácia dessas previsões e distribuir a carga de processamento necessária para realizar essas previsões; (iii) combinar e analisar tecnologias de IoT e RSSF para monitoramento de ambientes propensos à desastres naturais; (iv) identificar técnicas de tolerância a falhas que possam ser utilizadas para a previsão de desastres naturais, aumentando a confiabilidade e a disponibilidade dessas previsões; (v) avaliar o sistema de monitoramento e previsão de desastres naturais, sendo proposto com base no conhecimento adquirido nos itens anteriores.

Dessa forma, a **hipótese** desta tese é que a combinação de RSSF com tecnologias IoT e técnicas de AM pode gerar sistemas de grande acurácia, confiabilidade e disponibilidade e que, mesmo em momentos de falha de parte do sistema, ele possa manter previsões e alertas para a população.

A **justificativa** principal deste projeto é que métodos computacionais voltados à previsão são cada dia mais utilizados em várias áreas que necessitam primordialmente do conhecimento humano, tais como: a inadimplência bancária, a construção de ambientes inteligentes, a vida útil de geradores elétricos, a confiabilidade e a necessidades de manutenção de sistemas industriais e novos problemas computacionais, como a alocação de recursos em *grids* computacionais (AKHLAGHINIA; LOTFI; LANGENSIEPEN, 2007; BARCENA *et al.*, 2010; DABHI; PRAJAPATI, 2008; RUIHUA *et al.*, 2006; SALGADO; CAMINHAS; MENEZES, 2009).

A utilização de RSSF para previsão de desastres naturais é outro ponto que justifica pesquisas na área e está muito presente na literatura. Como exemplo, temos a RSSF

utilizada nesse projeto e apresentada em Ueyama *et al.* (2010) e Seal *et al.* (2012), onde é proposta uma RSSF combinada com regressão linear robusta para realizar a previsão de enchentes. Além disso, rede de sensores desempenha um papel crucial na arquitetura IoT, assim, a arquitetura da rede de sensores afeta diretamente a arquitetura IoT e, consequentemente, o sistema e as funções para os quais foi desenvolvida (PERERA *et al.*, 2014). Além disso, o fato da IoT ser uma evolução natural de RSSF (MAINETTI; PATRONO; VILEI, 2011) também justifica um estudo sobre a utilização de RSSF e IoT a fim de conseguir uma arquitetura tolerante a falhas para a previsão de desastres naturais.

Outro fator que deve ser abordado é a grande complexidade na análise de dados coletados em ambientes naturais que, muitas vezes, necessita de intervenção humana para tratamento. Essa intervenção acaba gerando imprecisão e falta de confiabilidade nos resultados. No entanto, humanos possuem grande capacidade de fazer generalizações e encontrar padrões e, em computação, essa habilidade pode ser simulada para realizar previsões de desastres naturais de maneira mais precisa e com grande confiabilidade e disponibilidade. Tal simulação se dá através de técnicas de AM empregadas juntamente com dados coletados por RSSF e tecnologias voltadas à IoT.

A influência do par $\langle \text{dado_coletado}, \text{local_do_sensor} \rangle$ em cada ponto de coleta do sistema REDE é outra justificativa deste trabalho. Em outras palavras, o modo que a leitura de um sensor x , localizado em uma região $r0$, influencia a leitura de um sensor y , localizado em uma região $r1$ à frente de $r0$, considerando o fluxo do rio. Além disso, o tempo para que essa influência ocorra e como tecnologias voltadas à IoT podem ajudar na distribuição dos dados, na padronização dos meios de comunicação e na disponibilização do poder de processamento e da capacidade de armazenamento necessários para a manipulação dos dados.

1.3 Abordagem de Pesquisa

Para alcançar o objetivo desta tese, desenvolveu-se um sistema tolerante a falhas chamado *System for dEctecting and forecasting Natural Disasters based on IoT* (SENDI). Esse sistema é baseado em RSSF, IoT e AM e ainda proporciona flexibilidade para se adaptar a diversos ambientes e situações, ao passo que os mecanismos de tolerância a falhas garantem que as previsões serão realizadas, mesmo com diferentes níveis de acurácia. O sistema SENDI foi modelado e testado a partir do *network simulator* (ns-3) (RILEY; HENDERSON, 2010) e um estudo de casos sobre enchentes foi realizado a partir dos dados coletados pelo sistema REDE sobre o modelo. Mais detalhes sobre os procedimentos metodológicos, a realização da modelagem e a análise do SENDI são apresentados no Capítulo 4.

A fim de investigar os mecanismos utilizados para a previsão de desastres natu-

rais presentes na literatura que utilizam ou não as tecnologias presentes no SENDI, foi realizada uma revisão sistemática segundo os protocolos descritos no trabalho de [Pisani e Lorena \(2013\)](#). Tais protocolos devem ser seguidos afim de garantir uma revisão completa e passível de reprodução. Os detalhes da revisão sistemática são apresentados no Anexo A.

A revisão sistemática indica que, apesar de ser uma área muito explorada, soluções que visam à previsão de desastres naturais e que fazem uso das tecnologias e das características combinadas do sistema SENDI ainda necessitam de estudos. Assim, o SENDI busca contribuir com uma abordagem tolerante a falhas e que possibilita a previsão de desastres naturais mesmo em ambientes inóspitos, a partir de tecnologias e de padrões encontrados na literatura.

1.4 Organização da Tese

Esta tese de doutorado está dividida nos seguintes capítulos.

O Capítulo 1 contextualiza o projeto de pesquisa e apresenta o problema, a hipótese, o objetivo e a justificativa a ele relacionados.

No Capítulo 2, são apresentados os principais conceitos e teorias relacionadas à RSSF e à IoT, utilizadas no desenvolvimento do SENDI.

O Capítulo 3 aborda conceitos e teorias fundamentais em AM e necessários para o entendimento do SENDI.

O Capítulo 4 apresenta princípios fundamentais de tolerância a falhas, a proposta desta tese, detalhes do funcionamento, da arquitetura e da modelagem do SENDI, assim como os mecanismos de tolerância a falhas incorporados ao sistema. Também são apresentados trabalhos relacionados à utilização de IoT e de tolerância a falhas para a previsão ou o gerenciamento de desastres naturais. Além disso, é descrito como foram criados os modelos de previsão de enchentes e como ocorreu a validação, os experimentos e os resultados obtidos por esses modelos e pelo SENDI. Esses experimentos foram realizados com os dados do estudo de casos sobre previsão de enchentes, coletados pelo sistema REDE nos rios da sub-bacia do rio Monjolinho na cidade de São Carlos - SP, Brasil.

E, finalmente, o Capítulo 5 apresenta as conclusões, as principais contribuições, as publicações decorrentes do estudo, suas limitações e propostas de trabalhos futuros.

CONCEITOS DE RSSF E IOT

2.1 Conceitos de Redes de Sensores sem Fio

Neste capítulo, são apresentados os fundamentos de RSSF e IoT, assim como uma descrição do sistema REDE, a RSSF utilizada para coletar os dados utilizados no estudo de caso sobre previsão de enchentes deste Tese.

Seal *et al.* (2012) descrevem RSSF como sistemas cooperativos de baixo custo e de baixo consumo de energia e que normalmente utilizam comunicação *multihop*, sendo independentes de serviços externos. Essas redes são constituídas por nós que, por sua vez, consistem em uma interface sem fio, dispositivos de sensoriamento, bateria, processador e memória.

Segundo esses estudos (SEAL *et al.*, 2012; UHEYAMA *et al.*, 2010; HUGHES *et al.*, 2011), os principais objetivos de uma RSSF são: (i) o monitoramento de fenômenos físicos e (ii) a transmissão de dados coletados sobre esses fenômenos. Uma das características dessas redes é a possibilidade de qualquer nó atuar como roteador, agregador ou na aquisição de dados, o que possibilita a utilização de várias topologias. A modificação da topologia e das rotas de transmissão em qualquer instante maximiza a redundância e aumenta a resiliência das RSSF, proporcionando uma melhora na confiabilidade do sistema.

Quando pensamos em distribuir modelos de previsão de desastres naturais em nós de uma RSSF, a questão da resiliência é muito vantajosa, tanto nas questões voltadas para a comunicação entre os nós quanto nas voltadas para a falha ou destruição de algum dos nós. Porém, essa mesma resiliência é um ponto preocupante para o desenvolvimento deste projeto já que, para que o sistema REDE tenha resiliência também na questão de previsão de enchentes e continue operacional mesmo em caso de falhas (como na ocorrência de perda de nós, falhas na comunicação ou quando o servidor estiver indisponível) a distribuição dos modelos deve garantir que as previsões continuem a serem realizadas e que tenham

boa acurácia. Esse ponto é importante, já que os nós do sistema REDE, assim como outras RFFS utilizadas para o monitoramento de ambientes naturais, estão em áreas de risco e podem ser levados ou destruídos durante enchentes, deslizamentos ou explosões.

Na Figura 1, ilustram-se os componentes básicos de um nó de uma RSSF, segundo Akyildiz *et al.* (2002). Os componentes principais são representados por linhas contínuas e os componentes adicionais, por linhas pontilhadas. Entre os componentes principais, temos: (i) a unidade de sensoriamento, constituída por um ou mais sensores e por um conversor analógico/digital (ADC), (ii) a unidade de processamento, constituída por um processador e por unidades de armazenamento de dados, (iii) um *transceiver*, responsável pela comunicação com o mundo exterior e (iv) uma unidade de energia, responsável por fornecer energia aos demais componentes.

Figura 1 – Componentes de um nó sensor - Figura adaptada

Fonte: Akyildiz *et al.* (2002).

Redes de sensores podem ser aplicadas para vários fins, desde o monitoramento de saúde ou uso em atividades militares e domésticas e, em cada um dos casos, apresentam desafios diferentes (AKYILDIZ *et al.*, 2002). De forma geral, os desafios relacionados a RSSF, consistem em: (i) realizar a instalação e os experimentos em ambientes reais, ao invés de utilizar simulações, (ii) necessitar de alta tolerância a falhas, visto que os sensores são normalmente instalados em áreas de risco, (iii) se adaptar a mudanças ambientais em tempo de execução, (iv) reduzir o consumo de bateria, visto que os nós das RFFS operam normalmente em regiões sem infra-estrutura elétrica e (v) assimilar técnicas e métodos de engenharia de *software*, a fim de facilitar a construção e a manutenção de software em sistemas de RSSF.

Derr e Manic (2015) descrevem que abordagens comuns de classificação de RSSF, são:

- Centralizadas x Distribuídas, onde a comunicação deve ser roteada necessariamente por um nó central ou não;
- Determinística x Não-determinística, onde a rota de comunicação é ou não determinística;

- Uma camada x Duas camadas, onde existem várias camadas de nós por onde a informação é roteada.

[Hughes et al. \(2011\)](#) ainda apresentam uma RSSF, centralizada, determinística e de uma camada para o monitoramento de rios chamada REDE. Apesar de realizar avaliações sobre o consumo de energia, o suporte a sensores, o custo de desenvolvimento e a ocupação da memória, o sistema REDE precisa ser avaliado quanto à confiabilidade e à rapidez dos avisos sobre enchentes e poluição, o que foi listado entre os trabalho futuro. Além disso, também como trabalho futuro, [Hughes et al. \(2011\)](#) pretendem gerar esses avisos com confiabilidade e disponibilidade para a evolução do sistema REDE.

Essa avaliação é difícil de ser realizada devido a três fatores: (i) o conflito existente entre o fornecimento de alertas a tempo de se tomar alguma atitude e a confiabilidade dos mesmos, (ii) a necessidade de muitos dados experimentais para dar suporte à avaliação, e (iii) a necessidade de coletar dados durante um longo período de tempo a fim de captar enchentes.

Esta tese tem como objetivo principal, dentre os citados por [Hughes et al. \(2011\)](#), o primeiro item, ou seja, utilizar AM sobre os dados coletados pelo sistema REDE a fim de gerar previsões confiáveis a tempo de fornecer alertas de enchentes. Além disso, este trabalho estende os itens citados por [Hughes et al. \(2011\)](#) para o sistema REDE e acrescenta a utilização de novas tendências tecnológicas, como IoT e RFFS na criação de uma arquitetura mais genérica para prever desastres naturais e tolerante a falhas ao realizar essas previsões.

Em ambientes inteligentes é comum a utilização de redes de sensores, em conjunto com técnicas de AM, para previsões e para a adequação do ambiente às necessidades e gostos dos ocupantes. [Akhlaghinia, Lotfi e Langensiepen \(2007\)](#) afirmam que vários fatores influenciam diretamente nas características de redes de sensores e em suas previsões, como: (i) a topologia de uma rede de sensores, (ii) a utilização de abordagens centralizadas ou distribuídas para o tratamento de dados e tomada de decisões, (iii) além de técnicas de tratamento das informações coletadas pelos sensores.

[Akhlaghinia, Lotfi e Langensiepen \(2007\)](#) afirmam que RFFS que seguem abordagens distribuídas utilizam uma combinação do poder de processamento dos nós sensores na manipulação dos dados coletados. Já em abordagens centralizadas, todos os dados coletados são armazenados e manipulados em uma base central. Existem também abordagens mistas, nas quais, por exemplo, alguns nós intermediários podem realizar o processamento parcial dos dados ou implementar algoritmos completos e, nesses casos, a verificação dos resultados, a redundância e/ou os cálculos finais são responsabilidade da base central.

Embora seja possível aumentar a capacidade de processamento computacional utilizando uma abordagem distribuída, isso nem sempre é uma alternativa viável para RFFS,

devido à necessidade de economia de energia. Já o ganho obtido na utilização de técnicas de tratamento de dados nos nós de uma RSSF e a consequente redução da quantidade de dados transmitidos são tópicos que devem ser analisados. Dentre as técnicas que podem ser utilizadas para o tratamento de dados encontram-se a compressão, a regressão e a *fuzzy*, as quais possibilitam a minimização da quantidade de informação transmitida pelo nó com o encargo do uso do poder de processamento do mesmo (AKHLAGHINIA; LOTFI; LANGENSIEPEN, 2007).

RFFS também são aplicadas no monitoramento de fenômenos físicos em ambientes não controlados. Em Seal *et al.* (2012), foi proposta uma RSSF para monitorar múltiplos parâmetros, como a temperatura e a pluviosidade em rios. A partir da coleta desses dados, utilizou-se regressão linear robusta para prever a possibilidade de enchentes de rios e alertar a comunidade local.

Em Ueyama *et al.* (2010), apresentou-se uma plataforma de *hardware* e *software* utilizada para o monitoramento de rios no estado de São Paulo, tendo em vista os requisitos específicos de tais rios. Na área de engenharia de *software*, ocorreu o primeiro exemplo prático de abordagem de programação multiparadigma utilizando RFFS.

Ueyama *et al.* (2010) também afirma que um dos principais pontos no qual são desprendidos a maioria dos esforços para o estudo de enchentes é o monitoramento de rios. Este projeto tem como principal questão um dos trabalhos futuros propostos em Ueyama *et al.* (2010), no qual técnicas de AM são utilizadas para realizar previsões confiáveis e em tempo para o fornecimento de alertas.

2.2 RSSF para a Detecção de Enchentes - REde de sensores sem fio para Detectar Enchentes (REDE)

RFFS são sistemas de baixo custo e consumo de energia, constituídos por nós formados basicamente por processador, memória, bateria, interface de rede sem fio e dispositivo(s) de sensoriamento. Como mencionado no Capítulo 1, essas redes apresentam características vantajosas, o que as torna fortes candidatas a monitorar ambientes naturais, como a coleta de dados para prever enchentes (SEAL *et al.*, 2012; FAIÇAL *et al.*, 2014; UYAMA *et al.*, 2010).

A RSSF REDE, desenvolvida pelo Instituto de Ciências Matemáticas e de Computação (ICMC) da Universidade de São Paulo (USP), e descrita no trabalho de Ueyama *et al.* (2010), realiza unicamente a *detecção de enchentes* e necessita de mecanismos para prever tais eventos. Considerando a composição dos nós de uma RSSF, cada nó do sistema REDE é formado por: (i) um microcontrolador Atmega328 (16MHz), (ii) memória flash 32KB, (iii) memória SRAM 2KB, (iv) memória EEPROM 1KB, (v) interface de rede

(Padrão ZigBee e/ou 3G), (vi) timers, (vii) conversores A/D, (viii) entrada serial para sensores (resistivos (i.e. temperatura), capacitivos (i.e. umidade) e sensores com interface serial (i.e. pHkit da Atlas Scientific)), (ix) baterias e (x) painel de energia solar.

Para este projeto, as características mais importantes do sistema REDE são: (i) a flexibilidade de sensores que o sistema permite utilizar, visto que, o uso da entrada serial amplia a gama de sensores a serem utilizados e (ii) a independência existente nos nós da rede, fator agregado pela captação de energia solar e o uso de bateria, que permite que os nós continuem funcionando por dias, mesmo com tempo nublado. Um nó do sistema REDE é ilustrado na figura 2.

O sistema REDE está instalado na bacia hidrográfica do rio Monjolinho, que abrange uma área de aproximadamente 275 KM², dos quais a maior área encontra-se no município de São Carlos - SP e uma pequena parte no município de Ibaté. Sua localização está entre os paralelos 21° 57' e 22° 06' de latitude sul e os meridianos 47° 50' e 48° 05' de longitude oeste. Essa é uma sub-bacia da bacia hidrográfica do Jacaré-Guaçú que, por sua vez, é uma sub-bacia da bacia hidrográfica do Tietê-Jacaré, um dos principais afluentes da margem direita do Tietê (DORNELLES, 2006).

Figura 2 – Exemplo de nó do sistema REDE.

Fonte: Adaptada de Furquim *et al.* (2017).

Esses sensores estão dispostos na cidade de São Carlos como ilustra a figura 3. Os pontos amarelos representam os sensores sem fio (sensor 1, sensor 2 e sensor 3), o ponto azul claro representa a localização do roteador e do pluviômetro (instalados no mesmo local) e o ponto vermelho identifica a localização da base central de coleta de dados. Os sensores 1 e 2 utilizam o padrão ZigBee para comunicação, fazem a leitura da pressão da água no fundo do rio e convertem esse valor em altura (nível) do rio. Já o sensor 3 (localizado na região do shopping da cidade) utiliza 3G e, além de realizar a leitura da altura do rio, também pode enviar fotos de sua situação atual. A base de coleta de

dados centraliza e armazena os dados enviados pelos três sensores, enquanto o roteador os redireciona, o que permite aumentar o alcance da área onde os sensores podem ser posicionados.

Figura 3 – Posição dos sensores (pontos amarelos), pluviômetro (ponto azul claro - instalado juntamente com o roteador), estação base (ponto vermelho) e roteador (ponto azul claro) na cidade de São Carlos

Fonte: [Furquim et al. \(2014b\)](#).

A figura 4 exemplifica uma imagem capturada pela câmera do sensor do shopping. Essas imagens podem, futuramente, ser combinadas com outros trabalhos do grupo de pesquisa sobre RFFS no ICMC – USP coordenado pelo Prof. Dr. Jó Ueyama, para a análise do estado dos rios.

Atualmente, o sistema REDE conta com um componente extra de software chamado *Middleware*, projetado e desenvolvido por [Coelho \(2014\)](#) e que provê confiabilidade nas informações em RFFS (MiRe). O MiRe é um *middleware* baseado no OpenCom, cuja função é analisar os dados coletados pelos sensores do sistema REDE. Essa análise visa a detectar erros de leitura e a aplicar técnicas de recuperação sobre esses dados. Assim, os dados são repassados de forma transparente para as aplicações, o que provê maior confiabilidade nas leituras e na tolerância a falhas.

Este projeto se beneficia desse tratamento das leituras dos sensores, já que, por

Figura 4 – Imagem capturada pelo sensor 3.

Fonte: [Furquim et al. \(2014b\)](#).

se tratar de dados colhidos em ambientes naturais, os ruídos normalmente presentes são minimizados. Considerando que algumas técnicas de AM podem ser prejudicadas por ruídos presentes no conjunto de treinamento, esse tratamento melhora a acurácia dos modelos de previsão gerados por tais técnicas.

O tratamento e a análise dos dados coletados pelo sistema REDE são de extrema importância para este projeto, pois só assim é possível verificar a situação atual dos rios da bacia hidrográfica, prever seu comportamento futuro e gerar alertas de forma rápida e precisa, com o intuito de diminuir os danos materiais e humanos causados por desastres naturais.

2.3 Conceitos sobre Internet of Things

Duas grandes evoluções ocorreram na área de tecnologia da informação que tiveram grande impacto em como cidades e áreas urbanas são gerenciadas: (i) a *Internet of Things* e a comunicação móvel. IoT é um novo conceito na área de telecomunicações e o termo foi introduzido por Kevin Ashton em 1999 enquanto trabalhava no Auto-ID lab ([ASHTON, 2009](#)). O conceito foi, então, explicado pelo *Massachusetts Institute of Technology* (MIT) *Auto-ID Center* e interligado com *Radio Frequency Identifier* (RFID) e *Electronic Product Code* (EPC) in 2001 ([BROCK, 2001](#)). Depois disso, a IoT foi introduzida pela *International Telecommunication Union* (ITU) em 2005 ([STRATEGY; UNIT, 2005](#)). A ideia por trás desse paradigma é a existência ubíqua de objetos ou coisas como tags, RFIDs, sensores e telefones móveis entre nós e o nosso ambiente ([ATZORI; IERA; MORABITO, 2010](#)). O objetivo principal da IoT é conectar objetos que podem agir de forma independente e inteligente e se comunicar uns com os outros com o mínimo de interferência humana. Semanticamente, IoT significa coisas e objetos interconectados,

individualmente endereçados e identificados e que interagem por protocolos padrões de comunicação (ATZORI; IERA; MORABITO, 2010).

O nome *Internet of Things*, ou Internet das Coisas, é composto por duas palavras: Internet e “*things*”, o que introduz uma visão orientada à Internet e a coisas (ATZORI; IERA; MORABITO, 2010). A simples definição de IoT vem dessas duas perspectivas, que consideram que as “*things*” são itens muito simples, como RFID tags, e capazes de se conectar e comunicar uns com os outros (ATZORI; IERA; MORABITO, 2010). A visão orientada à Internet relaciona-se a redes e comunicações e ao modo como essas “*things*” e objetos utilizam diferentes protocolos de rede para se comunicarem, enquanto a visão voltada a “*things*” se concentra nos objetos e em como eles podem ser combinados em um *framework* comum (ATZORI; IERA; MORABITO, 2010; GUBBI *et al.*, 2013). Na visão orientada à rede da IoT, varias alianças e conselhos tentam adaptar a IoT à tecnologia *Internet Protocol* (IP). Como exemplo, temos o *IPv6 over Low power Wireless Personal Area Networks* (6LoWPAN) e o Internet0, que seguiram uma abordagem de diminuição da complexidade da pilha do protocolo IP a fim de obter um protocolo desenvolvido para rotear IP sobre quaisquer “*things*”. Diferenças entre definições de IoT, muitas vezes, provêm de diferentes abordagens e aspectos selecionados por pesquisadores, indústrias e alianças de empresas. A combinação entre Internet e “*things*” revela uma terceira visão, conhecida como versão semântica. A ideia principal por trás dessa versão é que o número de objetos da Internet vai crescer dramaticamente no futuro e, portanto, organizar essa quantidade de “*things*” é um dos desafios em IoT e a versão semântica tem um papel importante na resolução do mesmo (ATZORI; IERA; MORABITO, 2010).

Devido à expansão e ao desenvolvimento da Tecnologia da Informação (TI), a quantidade de dados tem aumentado em escala em várias áreas. O montante de dados produzidos a cada dia no cenário mundial está crescendo firmemente como consequência do aumento do uso de Novas Tecnologias de Informação e Comunicação (NTIC), sensores e outros dispositivos.

Com o rápido avanço da IoT e da computação em nuvem (ARMBRUST *et al.*, 2010), existe o potencial de desenvolver aplicações para o monitoramento e a mineração de dados em vários tipos de ambientes, incluindo os naturais e através da utilização de vários objetos. Existem também alguns desafios a serem enfrentados, como a maneira de coletar e de analisar a grande quantidade de dados provenientes de diferentes fontes de informação, com diferentes tipos de tecnologia e a maneira de entregar esses dados para as aplicações de forma apropriada, considerando os requerimentos de cada aplicação.

O forte gerenciamento de áreas urbanas ou populadas e o desenvolvimento de cidades depende do monitoramento adequado à tomada de decisões. Uma abordagem possível é elevar a distribuição de sensores, como RFFS, para obter as condições encontradas no mundo real (GUO *et al.*, 2014). Cada objeto pode gerar uma grande quantidade de infor-

mação e compartilhá-las com outros objetos que, por sua vez, em conjunto, formam a IoT. A base da maioria das aplicações e serviços IoT é a disseminação e o compartilhamento de informações entre as partes envolvidas. O desafio é que a maioria das arquiteturas IoT atuais foram desenvolvidas para lidar com comunicação centrada no *host* e isso não suporta a comunicação centrada em dados de forma eficiente (YUE *et al.*, 2014).

RFFS é uma das tecnologias cruciais que afetam a realização da IoT. “RFFS são feitas por sensores distribuídos autônomos que controlam e verificam condições físicas e ambientais”. Esses sensores são equipados com transmissores sem fio que possibilitam a comunicação uns com os outros (YICK; MUKHERJEE; GHOSAL, 2008). Avanços na tecnologia de sensores e na arquitetura de microprocessadores auxiliam na criação de sensores mais duráveis e de menor tamanho para RFFS e podem acelerar o desenvolvimento da IoT.

A fim de projetar arquiteturas para IoT e para ambientes inteligentes, devemos considerar as características e as limitações dos objetos em IoT, como: a capacidade de troca de mensagens, energia, conectividade e poder de processamento. Redes de sensores desempenham papéis cruciais na arquitetura de sistemas IoT. Desta forma, a arquitetura de redes de sensores afeta diretamente a arquitetura IoT e, conseqüentemente, a arquitetura de ambientes e de cidades inteligentes. Sendo assim, existe uma forte conexão entre RFFS e IoT, como ilustra a figura 5.

Figura 5 – Relação entre redes de sensores e IoT.

Fonte: Perera *et al.* (2014).

As redes de sensores consistem de três arquiteturas principais basicamente: arquitetura plana, arquitetura de duas camadas e arquitetura de três camadas (PERERA *et al.*, 2014). Em arquiteturas planas, os sensores são conectados entre si por *multihop* e os dados são transferidos para os nós de concentração. Na arquitetura de duas camadas, existe mais de um nó de concentração móvel capaz de coletar dados e de enviá-los a níveis superiores da arquitetura. E a arquitetura de três camadas faz uso da Internet ou de algum tipo de *Wide Area Network* (WAN) para conectar múltiplas redes de sensores

umas com as outras (PERERA *et al.*, 2014). A IoT segue a arquitetura de três camadas na sua camada de sensoriamento a fim de superar a maioria dos seus desafios, como a escalabilidade e a heterogeneidade.

Devido ao grande número de sensores e à quantidade de informações produzidas por eles em IoT, a consciência de contexto desempenha um papel crucial e decide quais dados devem ser processados e qual o nível de consciência existente nos sistemas. Os dados coletados pelos sistemas não apresentam nenhum valor a não ser que sejam processados e analisados, tornando-se compreensíveis (PERERA *et al.*, 2014). Assim, a computação voltada para a consciência de contexto nos permite salvar as informações ligadas aos dados dos sensores e facilitar a sua análise.

2.4 Considerações Finais

Neste capítulo, foram apresentados conceitos fundamentais sobre RFFS e IoT, assim como apresentado a RSSF REDE, suas características e limitações, sendo que esses conceitos serviram como base para o desenvolvimento desta Tese. Também foram comentados algumas melhorias à serem feitas no sistema REDE e no uso de RSSF para monitoramento de ambientes e previsão de desastres naturais. Uma dessas possíveis melhorias é a utilização de RFFS em conjunto com técnicas de AM e com tecnologias de IoT para a previsão de desastres naturais de forma tolerante a falhas, as quais têm grande probabilidade de ocorrerem, já que esse tipo de sistema encontra-se em ambientes hostis. Este trabalho prevê que essas falhas podem acontecer, principalmente em casos em que ocorra alguma falha na comunicação ou em algum nó da rede, e apresenta mecanismos que permitem continuar com as previsões e as tomadas de decisões sobre a geração de alertas, mesmo que essas sejam feitas por um único nó isolado dos demais ou com quantidade limitada de informações.

APRENDIZADO DE MÁQUINA PARA PREVISÃO DE ENCHENTES

O Capítulo 2.1 demonstra que a aquisição de dados em ambientes naturais por meio de RFFS é algo comum e vantajoso. Porém, esses dados ainda necessitam ser tratados de forma eficiente para que previsões com boa acurácia possam ser geradas. Assim, técnicas de AM desempenham um papel de destaque nos dias de hoje, como afirma [Camps-Valls \(2009\)](#), sendo que tais técnicas podem ser encontradas em vários trabalhos, como [See e Openshaw \(1999\)](#), [Boukharouba et al. \(2013\)](#), [Laio et al. \(2003\)](#), [Toth, Brath e Montanari \(2000\)](#), [Tiwari e Chatterjee \(2010\)](#), [Toukourou, Johannet e Dreyfus \(2009\)](#), [Brath, Montanari e Toth \(2002\)](#), [Nayak et al. \(2005\)](#), [Kar et al. \(2012\)](#), [Kant et al. \(2013\)](#), [Mitra et al. \(2016\)](#), [Arjun et al. \(2015\)](#), [Hossain et al. \(2014\)](#), [Deak et al. \(2013\)](#).

Já que o estudo de caso desta tese é relacionado à previsão de enchentes, é importante fazer uma comparação entre modelos hidrológicos e o uso de técnicas de AM para tal finalidade.

[Kar et al. \(2012\)](#) afirmam que os modelos de previsão de enchentes podem ser divididos em duas categorias: (i) modelos físicos, nos quais se enquadram os modelos hidrológicos, e (ii) modelos *black-box*, também chamados de orientados a dados (*data-driven*), nos quais se enquadram as técnicas de AM.

Segundo [Yu et al. \(2014\)](#), os modelos hidrológicos são importantes para o gerenciamento de recursos hídricos e de previsão de enchentes, porém sua implementação e sua calibração são mais complexas devido à necessidade de um conhecimento preciso sobre o processo hidrológico e variáveis físicas da região. Muitas vezes, os dados necessários para o bom desempenho desses modelos são difíceis de se obter ou não estão disponíveis em tempo real, como é o caso da evaporação, da umidade do solo e da temperatura ([KAR et al., 2012](#)).

Kar *et al.* (2012) Yu *et al.* (2014) destacam ainda que modelos orientados a dados são mais simples de serem obtidos. Isso se deve ao fato de que podem extrair a relação entre a entrada e a saída de dados das observações hidrológicas e meteorológicas, mesmo sem pleno conhecimento das características físicas do sistema.

Esta seção apresentada, então, conceitos fundamentais sobre AM, base para o desenvolvimento deste projeto, além do restante dos artigos selecionados durante a revisão bibliográfica.

3.1 Conceitos sobre Aprendizado de Máquina

Mitchell (1997) define “aprender” como “melhorar automaticamente com a experiência”. Então, para que uma máquina aprenda é necessário que, através de um “conjunto de experiências”, ela possa “melhorar”. O conteúdo desse “conjunto de experiências” e o conceito de “melhorar” varia conforme a situação. Como exemplo, Mitchell (1997) cita que, para a medicina, esse conjunto de experiências seria composto de registros médicos utilizados por técnicas de AM a fim de determinar um melhor tratamento para novas doenças. Já, em outras áreas, o conjunto poderia ser composto pelo histórico de uso de energia em uma casa, com o propósito de aprender a economizar. De maneira geral, Mitchell (1997) afirma que:

Definição 1. Um programa **aprende** a partir de experiências E , com respeito a alguma classe de tarefas T e medida de desempenho P , se o desempenho de tarefas em T , medido por P , melhora com experiências E .

No caso deste projeto, um dos objetivos é tentar melhorar a acurácia de previsões de enchentes através de AM. Ou seja, melhorar a tarefa (T) de prever enchentes, com base em uma medida de desempenho P , a partir de dados históricos sobre o comportamento de rios (E), coletados com RFFS.

Assim, esta seção apresenta os conceitos fundamentais sobre AM para que se entenda o projeto, os trabalhos relacionados e os resultados obtidos.

Considerando um conjunto utilizado para aprender um conceito, os elementos desse conjunto são chamados de *instâncias* (x), e o conjunto de todas as instâncias é denominado X (conjunto de instâncias). O conceito ou função que queremos aprender é conhecido por *conceito alvo* (*target concept*) ou c , sendo que $c(x)$ é a classe da instância x e não faz parte de X (MITCHELL, 1997).

O aprendizado ocorre quando um conjunto de exemplos de treinamento é passado para a técnica de AM a fim de estimar c . Quando esses exemplos são formados pelo par ordenado $\langle x, c(x) \rangle$, a técnica é dita *supervisionada*. Quando os exemplos são formados

somente por x , a técnica é dita *não-supervisionada* (LUXBURG; SCHÖLKOPF, 1997; MITCHELL, 1997).

Problemas comuns quando lidamos com AM são o *overfitting* e o *underfitting*. Como mencionado em Silva, Spatti e Flauzino (2010), no caso de *overfitting*, a técnica de AM memoriza os dados durante o treinamento e alcança bons resultados nessa fase. Porém, perde a capacidade de generalização frente a outros conjuntos de dados, como o de teste ou conjuntos formados por novas instâncias.

Em contrapartida, Silva, Spatti e Flauzino (2010) descrevem que o *underfitting* não consegue extrair as características do comportamento do problema e apresenta resultados ruins tanto durante o treinamento quanto com novos conjuntos de teste. Isso ocorre quando, por exemplo, uma rede neural artificial tem poucos neurônios em sua camada escondida (Seção 3.1.1) ou quando o treinamento não ocorre de forma adequada.

Como afirmam Laio *et al.* (2003), pode-se evitar o *overfitting* através do método *early stopped*. Tal método consiste em seguir algumas regras que determinam quando o treinamento da técnica de AM terminará e é descrito mais detalhadamente em Yao, Rosasco e Caponnetto (2007).

O restante desta seção descreve algumas técnicas de AM utilizadas nesta tese ou necessárias para o entendimento de trabalhos relacionados. Vale ressaltar que, pela facilidade de serem integradas aos nós de uma RSSF, pelos resultados dos experimentos realizados e pela sua facilidade de implementação, as RNAs parecem ser bem promissoras.

3.1.1 Redes Neurais Artificiais

Silva, Spatti e Flauzino (2010) classificam as redes neurais artificiais como modelos computacionais inspirados no sistema biológico de aprendizado do sistema nervoso. O neurônio artificial foi construído através de modelos simplificados dos neurônios reais e consiste na unidade de processamento da rede. Silva, Spatti e Flauzino (2010) dizem que a arquitetura de redes neurais é definida pelo modo como os neurônios são organizados com relação aos demais e tal relação é descrita através de conexões sinápticas entre os neurônios artificiais. Essa técnica apresenta várias aplicações, dentre as quais podem ser citadas as previsões e a classificação de padrões de forma robusta, temas deste trabalho.

Outro ponto fundamental sobre redes neurais artificiais, apresentado em Berardi, Patuwo e Hu (2004), é a sua capacidade de aproximar o resultado de qualquer função de forma precisa, convergindo valores para o resultado esperado, o que as torna estimadores consistentes. Essa convergência pode ser obtida mesmo quando os dados apresentam ruídos. Essa característica é interessante já que descarta parte do trabalho realizado no pré-processamento para eliminação desses ruídos, visto que, na previsão e monitoramento de ambientes naturais, existem ruídos naturais nos dados coletados.

Dentre as redes neurais artificiais com maior versatilidade e aplicabilidade temos a MLP, descrita em [Silva, Spatti e Flauzino \(2010\)](#) como sendo composta por: (i) um ou vários neurônios de entrada, os quais formam uma camada correspondente aos parâmetros recebidos pela rede; (ii) um conjunto de neurônios de saída (podendo ser unitário), correspondente aos valores retornados pela rede; (iii) pelo menos uma camada escondida entre a camada de entrada e a de saída, sendo que a complexidade do problema é que determina a quantidade de camadas escondidas e o número de neurônios dessas camadas. Essa organização é ilustrada na Figura 6, onde temos um único neurônio de saída e uma única camada escondida que é, normalmente, a quantidade suficiente para resolver problemas de previsão, como descrito em [Oprea e Matei \(2010\)](#).

Segundo [Silva, Spatti e Flauzino \(2010\)](#), a MLP apresenta uma arquitetura *feed-forward*, na qual o fluxo de informações ocorre sempre em uma única direção (da camada de entrada para a de saída) e o seu aprendizado é realizado, normalmente, com o algoritmo de treinamento *backpropagation*, descrito mais detalhadamente em [Mitchell \(1997\)](#).

Figura 6 – Exemplo de uma rede neural genérica.

Fonte: Elaborada pelo autor.

Outra arquitetura de redes neurais artificiais são os mapas auto-organizáveis de Kohonen (em inglês, *Self-Organizing Map* (SOM)). Essa estrutura de RNA utiliza um processo de aprendizado competitivo, que ocorre de forma não-supervisionada e encontra-se detalhada em [Silva, Spatti e Flauzino \(2010\)](#).

3.1.1.1 Arquiteturas Recorrentes

Existem outros tipos de redes neurais artificiais e cada uma delas apresenta características próprias que melhoram a sua performance em áreas e problemas específicos. Como exemplo, [Silva, Spatti e Flauzino \(2010\)](#) e [Mitchell \(1997\)](#) citam as redes neurais artificiais com arquitetura recorrente, onde as saídas de neurônios em um tempo t são

utilizadas como sinais de entrada de outros neurônios em um tempo $t + 1$, ocorrendo assim uma realimentação. Essa característica torna esse tipo de rede apropriada para o tratamento de sinais variantes no tempo como, por exemplo, previsão de séries temporais. Desta forma, como existe uma realimentação, as saídas atuais de uma RNR leva em consideração os valores de saída calculadas anteriormente (SILVA; SPATTI; FLAUZINO, 2010). Essa arquitetura é exemplificada na Figura 7.

Figura 7 – Exemplo de uma rede neural recorrente genérica.

Fonte: Elaborada pelo autor.

3.1.2 Árvores de Decisão

Outra técnica de AM que pode ser aplicada à resolução de problemas de previsão são as árvores de decisão. Rezende (2003) define uma árvore de decisão, recursivamente, como: (i) um nó folha, que corresponde a uma classe ou; (ii) um nó de decisão, que consiste no teste de um atributo, e (iii) vértices que levam a outras árvores, dependendo do resultado obtido no teste.

As árvores de decisão utilizam conjuntos de teste para criar um modelo de classificação, o qual, em seguida, é utilizado para classificar novos dados. Mitchell (1997) cita que árvores de decisão estão entre os métodos mais amplamente utilizados e, além de serem práticas e robustas com relação a ruídos, podem ser facilmente convertidas em um conjunto de regras *if-then*.

Basicamente, algoritmos de treinamento de árvores de decisão diferem (i) no modo pelo qual é decidido o atributo testado para realizar a divisão do conjunto de teste em subgrupos e (ii) no método utilizado para a poda, no qual folhas são retiradas da árvore com o objetivo de evitar superajuste e melhorar o seu desempenho. Essa é uma técnica poderosa, que permite com que as classes de novos dados sejam previstas de forma fácil, rápida e sem a necessidade de grande poder de processamento (REZENDE, 2003).

Um exemplo de uma árvore de decisão, retirado de Shi (2007), é apresentado na Figura 8, onde *leaf* são nós folhas (classes) e N_x são nós de decisão, ligados a vértices que levam a outras árvores. Seus algoritmos de treinamento mais utilizados são o ID3, ASSISTANT e C4.5, os quais são descritos detalhadamente em Mitchell (1997).

Figura 8 – Exemplo de uma árvore de decisão.

Fonte: Shi (2007).

Ambas as técnicas de AM apresentadas até o momento (redes neurais artificiais e árvores de decisão) foram utilizadas para obter os resultados iniciais deste trabalho. Nos experimentos, a capacidade de classificação desses métodos foi utilizada para determinar o estado futuro e o nível da água de rios da cidade de São Carlos. Para o treinamento dos modelos foram utilizados os dados coletados pelos sensores do sistema REDE.

3.1.3 Support Vector Machine - SVM

Até o momento, as técnicas de AM apresentadas nesta tese não necessitaram de uma fundamentação teórica mais profunda. Porém, essa abordagem não será possível para descrever *Support Vector Machine* (SVM), devido à sua complexidade.

Considerando o problema de classificação, em que X é o conjunto de instâncias (também chamado de espaço de instâncias) e y suas classes (espaço de rótulos), Luxburg e Schölkopf (1997) definem o problema de aprendizado como encontrar uma função de mapeamento $f: X \rightarrow y$ que tenha a menor quantidade possível de erros. Essa função f é denominada *classificador*.

Cada técnica de AM faz a busca dessa função f em uma determinada classe de funções e Luxburg e Schölkopf (1997) demonstram que é crucial limitar a classe de funções para que ocorra a aprendizagem. Dito isso, classificadores SVM são baseados na classe de hiperplanos (HEARST, 1998) onde, em um hiperespaço de n dimensões, um hiperplano é um objeto de $n - 1$ dimensões utilizado para separar esse espaço. A Figura 9 representa como duas classes podem ser separadas através de diferentes retas.

A ideia por trás da SVM é encontrar o hiperplano ótimo, definido como o hiperplano que tem a máxima margem de separação entre as duas classes (HEARST, 1998; LORENA; CARVALHO, 2007). No caso da Figura 9, é a reta H_0 .

Detalhes de como esse processo é realizado, assim como mais informações sobre SVM, podem ser obtidos em Hearst (1998), Lorena e Carvalho (2007), Luxburg e Schölkopf

Figura 9 – Exemplo de separação de 2 classes em R^2 .

Fonte: Elaborada pelo autor.

(1997).

3.1.4 Rede Bayesiana

Chen, Liao e Tang (2007) descrevem uma Rede Bayesiana como um modelo estatístico que utiliza o teorema Bayesiano para realizar a classificação. Nessa técnica de AM, um conjunto de dados é utilizado para a criação da estrutura da rede e da tabela de probabilidade condicional e, a partir dessas estruturas, é possível realizar a classificação utilizando o teorema Bayesiano.

Mitchell (1997) exemplifica a representação de uma Rede Bayesiana (Figura 10) e afirma que tais redes descrevem a distribuição de probabilidade sobre um conjunto de variáveis. Em seu exemplo, cada variável é representada por um nó na rede (como a variável *Campfire*, por exemplo) e os arcos na rede representam a dependência condicional entre as variáveis. Para cada variável, uma tabela de probabilidade condicional também é especificada. No exemplo, temos a tabela de probabilidade condicional para a variável *Campfire*, onde o autor exemplifica que a probabilidade da variável *Campfire* (*C*) ser verdadeira, dado que as variáveis *Storm* (*S*) e *BusTourGroup* (*B*) são verdadeiras, é de 0,4, ou seja:

$$P(\text{Campfire} = \text{Verdadeiro} | \text{Storm} = \text{Verdadeiro}, \text{BusTourGroup} = \text{Verdadeiro}) = 0,4$$

Assim, é possível determinar o valor de uma das variáveis com base nos valores observados de outras. Mais detalhes de como esse processo é realizado, do Teorema Bayesiano

Figura 10 – Exemplo de uma Rede Bayesiana.

Fonte: [Mitchell \(1997\)](#).

e de como ocorre o treinamento de Redes Bayesianas podem ser encontrados em [Mitchell \(1997\)](#).

Outras técnicas de AM poderiam ser descritas nesta seção, no entanto, as técnicas mencionadas são suficientes para que o leitor acompanhe os trabalhos realizados neste projeto e os artigos descritos na revisão sistemática. A próxima seção apresenta o restante dos artigos selecionados durante a revisão sistemática.

3.2 Considerações Finais

Fica claro que existe um aumento de trabalhos relacionados à RSSF e a AM para a previsão de desastres naturais, principalmente enchentes. Sendo que, de todos os trabalhos listados, somente dois ([ROY; GUPTA; GOSWAMI, 2012](#); [HOSSAIN *et al.*, 2014](#)) empregam RSSF juntamente com AM para realizar essas previsões e nenhum faz uso de uma RSSF fisicamente instalada em ambientes reais.

Também é possível notar, pelos trabalhos de [Hughes *et al.* \(2008\)](#), [Roy, Gupta e Goswami \(2012\)](#), [Seal *et al.* \(2012\)](#), [Hossain *et al.* \(2014\)](#), que a utilização de RFFS na coleta de dados para realizar previsão de enchentes é um tópico recorrente, principalmente quando pensamos em integrar os modelos de previsão nas RFFS. Mas essa área ainda necessita de mais estudos, devido à falta de implementações reais das RSSF propostas, à falta de um estudo mais aprofundado das técnicas de AM para realizar as previsões e gerar alertas sobre enchentes, à falta de uma distribuição real dos modelos de previsão nos nós da RSSF e à verificação do impacto dessa integração no sistema.

Como visto neste capítulo, técnicas de AM são muito úteis para realizar a modelagem de sistemas, principalmente, quando características físicas do sistema são complexas para serem modeladas ou não são tão bem compreendidas. A utilização de AM para a previsão de enchentes e comportamento de rios é um tópico já bastante estudado e difundido. Porém, poucos dos trabalhos fazem uso de RSSF e IoT na captura dos dados ou

inserem mecanismos de tolerância a falhas em suas abordagens de previsão de desastres naturais, como será visto no Capítulo [4](#), a seguir.

TOLERÂNCIA A FALHAS EM PREVISÃO DE DESASTRES NATURAIS

4.1 Conceitos sobre Tolerância a Falhas

Sistemas computacionais estão sujeitos a falhas, principalmente quando voltados para o monitoramento de áreas de risco, como ambientes naturais sujeitos a desastres. Apesar de inevitáveis, as consequências dessas falhas podem ser reduzidas e, consequentemente, aumentar a confiabilidade e a disponibilidade desses sistemas. Nesse contexto, sistemas que incorporam o fato da existência de falhas, que são capazes de utilizar técnicas para tolerá-las e, concomitantemente, realizar os serviços para os quais foram projetados são considerados sistemas tolerantes a falhas (KOREN; KRISHNA, 2007). Em outras palavras, podemos alcançar a tolerância a falhas quando um sistema, programa ou hardware, por exemplo, consegue ser executado propriamente apesar da ocorrência de falhas (AVIŽIENIS, 1967).

A fim de melhor entender, devemos definir alguns conceitos básicos como falha, erro e defeito e, principalmente, a relação “*falha* $\rightarrow$ *erro* $\rightarrow$ *defeito*”. Uma falha em um sistema pode se manifestar como erros, os quais acabam por gerar possíveis defeitos no sistema. Esses defeitos acabam por desviar o sistema do seu comportamento normal, o que ocasiona um funcionamento diferente do que foi especificado para tal sistema (BEDER, 2001). Assim, falha pode ser definida como a causa física ou algorítmica de um erro, erro como um estado errôneo do sistema, o que leva a um estado inválido em sua especificação e a um defeito quando ocorre um comportamento fora da especificação do sistema (GIUNTINI, 2016).

O objetivo da tolerância a falhas é prevenir que erros e falhas levem a uma parada ou a uma anomalia no sistema (LEE; ANDERSON, 1990). Isso pode ser alcançado através de técnicas de tolerância a falhas, que podem ser divididas em duas categorias: (i) detec-

ção, localização e reconfiguração, e (ii) mascaramento (GIUNTINI, 2016). Na primeira categoria, temos as fases definidas para o tratamento de falhas, as quais levam o sistema a um estado válido anterior ou posterior à falha, porém, essa abordagem não é muito utilizada em sistemas de tempo real críticos devido à sobrecarga gerada. Já a segunda abordagem, o mascaramento, procura mascarar a falha, não permitindo com que ela se torne aparente. Dessa forma, o mascaramento impede que o sistema entre em um estado inválido, sendo desnecessário passar pelas fases citadas anteriormente (GIUNTINI, 2016).

Esse mascaramento pode ser obtido de diversas formas e a redundância é uma das mais comumente encontradas. Nesta tese, foram utilizadas redundâncias dos modelos de previsão de desastres naturais para obter a tolerância a falhas. Modelos capazes de operar com diferentes tipos e quantidades de dados foram distribuídos por todo o sistema e, dependendo do seu grau de degradação, as previsões são realizadas com maior ou menor acurácia. Isso garante que as previsões sejam realizadas e que os alertas sejam gerados mesmo em casos em que ocorra falha na comunicação ou a destruição dos nós do SENDI.

Nas próximas seções, são apresentados o cenário, as ferramentas e os detalhes de implementação do SENDI, o sistema tolerante a falhas baseado em IoT e AM para a previsão e detecção de desastres naturais proposto nesta tese.

4.2 Arquitetura do Sistema SENDI

O SENDI é composto por 3 *tiers*, como ilustra a Figura 11, sendo o primeiro *tier* constituído por uma RSSF, responsável por coletar os dados do ambiente monitorado. Já o segundo *tier* (*fog computing*) é responsável por concentrar os dados dos nós sensores, bem como realizar a distribuição e o processamento dos dados coletados pelo primeiro *tier* quando necessário. Vale destacar que, pelo segundo *tier* ser baseado no paradigma de *fog computing*, ele fornece recursos computacionais (processamento e armazenamento) mais próximos do utilizador final, a fim de melhorar a latência dos serviços. Finalmente, o terceiro *tier* é composto pela *cloud*, na qual todos os dados do sistema são concentrados, provendo poder de processamento e de armazenamento de dados de forma abstrata. Dessa forma, a *cloud* obtém uma visão geral do ambiente monitorado, podendo combinar os dados colhidos pelos sensores com dados disponíveis na Internet e gerar previsões com maior acurácia e alertas online para a população. A concepção dessa solução surgiu do projeto e-Noe, implantado na cidade de São Carlos, que consiste em prever e detectar enchentes, alertando os moradores da região da ocorrência de uma possível inundação. Em razão disso, o objetivo principal do SENDI é melhorar a previsão de desastres naturais e deixá-la tolerante a falhas. No último caso, é fundamental que as previsões continuem a ser realizadas, independentemente da situação do sistema.

A base do SENDI são os nós do *tier* 1/RSSF, modelados com base na RSSF

Figura 11 – Arquitetura do sistemas SENDI vista como *tiers*.

Fonte: Adaptada de [Furquim et al. \(2017\)](#).

existente em São Carlos. É considerado que esses nós possuem suprimento limitado de energia e suas baterias são carregadas via energia solar. Nesse *tier*, a função básica dos nós é a coleta de dados. Esses nós possuem vários sensores capazes de coletar diferentes características sobre o ambiente monitorado e enviá-las ao *tier 2/fog computing*. Os nós desse *tier* também possuem dispositivos sonoros e visuais (como luzes e sirenes) capazes de alertar a população local sobre o perigo de desastres. Por sua vez, os nós do *tier 2/fog computing* ficam responsáveis pela agregação dos dados coletados pelos nós sensores e pelo envio desses dados ao *tier 3/cloud*. É importante destacar que, dado um nó do *tier 2*, esse é responsável por coletar dados somente de nós do *tier 1* que são seus vizinhos. Definimos aqui que dois nós são vizinhos quando se comunicam diretamente, sem a necessidade de *hops*.

Como ilustrado na Figura 11, um nó do *tier 2* é responsável pelos nós do *tier 1*, que estão dentro do seu raio de comunicação direta. Os nós do *tier 2* se caracterizam pelo fornecimento ilimitado de energia e por ajudarem a diminuir o consumo de energia dos nós sensores (*tier 1/RSSF*) durante o envio de mensagens ao *tier 3/cloud*. Assim como os nós do *tier 1/RSSF* e por motivo de redundância, os nós do *tier 2/fog computing* também possuem dispositivos sonoros e visuais capazes de alertar a população local sobre o perigo de desastres. Finalmente, o *tier 3* é formado pela *cloud* que provê os recursos necessários para os dados serem armazenados, combinados entre si e com outras informações sobre o ambiente monitorado disponíveis online e, em seguida, processados através de modelos de previsão criados via AM.

Caso haja algum perigo de desastre, a *cloud* envia alertas online aos nós do *tier 2/fog computing*, que por sua vez, soam o alerta e os repassam aos nós do *tier 1/RSSF*. Essa configuração agrega grande flexibilidade ao SENDI, pois além de nós mais simples, que realizam somente leituras de dados, sua arquitetura contém o suporte para o processamento de grandes quantidades e de combinação de dados, disponibilizando também,

formas de economia de energia. Essa arquitetura pode ser facilmente modificada dependendo do ambiente monitorado e dos recursos disponíveis no local.

A fim de entender melhor o SENDI, sua arquitetura e implementação são detalhadas na Seção 4.3. A Seção 4.4 contém uma lista de trabalhos relacionados, seguida pelas Seções 4.5, que descreve a criação dos modelos de previsão, experimentos realizados e resultados conseguidos com esses modelos e a Seção 4.6, que apresenta os experimentos e resultados obtidos pelo SENDI.

4.3 SENDI - Detalhes de Implementação

O SENDI foi modelado a partir do ns-3 (RILEY; HENDERSON, 2010), um simulador de redes de eventos discretos, com licença GNU GPLv2, o módulo 6LoWPAN desse simulador e dados coletados pelo sistema REDE. A implementação do módulo 6LoWPAN foi baseada nos RFC4944 e RFC6282 e, apesar de não trazer todos os padrões descritos nesses documentos, ele serve de base para outras funcionalidades, além de ser um módulo oficial integrado ao ns-3 desde a versão ns-3.19.

Nesta Seção, serão apresentados detalhes sobre a implementação do SENDI, incluindo sua pilha de protocolos (Subseção 4.3.1), algoritmo de roteamento utilizado (Subseção 4.3.2), comportamento dos nós de ambos os *tiers* (Subseção 4.3.3 e 4.3.4), protocolo de eleição de líder (Subseção 4.3.5), mecanismo de tolerância a falhas (Subseção 4.3.6) e dados utilizados no estudo de caso sobre previsão de enchentes (Subseção 4.3.7).

4.3.1 Configuração da Pilha de Protocolos IoT nos Nós

Durante a programação da simulação do SENDI no ns-3, alguns módulos e parâmetros de configuração foram utilizados para a criação dos nós do SENDI. Essa Seção descreve essa configuração.

Por se tratar de uma arquitetura baseada em IoT, a camada física, assim como todos os parâmetros necessários para a simulação dessa camada, estão definidos pelo padrão IEEE 802.15.4, sendo utilizada a radio frequência de curto alcance (IEEE..., 2011). Já a camada de enlace é constituída também pelo 6LoWPAN (IPv6 with LoWPAN), responsável pela compatibilidade entre os pacotes IPv6 de redes sem fio de baixa potência e de outras redes que também utilizam o mesmo protocolo na camada de rede (THUBERT; HUI, 2011). Portanto, a pilha de protocolos utilizada pelo SENDI é apresentada na Figura 12.

No ns-3 a classe responsável por simular a pilha de protocolos para o padrão IEEE 802.15.4 é a *LrWpanHelper*. Essa classe contém as configurações e parâmetros necessários para a camada física e de enlace, relativos a especificação. Após a definição das

Figura 12 – Pilha de protocolos utilizada na simulação do sistema SENDI e implementada em ns-3.

Fonte: Adaptada de [Furquim et al. \(2017\)](#).

duas primeiras camadas, a classe *SixLowPanHelper* é utilizada para instalar o 6LoWPAN, fundamental para IoT, na pilha de protocolos. Finalizado esse procedimento, o nó está pronto para comunicação utilizando o IPv6 e demais protocolos compatíveis, nas camadas superiores.

O protocolo UDP foi utilizado na camada de transporte devido à falta de necessidade de conexão característica do protocolo, o que não bloqueia a comunicação caso o seu destino esteja inoperante, e também devido à sua simplicidade, que gera menos sobrecarga na comunicação.

No topo da pilha de protocolos está a aplicação para o monitoramento e a previsão de desastres naturais, incluindo a abordagem de tolerância a falhas que utilizam os vários modelos de previsão criados e a eleição de líder.

4.3.2 Algoritmo de Roteamento

Como protocolo de roteamento, o RFC 6550 define o *IPv6 Routing Protocol for Low-Power and Lossy Networks* (RPL) para *Low-Power and Lossy Networks* (LLNs). LLNs são redes que apresentam restrições quanto ao poder de processamento, energia e/ou memória dos nós. Nelas, os nós agem como roteadores e, normalmente, suportam somente taxas baixas de transferência. Essas redes também apresentam outras características específicas, como padrões de comunicação *point-to-multipoint* ou *multipoint-to-point*, e podem ser formadas por centenas de nós. Essas particularidades fazem que o roteamento em tais redes seja diferenciado e apresente seus próprios desafios e soluções ([BRANDT et al., 2012](#)). Diferentemente de redes cabeadas, onde encontramos topologias predefinidas devido ao *point-to-point*, as LLNs normalmente não possuem essa propriedade e, portanto, a RPL necessita descobrir os links existentes entre os nós e selecioná-los cuidadosamente. Dessa forma, a RPL organiza a topologia da rede como um conjunto de um único ou vá-

Figura 13 – Exemplo de instância RPL.

Fonte: Adaptada de [Furquim et al. \(2017\)](#).

rios *Destination Oriented DAGs* (DODAG), formando um *Directed Acyclic Graph* (DAG), sendo um DODAG por nó *sink* (root). A Figura 13 exemplifica a configuração de uma instância RPL, na qual os nós R1, R2 e R3 são os *roots* de suas DODAG e interligam o restante dos nós à Internet ou à outra rede, por exemplo. Os nós que apresentam um *path* direto ao root, como os nós A e B, são considerados *Rank 2*. Já os nós que não conseguem se comunicar diretamente o root, mas conseguem ver nós *Rank 2*, são considerados *Rank 3*, como o nó C da outra DODAG, e assim por diante.

Nenhum módulo que implemente um protocolo de roteamento para o módulo *lr-wpam* do ns-3 está disponível. Nesta tese, foi implementado um protocolo de roteamento baseado na posição de cada nó, onde um pacote, cujo destino seja um nó não vizinho, é enviado a algum nó vizinho cuja posição seja mais próxima à posição dos nós destino. Essa transferência é realizada até que o pacote chegue a um nó vizinho ao nó destino e esse o envie diretamente ao nó destino. Esse procedimento é descrito na Figura 14.

É importante destacar que existe um limite definido de vezes que um pacote pode ser roteado por um mesmo nó, que não seu destino. Esse limite é descrito na Figura 14 como n , serve para evitar ciclos infinitos durante o roteamento, pode ser ajustado com qualquer valor desejado e foi definido com o valor 5 durante as simulações realizadas. Devido a isso, todo pacote tem um identificador único que é calculado com base no nó que criou o pacote.

Outro ponto que merece ser ressaltado é a necessidade de cada nó manter uma lista atualizada de seus vizinhos ativos (que estão em funcionamento). Isso é realizado enviando uma mensagem em intervalos de tempo constante a seus vizinhos, sinalizando

Figura 14 – Roteamento de um pacote até seu destino.

Fonte: Elaborada pelo autor.

seu funcionamento. Esse tempo foi definido como o intervalo entre leituras dos nós sensores e, todo nó que não enviar essa mensagem, será considerado inativo até o próximo ciclo de leitura de dados.

Portanto, para o funcionamento desse protocolo de roteamento, os nós devem saber a posição de seus nós vizinhos e a posição do nó para o qual será enviada alguma mensagem. Adicionalmente, o endereço do nó destino é embutido no pacote enviado para que os demais nós, pelos quais o pacote for roteado, possam conhecer a posição do nó destino do pacote.

Desta forma, cada nó deve saber: (i) a posição onde foi instalado, (ii) posição dos seus nós vizinhos, definidos como os nós que estão em sua área de comunicação direta (sem necessidade de roteamento) e (iii) endereço dos nós com que queira se comunicar. Para tanto, é considerado que, na instalação de um nó, sua localização seja configurada (através de GPS ou outra tecnologia) e as localizações dos nós com que possa querer se comunicar também sejam também informadas. Qualquer nó fora dessa lista fica incomunicável. Portanto, devem ser incluídos na lista nós que coletem dados de interesse ao novo nó que está sendo instalado permitindo, assim, a disseminação desses dados.

Figura 15 – Instalação de um novo nó e disseminação das informações necessárias para o protocolo de roteamento utilizado.

Fonte: Elaborada pelo autor.

Nas simulações realizadas, um novo nó, ao ser instalado, faz uma requisição de endereço e localização de seus nós vizinhos, montando a lista de vizinhos, necessária ao roteamento. Esse procedimento é descrito na Figura 15.

Esse é um protocolo de roteamento simples, utilizado somente pela falta de uma implementação oficial nos módulos do ns-3. Devendo ser substituído em redes reais pelo RPL ou em futuras simulações, quando um módulo de roteamento para o lr-wpan estiver disponível.

4.3.3 Comportamento dos Nós do Tier 2

Sendo uma camada intermediária entre o *tier 1* e a *cloud*, o comportamento dos nós do *tier 2*, em situações onde existe comunicação normal com a *cloud*, se restringe a coletar os dados lidos pelos nós sensores (*tier 1*) e enviar esses dados para a *cloud*.

Em caso de falha na comunicação com a *cloud*, os nós do *tier 2*, além de coletar os dados dos nós sensores, ficam responsáveis pela distribuição dos dados lidos entre os demais nós do *tier 2*. Essa distribuição é realizada considerando os dados de interesse de cada nó do *tier 2*, sendo que, os dados de interesse de cada nó são definidos seguindo o modelo de previsão distribuído, apresentado na Seção 4.5.

Dessa forma, a cada ciclo de leitura e previsão, caso a *cloud* não esteja disponível, cada nó do *tier 2* envia mensagens requisitando seus dados de interesse aos demais nós do mesmo *tier*. Para tanto, cada nó do *tier 2* contém uma lista de nós que realizam a leitura de seus dados de interesse.

Esses procedimentos são representados na Figura 16 e 17, sendo que são utilizados dois parâmetros na configuração dessas operações: (i) quantidade máxima de mensagens consecutivas enviadas para a *cloud* sem que haja confirmação de recebimento, o que determina a inatividade da *cloud* e (ii) tempo de espera pela mensagem contendo os dados de interesse requisitados à outros nós, parâmetro também utilizado para o tempo de espera de confirmação de recebimento de dados pela *cloud*. Esses parâmetros são configuráveis para se adequarem às características do ambiente monitorado.

Figura 16 – Fluxograma de comportamento dos nós do *tier 2* - definição de *cloud* inativa.

Fonte: Elaborada pelo autor.

Figura 17 – Fluxograma de comportamento dos nós do *tier 2* quando *cloud* está inacessível - distribuição dos dados e seleção do modelo de previsão.

Fonte: Elaborada pelo autor.

Uma vez que a *cloud* é considerada inativa por um nó, essa continuará sendo considerada inativa para o mesmo nó até que ele receba uma mensagem vinda da *cloud*. Isso evita que segundos preciosos sejam utilizados a cada ciclo para determinar se a *cloud* está ativa novamente ou não.

Ao receber uma mensagem de solicitação de dados, um nó do *tier 2* irá comparar a lista de nós de interesse, recebida na mensagem de requisição, com sua lista de nós do *tier 1*, dos quais realizou a coleta de dados. Caso algum dos nós coincida, os dados são processados localmente e o resultado enviado para o nó que realizou a requisição.

Tanto o processamento local dos dados, quanto a criação dos modelos de previsão distribuído e a criação dos modelos locais de previsão são descritos na Seção 4.5. O comportamento dos nós do *tier 2* foi simulado no ns-3 criando uma nova aplicação, baseada nas aplicações modelo *udp-server* e *udp-client*, contendo o comportamento acima descrito, além dos métodos necessários para envio e recebimento das mensagens e colocada no topo

Figura 18 – Fluxograma de comportamento dos nós do *tier* 1.

Fonte: Elaborada pelo autor.

da pilha de protocolos IoT.

4.3.4 Comportamento dos Nós do Tier 1

Sendo constituído por uma RSSF equipada com a pilha de protocolo IoT em seus nós, o *tier* 1 do SENDI faz a coleta dos dados do ambiente monitorado e envia esses dados para os nós do *tier* 2. Esse comportamento permanece inalterado, desde que todos os nós do *tier* 1 tenham um nó vizinho do *tier* 2 como líder e seus líderes respondam aos envios de dados com uma mensagem de confirmação. Para tanto, todos os nós do *tier* 1 armazenam o endereço de seu líder e, a cada ciclo de leitura e previsão, os nós do *tier* 2 enviam uma mensagem contendo seu endereço, sinalizando seu funcionamento e renovando sua liderança aos nós vizinhos do *tier* 1.

Caso um nó n não receba uma mensagem de renovação de liderança ou seu nó líder não confirme o recebimento dos dados n vezes consecutivas, dentro de um mesmo ciclo de leitura e previsão, o nó n é considerado como não tendo um líder e é iniciado o processo de eleição de líder, descrito na Subseção 4.3.5. O valor de n pode ser definido e adaptado dependendo das características do ambiente a ser monitorado, sendo que foi definido com valor igual a 2 (dois) nas simulações realizadas com o SENDI. Esses passos são ilustrados na Figura 18, que representa o fluxograma de comportamento de um nó do *tier* 1.

Quando um nó do *tier* 1 é definido como líder, ele envia uma mensagem aos seus nós vizinhos avisando sobre sua liderança, e passa a ser o novo líder, unicamente de seus nós vizinhos que ainda não têm um líder definido. Após assumir a liderança, seu comportamento e funções passam a ser de um nó do *tier* 2, sendo que, sua liderança

Figura 19 – Modelo de dissipação de energia de rádio.

Fonte: [Heinzelman, Chandrakasan e Balakrishnan \(2002\)](#).

permanece por um tempo limitado, dependendo das configurações do SENDI.

No caso das simulações realizadas durante esta Tese, um nó do *tier* 1 eleito como líder, permanece nessa condição durante um *round*, ou seja, durante um ciclo de leitura e previsão. O comportamento dos nós do *tier* 1 foi simulado no ns-3 criando uma nova aplicação, também baseada nas aplicações modelo *udp-server* e *udp-client*, contendo o comportamento acima descrito, além dos métodos necessários para envio e recebimento das mensagens e colocada no topo da pilha de protocolos IoT.

4.3.5 Protocolo de Eleição dos Nós Líderes

O processo de eleição dos líderes e de formação dos *clusters*, adotado nesta Tese, é baseado na energia remanescente da bateria de cada nó. No entanto, o SENDI tem flexibilidade para adotar outros mecanismos de eleição de líder que melhor se adaptem as necessidades e características do ambiente monitorado.

É importante destacar que esses procedimentos só iniciam quando a possibilidade de desastres é detectada e quando uma parcela ou a totalidade dos nós sensores (*tier* 1) perdem contato com os nós do *tier* 2 (nó *sink*). A cada *round*, todos os nós do *tier* 1 devem ter um líder pertencente ao *tier* 2 que agrupe os dados coletados e os envie à *cloud* ou, em caso de falha na comunicação com a *cloud*, faça o processamento desses dados e preveja os desastres. O líder natural de um nó sensor, na arquitetura descrita, é um dos nós *sink* do *tier* 2. Porém, em caso de falha do nó *sink* local, os nós sensores podem eleger, entre si, um novo líder e dar continuidade ao funcionamento do sistema.

Assim como o protocolo de clusterização LEACH ([HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2002](#)), consideramos o modelo simplificado para a dissipação de energia do hardware de rádio (Figura 19) e utilizamos probabilidade para determinar se um nó será ou não líder de um *cluster* e passará a fazer parte do *tier* 2. No entanto, diferente do protocolo LEACH, não consideramos que: (i) inicialmente os nós tenham a mesma quantidade de energia, (ii) um nó seja líder várias vezes consecutivas, (iii) existe limite inferior ou superior de líderes, (iv) o processo de eleição ocorre várias vezes em um mesmo *round* até que todos os nós tenham um líder, (v) existe um limite de vezes

em que o processo de eleição ocorre em um nó durante um mesmo *round* e após esse limite ser alcançado, o nó é forçado a se tornar líder, e (vi) um nó somente é aceito como líder se pertencer à vizinhança de outro nó, o que diminui a necessidade de *hops* e de comunicações com nós muito distantes. De acordo com o modelo de dissipação de energia de rádio, a energia gasta no envio de uma mensagem ($E_{Tx}(l, d)$) contendo l – *bits* para um receptor a uma distância d , para que a transmissão tenha um Signal-to-Noise Ratio (SNR) aceitável, é dado por:

$$E_{Tx}(l, d) = \begin{cases} l.E_{elect} + L.\epsilon_{fs}.d^2 & \text{if } d < d_0 \\ l.E_{elect} + L.\epsilon_{mp}.d^4 & \text{if } d \geq d_0 \end{cases} \quad (4.1)$$

onde ϵ_{fs} e ϵ_{mp} são dependentes do modelo do amplificador de transmissão para espaço livre (free space - *fs*) e *multipath* (*mp*), d_0 é o *threshold* que determina qual modelo de transmissão (espaço livre ou *multipath*) será utilizado e E_{elect} é o consumo de energia para cada bit transmitido. Essa equação é utilizada pelo LEACH para alcançar a probabilidade ótima de um nó ser líder de um *cluster*, conforme Equação 4.2. Essa probabilidade é, então, utilizada na Equação 4.4, que define um *threshold*. Nesse momento, dado um nó, é gerado um número aleatório e, caso esse número randômico seja menor que o *threshold* calculado, o nó se torna líder de um *cluster*. Detalhes de como chegar a essa probabilidade encontram-se em (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2002).

$$p_{opt} = \frac{k_{opt}}{n} \quad (4.2)$$

onde, p_{opt} é a probabilidade ótima de um nó ser líder, n é o número de nós e k_{opt} é definido como a quantidade ótima de *clusters*, calculado como:

$$k_{opt} = \frac{2}{0.765} \sqrt{\frac{n}{2\pi}} \quad (4.3)$$

Como dito anteriormente, o LEACH não permite que um nó seja líder várias vezes seguidas. Para que um nó s possa ser líder novamente, ele deve entrar novamente no conjunto G da Equação 4.4. Esse conjunto contém somente os nós que não foram líderes nos mais recentes $(r * \text{mod}(n/k))$ *rounds*, onde k é o número de líderes esperado e r é o número de *rounds* em que o nó s não foi definido como líder.

$$T(s) = \begin{cases} \frac{p_{opt}}{1 - p_{opt} \cdot (r * \text{mod}(\frac{1}{p_{opt}}))} & \text{if } s \in G \\ 0 & \text{caso contrário} \end{cases} \quad (4.4)$$

Neste trabalho, o *threshold* definido pelo LEACH foi modificado, considerando a quantidade de energia remanescente no nó. Sendo assim, dado um nó s , o seu novo *threshold* é definido da seguinte forma:

$$T'(s) = T(s) * \alpha + \frac{E}{E_f} * (1 - \alpha) \quad (4.5)$$

onde $T(s)$ é o valor de *threshold* original do LEACH, E é a energia atual contida na bateria do nó, E_f é a capacidade máxima da bateria (permitindo uma heterogeneidade de baterias entre os nós), α corresponde ao peso dado à cada parte da equação, onde o valor 1 acaba por utilizar somente o LEACH modificado/implementado na simulação e o valor 0 utiliza somente as informações sobre energia.

Após o cálculo do *threshold*, o procedimento continua o mesmo, com s gerando um número aleatório e definindo se é ou não líder com base no valor comparado ao *threshold*. A quantidade de *rounds* r em s é zerada toda vez que s se torna líder. Além dessas diferenças, não existe mais a limitação dos $(r * \text{mod}(n/k))$ *rounds*, portanto qualquer nó pode ser líder independentemente de quantas vezes os demais já foram ou do número de *rounds*. Em um mesmo *round*, o processo de eleição pode ocorrer repetidas vezes, caso uma primeira eleição não seja suficiente para que todos os nós tenham um líder definido. Nesse caso, somente os nós que ainda não tiveram seus líderes definidos, continuam repetindo o processo de eleição por uma quantidade limitada e definida de repetições. A quantidade de repetições pode ser estipulada e, caso um nó não tenha um líder após atingir a quantidade definida de eleições, o próprio nó se torna líder independentemente do nível de sua bateria. Durante os experimentos realizados com o SENDI, o limite de repetições foi estipulado como sendo 5 (cinco).

O Algoritmo 1 ilustra o procedimento para a eleição de líder realizada em cada nó. Após cada tentativa de eleição, caso o nó não seja definido como líder, ele aguarda um tempo aleatório para que os demais nós em sua vizinhança também possam executar o algoritmo de eleição de líder. No caso de um novo líder não aparecer em sua vizinhança, o algoritmo continua a ser executado até que atinja o limite máximo estipulado, que é quando o próprio nó é forçado a se tornar líder, não importando o nível de sua bateria.

4.3.6 Abordagem de Tolerância a Falhas

O mecanismo de tolerância a falhas do SENDI considera a falha na comunicação e na transmissão de dados entre nós de quaisquer *tiers* e reage a essa falha garantindo que as previsões de desastres e alertas sejam realizados. Essa falha pode ser causada pela falta de energia ou pela destruição parcial ou total de nós ou ainda pela perda temporária ou permanente da capacidade de comunicação entre os nós. Para tanto, os nós dos *tiers* 1 e 2 são munidos com inteligência (descritas nas Seções 4.3.3 e 4.3.4) para se reorganizarem

Algoritmo 1 – Algoritmo de eleição de líder.

```

1:  $Kopt \leftarrow \sqrt{\frac{n}{2 \times \pi}} \times \frac{2}{0.765}$ 
2:  $Popt \leftarrow \frac{Kopt}{n}$ 
3:  $ThresholdLEACH \leftarrow \frac{Popt}{1 - Popt \times (rmod(\frac{1}{Popt}))}$ 
4:  $MaxNumberElection \leftarrow \text{GetMaxNumberOfElections}()$ 
5:  $ElectionCounter \leftarrow 0$ 
6: enquanto NoSemLider() faça
7:    $ThresholdEnergy \leftarrow \frac{\text{GetCurrentBatteryPowerLevel}()}{\text{GetFullBatteryPowerLevel}()}$ 
8:    $Threshold \leftarrow ThresholdLEACH \times \alpha + ThresholdEnergy \times (1 - \alpha)$ 
9:    $RandomNumber \leftarrow \text{random}(0, 1)$ 
10:   $ElectionCounter \leftarrow ElectionCounter + 1$ 
11:  se  $RandomNumber < Threshold$  or  $ElectionCounter > MaxNumberElection$  então
12:    DefineNóComoLider()
13:  senão
14:    wait( $\text{random}(0, 1)$ )
15:  fim se
16: fim enquanto

```

em casos assim, e possuem modelos embutidos de previsão de desastres (criados como descrito na Seção 4.5 e carregados no código do nó) mais simples que os processados pela *cloud*. Esses modelos embutidos são mais simples devido às capacidades energéticas e de processamento dos nós, e geram previsões com menor acurácia, porém, possibilitam que as previsões continuem a ser realizadas mesmo com falhas de comunicação com a *cloud* ou com os demais nós do sistema.

Em caso de falha de comunicação com a *cloud*, os nós do *tier 2* assumem o papel da *cloud*, armazenando os dados coletados pelos nós do *tier 1*, distribuindo esses dados entre si e realizando as previsões de enchentes através de modelos distribuídos de previsão de desastres, como proposto em Furquim *et al.* (2015) e descrito na Subseção 4.5.4. Nesse trabalho, as previsões realizadas por nós (similares aos nós do *tier 2*) em uma determinada região são repassadas a nós de outras regiões do ambiente monitorado. Por sua vez, as previsões recebidas são utilizadas para realizar novas previsões e para melhorar a acurácia das previsões na região desses nós. Como os nós do *tier 2* realizam a distribuição dos dados foi discutido na Subseção 4.3.3 e como foi criado o modelo de previsão distribuído e quais são os dados utilizados nesse modelo será discutido na Seção 4.5.

Caso ocorram falhas em nós pertencentes ao *tier 2*, gerando áreas sem cobertura e que nós do *tier 1* fiquem sem um representante no *tier 2* para enviar suas leituras, os nós do *tier 1* se reorganizam, elegendo entre eles nós para ocupar temporariamente os lugares dos nós defeituosos do *tier 2*. Essas eleições ocorrem a cada *round*, aqui considerado como um ciclo de leitura, envio/distribuição de dados, realização das previsões e, caso necessário, geração de alertas. São eleitos, durante esse processo, quantos nós sejam necessários até

que todos os nós do *tier* 1 tenham um representante no *tier* 2. Basicamente, os nós eleitos assumem o papel de nós do *tier* 2, coletam dados do *tier* 1 (incluindo seus próprios) e os enviam à *cloud*. Em caso de falha da *cloud*, realizam a distribuição dos dados e as previsões dos desastres no ambiente monitorado. O funcionamento e detalhes sobre o algoritmo de eleição foram descritos com detalhes na Seção 4.3.5.

O funcionamento do algoritmo para o sistema de tolerância a falhas realizado pelos sensores do *tier* 1/RSSF é descrito na Figura 20. Os nós do *tier* 2/*fog computing* realizam parte desse procedimento, exceto os estágios de decisão do líder, já que esses nós são considerados líderes por padrão.

Em situações graves, onde ocorre a perda ou a falha na comunicação entre muitos nós, podendo chegar a casos de nós completamente isolados, os nós isolados do *tier* 1, além de serem considerados seus próprios líderes, são embarcados com modelos de previsão simples, que utilizam somente os dados locais do próprio nó. Dessa forma, apesar de realizar previsões com menor acurácia e sem poder gerar alertas online, os nós ainda podem gerar alertas locais (como luzes e sirenes) e avisar a população próxima da região afetada. Como foram criados os modelos de previsão utilizados pelos nós do *tier* 1, quando estão isolados dos demais, também será descrito na Seção 4.5.

De modo geral, o SENDI garante que as previsões continuaram a ser realizadas, mesmo em casos de falha parcial dos nós ou em casos extremos, nos quais temos somente nós isolados funcionando no sistema.

4.3.7 Dados Utilizados no Estudo de Caso sobre Enchentes

Nesta Subseção, são descritas as características dos dados utilizados no estudo de caso sobre a previsão de enchentes e como esses dados foram tratados.

Como mencionado no Capítulo 2 a RSSF REDE é composta por nós instalados nos rios da cidade de São Carlos. Todos os nós utilizam sensores que fazem a leitura da pressão da água no fundo dos rios e convertem esse valor para a altura em centímetros. Essas leituras são realizadas em intervalos de 5 minutos, com exceção do sensor do shopping. A Figura 21 exemplifica a leitura realizada por um dos sensores, onde o nível de um rio em centímetros (eixo y) é apresentado em função do tempo (eixo x), e utilizada em alguns dos experimentos desta Tese.

Os dados considerados nos experimentos desta Tese foram coletados durante vários períodos, desde o início do funcionamento da RSSF REDE, de acordo com a necessidade de cada experimento. No estudo de caso sobre enchentes, o SENDI foi analisado com os dados de outubro de 2013 e outubro de 2014, coletados pelo nó do shopping, o qual realiza a leitura do nível do rio em intervalos de 1 minuto, além de ser o ponto onde ocorrem enchentes com mais frequência na região. As características do padrão de leitura do sis-

Figura 20 – Fluxograma representando os *rounds* de um nó sensor.

Fonte: Adaptada de Furquim *et al.* (2017).

Figura 21 – Exemplo de leitura realizada pelos sensores da RSSF REDE.

Fonte: [Furquim et al. \(2014a\)](#).

tema REDE foram definidas juntamente com o Núcleo Integrado de Bacias Hidrográficas (NIBH) - EESC - USP/São Carlos, os quais também fazem uso dos dados coletados em seus experimentos hidrológicos.

Nos primeiros experimentos com AM para prever o comportamento dos rios e enchentes, foram utilizados períodos contínuos de leituras dos sensores, incluindo níveis altos, constantes e baixos dos rios.

Já para o treinamento dos modelos do estudo de caso sobre enchentes utilizando o SENDI, as leituras selecionadas foram somente as que representam os momentos de subida e de descida do nível do rio. Essas leituras foram separadas das demais e foi calculada a média de cada 5 leituras consecutivas, formaram o *dataset* da Figura 22.

Outro formato dos dados coletados pelo sistema REDE e utilizado nos experimentos é relativo ao sensor de pluviosidade, conforme Figura 23. Esse tipo de informação, além de proporcionar uma visão mais abrangente do estado do ambiente monitorado, pode ser combinada com o sensor de nível do rio para melhorar a acurácia das previsões de enchente [Furquim et al. \(2016\)](#).

O modo como esses dados foram utilizados, para criar os modelos de previsão e os resultados obtidos por esses modelos e pelo SENDI, são apresentados no Capítulo 4.5.

Figura 22 – *Dataset* utilizado no estudo de caso sobre enchentes.

Fonte: Adaptada de [Furquim *et al.* \(2017\)](#).

Figura 23 – Exemplo de dados coletados pelo sensor de pluviosidade - sistema REDE.

Fonte: Adaptada de [Furquim *et al.* \(2016\)](#).

4.4 Trabalhos relacionados

Nesta Seção, são apresentados os artigos relacionados que apresentam foco em IoT e em tolerância a falhas ou apresentam arquiteturas diferenciadas das demais já apresentadas, e todos são voltados à previsão de desastres naturais.

Um trabalho que utiliza uma visão interessante com RFFS e computação em *cloud* para gerar alertas durante desastres naturais é [Arjun et al. \(2015\)](#), no qual é proposto um *framework Software as a Service* (SaaS) com integração entre RFFS e computação em *cloud*. Em seu sistema, [Arjun et al. \(2015\)](#) aplicam virtualização em nível de rede e de nó da RSSF, o que torna essa estrutura passível de utilização em vários tipos de desastres naturais. Para a análise dos dados, a tomada de decisão e as previsões, utiliza-se o ID3 sobre os dados coletados pelos sensores no ambiente da *cloud*. Os autores não deixam claro como os dados foram coletados, focando em como os dados são analisados e recuperados pelo sistema. Outro ponto não coberto pelo trabalho é como o sistema se comporta em caso de falhas como, por exemplo, problemas na comunicação com a *cloud*.

Outro trabalho sobre desastres naturais é o de [Ryu \(2015\)](#), que apresenta um sistema para a detecção e a análise dos dados sobre incêndios em prédios e construções utilizando IoT e RSSF. Esse sistema prevê a construção dinâmica de rotas de fuga, dependendo da localização do incêndio e das características da construção, as quais são criadas com base nos dados dos sensores, características cognitivas humanas e equipamentos inteligentes para a evacuação de construções. A base da solução proposta por [Ryu \(2015\)](#), assim como a abordagem apresentada nesta Tese, utiliza como base uma rede de sensores sem fio para coleta de dados. No entanto, todos os cálculos são realizados por um servidor central, não contemplando a possibilidade de falha desse servidor ou perda de nós. O tipo de dado utilizado também diferencia dos apresentados nesta Tese, visto que as abordagens para tratamento de desastres naturais são diferente.

Voltando à utilização de RFFS em enchentes, [Andersson e Hossain \(2015\)](#) propõem uma arquitetura de RFFS heterogêneas para ajudar na tomada de decisões em cenários onde ocorrem enchentes. Para tanto, em sua arquitetura, eles padronizam a camada de dados, a camada de transporte e a camada de rede da rede de sensores. Os dados coletados e padronizados podem, então, ser utilizados para alimentar um *Belief Rule Based Expert System* (BRBES), responsável pela ajuda na tomada de decisões (sobre previsão de enchentes, no estudo de caso apresentado no artigo). Considerando que um dos principais pontos do conceito de IoT é assegurar a padronização na comunicação entre objetos, permitindo a utilização de IPv6, o diferencial desse trabalho se encontra nas aplicações das camadas superiores. [Andersson e Hossain \(2015\)](#) descrevem, em sua arquitetura, que os dados são processados na *cloud*, como os dados são processados e quais dados podem ser incluídos, entre eles, características das bacias hidrográficas, o que aumenta a complexidade do modelo de previsão, podendo ser comparável a modelos hidrológicos. Porém,

nenhum dado ou resultado é apresentado. Novamente, situações de falha na *cloud* ou em nós não são contemplados.

Usando uma versão anterior do simulador utilizado nesta Tese, [Lara-Cueva et al. \(2014\)](#) fazem uso do ns-2 para estudar a quantidade necessária de nós de uma RSSF para o monitoramento de vulcões e usam dados reais para validar sua simulação, coletados no vulcão Cotopaxi, Equador. Em seu artigo, [Lara-Cueva et al. \(2014\)](#) consideram algumas métricas de performance, como o *throughput* normalizado, em vários cenários de simulação, variando a posição e o número de nós existentes. Apesar de não utilizarem IoT ou fazerem previsões, esse é um bom exemplo do emprego de RFFS para o gerenciamento de desastres naturais.

Outro estudo encontrado na literatura é apresentado em [Huang et al. \(2016\)](#) e, apesar de apresentarem somente uma arquitetura IoT para o monitoramento dinâmico de declives e de deslizamentos de terra (sem detalhes sobre o equipamento ou modelos de detecção ou previsão), eles realizam coleta de dados em ambiente real e descrevem pontos importantes sobre a aquisição desses dados, sobre a conexão entre os nós, seu posicionamento e instalação. No entanto, não apresentam soluções para falhas que possam ocorrer no sistema nem resultados de predições para comparações.

[Deak et al. \(2013\)](#) apresentam um sistema utilizado para a detecção de pessoas em um ambiente monitorado após a ocorrência de um desastre, com a intenção de ajudar em situações de emergência e resgates. Esse sistema utiliza RSSF Device-free Passive Localization (DfPL) em um cenário IoT, além de filtros sobre os dados coletados que, então, servem como entrada para classificadores tradicionais, como Naive Bayes e TreeBagger, para a detecção da presença de pessoas no ambiente. [Deak et al. \(2013\)](#) destacam que nenhuma computação é realizada nos nós, e sim nos servidores e, apesar de não realizar a previsão dos desastres, serve como uma opção viável para o gerenciamento pós-desastre, porém, não vislumbra falhas nos servidores.

Também com foco nos habitantes das regiões de risco, uma arquitetura interessante foi proposta por [Zambrano et al. \(2016\)](#). Tal arquitetura utiliza a noção de Mobile Crowd-Sensing (MCS), onde uma grande quantidade de *Smartphones* coletam os dados para a previsão de terremotos, geração de alertas e o gerenciamento pós-desastre. Os *Smartphones* dos habitantes da região têm uma aplicação de sensoriamento instalada, para realizar a coleta e o processamento inicial dos dados. Enquanto, na camada 2, existe um servidor intermediário que realiza a análise espacial e temporal dos dados coletados e processados, na camada 3, a arquitetura prevê a utilização de um centro de controle, que distribui informações sobre o risco de terremoto e ajuda no gerenciamento pré e pós-evento. [Zambrano et al. \(2016\)](#) enfatizam que existe uma grande heterogeneidade entre os *Smartphones* e seus sensores, o que resulta em menor acurácia dos resultados e que o consumo de energia, pela aplicação de previsão, é tão importante para o usuário quanto

sua segurança pessoal. Esse fato, apesar de não ser tratado, pode acabar por gerar falta de dados e informações providas dos usuários, já que esses priorizam a vida da bateria de seus *Smartphones*.

Outro trabalho que utiliza *Smartphones* para detectar e reduzir os danos causados por desastres em geral é apresentado em [Konomi et al. \(2016\)](#). Outra vez, o consumo de bateria dos *Smartphones* é considerado um problema, porém é tratado através do uso do acelerômetro de 3-eixos, que auxilia o procedimento de *Bluetooth beacon* para localizar pessoas. Além disso, a arquitetura de [Konomi et al. \(2016\)](#) não se limita à localização de pessoas após desastres, mas também prevê o envio de rotas de fuga calculadas dinamicamente para pedestres, baseadas em diferentes informações, incluindo imagens de câmeras. Apesar de se limitar a apresentar a arquitetura proposta e não apresentar dados ou resultados, algo interessante de ser notado é que o sistema também se originou de uma RSSF, assim como o trabalho desenvolvido nesta tese.

[Fang et al. \(2015\)](#) apresentam um *Integrated Information System* (IIS) para alarmes sobre enchentes provenientes do derretimento de neve. Sua abordagem é baseada em geoinformática e inclui sistemas de informação geográfica (GIS), sistemas de posicionamento global (GPS), IoT e serviços de *cloud*. Na base de seu sistema, [Fang et al. \(2015\)](#) propõem o uso de sensores diversos e nas partes mais altas fazem uso de computação em *cloud*, assim como esta Tese. Como modelos de previsão, são utilizados modelos de escoamento de neve, modelos de análise espacial, modelos de interpolação, modelos de análise estatística, entre outros. O nível do alerta é calculado levando-se em conta diversos fatores, dentre eles a perda econômica e de vidas. Um protótipo do sistema foi instalado na bacia hidrográfica de Quergou, China, e os dados coletados serviram para avaliar e validar o sistema. No entanto, diferente da abordagem proposta nesta Tese, não existe mecanismos de tolerância a falhas, caso ocorra falha de componentes do sistema.

[Mitra et al. \(2016\)](#) apresentam um projeto bem semelhante ao desenvolvido nesta tese, que utiliza RSSF, IoT e AM para a previsão de enchentes, porém é voltado somente para enchentes e não leva em conta a tolerância à falhas. Eles citam que enchentes são o desastre natural mais comum e que causam danos significativos em diversos setores. Como modelo de previsão de enchentes, [Mitra et al. \(2016\)](#) utilizam RNA e, para a simulação, fazem uso do ns-2. Sendo assim, eles não puderam utilizar nenhum módulo que simulasse a pilha de protocolos para IoT, devido ao fato do ns-2 ter sido descontinuado e substituído pelo ns-3. Como experimentos e resultados, [Mitra et al. \(2016\)](#) analisam a relação entre a quantidade de nós e a perda de pacotes e apresentam, como medida de desempenho dos modelos de previsão, a correlação (R). Sendo o coeficiente de determinação o quadrado da correlação, ou seja, R^2 , os resultados obtidos por [Mitra et al. \(2016\)](#) são bem próximos aos obtidos nesta Tese. No entanto, no artigo, a forma como os dados foram selecionados e utilizados não foi descrita com detalhes, não permitindo saber se a base de dados foi

restrita somente a situações de enchentes, ou não. Esse fato, juntamente com a falta de uma porcentagem de erros e acertos, impediu uma melhor análise e comparação dos resultados obtidos.

Em um último trabalho, [Xu, Huang e Fang \(2015\)](#) introduzem o conceito de *cloud asset*, que consiste em um *asset* estendido e com capacidade de identificação, sensoria-mento, comunicação, raciocínio e ação, além de poder ser armazenado, controlado e com-partilhado pela *cloud*. Em seu artigo, [Xu, Huang e Fang \(2015\)](#) descrevem *assets* como barragens, bombas, bacias de retenção, geradores emergenciais, sacos de areia e barracas, visto que o seu sistema é voltado para o controle de enchentes. Nesse ponto, um *cloud asset* seria a combinação dos serviços de rede e de um *smart asset*, que seria uma extensão dos *smart objects* encontrados em IoT, segundo o autor. Para validar e testar a abordagem proposta, [Xu, Huang e Fang \(2015\)](#) utilizam modelos de LEGO que representam *assets* físicos comumente encontrados em sistemas de controle de enchentes urbanas.

A Tabela 1 contém uma comparação entre os artigos relacionados nesta Seção quanto ao uso de: redes de sensores sem fio para coleta de dados (coluna *RSSF*), uso de dados coletados em ambientes reais (*Amb. Real*), uso de técnicas de aprendizado de máquina para criar modelos previsão (*AM*), geração de alertas para a população (*Aler-tas*), se esses alertas estão embutidos nos nós sensores, não sendo somente alertas online (*Integrados*), se o sistema ou modelos de previsão são distribuídos entre os nós (*Distrib.*), fazem uso de tolerância a falhas em suas abordagens (*TF*), fazem uso de tecnologias IoT (*IoT*) e se realizam previsões de desastres naturais (*Previsão*).

Como pode ser observado, existe um aumento na quantidade de trabalhos que utili-zam IoT e AM para previsão de desastres naturais. Porém, poucos trabalhos se preocupam em criar mecanismos de tolerância a falhas, que permitam a continuidade das previsões, mesmo quando existe degradação das suas abordagens, sistemas ou arquiteturas.

Tabela 1 – Tabela de comparação dos artigos relacionados.

Artigo	RSSF	Amb. Real	AM	Alertas	Integrados	Distrib.	TF	IoT	Previsão
Arjun et al. (2015)	✓	✗	✓	✓	✗	✗	✗	✗	✗
Ryu (2015)	✓	✗	✗	✗	✗	✗	✗	✓	✗
Andersson e Hossain (2015)	✓	✓	✓	✗	✗	✗	✗	✓	✗
Lara-Cueva et al. (2014)	✓	✓	✗	✗	✗	✗	✗	✗	✗
Huang et al. (2016)	✗	✓	✗	✓	✗	✗	✗	✓	✓
Deak et al. (2013)	✓	✗	✓	✗	✗	✗	✗	✓	✗
Zambrano et al. (2016)	✗	✓	✗	✓	✓	✗	✗	✓	✗
Konomi et al. (2016)	✓	✓	✗	✓	✓	✗	✗	✓	✗
Fang et al. (2015)	✓	✓	✗	✓	✗	✗	✗	✓	✓
Mitra et al. (2016)	✓	✗	✓	✓	✗	✗	✗	✓	✓
Xu, Huang e Fang (2015)	✓	✗	✗	✗	✗	✗	✗	✓	✗
Arq. Proposta	✓	✓	✓	✓	✓	✓	✓	✓	✓

Fonte: Elaborada pelo autor.

4.5 Criação dos Modelos de Previsão, Análise e Resultados

Esta seção apresenta como foram criados os modelos que executam nos *tiers* 1 e 2 do SENDI, a análise realizada em cada modelo e os resultados obtidos. Esses modelos foram criados a partir dos dados utilizados no estudo de caso sobre previsão de enchentes.

Inicialmente foram realizados estudos sobre a utilização de métodos de aprendizado de máquina para a previsão do comportamento de rios, apresentados na Subseção 4.5.1. Na sequência, foi utilizada a Teoria do Caos para analisar os dados, antes de serem alimentados em uma técnica de AM. Esses experimentos resultaram nos modelos utilizados no *tier* 1 e permitiram selecionar melhor os dados que seriam utilizados para realizar as previsões, sendo que esses resultados são apresentados na Subseção 4.5.2. Finalmente, a Subseção 4.5.3 apresenta a primeira tentativa de uso de múltiplos sensores para a criação de modelos de previsão, que originou a criação do modelo de previsão distribuído, utilizado pelo *tier* 2, descrito na Subseção 4.5.4.

4.5.1 Combinando AM e RSSF para a Previsão de Enchentes

Nesta Subseção, serão apresentados os resultados obtidos através da utilização dos dados coletados pelo sistema REDE e por diversas técnicas de AM. Esse estudo foi realizado e publicado em Furquim *et al.* (2014b) e teve o intuito de comparar e de averiguar a possibilidade da utilização de diversas técnicas de AM para a previsão de enchentes utilizando somente os dados coletados por uma RSSF.

Nesse estudo, as técnicas de AM foram alimentadas com a leitura de um único sensor e a saída desejada era a classificação do estado do rio em níveis predefinidos. Para tanto, foram sugeridas cinco categorias: estável, leve aumento, leve diminuição, grande aumento, grande diminuição. Essas classes são apresentadas na Figura 24, onde encontra-se a correspondência em centímetros e as grades de cada classe.

Foram comparadas sete técnicas de AM e priorizado o estudo de dois fatores principais: (i) número de leituras utilizadas nas previsões, chamada de *tamanho da janela*, e (ii) o tipo de dado utilizado - dados brutos ou atributos dos dados. As técnicas de AM estudadas foram cinco tipos de árvore de decisão: J48, *Random Forest*, *Random Tree*, *BFTree* e *Simple Cart*, um tipo de rede neural artificial - a MLP, e um representante do aprendizado Bayesiano (*Bays Net*). Todas presentes no Weka (HALL *et al.*, 2009), treinadas e analisadas utilizando *10-fold cross-validation* e deixadas com as configurações padrões do *software*.

Os atributos dos dados escolhidos como entrada para as técnicas de AM foram: média, mediana, desvio padrão, soma e diferença entre a primeira e última leitura. A

Figura 24 – Classes para a previsão.

Fonte: Adaptada de Furquim *et al.* (2014b).

Figura 25 – Exemplo de tamanho de janela (W) e tempos de previsão em uma série temporal fictícia de nível de água.

Fonte: Adaptada de Furquim *et al.* (2014b).

janela de dados variou de 5 a 10 leituras. Dessa forma, foram analisadas 4 combinações de tamanhos de janela diferentes e utilizados os dados brutos ou seus atributos.

Como saída, esperava-se o estado do rio nos tempos $t+1$, $t+2$ e $t+3$, considerando t como tempo atual. A combinação de janelas (W), leituras e tempo de previsão são retratados na Figura 25.

Os resultados são apresentados na Tabela 2 e nas Figuras 26, 27 e 28, onde Sx corresponde à previsão em $t+x$, Wy corresponde à janela de tamanho y , A denota a utilização dos atributos dos dados e B a utilização dos dados brutos. Por exemplo, $S2W5_A$ simboliza a previsão em $t+1$, utilizando uma janela de dados de tamanho igual a cinco e uso dos atributos dos dados.

Esses resultados mostram que diferentes técnicas de AM podem gerar previsões

Tabela 2 – Porcentagem de instâncias corretamente classificadas. Variações do tamanho da janela de dados (W) e do tempo de previsão do estado (E) do rio

	W = 5			W = 10		
	E_{t+1}	E_{t+2}	E_{t+3}	E_{t+1}	E_{t+2}	E_{t+3}
Usando dados brutos						
Bays Net	79.754	71.779	62.576	78.797	70.253	60.126
MLP	83.128	73.006	61.656	84.810	72.151	63.291
J48	83.435	68.711	60.736	83.860	71.835	59.493
Random Forest	80.368	69.325	58.282	81.329	68.987	60.126
Random Tree	79.754	68.711	53.067	76.265	63.607	49.367
BFTree	80.981	69.018	60.736	80.379	69.936	59.810
Simple Cart	80.368	67.484	60.122	81.962	68.987	61.708
Usando atributos dos dados						
Bays Net	81.901	74.539	62.883	80.063	74.050	64.240
MLP	*85.276	74.846	62.883	83.544	73.417	*66.772
J48	82.822	75.153	61.656	82.278	71.519	62.974
Random Forest	80.981	71.165	59.202	80.696	73.417	62.658
Random Tree	79.447	71.779	57.668	80.063	71.202	63.924
BFTree	81.595	*75.460	64.110	81.329	73.734	61.075
Simple Cart	82.515	74.539	64.110	82.594	72.784	60.126

Figura 26 – Instâncias corretamente classificadas para $t + 1$. Fonte: Adaptada de Furquim *et al.* (2014b).

com maior acurácia dependendo do cenário. No entanto, janelas que contêm cinco leituras obtiveram melhores resultados, assim como a utilização dos atributos dos dados ao invés de dados brutos somente. Além disso, a técnica de AM que conseguiu maior consistência em seus resultados foi a MLP.

4.5.2 Análise de Dados Utilizando a Teoria do Caos

Os experimentos realizados com técnicas de AM para a previsão de enchentes evidenciaram a necessidade de uma melhor análise dos dados colhidos pelo sistema REDE antes de serem utilizados para alimentar/treinar os modelos de previsão. Tendo isso em

Figura 27 – Instâncias corretamente classificadas para $t + 2$.

Fonte: Adaptada de Furquim *et al.* (2014b).

Figura 28 – Instâncias corretamente classificadas para $t + 3$.

Fonte: Adaptada de Furquim *et al.* (2014b).

mente, os valores do nível dos rios são coletados pelo sistema REDE em intervalos constantes de tempo, o que pode ser considerado uma série temporal. A área de análise de séries temporais permite estudar, modelar e prever sistemas com essas características. Séries temporais são definidas como dados organizados em termos de suas variáveis e de suas observações ao longo do tempo (ISHII; MELLO, 2012; MELLO, 2011), ou seja, uma série temporal é uma sequência de observações ordenadas no tempo e capturadas em intervalos constantes. Portanto, os dados coletados pelo sistema REDE constituem uma série temporal, o que permite analisá-los sob a luz dos conceitos dessa área.

Nesses experimentos, assumimos que existe uma relação temporal entre as observações sobre o nível do rio. Sendo assim, o valor de uma observação no instante atual apresenta alguma dependência em relação aos valores passados. O foco dos experimentos foi em como encontrar essa relação temporal, modelar e prever o comportamento do nível dos rios e, desta forma, melhorar a acurácia das previsões. Para isso, foram utilizadas ferramentas da área de Análise de Séries Temporais, mais precisamente da subárea de Teoria

do Caos. Os resultados foram publicados em [Furquim et al. \(2014a\)](#) e são utilizados no *tier* 1 do SENDI quando o sistema está em nível crítico, ou seja, quando tiver perdido a comunicação com a *cloud* e com os demais nós do *tier* 2.

Um dos principais resultados dessa subárea é o Teorema de Imersão, proposto em [Takens \(1981\)](#), o qual permite que uma série temporal seja reconstruída (desdobrada) em vetores com m valores, onde m é denominado dimensão embutida, e cada valor corresponde a uma observação espaçada de acordo com um deslocamento temporal (*time delay*) ou dimensão de separação, denominada τ . Conforme apresentado em [Mello e Yang \(2009\)](#), esses vetores representam as relações de interdependência entre as observações, o que descreve melhor a série temporal e aumenta a acurácia da previsão em modelagens ([MELLO, 2011](#)). Por exemplo, considerando uma série temporal $X = (x_{-\beta}, \dots, x_{-1}, x_0)$, observada no período de tempo $[-\beta, 0]$, em que $-\beta > 0$, cada vetor reconstruído pelo Teorema de Imersão corresponde a $I = (x_{-\tau \times (m-1)}, \dots, x_{2 \times \tau}, x_0)$. Após a obtenção desses vetores, pode-se empregar técnicas de modelagem que permitam relacionar observações ao longo do tempo e, portanto, realizar predições. Contudo, antes de se obter esses vetores, é necessário estimar valores para as dimensões embutidas (m) e de separação (τ).

De acordo com [Fraser e Swinney \(1986\)](#), a técnica Auto-Mutual Information technique (AMI) apresenta bons resultados para estimar a dimensão de separação (τ). Para tanto, devemos empregar a técnica AMI com vários valores de deslocamento temporal, começando por 1 e incrementando, além de plotar a curva gerada em função de cada valor de deslocamento temporal (*lag*) considerado. O primeiro mínimo obtido em cada gráfico, antes que a curva comece uma nova ascensão, é utilizado como candidato para a dimensão de separação ([MELLO; YANG, 2009](#)). Além disso, outros valores podem ser considerados para a dimensão de separação, tendo em vista as características das curvas geradas.

Após a dimensão de separação ter sido definida, devemos encontrar a dimensão embutida. Para tanto, [Kennel, Brown e Abarbanel \(1992\)](#) propõem o uso do False Nearest Neighbors Method (FNN) para estimar m . Essa técnica calcula os vizinhos mais próximos para cada ponto no espaço, dado um determinado valor de deslocamento temporal, começando por τ igual a 1 e incrementando. Conforme a distância aumenta, dados que antes foram considerados vizinhos, passam a ser considerados falsos vizinhos, o que evidencia a necessidade de mais dimensões para realizar o desdobramento da série. Tipicamente, o valor estimado para m é o valor da dimensão embutida em que temos o FNN igual a zero ([MELLO; YANG, 2009](#)).

Seguindo a abordagem e utilizando os dados de um dos sensores do sistema REDE, o primeiro passo foi calcular o AMI para determinar o valor da dimensão de separação (τ). O resultado desse cálculo é apresentado na Figura 29, onde temos o deslocamento temporal no eixo x (*lag*) e o valor calculado de AMI no eixo y . Com base no gráfico, foi estimado

Figura 29 – AMI para a série temporal do nível do rio.

Fonte: Furquim *et al.* (2014a).

$\tau = 21$. Porém, devido às características do gráfico gerado, onde não existe um mínimo bem definido, podemos dizer que existe uma dependência entre a observação atual e todas as anteriores. Nesses casos, tipicamente, também se considera $\tau = 1$ (MELLO; YANG, 2009). O passo seguinte ao cálculo da dimensão de separação, é o cálculo da dimensão embutida (m) através do método FNN. A Figura 30 apresenta o resultado do cálculo para o conjunto de dados coletados considerando $\tau = 1$ (Figura 30a) e $\tau = 21$ (Figura 30b). No entanto, podemos verificar que, considerando as 12 dimensões, o valor zero não é alcançado, o que dificulta determinar um valor preciso para a dimensão embutida. Esse fato se deve aos ruídos de séries obtidas em ambientes reais, como é o caso deste trabalho. Liebert, Pawelzik e Schuster (1991) sugerem que, em casos como esse, a dimensão seja determinada quando a fração dos falsos vizinhos esteja abaixo de 30% e o valor obtido a partir de 4 para m , quando $\tau = 1$ e a partir de 11 para m , quando $\tau = 21$. Valores maiores que os encontrados para a dimensão embutida (4 e 21, neste caso) podem ser considerados, contudo, segundo Kennel, Brown e Abarbanel (1992), isso não teria tanta influência pois, uma vez desdobrada, o comportamento da série temporal já pode ser compreendido e estudado. Após a reconstrução da série foi empregada a RNA a fim de gerar os modelos de previsão, devido aos resultados obtidos em Furquim *et al.* (2014b).

Os resultados são apresentados nas Figuras 31 e 32 e na Tabela 3, e demonstram que os valores $m = 4$ e $\tau = 1$ apresentaram um melhor resultado, principalmente nas regiões de pico do gráfico do nível do rio, uma parte crítica, pois nos picos é que ocorrem as enchentes. Em ambos os resultados, a MLP utilizou a configuração padrão do software Weka e o *10-fold cross-validation* para teste e análise dos modelos.

Aqui, podemos comparar os resultados obtidos com o trabalho e resultados de Mitra *et al.* (2016), onde foram utilizados sete modelos de previsão de enchentes diferentes, cada um combinando informações de sensores diferentes. Em seu trabalho, Mitra *et al.*

Figura 30 – % de vizinhos próximos falsos considerando.

Fonte: Adaptada de [Furquim et al. \(2014a\)](#).

Tabela 3 – Erro quadrático médio da MLP treinada com a série desdobrada

	Erro médio quadrático da MLP	R^2
$m = 1$ and $\tau = 4$	1.5732	0.9618
$m = 11$ and $\tau = 21$	9.4761	0.1687

Fonte: Adaptada de [Furquim et al. \(2014a\)](#).

(2016) utilizou a MLP para a criação dos modelos, tendo uma camada escondida contendo 12 neurônios, mas não descreve com detalhes como selecionou a quantidade de dados de entrada para cada modelo. Esses resultados são apresentados na Tabela 4, onde o coeficiente de correlação (R^2) não é apresentado no trabalho original, mas é calculado a partir da correlação (R), por motivos de comparação.

Tabela 4 – Correlação e Coeficiente de correlação da MLP

Modelo	Descrição	R	R^2
A	Somente chuva como entrada (RN)	0.5745	0.3301
B	Somente umidade como entrada (HM)	0.2521	0.0636
C	Somente fluxo de água como entrada (WF)	0.8512	0.7245
D	RN + HM	0.9713	0.9434
E	HM + WF	0.8914	0.7946
F	RN + WF	0.9891	0.9783
G	RN + HM + WF	0.9912	0.9825

Fonte: Adaptada de [Mitra et al. \(2016\)](#).

Como podemos comparar, baseado no R^2 , os resultados obtidos e apresentados pela abordagem desta Tese superam cinco dos modelos apresentados por [Mitra et al. \(2016\)](#), perdendo somente para os dois últimos modelos, que utilizam dados de mais de um sensor.

Figura 31 – Nível esperado do rio e nível previsto para $\tau = 1$ com $m = 4$. Figura 32 – Nível esperado do rio e nível previsto para $\tau = 11$ com $m = 21$.

4.5.3 Previsões Utilizando Combinação de Diferentes Tipos de Sensores

A primeira tentativa de utilizar mais de um dos sensores do sistema REDE para realizar a previsão de enchentes é descrita em [Furquim *et al.* \(2016\)](#), que estende [Furquim *et al.* \(2014a\)](#), e utiliza os dados coletados pelo pluviômetro na tentativa de aumentar a acurácia dos modelos de previsão. Nesses experimentos, é feita uma comparação com os resultados obtidos anteriormente, os quais não levaram o pluviômetro em conta. Além disso, a Rede Neural Recursiva (RNR) de Elman também foi estudada para prever enchentes e os resultados obtidos foram comparados à MLP, visto que a RNR não havia sido incluída em [Furquim *et al.* \(2014b\)](#) e é conhecida por ser uma opção viável para modelar séries temporais.

Os mesmos passos de desdobramento de séries temporais descritos em ([FURQUIM *et al.*, 2014a](#)) foram utilizados nos dados coletados para esses experimentos.

A Tabela 5 mostra a relação entre o nível do rio e o volume de chuva ao longo do

tempo, durante um período de pico em que o nível do rio chegou a 3 m, ocasionando uma enchente na região. O ápice da enchente não aparece na tabela por não conter volume de chuva significativa durante o evento. Como pode ser observado na Tabela, dado um instante t_0 no tempo, no qual o pluviômetro indicou algum volume de chuva, o nível do rio parece ser influenciado por esse volume de chuva no instante seguinte (t_1), porém isso varia quando analisamos outras partes das séries. Devido a essa característica, foram utilizados valores de tamanho de janela entre 1 e 10 leituras do pluviômetro.

Tabela 5 – Nível do rio e volume de chuva na enchente de 13/04/2014.

Tempo (hh:mm)	Nível do rio (cm)	Volume de chuva (mm)
16:31	23.24	0
16:36	23.55	0
16:41	23.71	0
16:46	22.45	0
16:51	23.55	6.4
16:56	36.74	5.8
17:01	75.99	5.2
17:06	100.95	8.2
17:11	139.42	8
17:16	200.18	4.8
17:21	231.10	2.8
17:26	242.09	2.4
17:31	244.29	1.4
17:36	236.44	0
17:41	238.48	0
17:46	217.92	0

Fonte: Adaptada de [Furquim et al. \(2016\)](#).

A Tabela 6 apresenta os resultados dos experimentos que alimentaram a MLP com entradas da série temporal desdobrada do nível do rio, juntamente com os dados do volume de chuva, variando o tamanho da janelas de dados do pluviômetro. Os parâmetros utilizados para a análise dos resultados, foram: Mean Absolute Error (MAE), Root Mean Square Error (RMSE) e Coeficiente de Determinação (R^2). A MLP foi configurada com uma camada escondida contendo 3 neurônios, taxa de aprendizagem = 0.3 e *10-fold cross-validation* para teste e análise. Por motivos de comparação, a linha “Sem volume de chuva” apresenta os resultados obtidos sem o uso da série temporal do volume de chuva.

Novamente, podemos comparar os resultados obtidos nesses experimentos com os obtidos por [Mitra et al. \(2016\)](#) (Tabela 7). Desta vez, não somente a inclusão do pluviômetro, como também a utilização somente dos dados de nível do rio, superaram os resultados obtidos por todos os modelos de previsão de [Mitra et al. \(2016\)](#), incluindo os que utilizaram mais de um tipo de sensor. Desta forma, podemos constatar que, como esperado, a acurácia dos modelos de previsão dependem da série utilizada para o treinamento das

Tabela 6 – Nível do rio e volume de chuva MLP: MAE, RMSE e R^2 .

Tamanho da janela	MAE	RMSE	R^2
Sem volume de chuva	1.1058	1.4029	0.9944
1	1.1037	1.4028	0.9944
2	1.1241	1.4234	0.9944
3	1.1126	1.4143	0.9944
4	1.1196	1.4230	0.9944
5	1.1220	1.4255	0.9942
6	1.1236	1.4344	0.9942
7	0.9189	1.1806	0.9960
8	0.9881	1.2748	0.9954
9	1.0624	1.4236	0.9946
10	1.0324	1.3654	0.9948

Fonte: Adaptada de [Furquim et al. \(2016\)](#).

técnicas de AM. Desta forma, séries diferentes, mesmo contendo o mesmo tipo de dado sobre um mesmo ambiente, podem gerar modelos com acurácia diferente. Outro ponto, demonstrado pela comparação entre as Tabelas, é que os resultados apresentados nesta Tese se assemelham aos de [Mitra et al. \(2016\)](#), apresentando acurácia tão boa ou superior.

Tabela 7 – Correlação e Coeficiente de correlação da MLP

Modelo	Descrição	R	R^2
A	Somente chuva como entrada (RN)	0.5745	0.3301
B	Somente umidade como entrada (HM)	0.2521	0.0636
C	Somente fluxo de água como entrada (WF)	0.8512	0.7245
D	RN + HM	0.9713	0.9434
E	HM + WF	0.8914	0.7946
F	RN + WF	0.9891	0.9783
G	RN + HM + WF	0.9912	0.9825

Fonte: Adaptada de [Mitra et al. \(2016\)](#).

Outro conjunto de experimentos visava avaliar a aplicação da RNR para prever enchentes e comparar seus resultados aos dos modelos criados pela MLP. Para esses experimentos, foi considerado o melhor caso de desdobramento obtido pela MLP, ou seja, com $\tau = 1$ e $m = 4$ (vide Tabela 8). Tanto a MLP quanto a RNR foram configuradas com uma camada escondida contendo 3 neurônios, taxa de aprendizagem = 0.3 e *10-fold cross-validation* para teste e análise.

Experimentos com a RNR e os dados sobre o volume de chuva também foram realizados. Os resultados encontram-se na Tabela 9, utilizando a mesma configuração de RNR do experimento anterior.

Uma informação importante no que tange a previsão de enchentes é o desempenho das técnicas de previsão em momentos de pico no nível do rio. A Figura 33 compara os

Tabela 8 – Nível do rio em abril de 2014 - MLP e RNR: MAE, RMSE e R^2 ($\tau = 1$ e $m = 4$).

	MAE	RMSE	R^2
MLP	1.1058	1.4029	0.9944
RNR	1.0540	1.3392	0.9950

Fonte: Adaptada de [Furquim et al. \(2016\)](#).

Tabela 9 – Nível do rio e volume de chuva em abril de 2014 - RNN: MAE, RMSE and R^2 .

Tamanho da janela	MAE	RMSE	R^2
Sem volume de chuva	1.0540	1.3392	0.9950
1	1.0621	1.3500	0.9950
2	1.0504	1.3370	0.9952
3	1.0802	1.3718	0.9950
4	1.0094	1.2859	0.9954
5	1.0249	1.3066	0.9952
6	1.0396	1.3299	0.9952
7	1.1077	1.4132	0.9944
8	1.0062	1.3147	0.9952
9	1.0304	1.3299	0.9954
10	0.9460	1.2770	0.9954

Fonte: Adaptada de [Furquim et al. \(2016\)](#).

valores reais (obtidos pelo sensor de nível) aos previstos pelas técnicas utilizadas durante o maior pico da série temporal, momento em que ocorreu uma enchente em São Carlos. Como observado, as melhores representações do pico da série temporal foram obtidas pela MLP, com e sem a utilização dos dados do pluviômetro. A RNR, apesar de acompanhar as mudanças na série, subestima os valores de pico do nível do rio e não representa com tanta fidelidade o fenômeno da enchente.

Com esses experimentos, concluímos que, na maioria dos casos, o uso do pluviômetro melhorou o desempenho dos modelos criados pela MLP, porém não de maneira constante. Em alguns casos, o ganho na acurácia pela inserção dos dados do pluviômetro não foi significativo. Isso pode ter ocorrido devido à existência de somente um pluviômetro no sistema REDE, o que impede uma análise mais completa da chuva, visto que essa pode vir de várias direções. Além disso, a RNR obteve bons resultados e superou a MLP na maioria dos casos. No entanto, em momentos críticos, como nos pico da série temporal, a MLP superou a RNR e continua a ser uma boa opção para prever enchentes.

Figura 33 – Nível do rio esperado e previsto durante a enchente.

Fonte: Adaptada de [Furquim et al. \(2016\)](#).

4.5.4 Previsões Utilizando Combinação de Diferentes Sensores de Nível

Com base nos experimentos anteriores, foi criado o modelo de previsão utilizado pelo *tier 2* do SENDI quando ocorre a perda da comunicação com a *cloud*. Nesses novos experimentos, vários sensores combinam seus dados e previsões para gerar previsões mais precisas sobre o comportamento do ambiente e distribuir a carga de processamento necessária para o tratamento desses dados e para a execução dos modelos de previsão.

A abordagem proposta, publicada em [Furquim et al. \(2016\)](#), explora as leituras e a capacidade de processamento de múltiplos sensores para realizar previsões de enchentes, ao invés de apenas um único sensor. Dessa forma, podemos explorar melhor a capacidade de processamento dos nós de uma RSSF, além de dar um grau maior de liberdade aos sensores, principalmente em casos onde a comunicação com a base central seja perdida. Nessa abordagem, representada na Figura 34, os sensores *s1* e *s2* realizam parte do processamento e enviam seus resultados para o sensor *s3*. Esse processamento consiste em prever o nível do rio para o instante $t + 1$ (5 minutos no futuro), para as suas respectivas localizações através das próprias leituras somente.

Após enviadas ao sensor *s3*, as previsões locais em $t + 1$ dos sensores *s1* e *s2* foram combinadas com as leituras locais do sensor *s3*, onde foi executado o restante do

Figura 34 – Representação da abordagem de previsão distribuída.

processamento. Nesse estudo, as técnicas de AM, SVM, GP e MLP foram aplicadas a fim de gerar modelos de previsão para o sensor $s3$ e, dessa vez, os dados e as previsões dos 3 sensores foram utilizados.

Primeiramente, foram comparadas as 3 técnicas de AM para determinar qual delas seria mais apropriada para gerar previsões em $t + 1$ nos sensores $s1$ e $s2$. O resultado dessa comparação é apresentado na Tabela 10, onde todas as técnicas utilizaram o *10-fold cross-validation* para teste e análise, sendo que a MLP foi configurada com taxa de aprendizagem = 0.3 e uma camada escondida com 6 neurônios. Como pode ser visto, a técnica GP conseguiu melhores resultados em ambos os sensores e passou a ser utilizada nos experimentos seguintes.

Tabela 10 – Performance das técnicas de AM para análise e uso na abordagem distribuída.

	SVM (epsilon-SVR)		SVM (nu-SVR)		GP		MLP	
Sensor	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE
$s1$	0.5456	0.2132	0.5439	0.1956	0.6069	0.1804	0.5935	0.1836
$s2$	0.3484	1.4142	0.3481	1.4257	0.6051	1.0934	0.5679	1.1547
$s3$	0.6059	1.7499	0.6040	1.7486	0.8731	0.9796	0.8794	0.9334

Fonte: Adaptada de [Furquim et al. \(2016\)](#).

A abordagem distribuída de previsão de enchentes proposta também visa a analisar se as previsões realizadas por grupos de sensores ($s1$ e $s2$, no caso) podem melhorar a acurácia de previsões mais distante no futuro, realizadas por outros sensores ($s3$, no caso). Para isso, no sensor $s3$, além de serem gerados modelos de previsão para o instante $t + 1$

Tabela 11 – Resultado das técnicas de AM para a previsão distribuída em $t + 1$.

	SVM (epsilon-SVR)		SVM (nu-SVR)		GP		MLP	
	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE
s3	0.6059	1.7499	0.6040	1.7486	0.8731	0.9796	0.8794	0.9334
Distribuída	0.5864	1.7979	0.5844	1.7986	0.8858	0.9534	0.8805	0.9270

Fonte: Adaptada de [Furquim et al. \(2016\)](#).Tabela 12 – Resultado das técnicas de AM para a previsão distribuída em $t + 2$.

	SVM (epsilon-SVR)		SVM (nu-SVR)		GP		MLP	
	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE
s3	0.5487	1.8463	0.5482	1.8453	0.7661	1.3167	0.7784	1.2613
Distribuída	0.5872	1.7943	0.5856	1.7937	0.7961	1.2302	0.8065	1.1787

(5 minutos no futuro), utilizou-se a mesma combinação de dados e de técnicas de AM para gerar modelos de previsão para o instante $t + 2$ (10 minutos no futuro).

Os resultados obtidos com esses experimentos constam na Tabela 11, que apresenta o desempenho das técnicas de AM utilizadas para gerar os novos modelos de previsão da abordagem distribuída para o tempo $t + 1$ nos sensores $s1$ e $s2$. A primeira linha da tabela apresenta os resultados obtidos sem o uso da distribuição, devido à comparação, e o melhor desempenho foi destacado em negrito. As configurações das técnicas de AM se mantiveram as mesmas.

Como pode ser observado na Tabela 11, os índices de desempenho discordam quanto ao melhor modelo criado pelas técnicas de AM. O GP obteve melhor desempenho quando consideramos o R^2 e a MLP, quando consideramos o RMSE. Outro ponto a ser destacado é que ambos os modelos de previsão para $t + 1$, gerados pela MLP ou pelo GP, melhoraram seu desempenho quando utilizados na abordagem distribuída. No entanto, ambos os *kernels* da SVM pioram seu desempenho.

A Tabela 12 ilustra a mesma abordagem distribuída e utiliza as previsões para o tempo $t + 1$ dadas pelos sensores $s1$ e $s2$. No entanto, nesses experimentos, os modelos de previsão gerados pelas técnicas de AM para o sensor $s3$, realizam previsões para o tempo $t + 2$. Dessa forma, as previsões foram realizadas para 10 minutos no futuro, o que permite gerar alertas com maior tempo para a tomada de decisões. Para comparação, a primeira linha da tabela apresenta os resultados obtidos sem a utilização da abordagem distribuída e o melhor desempenho foi destacado em negrito. As configurações das técnicas de AM se mantiveram as mesmas.

Diferentemente do experimento anterior, todos os modelos gerados pelas técnicas de AM obtiveram melhor desempenho quando utilizamos a abordagem distribuída para previsões do instante $t + 2$ e os melhores resultados foram obtidos pela MLP, dessa vez,

Figura 35 – Comparação gráfica entre as leituras reais e as previsões realizadas para o instante $t + 1$.

em ambos os índices de desempenho.

Podemos verificar o desempenho dos modelos de previsão gerados pelas técnicas de ML, tanto para $t + 1$ quanto para $t + 2$ nas Figuras 35 e 36. Como observado, em ambos os casos ocorre uma defasagem entre os valores estimados pela MLP e as leituras reais do nível do rio quando não utilizamos a abordagem distribuída. Essa defasagem é corrigida quando a abordagem distribuída é aplicada e, além disso, os valores de pico são melhor representados, o que não ocorre quando utilizamos GP.

4.6 SENDI - Experimentos e Resultados

A simulação realizada com o SENDI, serviu tanto para o estudo de caso sobre enchentes quanto para a validação do sistema. Em seu *tier* 1, o SENDI utiliza os modelos de previsão descritos na Subseção 4.5.2, enquanto que em seu *tier* 2 é utilizada a abordagem de previsão distribuída, descrita na Subseção 4.5.4.

Como o módulo lr-wpan do ns-3 ainda não possui métricas para energia, o modelo simples de dissipação de energia de rádio, descrito na Subseção 4.3.6 foi adotado na simulação a fim de determinar o consumo de energia durante a transmissão e a recepção dos dados. Na Tabela 13 encontram-se os parâmetros adotados na simulação nas equações descritas na Subseção 4.3.6, assim como a quantidade de energia utilizada pelo nó líder

Figura 36 – Comparação gráfica entre as leituras reais e as previsões realizadas para o instante $t + 2$.

Tabela 13 – Parâmetros de simulação.

E_{elect}	$5 * 10^{-8}$ nJ/bit
ϵ_{fs}	$1 * 10^{-11}$
ϵ_{mp}	$1.3 * 10^{-15}$
E	15000 J
E_f	241920 J
d	87.7 m
n	100
Voltagem and Amperagem (inativo)	0.5 A / 5.0 V
Consumo nó líder	50 J

Fonte: Adaptada de [Heinzelman, Chandrakasan e Balakrishnan \(2002\)](#).

do *cluster* para cada nó pertencente ao seu *cluster*. Esse valor foi adotado por impactar a quantidade de energia armazenada no nó líder e gerar a necessidade de um mecanismo de controle de energia, além de permitir a utilização de várias técnicas de AM, a combinação dessas técnicas ou mesmo o tratamento de vários tipos de dados, coletados por diferentes sensores, como imagens, nível de rio ou volume de chuva, entre outros.

Os experimentos iniciais consideraram uma situação em que a comunicação com os nós *sink* e a *cloud* foi interrompida, fazendo-se necessário o uso do mecanismo de seleção de líder. Esses experimentos consistiram na repetição de *rounds* até que a energia total do sistema fosse consumida e todos os nós ficassem inativos. Definimos *round* como um

Tabela 14 – Número de nós ativos por *round*.

α	round 18	round 19	round 20
0.0	100	5	0
0.1	100	7	0
0.2	100	4	0
0.3	100	6	0
0.4	100	8	0
0.5	100	12	0
0.6	100	6	0
0.7	100	6	0
0.8	100	10	0
0.9	100	10	0
1	100	7	0
0.5 (500 nós)	500	464	36

Fonte: Adaptada de [Furquim et al. \(2015\)](#).

ciclo de leitura de dados, de disseminação dos dados e de processamento para realizar a previsão.

O modelo foi configurado com 100 nós no *tier* 1 e executado 33 vezes para cada valor de α adotado (Equação 4.5, Subseção 4.3.6) e as médias da quantidade de nós operantes em cada *round* são apresentadas na Tabela 14.

Como pode ser notado, devido às características da simulação, a energia consumida em cada *round* permanece quase constante e o que varia é o modo com que o consumo é distribuído entre os nós. Dessa forma, em todas as execuções os nós começaram a ficar inoperantes no mesmo instante de tempo e somente a quantidade de nós cuja energia foi totalmente drenada apresentou diferença. Assim, valores de α como 0.8, 0.9 e principalmente 0.5 distribuíram melhor a carga entre os nós e permitiram que mais nós permanecessem operantes nos *rounds* seguintes. Esse é um fator importante já que, com mais nós ativos, uma maior quantidade de dados pode ser coletada, o que abrange uma região maior que a área monitorada e, dessa forma, é possível realizar previsões com maior acurácia ou mesmo alertar um maior número de pessoas. A última linha da tabela ilustra o melhor valor de α com o intuito de estudar a escalabilidade da abordagem utilizada.

Para os experimentos relacionados à previsão de enchentes, foram utilizados os dados do sensor localizado na região do shopping de São Carlos, que realiza a leitura do nível do rio em intervalos de 1 minuto. Considerando o tempo atual como sendo t , foram considerados enchentes casos onde a média de $t + 1$ até $t + 5$ (5 minutos no futuro) ou a média de $t + 6$ até $t + 10$ (10 minutos no futuro) foram superiores a 2 metros (esse valor foi definido devido às características da região). Foram classificados como alerta vermelho, quando a média entre o intervalo de $t + 1$ até $t + 5$ superava o valor de 2 metros, considerados como alta probabilidade de enchentes. Já a média entre $t + 6$ e $t + 10$ foi

Tabela 15 – Porcentagem de acertos e erros na previsão (alerta vermelho).

		Classe prevista	
		positivo	negativo
Classe verdadeira	Positivo	28	4
	Negativo	6	14

Fonte: Adaptada de Furquim *et al.* (2017).

Tabela 16 – Porcentagem de acertos e erros na previsão (alerta amarelo).

		Classe prevista	
		positivo	negativo
Classe verdadeira	Positivo	20	14
	Negativo	2	11

Fonte: Adaptada de Furquim *et al.* (2017).

considerada para gerar alertas amarelos, que indicam a possibilidade de enchentes. Os resultados foram avaliados considerando as seguintes medidas de desempenho: sensibilidade, precisão, especificidade e acurácia e são apresentados nas Tabelas 15 e 16. Novamente, a MLP utilizou taxa de aprendizagem = 0.3, *10-fold cross-validation* para teste e análise, tendo uma camada escondida com 3 neurônios.

Segundo Fawcett (2006), tais medidas possuem características inerentes. A sensibilidade é o total de amostras cuja classe resultante é realmente positiva (verdadeiros positivos); a precisão é o total de exemplos classificados como positivo, mas que nem sempre são, ou seja, os verdadeiros negativos; já a especificidade é o oposto da sensibilidade, se importando somente com a classificação negativa dos exemplos que são negativos de fato (falsos positivos) e a acurácia permite analisar o quão precisamente o método classifica o comportamento de um equipamento ou ambiente. Seguindo esse método, a avaliação da capacidade de previsão do sistema SENDI é apresentada na Figura 37.

Como podemos notar na Figura 37 e nas Tabelas 15 e 16, a acurácia em ambos os casos (alerta vermelho e amarelo) supera os 65% e o alerta vermelho atinge níveis de acurácia maiores que 80%. De maneira similar, a porcentagem de falsos positivos do alerta vermelho é também inferior à porcentagem do alerta amarelo. Para um análise mais detalhada, podemos considerar a quantidade de falsos negativos, ou seja, lugares onde existem enchentes não detectadas pelo sistema. Nesse quesito, o alerta vermelho supera em muito os resultados obtidos pelo alerta amarelo, apresentando uma taxa de falhas muito menor.

Apesar da situação analisada descrever uma situação com inúmeras falhas, onde existe uma forte degradação do SENDI, o sistema ainda é capaz de gerar alertas e de avisar a população local. Em cenários melhores, nos quais é possível usar a previsão distribuída ou mesmo a *cloud*, os resultados, tanto para o alerta vermelho quanto para o

Figura 37 – Análise dos resultados obtidos pelo SENDI.

Fonte: Adaptada de [Furquim et al. \(2017\)](#).

alerta amarelo, podem ser melhorados.

4.7 Considerações Finais

Este Capítulo apresentou o SENDI, sua arquitetura, como são utilizados os vários modelos de previsão, como é realizada a organização ou reorganização dos nós em momentos críticos e seu mecanismo de tolerância a falhas. Essa abordagem explora as tecnologias voltadas para IoT, tendo como base uma RSSF para coleta de dados, constituindo, desta forma, um sistema flexível para previsão de desastres naturais.

Também neste capítulo, foram apresentadas e ilustradas as ferramentas utilizadas para simulação do SENDI, assim como detalhes sobre os dados utilizados na validação do sistema, no estudo de caso e na criação dos modelos de previsão, além dos trabalhos relacionados à utilização de IoT para previsão de desastres naturais.

Os experimentos com os modelos de previsão foram conduzidos de forma incremental até atingir um nível no qual foi obtido um sistema tolerante às falhas mais comumente encontradas em RSSFs e capaz de continuar realizando previsões de desastres naturais mesmo na ocorrência de falhas.

Inicialmente, foram realizados experimentos com os dados de um único sensor e

com várias técnicas de AM a fim de saber se ambos eram viáveis para prever desastres naturais, especificamente, no estudo de caso sobre enchentes. Esses experimentos demonstraram a importância de uma análise mais profunda antes da utilização de qualquer técnica de AM. Para realizar essa análise, utilizou-se a análise de séries temporais, mais especificamente, a teoria do Caos. Nela, foi definida a quantidade de leituras passadas necessária para realizar previsões sobre a série temporal do nível do rio. Nesse ponto, pudemos definir o funcionamento do *tier* mais baixo do SENDI e realizar as previsões com menor acurácia do sistema.

Após essas definições, a utilização de múltiplos sensores para realizar a previsão de desastres foi algo natural. Nesse ponto, vários sensores combinaram seu poder de processamento e dados para gerar previsões e a técnica foi empregada no *tier* 2 do SENDI. Os resultados obtidos indicam que previsões com maior acurácia podem ser obtidas através da combinação de dados de múltiplos sensores e que a distribuição da carga de processamento é possível e pode beneficiar a RSSF como um todo.

Finalmente, se beneficiando dos resultados anteriores e de novas tendências tecnológicas como a IoT, a arquitetura do SENDI foi criada. Essa arquitetura foi simulada e validada através do estudo de caso sobre enchentes. Nela, a tolerância a falhas é implementada através de vários *tiers*, mecanismos de eleição de líderes e múltiplos modelos de previsão, o que permite que as previsões continuem a ser realizadas mesmo em situações de falha no sistema.

CONCLUSÃO

Desastres naturais são eventos que acontecem em todo o mundo e são inevitáveis. Porém esses eventos podem ser, muitas vezes, previstos, o que viabiliza a análise de risco, tomada de decisão e gerenciamento e intervenções nas áreas de risco ou potencialmente perigosas. Todavia, a maioria das abordagens e sistemas utilizados para esse fim não vislumbram a possibilidade de falhas no sistema ou são muito complexos, exigindo uma grande quantidade de informações sobre a bacia hidrográfica dos rios estudados.

A abordagem proposta nesta Tese permite que essas previsões sobre desastres naturais sejam realizadas, utilizando tecnologias atuais, com uma acurácia comparável ou superior a outros resultados apresentados na literatura e aumenta, desta forma, as chances de mitigar os efeitos causados por esses desastres.

5.1 Síntese das Contribuições

Entre as principais contribuições desta Tese, pode-se ressaltar:

- Esta pesquisa permitiu a análise do uso de diferentes técnicas de aprendizado de máquina para previsão de desastres naturais, focando no estudo de caso sobre previsão de enchentes, realizando uma comparação entre essas técnicas em diferentes cenários. Com base nesses estudos, foi possível constatar que tipos de dados diferentes, o tratamento recebido por esses dados e a técnica de AM selecionada, influenciam de forma visível na acurácia da previsão. Esses resultados geraram a necessidade de definir um maneira mais precisa de selecionar a entrada de dados que gere previsões com maior acurácia, dado uma determinada técnica de AM.
- A análise de séries temporais foi a forma escolhida para selecionar e processar os dados coletados pelos sensores do sistema REDE, antes de serem usados para ali-

mentar as técnicas de AM. O uso desta análise, mais especificamente sua sub-área, a Teoria do Caos, permitiu visualizar as leituras dos sensores de uma forma diferente e selecionar de maneira não arbitrária os dados utilizados para gerar os modelos e, desta forma, aumentar a acurácia das previsões realizadas. Tendo resultados comparáveis ou superiores a outros apresentados na literatura, a aplicação e análise dessa técnica para a previsão de desastres naturais é mais uma contribuição desta Tese.

- A análise da combinação dos dados e previsões de múltiplos sensores a fim de aumentar a acurácia das previsões de desastres naturais e o estudo sobre a distribuição da carga de processamento necessária para realizar tais operações, foi outro tópico abordado por esta Tese. A partir dessa análise, foi possível determinar quais sensores e quais informações desses sensores foram mais relevantes para realizar previsões com maior acurácia, quando comparamos com o uso das informações de somente um sensor isolado.
- Durante o desenvolvimento desta Tese, novos caminhos e tecnologias para monitoramento de ambientes foram ganhando destaque na literatura. Um dos conceitos que mais se destacou foi a IoT e as tecnologias e padrões voltados para essa área. O uso de IPv6, em uma versão menos complexa abre diversas possibilidades, entre elas, a aplicação para monitoramento e previsão de desastres naturais. Por esse motivo, nesta Tese foi feito um estudo sobre a utilização de tecnologias IoT voltadas para previsão de desastres naturais, contribuindo, assim, com o estudo da arte nessa área.
- O desenvolvimento, análise e validação, através de um estudo de caso sobre enchentes, da arquitetura de previsão de desastres naturais baseado em IoT, AM e tolerante a falhas chamada SENDI, pode ser considerada como contribuição final e faz uso das demais contribuições desta Tese. Além disso, o SENDI ainda apresenta uma abordagem de reorganização dos nós e dos vários *tiers* presentes em sua arquitetura, o que garante que as previsões continuem a ser realizadas, mesmo em casos de degradação forte dos componentes do sistema.

5.2 Limitações e Trabalhos Futuros

Após realizar esse estudo, foi possível verificar falhas e consequentes áreas que podem ser exploradas em trabalhos futuros, como as seguintes:

- Os modelos de previsão utilizados nesta Tese necessitam de dados históricos para serem criados. Outras abordagens, como o aprendizado online, conseguem criar modelos sem essa limitação, podendo ser uma alternativa a ser estudada.
- Realizar um estudo sobre quais dados, disponíveis online, podem ser combinados com os dados coletados pelo sistema REDE para aumentar a acurácia das previsões.

- Utilizar as imagens coletadas pelos sensores portadores de câmeras para auxiliar as previsões.
- Técnicas de *Deep Learning* para análise de imagens ganharam bastante destaque durante o desenvolvimento desta Tese. E, apesar de exigirem muito processamento, essas técnicas podem ser executadas na camada superior do SENDI (*cloud*), juntamente com os leituras sobre o nível do rio e pluviosidade.
- Comparação entre o item anterior e abordagem que utilizam modelos hidrológicos.
- Empregar tolerância a falhas em outros níveis do SENDI como, por exemplo, no *hardware* do sistema.
- Análise do SENDI utilizando outros estudos de caso, com diferentes tipos de dados e desastres naturais.

5.3 Publicações e Trabalhos Desenvolvidos

FURQUIM, G.; NETO, F.; PESSIN, G.; UHEYAMA, J.; ALBUQUERQUE, J. D.; CLARA, M.; MENDIONDO, E.; SOUZA, V. D.; SOUZA, P. D.; DIMITROVA, D.; BRAUN, T. Combining wireless sensor networks and machine learning for flash flood now-casting. In: Advanced Information Networking and Applications Workshops (WAINA), 2014 28th International Conference on. [S.l.: s.n.], 2014. p. 67–72.

FURQUIM, G.; MELLO, R.; PESSIN, G.; FAIÇAL, B.; MENDIONDO, E.; UHEYAMA, J. An accurate flood forecasting model using wireless sensor networks and chaos theory: A case study with real wsn deployment in Brazil. In: MLADENOV, V.; JAYNE, C.; ILIADIS, L. (Ed.). Engineering Applications of Neural Networks. Springer International Publishing, 2014, (Communications in Computer and Information Science, v. 459). p. 92–102. ISBN 978-3-319-11070-7.

FURQUIM, G.; PESSIN, G.; GOMES, P. H.; MENDIONDO, E. M.; UHEYAMA, J. A distributed approach to flood prediction using a wsn and ml: A comparative study of ml techniques in a wsn deployed in brazil. In: . Intelligent Data Engineering and Automated Learning – IDEAL 2015: 16th International Conference, Wroclaw, Poland, October 14–16, 2015, Proceedings. Cham: Springer International Publishing, 2015. p. 485–492. ISBN 978-3-319- 24834-9.

FURQUIM, G.; PESSIN, G.; FAIÇAL, B. S.; MENDIONDO, E. M.; UHEYAMA, J. Improving the accuracy of a flood forecasting model by means of machine learning and chaos theory. Neural Computing and Applications, v. 27, n. 5, p. 1129–1141, 2016. ISSN 1433-3058.

REFERÊNCIAS

AKHLAGHINIA, M.; LOTFI, A.; LANGENSIEPEN, C. Soft computing prediction techniques in ambient intelligence environments. **Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International**, p. 1 – 6, July 2007. Citado nas páginas 28, 33 e 34.

AKYILDIZ, I.; SU, W.; SANKARASUBRAMANIAM, Y.; CAYIRCI, E. A survey on sensor networks. **Communications Magazine, IEEE**, v. 40, n. 8, p. 102–114, 2002. ISSN 0163-6804. Citado na página 32.

ANDERSSON, K.; HOSSAIN, M. S. Heterogeneous wireless sensor networks for flood prediction decision support systems. In: **2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)**. [S.l.: s.n.], 2015. p. 133–137. Citado nas páginas 69 e 72.

ARJUN, D.; BALA, A.; V, D.; S, S. K.; B, P. R. B.; PASUPULETI, H. Integrating cloud-wsn to analyze weather data and notify saas user alerts during weather disasters. In: **2015 IEEE International Advance Computing Conference (IACC)**. [S.l.: s.n.], 2015. p. 899–904. Citado nas páginas 25, 41, 69 e 72.

ARMBRUST, M.; FOX, A.; GRIFFITH, R.; JOSEPH, A. D.; KATZ, R.; KONWINSKI, A.; LEE, G.; PATTERSON, D.; RABKIN, A.; STOICA, I.; ZAHARIA, M. A view of cloud computing. **Commun. ACM**, ACM, New York, NY, USA, v. 53, n. 4, p. 50–58, abr. 2010. ISSN 0001-0782. Disponível em: <<http://doi.acm.org.ez67.periodicos.capes.gov.br/10.1145/1721654.1721672>>. Citado na página 38.

ASHTON, K. That ‘internet of things’ thing. **RFiD Journal**, v. 22, n. 7, p. 97–114, 2009. Citado na página 37.

ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. **Computer Networks**, v. 54, n. 15, p. 2787 – 2805, 2010. ISSN 1389-1286. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1389128610001568>>. Citado nas páginas 37 e 38.

AVIŽIENIS, A. Design of fault-tolerant computers. In: **Proceedings of the November 14-16, 1967, Fall Joint Computer Conference**. New York, NY, USA: ACM, 1967. (AFIPS '67 (Fall)), p. 733–743. Disponível em: <<http://doi.acm.org.ez67.periodicos.capes.gov.br/10.1145/1465611.1465708>>. Citado na página 51.

BARBOSA, R. V. R.; VECCHIA, F. A. S. Estudos de ilha de calor urbana por meio de imagens do landsat 7 etm+: Estudo de caso em são carlos (sp). **Revista Minerva - Pesquisa & Tecnologia**, FUNDAÇÃO PARA O INCREMENTO DA PESQUISA E DO APERFEIÇOAMENTO INDUSTRIAL, v. 6, n. 3, p. 273–278, 2009. Citado na página 25.

BARCENA, A. D. L.; MARCANO, A.; ANDINA, D.; PIÑUELA, J. Modeling logic and neural approaches to bankruptcy prediction models. **World Automation Congress (WAC)**, 2010, p. 1 – 6, Sept. 2010. Citado na página 28.

BEDER, D. M. **Uma arquitetura de software baseada em padrões de projeto para o desenvolvimento de aplicações concorrentes confiáveis**. Tese (Doutorado) — Universidade Estadual de Campinas. Instituto de Computação, 6 2001. Citado na página 51.

BERARDI, V. L.; PATUWO, B. E.; HU, M. Y. A principled approach for building and evaluating neural network classification models. **Decis. Support Syst.**, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 38, n. 2, p. 233–246, nov. 2004. ISSN 0167-9236. Disponível em: <[http://dx.doi.org/10.1016/S0167-9236\(03\)00093-9](http://dx.doi.org/10.1016/S0167-9236(03)00093-9)>. Citado na página 43.

BOUKHAROUBA, K.; ROUSSEL, P.; DREYFUS, G.; JOHANNET, A. Flash flood forecasting using support vector regression: An event clustering based approach. In: **Machine Learning for Signal Processing (MLSP), 2013 IEEE International Workshop on**. [S.l.: s.n.], 2013. p. 1–6. ISSN 1551-2541. Citado na página 41.

BRANDT, A.; VASSEUR, J.; HUI, J.; PISTER, K.; THUBERT, P.; LEVIS, P.; STRUIK, R.; KELSEY, R.; CLAUSEN, T. H.; WINTER, T. **RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks**. RFC Editor, 2012. RFC 6550. (Request for Comments, 6550). Disponível em: <<https://rfc-editor.org/rfc/rfc6550.txt>>. Citado na página 55.

BRATH, A.; MONTANARI, A.; TOTH, E. Neural networks and non-parametric methods for improving real-time flood forecasting through conceptual hydrological models. **Hydrology and Earth System Sciences (HESS) & Discussions (HESSD)**, 2002. Citado na página 41.

BROCK, D. L. The electronic product code (epc). **Auto-ID Center White Paper MIT-AUTOID-WH-002**, 2001. Citado na página 37.

CAMPS-VALLS, G. Machine learning in remote sensing data processing. In: **Machine Learning for Signal Processing, 2009. MLSP 2009. IEEE International Workshop on**. [S.l.: s.n.], 2009. p. 1–6. Citado na página 41.

CHEN, B.; LIAO, Q.; TANG, Z. A clustering based bayesian network classifier. In: **Fuzzy Systems and Knowledge Discovery, 2007. FSKD 2007. Fourth International Conference on**. [S.l.: s.n.], 2007. v. 1, p. 444–448. Citado na página 47.

COELHO, C. E. **MIRE: UM MIDDLEWARE PARA PROVER CONFIABILIDADE NAS INFORMAÇÕES EM REDES DE SENSORES SEM FIO**. Monografia (Trabalho de Conclusão de Curso) — University of São Paulo, 2014. Citado nas páginas 21 e 36.

DABHI, V.; PRAJAPATI, H. Soft computing based intelligent grid architecture. **Computer and Communication Engineering, 2008. ICCCE 2008. International Conference on**, p. 574 – 577, May 2008. Citado na página 28.

DEAK, G.; CURRAN, K.; CONDELL, J.; ASIMAKOPOULOU, E.; BESSIS, N. Iots (internet of things) and dfpl (device-free passive localisation) in a disaster management scenario. **Simulation Modelling Practice and Theory**, v. 35, p. 86 – 96, 2013. ISSN 1569-190X. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1569190X13000415>>. Citado nas páginas 41, 70 e 72.

DERR, K.; MANIC, M. Wireless sensor networks x2014;node localization for various industry problems. **IEEE Transactions on Industrial Informatics**, v. 11, n. 3, p. 752–762, June 2015. ISSN 1551-3203. Citado na página 32.

DORNELLES, C. T. A. **Percepção Ambiental: Uma Análise na Bacia Hidrográfica do Rio Monjolinho, São Carlos, SP**. Dissertação (Mestrado) — Escola de Engenharia de São Carlos - Universidade de São Paulo, 2006. Citado na página 35.

FAIÇAL, B. S.; COSTA, F. G.; PESSIN, G.; UEYAMA, J.; FREITAS, H.; COLOMBO, A.; FINI, P. H.; VILLAS, L.; OSÓRIO, F. S.; VARGAS, P. A.; BRAUN, T. The use of unmanned aerial vehicles and wireless sensor networks for spraying pesticides. **Journal of Systems Architecture**, v. 60, n. 4, p. 393 – 404, 2014. Citado na página 34.

FANG, S.; XU, L.; ZHU, Y.; LIU, Y.; LIU, Z.; PEI, H.; YAN, J.; ZHANG, H. An integrated information system for snowmelt flood early-warning based on internet of things. **Information Systems Frontiers**, v. 17, n. 2, p. 321–335, 2015. ISSN 1572-9419. Citado nas páginas 71 e 72.

FAWCETT, T. An introduction to {ROC} analysis. **Pattern Recognition Letters**, v. 27, n. 8, p. 861 – 874, 2006. ISSN 0167-8655. {ROC} Analysis in Pattern Recognition. Citado na página 90.

FRASER, A. M.; SWINNEY, H. L. Independent coordinates for strange attractors from mutual information. **Physical Review A**, American Physical Society, v. 33, p. 1134–1140, 1986. Citado na página 77.

FREITAS, C. M. d.; XIMENES, E. F. Floods and public health - a review of the recent scientific literature on the causes, consequences and responses to prevention and mitigation. **Ciência e Saúde Coletiva**, scielo, v. 17, p. 1601 – 1616, 06 2012. ISSN 1413-8123. Citado na página 25.

FURQUIM, G.; FILHO, G. P. R.; JALALI, R.; PESSIN, G.; PAZZI, R.; UEYAMA, J. How to improve fault tolerance in disaster predictions: a case study about flash floods using iot, ml and real data. **Submetido à PLOS ONE**, PLOS ONE, 2017. Citado nas páginas 35, 53, 55, 56, 66, 68, 90 e 91.

FURQUIM, G.; MELLO, R.; PESSIN, G.; FAIÇAL, B.; MENDIONDO, E.; UEYAMA, J. An accurate flood forecasting model using wireless sensor networks and chaos theory: A case study with real wsn deployment in brazil. In: MLADENOV, V.; JAYNE, C.; ILIADIS, L. (Ed.). **Engineering Applications of Neural Networks**. Springer International Publishing, 2014, (Communications in Computer and Information Science, v. 459). p. 92–102. ISBN 978-3-319-11070-7. Disponível em: <http://dx.doi.org/10.1007/978-3-319-11071-4_9>. Citado nas páginas 67, 77, 78, 79 e 80.

FURQUIM, G.; NETO, F.; PESSIN, G.; UEYAMA, J.; ALBUQUERQUE, J. D.; CLARA, M.; MENDIONDO, E.; SOUZA, V. D.; SOUZA, P. D.; DIMITROVA, D.; BRAUN, T.

Combining wireless sensor networks and machine learning for flash flood nowcasting. In: **Advanced Information Networking and Applications Workshops (WAINA), 2014 28th International Conference on**. [S.l.: s.n.], 2014. p. 67–72. Citado nas páginas 36, 37, 73, 74, 75, 76, 78 e 80.

FURQUIM, G.; PESSIN, G.; FAIÇAL, B. S.; MENDIONDO, E. M.; UEYAMA, J. Improving the accuracy of a flood forecasting model by means of machine learning and chaos theory. **Neural Computing and Applications**, v. 27, n. 5, p. 1129–1141, 2016. ISSN 1433-3058. Disponível em: <<http://dx.doi.org/10.1007/s00521-015-1930-z>>. Citado nas páginas 67, 68, 80, 81, 82, 83, 84, 85 e 86.

FURQUIM, G.; PESSIN, G.; GOMES, P. H.; MENDIONDO, E. M.; UEYAMA, J. A distributed approach to flood prediction using a wsn and ml: A comparative study of ml techniques in a wsn deployed in brazil. In: _____. **Intelligent Data Engineering and Automated Learning – IDEAL 2015: 16th International Conference, Wroclaw, Poland, October 14-16, 2015, Proceedings**. Cham: Springer International Publishing, 2015. p. 485–492. ISBN 978-3-319-24834-9. Disponível em: <http://dx.doi.org/10.1007/978-3-319-24834-9_56>. Citado nas páginas 64 e 89.

GARCIA, W. **Pensamentos e reflexões do Papa Francisco: 500 frases em 50 lições para você descobrir o que ele pensa**. ARTEM VIVENDI, 2015. (Coleção Pensamentos Biográficos). ISBN 9788569625001. Disponível em: <<https://books.google.com.br/books?id=09B3DAAQBAJ>>. Citado na página 9.

GIUNTINI, F. T. **Um Modelo de Rede de Sensores sem fio Auto-Organizada e Tolerante a Falhas para Detecção de Incêndios**. Dissertação (Mestrado) — UNIVERSIDADE FEDERAL DE SÃO CARLOS. CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA, 8 2016. Citado nas páginas 51 e 52.

GUBBI, J.; BUYYA, R.; MARUSIC, S.; PALANISWAMI, M. Internet of things (iot): A vision, architectural elements, and future directions. **Future Generation Computer Systems**, v. 29, n. 7, p. 1645 – 1660, 2013. ISSN 0167-739X. Including Special sections: Cyber-enabled Distributed Computing for Ubiquitous Cloud and Network Services amp; Cloud Computing and Scientific Applications — Big Data, Scalable Analytics, and Beyond. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0167739X13000241>>. Citado na página 38.

GUO, B.; YU, Z.; ZHOU, X.; ZHANG, D. From participatory sensing to mobile crowd sensing. In: **2014 IEEE International Conference on Pervasive Computing and Communication Workshops (PERCOM WORKSHOPS)**. [S.l.: s.n.], 2014. p. 593–598. Citado na página 38.

HALL, M.; FRANK, E.; HOLMES, G.; PFAHRINGER, B.; REUTEMANN, P.; WITTEN, I. H. The weka data mining software: An update. **SIGKDD Explor. Newsl.**, ACM, New York, NY, USA, v. 11, n. 1, p. 10–18, nov. 2009. ISSN 1931-0145. Disponível em: <<http://doi.acm.org.ez67.periodicos.capes.gov.br/10.1145/1656274.1656278>>. Citado na página 73.

HEARST, M. A. Trends controversies: Support vector machines. **IEEE Intelligent System**, v. 13, n. 4, p. 18–28, 1998. Citado nas páginas 46 e 47.

- HEINZELMAN, W. B.; CHANDRAKASAN, A. P.; BALAKRISHNAN, H. An application-specific protocol architecture for wireless microsensor networks. **IEEE Transactions on Wireless Communications**, v. 1, n. 4, p. 660–670, Oct 2002. ISSN 1536-1276. Citado nas páginas 61, 62 e 88.
- HOSSAIN, M.; TURNA, T.; SOHELI, S.; KAISER, M. Neuro-fuzzy(nf)-based adaptive flood warning system for bangladesh. In: **Informatics, Electronics Vision (ICIEV), 2014 International Conference on**. [S.l.: s.n.], 2014. p. 1–5. Citado nas páginas 41 e 48.
- HUANG, H.; YAN, Z.; ZHANG, X.; WU, Q.; ZHU, G.; ZHANG, Y.; HUANG, Y. Study on dynamic monitoring system of high slope based on the internet of things. In: **2016 IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)**. [S.l.: s.n.], 2016. p. 407–412. Citado nas páginas 70 e 72.
- HUGHES, D.; BENCOMO, N.; BLAIR, G.; COULSON, G.; GRACE, P.; PORTER, B. Exploiting extreme heterogeneity in a flood warning scenario using the gridkit middleware. In: **Proceedings of the ACM/IFIP/USENIX Middleware '08 Conference Companion**. New York, NY, USA: ACM, 2008. (Companion '08), p. 54–57. ISBN 978-1-60558-369-3. Disponível em: <<http://doi.acm.org/10.1145/1462735.1462748>>. Citado na página 48.
- HUGHES, D.; UHEYAMA, J.; MENDIONDO, E.; MATTHYS, N.; HORRÉ, W.; MICHELS, S.; HUYGENS, C.; JOOSEN, W.; MAN, K.; GUAN, S.-U. A middleware platform to support river monitoring using wireless sensor networks. **Journal of the Brazilian Computer Society**, Springer-Verlag, v. 17, n. 2, p. 85–102, 2011. ISSN 0104-6500. Disponível em: <<http://dx.doi.org/10.1007/s13173-011-0029-3>>. Citado nas páginas 26, 31 e 33.
- IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs). **IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006)**, p. 1–314, Sept 2011. Citado na página 54.
- ISHII, R. P.; MELLO, R. F. de. An online data access prediction and optimization approach for distributed systems. **IEEE Transactions on Parallel and Distributed Systems**, IEEE Computer Society, Los Alamitos, CA, USA, v. 23, n. 6, p. 1017–1029, 2012. ISSN 1045-9219. Citado na página 76.
- KANT, A.; SUMAN, P.; GIRI, B.; TIWARI, M.; CHATTERJEE, C.; NAYAK, P.; KUMAR, S. Comparison of multi-objective evolutionary neural network, adaptive neuro-fuzzy inference system and bootstrap-based neural network for flood forecasting. **Neural Computing and Applications**, Springer London, v. 23, n. 1, p. 231–246, 2013. ISSN 0941-0643. Disponível em: <<http://dx.doi.org/10.1007/s00521-013-1344-8>>. Citado na página 41.
- KAR, A.; WINN, L.; LOHANI, A.; GOEL, N. Soft computing–based workable flood forecasting model for ayeyarwady river basin of myanmar. **Journal of Hydrologic Engineering**, v. 17, n. 7, p. 807–822, 2012. Disponível em: <[http://dx.doi.org/10.1061/\(ASCE\)HE.1943-5584.0000505](http://dx.doi.org/10.1061/(ASCE)HE.1943-5584.0000505)>. Citado nas páginas 41 e 42.
- KENNEL, M. B.; BROWN, R.; ABARBANEL, H. D. I. Determining embedding dimension for phase-space reconstruction using a geometrical construction. **Phys. Rev.**

- A, American Physical Society, v. 45, p. 3403–3411, Mar 1992. Disponível em: <<http://link.aps.org/doi/10.1103/PhysRevA.45.3403>>. Citado nas páginas 77 e 78.
- KNOX, J. C.; KUNDZEWICZ, Z. W. Extreme hydrological events, palaeo-information and climate change. **Hydrological Sciences Journal**, v. 42, n. 5, p. 765–779, 1997. Disponível em: <<http://dx.doi.org/10.1080/02626669709492071>>. Citado na página 25.
- KONOMI, S.; WAKASA, K.; ITO, M.; SEZAKI, K. User participatory sensing for disaster detection and mitigation in urban environments. In: _____. **Distributed, Ambient and Pervasive Interactions: 4th International Conference, DAPI 2016, Held as Part of HCI International 2016, Toronto, ON, Canada, July 17-22, 2016, Proceedings**. Cham: Springer International Publishing, 2016. p. 459–469. ISBN 978-3-319-39862-4. Disponível em: <http://dx.doi.org/10.1007/978-3-319-39862-4_42>. Citado nas páginas 71 e 72.
- KOREN, I.; KRISHNA, C. M. **Fault-Tolerant Systems**. 1st. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007. ISBN 0120885255, 9780120885251. Citado na página 51.
- LAIO, F.; PORPORATO, A.; REVELLI, R.; RIDOLFI, L. A comparison of nonlinear flood forecasting methods. **Water Resources Research**, v. 39, n. 5, p. n/a–n/a, 2003. ISSN 1944-7973. Disponível em: <<http://dx.doi.org/10.1029/2002WR001551>>. Citado nas páginas 41 e 43.
- LARA-CUEVA, R.; BENÍTEZ, D.; CAAMAÑO, A.; ZENNARO, M.; ROJO-ÁLVAREZ, J. L. Performance evaluation of a volcano monitoring system using wireless sensor networks. In: **2014 IEEE Latin-America Conference on Communications (LATINCOM)**. [S.l.: s.n.], 2014. p. 1–6. ISSN 2330-989X. Citado nas páginas 70 e 72.
- LEE, P. A.; ANDERSON, T. **Fault Tolerance: Principles and Practice**. 2nd. ed. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1990. ISBN 0387820779. Citado na página 51.
- LIEBERT, W.; PAWELZIK, K.; SCHUSTER, H. G. Optimal embeddings of chaotic attractors from topological considerations. **EPL (Europhysics Letters)**, v. 14, n. 6, p. 521, 1991. Disponível em: <<http://stacks.iop.org/0295-5075/14/i=6/a=004>>. Citado na página 78.
- LORENA, A. C.; CARVALHO, A. C. P. L. F. de. Uma introdução às support vector machines. **RITA**, v. 14, n. 2, p. 43–67, 2007. Disponível em: <<http://dblp.uni-trier.de/db/journals/rita/rita14.html#LorenaC07>>. Citado nas páginas 46 e 47.
- LUXBURG, U. V.; SCHÖLKOPF, B. Statistical learning theory: Models, concepts, and results. In: **Handbook for the History of Logic, Vol. 10: Inductive Logic**. Elsevier, 2011. 13 A.C. Tamhane Y. Hochberg. **Multiple Comparison Procedures**. [S.l.]: John Wiley & Sons, 1997. Citado nas páginas 43, 46 e 47.
- MAINETTI, L.; PATRONO, L.; VILEI, A. Evolution of wireless sensor networks towards the internet of things: A survey. In: **Software, Telecommunications and Computer Networks (SoftCOM), 2011 19th International Conference on**. [S.l.: s.n.], 2011. p. 1–6. Citado nas páginas 26, 27 e 29.

MELLO, R. Improving the performance and accuracy of time series modeling based on autonomic computing systems. **Journal of Ambient Intelligence and Humanized Computing**, Springer-Verlag, v. 2, n. 1, p. 11–33, 2011. ISSN 1868-5137. Disponível em: <<http://dx.doi.org/10.1007/s12652-010-0028-9>>. Citado nas páginas 76 e 77.

MELLO, R.; YANG, L. Prediction of dynamical, nonlinear, and unstable process behavior. **The Journal of Supercomputing**, Springer US, v. 49, n. 1, p. 22–41, 2009. ISSN 0920-8542. Disponível em: <<http://dx.doi.org/10.1007/s11227-008-0215-z>>. Citado nas páginas 77 e 78.

MENDIONDO, E. Challenging issues of urban biodiversity related to ecohydrology. **Brazilian Journal of Biology**, scielo, v. 68, p. 983 – 1002, 11 2008. ISSN 1519-6984. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1519-69842008000500007&nrm=iso>. Citado na página 25.

MITCHELL, T. M. **Machine Learning**. New York, NY, USA: McGraw-Hill, Inc., 1997. ISBN 0070428077, 9780070428072. Citado nas páginas 42, 43, 44, 45, 47 e 48.

MITRA, P.; RAY, R.; CHATTERJEE, R.; BASU, R.; SAHA, P.; RAHA, S.; BARMAN, R.; PATRA, S.; BISWAS, S. S.; SAHA, S. Flood forecasting using internet of things and artificial neural networks. In: **2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)**. [S.l.: s.n.], 2016. p. 1–5. Citado nas páginas 41, 71, 72, 78, 79, 81 e 82.

NAYAK, P. C.; SUDHEER, K. P.; RANGAN, D. M.; RAMASASTRI, K. S. Short-term flood forecasting with a neurofuzzy model. **Water Resources Research**, v. 41, n. 4, p. n/a–n/a, 2005. ISSN 1944-7973. Disponível em: <<http://dx.doi.org/10.1029/2004WR003562>>. Citado na página 41.

OPREA, M.; MATEI, A. Applying artificial neural networks in environmental prediction systems. In: **Proceedings of the 11th WSEAS international conference on Automation & information**. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS), 2010. (ICAI'10), p. 110–115. ISBN 978-960-474-193-9. Disponível em: <<http://dl.acm.org/citation.cfm?id=1863281.1863301>>. Citado nas páginas 26 e 44.

PERERA, C.; ZASLAVSKY, A.; CHRISTEN, P.; GEORGAKOPOULOS, D. Context aware computing for the internet of things: A survey. **IEEE Communications Surveys Tutorials**, v. 16, n. 1, p. 414–454, First 2014. ISSN 1553-877X. Citado nas páginas 29, 39 e 40.

PISANI, P.; LORENA, A. A systematic review on keystroke dynamics. **Journal of the Brazilian Computer Society**, Springer London, v. 19, n. 4, p. 573–587, 2013. ISSN 0104-6500. Disponível em: <<http://dx.doi.org/10.1007/s13173-013-0117-7>>. Citado nas páginas 30 e 107.

POZZA, S. A. **Identificação das Fontes de Poluição Atmosférica na Cidade de São Carlos - SP**. Dissertação (Mestrado) — Federal University of São Carlos, 2005. Disponível em: <http://www.bdt.d.ufscar.br/htdocs/tedeSimplificado/tde_busca/arquivo.php?codArquivo=2621>. Citado na página 25.

REZENDE, S. **Sistemas inteligentes: fundamentos e aplicações**. [S.l.]: Manole, 2003. ISBN 9788520416839. Citado na página 45.

- RILEY, G. F.; HENDERSON, T. R. Modeling and tools for network simulation. In: _____. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. cap. The ns-3 Network Simulator, p. 15–34. ISBN 978-3-642-12331-3. Disponível em: <http://dx.doi.org/10.1007/978-3-642-12331-3_2>. Citado nas páginas 29 e 54.
- ROY, J.; GUPTA, D.; GOSWAMI, S. An improved flood warning system using wsn and artificial neural network. In: **India Conference (INDICON), 2012 Annual IEEE**. [S.l.: s.n.], 2012. p. 770–774. Citado na página 48.
- RUIHUA, L.; GUOXIANG, M.; ZHENGJIN, F.; YONGHONG, C.; NAIKUI, G. A hybrid computing intelligence approach for the stator insulation residual life predicting of large generator. **Properties and applications of Dielectric Materials, 2006. 8th International Conference on**, p. 271 – 275, June 2006. Citado na página 28.
- RYU, C.-S. Iot-based intelligent for fire emergency response systems. **International Journal of Smart Home**, v. 9, n. 3, p. 161–168, 2015. Citado nas páginas 69 e 72.
- SALGADO, M.; CAMINHAS, W.; MENEZES, B. Computational intelligence in reliability and maintainability engineering. **Reliability and Maintainability Symposium, 2009. RAMS 2009. Annual**, p. 304 – 309, Jan. 2009. Citado na página 28.
- SCHULZ, J.; REICHENBACH, F.; BLUMENTHAL, J.; TIMMERMANN, D. Low cost system for detecting leakages along artificial dikes with wireless sensor networks. In: **Proceedings of the Workshop on Real-world Wireless Sensor Networks**. New York, NY, USA: ACM, 2008. (REALWSN '08), p. 66–70. ISBN 978-1-60558-123-1. Disponível em: <<http://doi.acm.org/10.1145/1435473.1435491>>. Citado na página 25.
- SEAL, V.; RAHA, A.; MAITY, S.; MITRA, S. K.; MUKHERJEE, A.; NASKAR, M. K. A simple flood forecasting scheme using wireless sensor networks. **CoRR**, abs/1203.2511, 2012. Citado nas páginas 26, 29, 31, 34 e 48.
- SEE, L.; OPENSHAW, S. Applying soft computing approaches to river level forecasting. **Hydrol. Sci. J**, v. 44, p. 763–778, 1999. Citado na página 41.
- SHI, H. **Best-first decision tree learning**. [S.l.], 2007. Citado nas páginas 45 e 46.
- SILVA, I. D.; SPATTI, D.; FLAUZINO, R. **Redes Neurais Artificiais para Engenharia e Ciências Aplicadas**. [S.l.]: ARTLIBER, 2010. ISBN 9788588098534. Citado nas páginas 43, 44 e 45.
- STRATEGY, I.; UNIT, P. Itu internet reports 2005: The internet of things. **Geneva: International Telecommunication Union (ITU)**, 2005. Citado na página 37.
- TAKENS, F. Detecting strange attractors in turbulence. In: RAND, D.; YOUNG, L.-S. (Ed.). **Dynamical Systems and Turbulence, Warwick 1980**. Springer Berlin Heidelberg, 1981, (Lecture Notes in Mathematics, v. 898). p. 366–381. ISBN 978-3-540-11171-9. Disponível em: <<http://dx.doi.org/10.1007/BFb0091924>>. Citado na página 77.
- THUBERT, P.; HUI, J. **Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks**. RFC Editor, 2011. RFC 6282. (Request for Comments, 6282). Disponível em: <<https://rfc-editor.org/rfc/rfc6282.txt>>. Citado na página 54.

- TIWARI, M. K.; CHATTERJEE, C. Development of an accurate and reliable hourly flood forecasting model using wavelet–bootstrap–ann (wbann) hybrid approach. **Journal of Hydrology**, v. 394, n. 3–4, p. 458 – 470, 2010. ISSN 0022-1694. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0022169410006025>>. Citado na página 41.
- TJ, P.; SCOFIELD, R. A. Applications of remotely sensed data in flood prediction and monitoring: report of the ceos disaster management support group flood team. In: **Geoscience and Remote Sensing Symposium, 2002. IGARSS '02. 2002 IEEE International**. [S.l.: s.n.], 2002. v. 2, p. 768–770 vol.2. Citado na página 25.
- TOTH, E.; BRATH, A.; MONTANARI, A. Comparison of short-term rainfall prediction models for real-time flood forecasting. **Journal of Hydrology**, v. 239, n. 1–4, p. 132 – 147, 2000. ISSN 0022-1694. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0022169400003449>>. Citado na página 41.
- TOUKOUROU, M.; JOHANNET, A.; DREYFUS, G. Flash flood forecasting by statistical learning in the absence of rainfall forecast: A case study. In: PALMER-BROWN, D.; DRAGANOVA, C.; PIMENIDIS, E.; MOURATIDIS, H. (Ed.). **Engineering Applications of Neural Networks**. Springer Berlin Heidelberg, 2009, (Communications in Computer and Information Science, v. 43). p. 98–107. ISBN 978-3-642-03968-3. Disponível em: <http://dx.doi.org/10.1007/978-3-642-03969-0_10>. Citado na página 41.
- UEYAMA, J.; HUGHES, D.; MAN, K. L.; GUAN, S.; MATTHYS, N.; HORREÉ, W.; MICHIELS, S.; HUYGENS, C.; JOOSEN, W. Applying a multi-paradigm approach to implementing wireless sensor network based river monitoring. **Cryptography and Network Security, Data Mining and Knowledge Discovery, E-Commerce & #38 Its Applications and Embedded Systems (CDEE), 2010 First ACIS International Symposium on**, p. 187 – 191, Oct. 2010. Citado nas páginas 25, 26, 29, 31 e 34.
- XU, G.; HUANG, G. Q.; FANG, J. Cloud asset for urban flood control. **Advanced Engineering Informatics**, v. 29, n. 3, p. 355 – 365, 2015. ISSN 1474-0346. Citado na página 72.
- YAO, Y.; ROSASCO, L.; CAPONNETTO, A. On early stopping in gradient descent learning. **Constructive Approximation**, Springer-Verlag, v. 26, n. 2, p. 289–315, 2007. ISSN 0176-4276. Disponível em: <<http://dx.doi.org/10.1007/s00365-006-0663-2>>. Citado na página 43.
- YICK, J.; MUKHERJEE, B.; GHOSAL, D. Wireless sensor network survey. **Computer Networks**, v. 52, n. 12, p. 2292 – 2330, 2008. ISSN 1389-1286. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1389128608001254>>. Citado na página 39.
- YU, J.; QIN, X.; LARSEN, O.; CHUA, L. Comparison between response surface models and artificial neural networks in hydrologic forecasting. **Journal of Hydrologic Engineering**, v. 19, n. 3, p. 473–481, 2014. Disponível em: <[http://dx.doi.org/10.1061/\(ASCE\)HE.1943-5584.0000827](http://dx.doi.org/10.1061/(ASCE)HE.1943-5584.0000827)>. Citado nas páginas 41 e 42.
- YUE, H.; GUO, L.; LI, R.; ASAEDA, H.; FANG, Y. Dataclouds: Enabling community-based data-centric services over the internet of things. **IEEE Internet of Things Journal**, v. 1, n. 5, p. 472–482, Oct 2014. ISSN 2327-4662. Citado na página 39.

ZAMBRANO, A.; PEREZ, I.; PALAU, C.; ESTEVE, M. Technologies of internet of things applied to an earthquake early warning system. **Future Generation Computer Systems**, p. –, 2016. ISSN 0167-739X. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0167739X16303909>>. Citado nas páginas 70 e 72.

ZELENKAUSKAITE, A.; BESSIS, N.; SOTIRIADIS, S.; ASIMAKOPOULOU, E. Disaster management and profile modelling of iot objects: Conceptual parameters for interlinked objects in relation to social network analysis. In: **Intelligent Networking and Collaborative Systems (INCoS), 2012 4th International Conference on**. [S.l.: s.n.], 2012. p. 509–514. Citado nas páginas 26 e 27.

DESCRIÇÃO DO PROCESSO DE REVISÃO SISTEMÁTICA REALIZADO

Neste apêndice serão apresentados os passos utilizados para realizar a revisão sistemática desta Tese. Destacando que essa revisão sistemática foi complementada com outros artigos sugeridos pelo orientador deste projeto, colegas do grupo de pesquisa e demais alunos de curso, assim como outros professores do programa.

A revisão sistemática é um método formal de conduzir a revisão bibliográfica, onde são definidos protocolos para serem seguidos e, assim, tentar assegurar que a revisão será completa e possível de ser reproduzida (PISANI; LORENA, 2013).

Realizar uma revisão sistemática consiste em seguir 3 passos, descritos com detalhes em Pisani e Lorena (2013), sendo eles: (i) planejamento, (ii) condução e (iii) apresentação dos resultados. Na fase de planejamento é definida uma questão de pesquisa, juntamente com as estratégias de busca e seleção de referências. Na fase de condução, o protocolo de pesquisa é executado, sendo aplicado o critério de seleção em todas as referências retornadas pela busca. Ao final, na fase de apresentação dos resultados, é feito um relatório final contendo o resultado do processo. Assim sendo, nas seções seguintes, são descritos os protocolos e critérios utilizados em cada uma dessas fases para realizar a revisão sistemática desta tese.

A.1 Planejamento

Como ponto de partida da revisão sistemática, foi definida uma questão principal para a pesquisa, juntamente com algumas subquestões que focam no objetivo desta tese. Sendo elas:

- Como são utilizados as RSSF, IoT e/ou AM para realizar previsões de

desastres naturais?

Quais são as vantagens e desvantagens de utilizar AM, RSSF e IoT para realizar previsões de desastres naturais?

Quais são os dados coletados e utilizados para realizar tais previsões?

Quais as técnicas de AM são utilizadas nas previsões e como é realizada a comparação entre elas?

Quais as medidas de performance utilizadas nas comparações e qual o desempenho alcançado?

Após a questão de pesquisa ter sido definida, foi montada uma expressão de busca (utilizando os operadores lógicos *AND* e *OR*) com possíveis termos de artigos, que permitam responder essas questões. Alguns dos termos definidos estão relacionados ao mesmo tópico, sendo muitas vezes sinônimos, o que contribuiu para uma pesquisa mais completa. O termo genérico de busca é mostrado abaixo.

```

1 ("natural disaster"
2 OR "natural disaster nowcasting"
3 OR "natural disaster forecast"
4 OR "natural disaster forecasting"
5 OR "natural disaster system"
6 OR "natural disaster prediction"
7 OR "natural disaster alert"
8 OR "forecast flooding"
9 OR "flooding forecast"
10 OR "flood forecasting"
11 OR "flood forecast")
12 AND
13 ("wireless sensor networks"
14 OR "machine learning")
15 AND
16 ("internet of things"
17 OR "IoT")

```

Essa expressão de busca foi utilizada nas bases de artigos listadas abaixo no dia 04 de abril de 2017, sendo que, como cada base tem uma *sintaxe* diferente para realizar a busca, a expressão foi adaptada para cada uma delas. Na base de artigos *Science Direct* a pesquisa foi realizada somente nas áreas de *Computer Science* e *Engineering*.

- *ACM Digital Library* (<<http://dl.acm.org/>>)

- *IEEE Xplore* (<<http://ieeexplore.ieee.org/>>)
- *Science Direct* (<<http://www.sciencedirect.com/>>)
- *Web of Science* (<<http://isiknowledge.com/>>)
- *Scopus* (<<http://www.scopus.com/>>)
- *Springer* <<http://www.springer.com/>>

No próximo passo do planejamento, foram definidos critérios para a seleção de artigos, os quais foram utilizados para exclusão ou inserção dos artigos retornados pelas bases. Esses critérios definem quais artigos serão considerados ou desconsiderados nos demais passos do processo. Os critérios utilizados para exclusão foram:

- Publicações que não utilizam RSSF ou AM ou IoT para realizar previsões sobre desastres naturais;
- Publicações com uma página, pôsteres, apresentações, resumos e índices de anais de eventos e editoriais;
- Publicações hospedadas em serviços inacessíveis;
- Publicações em outras línguas que não inglês ou português;
- Publicações com notas sobre retratação;
- Publicações duplicadas durante a busca.

Como última parte dessa fase, foram especificadas informações a serem extraídas das publicações e que poderiam ser utilizadas por um critério de qualidade. Essas informações devem responder aos 11 itens listados abaixo, sendo que o valor final de “qualidade do artigo” é dado pela soma das notas dadas nesses itens. Em cada item é dado um valor variando entre 0 e 1, sendo que esse itens foram definidos segundo os critérios de pesquisa para esta tese.

- Os objetivos são apresentados de forma clara no início do trabalho?
- As abordagens aplicadas são detalhadas de forma que seja possível replicá-las?
- Foram conduzidos testes experimentais?
- Os resultados são comparados com trabalhos anteriores ou de outros autores?
- As limitações do estudo são apresentadas?

- Trabalha com RSSF? (peso 2)
- Trabalha com AM? (peso 2)
- Trabalha com IoT? (peso 2)
- Apresenta mecanismos de tolerância a falhas? (peso 2)
- São gerados alertas ou o sistema é embutido na RSSF ou é dito poder ser facilmente integrado em um sistema de alertas?
- Os dados utilizados são similares aos nossos?

Terminada a fase de planejamento, o próximo passo é colocar todo o protocolo definido em prática na fase de condução.

A.2 Condução e Resultados

Nesta seção é descrita a condução dos protocolos definidos na fase de planejamento (Seção A.1), juntamente com os resultados obtidos.

A quantidade de publicações retornada por cada base, depois de aplicada a expressão de busca, é mostrada na tabela 17. A penúltima linha da tabela representa o total de artigo, depois de aplicados todos os critérios de exclusão menos “Publicações que não utilizam RSSF ou AM para realizar previsões sobre enchente”. Após aplicado esse último critério de exclusão, a quantidade resultante de publicações para serem analisadas é mostrada na última linha.

Tabela 17 – Total de artigos retornado por cada base

Base	Total de resultados
<i>ACM Digital Library</i>	6
<i>IEEE Xplore</i>	0
<i>Science Direct - Computer Science</i>	54
<i>Science Direct - Engineering</i>	33
<i>Web of Science</i>	8
<i>Scopus</i>	102
<i>Springer</i>	112
Total	315
Total após aplicados os critérios de exclusão	269
Total de publicações para serem analisadas	26

Fonte: Elaborada pelo autor.

Após extrair as informações das publicações restantes e a partir dessas informações, as publicações foram pontuadas segundo os 11 itens do critério de qualidade, definido

na fase de planejamento. A figura 38 mostra a distribuição de pontuação conseguida após esse procedimento. Devido a quantidade de publicações encontradas, para a revisão bibliográfica desta tese somente as publicações que conseguiram valores maiores ou iguais a 7,5 no critério de qualidade foram consideradas. Essas publicações foram detalhadas durante as seções 4.4.

Figura 38 – Distribuição da pontuação segundo o critério de qualidade.

Fonte: Elaborada pelo autor.