



Pós-Graduação em Ciência da Computação

Rafael Roque de Souza

Um Framework Inteligente para Escalonamento de VNFs em Data center



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
<http://cin.ufpe.br/~posgraduacao>

Recife
2020

Rafael Roque de Souza

Um Framework Inteligente para Escalonamento de VNFs em Data center

Tese de Doutorado apresentada ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de Doutor em Ciência da Computação.

Área de Concentração: Redes de Computadores e Sistemas Distribuídos

Orientador: Prof. Dr. Stênio Flávio de Lacerda Fernandes

Coorientador: Prof. Dr. Kelvin Lopes Dias

Recife

2020

Catálogo na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

S729f Souza, Rafael Roque de
Um framework inteligente para escalonamento de VNFs em data center /
Rafael Roque de Souza. – 2020.
118 f.: il., fig., tab.

Orientador: Stênio Flávio de Lacerda Fernandes.
Tese (Doutorado) – Universidade Federal de Pernambuco. CIn, Ciência da
Computação, Recife, 2020.
Inclui referências e apêndice.

1. Redes de computadores. 2. Escalonamento. I. Fernandes, Stênio Flávio
de Lacerda (orientador). II. Título.

004.6 CDD (23. ed.) UFPE - CCEN 2020 - 142

Rafael Roque de Souza

“Um Framework Inteligente para Escalonamento de VNFs em Data center”

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação.

Aprovado em: 10/03/2020.

Orientador: Prof. Dr. Stênio Flávio de Lacerda Fernandes

BANCA EXAMINADORA

Prof. Dr. Ricardo Massa Ferreira Lima
Centro de Informática / UFPE

Profª. Dra. Erica Teixeira Gomes de Sousa
Departamento de Estatística e Informática / UFRPE

Prof. Dr. Marcelo Anderson Batista dos Santos
Instituto Federal de Pernambuco / Campus Salgueiro

Prof. Dr. Carlos Alberto Kamienski
Centro de Matemática, Computação e Cognição / UFABC

Prof. Dr. Marco Antonio de Oliveira Domingues
Instituto Federal de Pernambuco / Campus Recife

Decido este trabalho a minha família e minha namorada que foram porto seguro perante as dificuldades durante este percurso.

AGRADECIMENTOS

Agradeço primeiramente a Deus, pelo dom da vida, pelo seu amor incondicional, pela saúde, pela coragem, pela força que me proporcionou para nunca desistir dos meus objetivos. À minha família, especialmente meus pais, José Alípio (In Memory), Lourdes Roque, as minhas irmãs Andreia Roque e Raquel Roque. A minha amada esposa, Geysel Roque, por compreender minha ausência e apoiar mesmo nos momentos de dificuldade e estresse. Agradeço ao meu orientador, Stênio Fernandes, por ter me dado a oportunidade de crescimento e por ter acreditado em mim, além de ter me incentivado, sempre com bastante objetividade, sobre a pesquisa, escrita e lições de vidas que vão além e de igual forma ao meu coorientador Kelvin Dias. Os seus sábios conselhos, dedicação foram de grande contribuição em cada passo na construção deste trabalho, e para minha capacitação como pesquisador. Aos Professores(a) Marcelo Santos (IFSertão-PE), Erica Sousa (UFRPE), Marcos Antônio (IFPE), Carlos Kamienski (UFABC), Ricardo Massa (CIn/UFPE), por terem aceitado o convite para compor a banca examinadora. Ao Centro de Informática (CIn) da Universidade Federal de Pernambuco (UFPE), que forneceu suporte durante a realização desta pesquisa. À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), que me promoveu recursos financeiros para execução deste trabalho. Aos meus amigos, por toda informação passada e pelos muitos momentos de descontração: Pablo Tibúrcio, Michel Sales, Felipe Alencar, Flávio, Amarildo, Gunnar. Enfim, a todos que de forma direta ou indireta contribuíram para essa conquista. A todos, meu sincero agradecimento.

“A fé é o ápice da evolução humana. Com ela, conseguimos transformar nossa cognição e mudar paradigmas improváveis, transformando sonhos em realidade.” (ROQUE, 2019).

RESUMO

A Network Function Virtualization (NFV) visa desacoplar a funcionalidade de rede de dispositivos físicos dedicados, permitindo assim que as máquinas virtuais hospede serviços tradicionalmente realizados por roteadores, firewalls, ou balanceamento de carga. A hospedagem desses serviços em máquinas virtuais fornecerá novas oportunidades de negócios aos data center. Atualmente, os recursos físicos disponíveis são limitados, o que torna o escalonamento de recursos eficiente, uma questão desafiadora para a adoção bem-sucedida da NFV. Destacamos nesta tese três problemas relacionados à NFV aplicada em data center: (i) escalonamento de cadeias de serviços; (ii) questões de confiabilidade e (iii) otimização da utilização dos recursos disponíveis. O foco desses problemas está relacionado à necessidade de evitar picos inesperados de tempo de inatividade dos serviços de rede, levando a perdas financeiras e rescisão de contrato *Service Level Agreement* (SLA). Além disso, outros fatores, como o aumento da temperatura gerada pela alocação ineficiente de recursos, podem degradar ainda mais o desempenho dos serviços de rede em função de falhas provenientes de altas temperaturas. Assim, torna-se essencial investigar técnicas de tolerância a falhas e requisitos de alto desempenho para o NFV aplicado ao data center. Portanto, alcançar a alocação eficiente de recursos é crucial. Essa tese propõe um framework inteligente para o escalonamento de VNFs. Em particular, a solução de escalonamento autônomo proposta é baseada no modelo de Memória de Longo Prazo (LSTM), com base nas regras de afinidade em data center. Seguidamente, a ferramenta de diagnóstico proposta incorpora uma análise holística do desempenho, falhas e análise exploratória de dados (EDA), após o escalonamento de VNFs. Além de integrar-se com ferramenta de medições de importância dos dispositivos no data center. Os resultados experimentais, examinados com dados de referência, mostram que a técnica proposta de escalonamento com tolerância a falhas, baseada em LSTM e técnicas de afinidade, alcança resultados promissores. O framework de escalonamento inteligente é capaz de automatizar o escalonamento dos recursos com e sem intervenção humano. Assim, a abordagem proposta é um mecanismo auxiliar para alcançar a alocação eficiente de recursos e aumentar o desempenho e disponibilidade do data center.

Palavras chaves: NFV. SFC. Escalonamento. Data Center. Aprendizagem Profundo.

ABSTRACT

Network Function Virtualization (NFV) aims to decouple network functionality from dedicated physical devices, thus allowing Virtual Machine (VM) to host services traditionally performed by routers, firewalls, or load balancing. Hosting these services on VMs will provide new business opportunities for the data center. Currently, the physical resources available are limited, which makes efficient resource scheduling a challenging issue for the successful adoption of NFV. In this thesis, three problems related to NFV applied in the data center: (i) service chain, (ii) reliability issues, and (iii) optimization of available resources utilization. The focus of these problems is related to the need to avoid unexpected peaks in downtime, leading to financial losses, decreased employee satisfaction, and reduced customer retention. Thus, it becomes essential to investigate failure tolerance techniques and high-performance requirements for the NFV applied to the data center. In addition, other factors, such as increased temperature generated by inefficient resource allocation, can further degrade the performance of network services. Therefore, achieving efficient resource allocation is crucial. This thesis proposes an intelligent framework for the scheduling of VNFs. In particular, the proposed autonomous dispatching solution is based on the Long Term Memory (LSTM) model based on affinity rules in a data center; an LSTM will allow the new generation of a multi-step forecast. The proposed diagnostic tool incorporates a holistic analysis of performability, failure, and exploratory data analysis (EDA). In addition to integrating with a measurement tool, the importance of the system. Experimental results, examined with reference data, show that the proposed fault tolerance scaling technique, based on LSTM and affinity techniques, achieves promising results. The fault tolerance conscious scaling framework can accurately predict the required resource scaling. Thus, the proposed approach is an auxiliary mechanism to achieve efficient resource allocation and increase data center availability.

Keywords: NFV. SFC. Scheduling. Data Center. Deep Learning.

LISTA DE FIGURAS

Figura 1 – Tópico de Pesquisa	19
Figura 2 – Escopo da Pesquisa de Doutorado	21
Figura 3 – Descrição do Problema	22
Figura 4 – Metodologia de Pesquisa	26
Figura 5 – Dependências Lógicas entre os Elementos desta Tese.	29
Figura 6 – Infraestruturas de data center.	30
Figura 7 – Semântica de operação do data center	32
Figura 8 – Arquitetura de Referência <i>Network Functions Virtualization</i> (NFV)	34
Figura 9 – NFV data center	35
Figura 10 – Etapas do escalonamento de <i>Virtual Network Function</i> (VNF)s	39
Figura 11 – Estrutura de Rede Neural Artificial.	43
Figura 12 – Modelo Multilayer Perceptron.	44
Figura 13 – Representação RNN	45
Figura 14 – Representação das células de uma RNN	45
Figura 15 – Célula LSTM em Detalhes	46
Figura 16 – Avaliação de Performabilidade	50
Figura 17 – Problema de Pesquisa no Escalonamento de VNFs	61
Figura 18 – Framework Inteligente para Escalonamento de VNFs em Data Center	64
Figura 19 – Fluxograma do Framework Inteligente de Escalonamento de VNFs	69
Figura 20 – Módulo Diagnóstico	71
Figura 21 – Avaliação baseado no Teste Conover	83
Figura 22 – Avaliação baseado no Teste de Friedman	83
Figura 23 – Treinamento e validação durante o processo de aprendizagem de um <i>Long Short-Term Memory</i> (LSTM).	86
Figura 24 – Predição de Escalonamento de VNFs com Afinidade	86
Figura 25 – Delay no Escalonamento de VNFs	87
Figura 26 – Variação de Temperatura ao longo Tempo	88
Figura 27 – Aumento de Temperatura Devido à Dissipação de Energia	89
Figura 28 – Degradação do Desempenho em Relação a Temperatura	90
Figura 29 – Performabilidade baseada na Taxa de Falhas	90
Figura 30 – Performabilidade baseada no Impacto da Temperatura.	91
Figura 31 – Correlação entre o utilização de <i>Mean Time To Failure</i> (MTTF) e <i>Central Processing Unit</i> (CPU).	93
Figura 32 – Distribuição do consumo da CPU.	94
Figura 33 – Distribuição do Consumo de Memória.	94
Figura 34 – Distribuição de Consumo de Armazenamento.	95

Figura 35 – Metodologia para Análise de Medições de Importância	96
Figura 36 – Arquitetura para o modelo <i>Reliability Block Diagram</i> (RBD)	99
Figura 37 – RBD da Arquiteturas (A1)	99
Figura 38 – RBD da Arquitetura (A2)	100
Figura 39 – RBD da Arquitetura (A3)	100
Figura 40 – RBD da Arquitetura (A4)	101
Figura 41 – RBD da Arquitetura (A5)	101
Figura 42 – Análise de Disponibilidade das arquiteturas A1 a A5	103
Figura 43 – Análise de Falhas das arquiteturas A1 a A5	103
Figura 44 – Análise do Tempo de Uptime das Arquiteturas A1 a A5	104
Figura 45 – Análise de Downtime para as Arquiteturas A1 a A5	104

LISTA DE TABELAS

Tabela 1 – Estado da Arte do Problema de Escalonamento	56
Tabela 2 – Parâmetros Térmicos Utilizados	79
Tabela 3 – Classificação do Modelo LSTM.	84
Tabela 4 – Parâmetros de Entrada do RBD	98
Tabela 5 – Experimento I - sem medições de importância	100
Tabela 6 – Medições de Importância de Disponibilidade (A1)	102
Tabela 7 – Resultados das Arquiteturas A1 até A5	105

LISTA DE ABREVIATURAS E SIGLAS

AI	<i>Availability Importance</i>
ANN	<i>Artificial Neural Networks</i>
APE	<i>Absolute Percentage Error</i>
ARV	<i>Average Relative Variance</i>
BSS	<i>Business Support Systems</i>
CAI	<i>Critically Availability</i>
CAPEX	<i>Capital Expenditure</i>
CD	<i>Critical Distance</i>
CDF	<i>Cumulative Distribution Function</i>
COTS	<i>Commercial Off-The-Shelf</i>
CPU	<i>Central Processing Unit</i>
CRAC	<i>Computer Room Air Conditioning</i>
DPI	<i>Deep Packet Inspection</i>
EDA	<i>Exploratory Data Analysis</i>
ETSI	<i>European Telecommunications Standards Institute</i>
FG	<i>Forwarding Graph</i>
GA	<i>Genetic Algorithm</i>
GMPLS	<i>Generalized Multi-Protocol Label Switching</i>
GRU	<i>Gated Recurrent Unit</i>
HD	<i>Hard Disk</i>
IETF	<i>Internet Engineering Task Force</i>
ILP	<i>Integer Linear Programming</i>
IP	<i>Internet Protocol</i>
IRTF	<i>Internet Research Task Force</i>
KDN	<i>Knowledge-Defined Networking</i>
KNN	<i>K-Nearest Neighbors</i>
LSTM	<i>Long Short-Term Memory</i>
MAE	<i>Mean Absolute Error</i>
MANO	<i>Management and Orchestration</i>
MAPE	<i>Mean Absolute Percentage Error</i>

MC	<i>Markov Chain</i>
MILP	<i>Mixed Integer Linear Programming</i>
MLP	<i>Multi Layer Perceptron</i>
MPLS	<i>Multiprotocol Label Switching</i>
MSE	<i>Mean Squared Error</i>
MTBF	<i>Mean Time Between Failures</i>
MTR	<i>Mean Time to Restore</i>
MTTF	<i>Mean Time To Failure</i>
MTTR	<i>Mean Time To Repair</i>
NAS	<i>Network-Attached Storage</i>
NAT	<i>Network Address Translation</i>
NFV	<i>Network Functions Virtualization</i>
NFVI	<i>Network Functions Virtualization Infrastructure</i>
NFVO	<i>NFV Orchestrator</i>
NIC	<i>Network Interface Card</i>
OPEX	<i>Operational Expenditure</i>
OSS	<i>Operational Support Systems</i>
PaaS	<i>Platform as a Service</i>
PDU	<i>Power Distribution Unit</i>
PM	<i>Physical Machine</i>
POCID	<i>Prediction of Change in Direction</i>
QoE	<i>Quality of Experience</i>
QoS	<i>Quality of Service</i>
QP	<i>Questão de Pesquisa</i>
RBD	<i>Reliability Block Diagram</i>
ReLU	<i>Rectified Linear Unit</i>
RNN	<i>Recurrent Neural Network</i>
RW	<i>Random Walks</i>
SaaS	<i>Software as a Service</i>
SAN	<i>Storage Area Network</i>
SDN	<i>Software-Defined Networking</i>
SDT	<i>Step Down Transformer</i>
SFC	<i>Service Function Chain</i>

SFP	<i>Service Function Path</i>
SLA	<i>Service Level Agreement</i>
SPN	<i>Stochastic Petri Net</i>
SVM	<i>Support Vector Machines</i>
Theil's	<i>Theil's U statistic</i>
TTF	<i>Time To Failure</i>
TTR	Time To Repair
UPS	<i>Power Supplies</i>
VIM	<i>Virtualized Infrastructure Manager</i>
VLAN	<i>Virtual LAN</i>
VM	<i>Virtual Machine</i>
VN	<i>Virtual Network</i>
VNE	<i>Virtual Network Embedding</i>
VNF	<i>Virtual Network Function</i>
VNFM	<i>VNF Manager</i>
VPN	<i>Virtual Private Networks</i>

SUMÁRIO

1	INTRODUÇÃO	17
1.1	ESCOPO DA PESQUISA E MOTIVAÇÃO	20
1.2	VISÃO GERAL DO PROBLEMA	21
1.3	QUESTÕES DE PESQUISA	23
1.4	OBJETIVO E PROPÓSITO DA PESQUISA	25
1.5	MÉTODOS E METODOLOGIAS	25
1.6	CONTRIBUIÇÕES DA TESE	27
1.7	ESTRUTURA DA TESE	28
2	CONTEXTO TEÓRICO	30
2.1	DATA CENTER	30
2.1.1	Virtualização de Rede	32
2.1.2	Virtualização das Funções de Rede	33
2.1.3	Virtualização das Funções de Rede em data center	35
2.2	ESCALONAMENTO INTELIGENTE	37
2.2.1	Modelos Afinidades	40
2.2.2	Aprendizado de Máquina	42
2.3	MÓDULO DIAGNÓSTICO	47
2.3.1	Dependabilidade	48
2.3.2	Performabilidade	49
2.3.3	Análise Exploratória de Dados	50
2.3.4	Medições de Importâncias	51
2.4	CONSIDERAÇÕES FINAIS	53
3	ESTADO DA ARTE	54
3.1	PROBLEMA CONVENCIONAL	54
3.2	PROBLEMA DE ESCALONAMENTO DE VNFS	56
3.3	CONSIDERAÇÕES FINAIS	58
4	ESCALONAMENTO INTELIGENTE	60
4.1	DESCRIÇÃO DO PROBLEMA DE ESCALONAMENTO	60
4.2	ABORDAGEM PROPOSTA	63
4.2.1	Módulo Analítico	67
4.2.2	Módulo Escalonamento de VNFs	68
4.2.3	Abordagem do Módulo Diagnóstico	70
4.2.3.1	Modelo de CPU	71

4.2.3.2	Modelo de Temperatura	72
4.2.3.3	Gerador do Dataset	73
4.2.3.4	Módulo de Avaliação	73
4.2.4	Cenários Realistas	75
4.3	CONSIDERAÇÕES FINAIS	77
5	EXPERIMENTOS E AVALIAÇÕES	78
5.1	DESCRIÇÃO DA BASE DE DADOS	78
5.2	NORMALIZAÇÃO DE DADOS	79
5.3	VALIDAÇÃO DO MODELO	80
5.4	AVALIAÇÃO DE DESEMPENHO	82
5.4.1	Análise de Acurácia de Previsões	82
5.4.2	LSTM com Previsão Multi-passo à Frente	84
5.4.3	Ferramenta de Diagnóstico	87
5.4.3.1	Validação do Impacto da Temperatura	88
5.4.3.2	Avaliação de Performabilidade	89
5.4.3.3	Análise Exploratória de Dados	92
5.4.4	Análise de Importância de Disponibilidade e Criticidade de Dispositivos	95
5.4.5	Discussão	105
5.5	CONSIDERAÇÕES FINAIS	106
6	CONCLUSÃO E DIREÇÕES FUTURAS	107
6.1	CONCLUSÃO	107
6.2	AMEAÇAS À VALIDADE	107
6.3	TRABALHOS FUTUROS	109
	REFERÊNCIAS	110
	APÊNDICE A – APPENDIX	119

1 INTRODUÇÃO

Atualmente, existem mais de 4 bilhões de usuários conectados à Internet¹. De acordo com algumas previsões, o número total de dispositivos que estarão conectados à Internet atingirá uma marca de 5 bilhões até 2021. Ao mesmo tempo, espera-se que o tráfego global da Internet cresça em proporções semelhantes ao número de dispositivos (INDEX, 2013). O tráfego de dados aumentou exponencialmente nos últimos anos, e o tráfego anual quase quadruplicou, representando 92% do tráfego total do data center na nuvem até 2021. A Cisco atribui esse aumento da migração de empresas para infraestruturas de computação em nuvem à capacidade de expandir de forma eficiente e suportar mais cargas de trabalho do que os data center tradicionais (CISCO, 2016). Essa multiplicação de dispositivos, serviços de Internet e tráfego de dados tem contribuições significativas para o desenvolvimento do data center.

Os data center são formados por um conjunto de tecnologias em servidores e comunicação de rede que fornecem serviços especializados de valor agregado. Esses serviços incluem o processamento e armazenamento de grandes massas de dados, de forma escalável, para empresas de diversos portes. Essas organizações investem pesadamente no planejamento de data center para evitar, por exemplo, picos inesperados no tempo de inatividade ou sobrecarga. Vários artigos que abordam a alocação de recursos em ambientes de rede virtual negligenciaram aspectos de dependabilidade que envolvem disponibilidade e tolerância a falhas (NARTEN et al., 2014). Muitos dos problemas de falha de um data center acontecem, porque não há uma estratégia de tolerância a falhas bem definida, o que é comprovado em estudos realizados pelo Ponemon Institute, em 2013. O estudo afirma que 81% dos casos de falha se devem a projetos realizados com pouco planejamento de desempenho, disponibilidade no data center, o que leva ao aumento das taxas de inatividade. A mesma pesquisa mostra que o custo máximo da inatividade dobrou de 2010 para 2016, pouco mais de 1 milhão para 2,4 milhões, um aumento de 34%. O preço médio do tempo de inatividade do data center está próximo de 9.000 dólar americano(USD) por minuto, em média (PONEMON, 2016).

Um estudo realizado pela Intel² mostrou que a energia consumida pelos data center quase dobrou entre 2000 e 2005, passando de 12 para 23 bilhões de kWh. Outro relatório mais recente do NRDC³, indica que os data center localizados nos EUA consumiram aproximadamente 91 bilhões de kWh de eletricidade em 2013 e devem atingir 140 bilhões de kWh até 2020. Além disso, espera-se que os custos de consumo de energia representem quase 50% das despesas operacionais do data center e estejam aumentando mais rapidamente

¹ <http://www.internetlivestats.com/internet-users/>

² <https://software.intel.com/en-us/articles/the-problem-of-power-consumption-in-servers>

³ <https://www.nrdc.org/resources/americas-data-centers-consuming-and-wasting-growing-amounts-energy>

do que os custos de hardware (CHUN et al., 2010). Fazer algo inovador relacionado a redes de computadores e data center é uma tarefa difícil. As tecnologias desses ambientes - como switches e roteadores - são fechadas. Esses tipos de hardware têm firmware específico do fornecedor. Além da questão do hardware, a virtualização ao longo do tempo permitiu virtualizar servidores e funções de rede em máquinas de uso geral. Assim, a virtualização ajuda os data center, tornando-os mais dinâmicos. O avanço significativo da virtualização permite que serviços virtuais sejam migrados sob demanda entre hosts físicos. A migração permite a otimização de recursos enquanto fortalece o SLA. Dessa forma, é possível, por exemplo, automatizar processos, com uma redução em *Operational Expenditure* (OPEX) e *Capital Expenditure* (CAPEX). Nesta perspectiva, têm surgido várias abordagens como propostas para enfrentar os desafios que os sistemas modernos de OPEX e CAPEX enfrentam em relação às infraestruturas de rede (CHOWDHURY; BOUTABA, 2009). A principal tendência entre os vários métodos é a virtualização de redes (e.g., redes de sobreposição - *overlay*, ativas e programáveis) (CHOWDHURY; BOUTABA, 2010). As infraestruturas de rede atuais, estão adotando cada vez mais a programabilidade e virtualização de redes através dos paradigmas de *Software-Defined Networking* (SDN) e NFV (FEAMSTER; REXFORD; ZEGURA, 2013). Tanto o SDN como o NFV permitem maior dinamismo e flexibilidade na gestão de redes no data center, proporcionando um cenário evolutivo das redes atuais (FEAMSTER; REXFORD; ZEGURA, 2013).

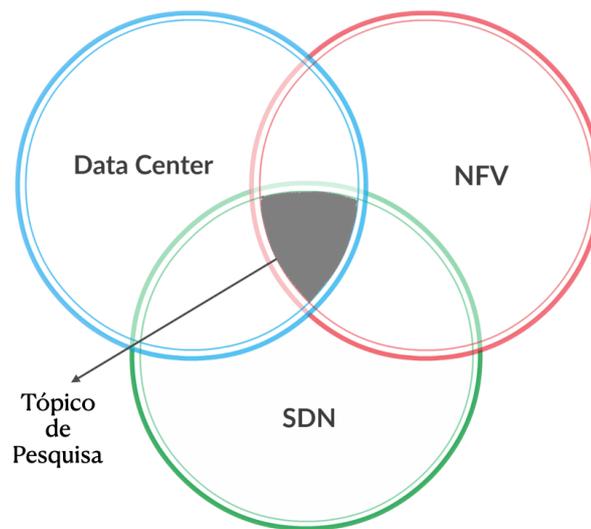
SDN é uma arquitetura de rede que permite desacoplar o plano de controle (lógico) do plano de dados. Por exemplo, SDN cria uma camada programável de software na rede, que pode ser vista como um sistema. O plano de dados consiste em dispositivos de roteamento (switches e roteadores físicos ou virtuais). O plano de controle é logicamente centralizado na entidade de software, chamada de controlador SDN (MCKEOWN et al., 2008). O NFV é um paradigma que utiliza o conceito de virtualização, herdado do mundo dos data center, para executar funções de rede como firewalls, balanceamento de carga e *Deep Packet Inspection* (DPI), tipicamente embutidos em dispositivos de rede dedicados, em *Commercial Off-The-Shelf* (COTS) (SCHAFFRATH et al., 2009; CHIOSI et al., 2012). É importante notar que essas abordagens permitem o gerenciamento dos recursos da rede de uma maneira completamente diferente das redes tradicionais.

Normalmente, as operadoras de rede gerenciam vários dispositivos através de comandos específicos de baixo nível em equipamentos integrados verticalmente com arquiteturas proprietárias. Através da virtualização de recursos, a operadora de rede pode gerenciar recursos de forma mais flexível para atingir objetivos específicos. O campo da virtualização de redes é recente em relação a virtualização de servidores, e há várias abordagens e tecnologias concorrentes (BARI et al., 2013; CHOWDHURY; BOUTABA, 2010). Essa multiplicidade de alternativas concorrentes contribui para aumentar a complexidade desses sistemas de virtualização. É importante notar que, além da virtualização, o desempenho do data center é influenciado por diversos fatores, como arquitetura de software e recursos disponíveis.

A ideia da arquitetura NFV permite virtualizar funções de rede como firewalls e balanceamento de carga. Além disso, o NFV permite organizar recursos físicos e virtuais, otimizando provisionamento, configuração de rede, alocação de largura de banda, automação de operações, monitoramento, segurança e políticas de controle. Essa cooperação tende a reduzir os custos da empresa na aquisição de novos hardwares de rede. Além disso, uma pesquisa realizada em 2015 pelo IHS Markit⁴ sobre o crescimento do NFV em 2015-2019 projetou um investimento próximo a US\$ 11,6 bilhões. O mesmo estudo afirma que 100% dos prestadores de serviços consultados pretendem adaptar sua infraestrutura para usar NFV. Destes, 81% planejavam fazer essa modificação até 2019.

No cenário específico do data center, a adoção da NFV transfere funções de rede para um COTS ao invés de dispositivos proprietários especializados. NFV permite que hardware de uso geral seja usado para reduzir custos através da interoperabilidade de diferentes sistemas. Para tanto, conforme ilustrado na Figura 1, esta pesquisa de doutorado tem focado no impacto da SDN/NFV em data center.

Figura 1 – Tópico de Pesquisa



Fonte: O Autor

A alocação e gestão eficiente dos recursos da rede é uma questão que deve ser considerada pelos profissionais de TI. No entanto, quando se trata de redes virtualizadas, uma questão se destaca: Quais são os riscos? Os riscos são geralmente inerentes à infraestrutura física e virtualizadas utilizada (nós e enlace). Portanto, os riscos devem ser considerados em ambas as camadas, infraestruturas virtualizadas em execução na infraestrutura física. A maioria das abordagens de NFV em data center, foca na interconexão de recursos virtuais com a estrutura física (MAINI; MANZALINI, 2014; RICCOBENE et al., 2014; SHEN et al., 2014; CLAYMAN et al., 2014). Outras pesquisas estão relacionadas ao desempenho da NFV em

⁴ <https://technology.ihs.com/545830/nfv-market-to-grow-more-than-5-fold-through-2019-as-operators-virtualize-networks-says-ihs>

função da variação da carga de trabalho (SCHAFFRATH et al., 2009; WANG; HU; YANG, 2014). Assim, um dos desafios para o gerenciamento da VNF no data center está na alocação de recursos com serviço tolerante a falhas. Para este fim, realizamos uma revisão sistemática (SOUZA; DIAS; FERNANDES, 2020) para permitir um profundo conhecimento, sobre um subproblema de alocação de recursos que é o escalonamento de VNFs em data center.

1.1 ESCOPO DA PESQUISA E MOTIVAÇÃO

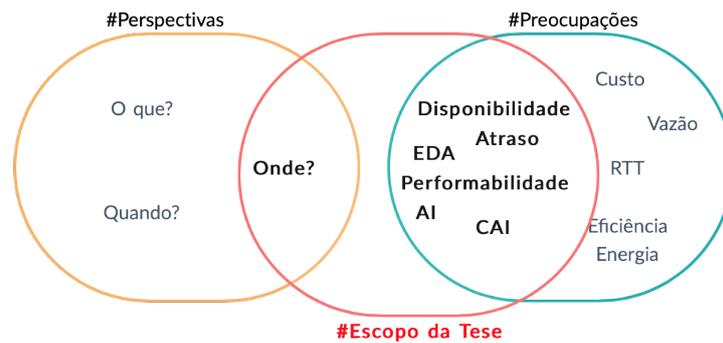
A maioria dos esforços de pesquisa de NFV aplicado em data center, concentra-se em posicionamento de VNFs. Um número menor de artigos, se concentra no problema de escalonamento de VNFs para o data center (ver Tabela 1). O escalonamento de VNFs pode ajudar a evitar falhas, degradação do desempenho ou violação dos requisitos do SLA, causados por excesso ou falta de fornecimento de serviços (ISLAM; LIU, 2012; ABU-JODA; KOUCHAKSARAEI; PAPADIMITRIOU, 2016; BECK; BOTERO, 2017). Os sistemas de escalonamento geralmente exploram uma ou mais das três perspectivas seguintes:

- **O que escalonar?** é um aspecto crucial para o sucesso das aplicações NFV em data center. Escalonamento refere-se à capacidade do data center de lidar com uma quantidade crescente de trabalho, enquanto mantém a disponibilidade, confiabilidade e desempenho à medida que a quantidade de tráfego aumenta. Neste sentido, a escalabilidade também pode ajudar a lidar com picos inesperados de tráfego.
- **Quando escalonar?** corresponde ao processo para decidir o momento mais apropriado para realizar o escalonamento. Para isso, é necessário considerar o tráfego de dados e a condição de funcionamento da rede. A condição de trabalho em rede tem um papel importante no processo de tomada de decisão.
- **Onde escalonar?** Normalmente, o escalonamento de VNFs, envolve decidir para qual *Physical Machine* (PM), as VNFs devem ser enviadas. Há muitos critérios, com base nos quais o escalonamento pode ser feito, por exemplo, minimizar o atraso (fim a fim), reduzir o tempo total de execução do mapeamento obtido. Um data center tem diferentes tipos de servidores em termos de potência de recursos e tecnologia. As características de tal infraestrutura devem ser consideradas para construir uma solução de escalonamento.

A Figura 2 mostra o escopo dessa tese. Com a intenção de fornecer conclusões significativas, esta pesquisa de doutorado se concentra principalmente na perspectiva “onde”, devido às suas características desafiadoras e importantes. **O ponto focal está no escalonamento de VNFs, considerando a estratégia de tolerância a falhas.** Isso requer que o data center e operadores tenham um melhor gerenciamento de rede e (re)planejamento para garantir o desempenho e requisitos de alta disponibilidade. Os aspectos discutidos

acima estão diretamente relacionados à perspectiva de escalonamento de VNFs com os conceitos de tolerância a falhas, performabilidade, disponibilidade, atraso(fim a fim), *Availability Importance* (AI), *Critically Availability* (CAI), *Exploratory Data Analysis* (EDA).

Figura 2 – Escopo da Pesquisa de Doutorado



Fonte: O Autor

Portanto, desenvolvemos modelos de aprendizagem profunda para permitir um processo inteligente de escalonamento com a rede de substrato.

1.2 VISÃO GERAL DO PROBLEMA

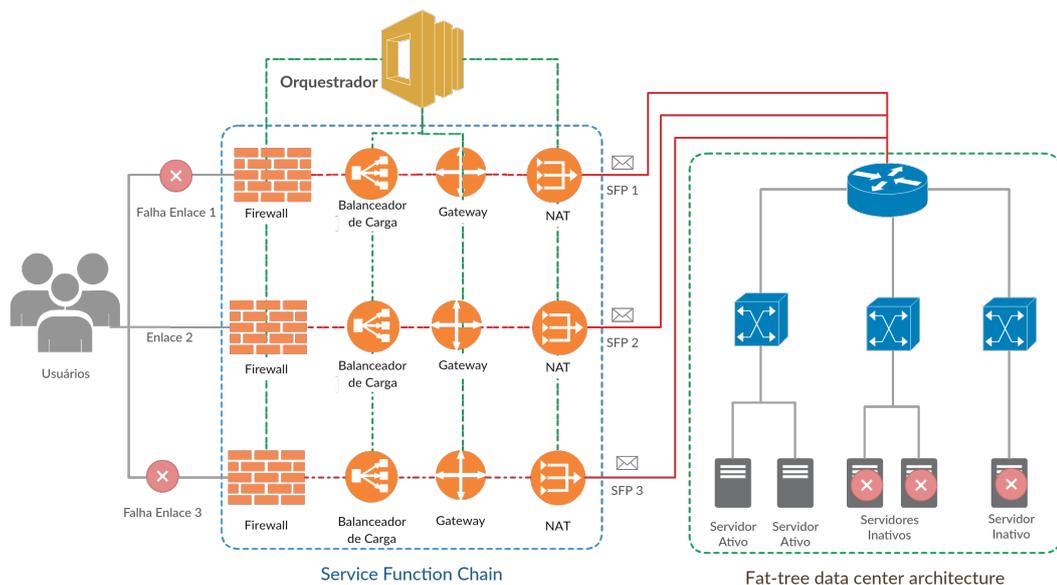
A ocorrência de falhas no data center é comum, dada a sua complexidade em termos de uma série de tecnologias. Isso é ainda mais crítico em grandes empresas como a Google ou a Amazon, as quais têm impactos financeiros. Para auxiliar em uma melhor gerenciamento de recurso, existem algumas métricas de monitoramento, tais como *Mean Time Between Failures* (MTBF), *Mean Time to Restore* (MTR), MTTF, *Mean Time To Repair* (MTTR) para aumentar a confiabilidade nesses ambientes. Estas métricas são, em muitos casos, contempladas no SLA assinado entre provedores de serviços de Internet e clientes (URIARTE; TIEZZI; NICOLA, 2014). Embora impactantes, as falhas nos data center são frequentes ao longo do ano. As falhas de enlace, conhecidas como falhas de curto prazo, são mais comuns. Segundo (GILL; JAIN; NAGAPPAN, 2011a), esse tipo de falha corresponde a 41% do total. Quando ocorre, essa falha geralmente afeta apenas alguns (2-4) enlaces, mas em 10% dos casos, afeta mais de quatro dispositivos. As falhas de curto prazo são mais frequentes no data center, e o tempo médio de falha é de 8,6 minutos.

Normalmente, os problemas estão relacionados com *bugs* de *firmware* que gera baixa disponibilidade do dispositivo. Por outro lado, as falhas no nível de pacotes são decorrentes de um fluxo que envia uma sequência de pacotes através do enlace de rede. O número de fluxos impactados é tipicamente uma questão de aplicações que são sensíveis à latência de rede. Mesmo alguns fluxos atrasados podem afetar significativamente a realização

de sistema de ponta a ponta. No caso de falhas de hardware raramente ocorrem, mas contribuem fortemente para a indisponibilidade da rede (GILL; JAIN; NAGAPPAN, 2011a) mostram que uma redundância de hardware um-para-um, dedicada à recuperação de falhas, forneceu cerca de 90% da quantidade de tráfego entregue no caso de uma falha média.

O desempenho é pior no final da cauda da distribuição de probabilidade do tráfego ao longo do tempo, com apenas 60% do tráfego entregue durante 20% das falhas. Isto sugere, que são necessários melhores métodos para explorar o escalonamento existente. Neste contexto, o conceito de dependabilidade pode ser definido a partir de um conjunto de métricas compostas da seguinte forma: disponibilidade, confiabilidade, integridade, segurança, manutenção para elevar os níveis de tolerância a falhas (AVIZIENIS et al., 2004). Para tal, é essencial realizar uma análise de redes virtualizadas sob métricas de dependabilidade, para conhecer o nível de disponibilidade de NFV em data center, por exemplo, pode garantir o sucesso de um negócio. Para isso, considere o cenário ilustrado na Figura 3 para representar o nosso problema de pesquisa.

Figura 3 – Descrição do Problema



Fonte: O Autor

Suponha que um operador de rede pretenda construir uma infraestrutura para NFV aplicada ao data center. A partir da escalonamento de VNFs em data center, considere que uma ou várias VNFs falham ou vários enlaces falham. Portanto, é necessário ter redundância para garantir a alta disponibilidade necessária, mesmo em caso de falha. Devido à redundância, é possível fazer a comunicação, através do *Service Function Path* (SFP)₂ interagindo com o servidor um e o servidor dois. Esses eventos de indisponibilidade indicam que o processo de escalonamento e planejamento de alta disponibilidade é crucial

para o desenvolvimento de estratégias de comunicação, a fim de fornecer um serviço de alta qualidade e ininterrupto em um ambiente NFV aplicado a data center.

No entanto, há vários desafios sobre um escalonamento inteligente, embora as técnicas de aprendizado de máquina e aprendizagem profunda forneçam ferramentas flexíveis para a aprendizagem de um sistema inteligente de escalonamento de VNFs em data center. Atualmente, existe na literatura um novo paradigma chamado *Knowledge-Defined Networking* (KDN) (MESTRES et al., 2017), que proporciona automação, recomendação, otimização, validação e estimação. No entanto, não foi abordado para escalonamento de VNFs. Atualmente, o KDN considera os três planos tradicionais de SDN: (i) O Plano de Dados: responsável pelo armazenamento, encaminhamento e processamento dos pacotes de dados, (ii) O Plano de Controle: troca o estado operacional a fim de atualizar as regras correspondentes e processamento do plano de dados, e (iii) O Plano de Gerenciamento: assegura a operação correta e o desempenho da rede. A partir disso, levantam-se alguns benefícios relacionados ao problema de pesquisa, os quais são descritos nas hipóteses a serem investigadas:

- H1:** Se aplicarmos o aprendizagem profunda em vez da *Integer Linear Programming* (ILP) para otimizar o escalonamento de VNFs, então será possível encontrar resultado próximo do ótimo em um tempo reduzido e com qualidade adequada. Por exigir um alto custo computacional, é difícil encontrar uma solução ótima com ILP para problemas NP-difícil.
- H2:** Se automatizar as políticas de alocação de data center com KDN, haverá menos desperdício de recursos e problemas de escalonamento de VNFs, o que permitirá aumentar o desempenho.
- H3:** Se construirmos uma ferramenta de diagnóstico, será possível auxiliar o projetista no melhor ajuste do framework de escalonamento de VNFs.
- H4:** Se modelos de RBD forem adotados no lugar de outros modelos analíticos, então pode-se extrair equações de forma fechada, que proporciona a resolução do problema do explosão de espaço de estados e auxiliar no replanejamento, baseado em estratégias de redundância do data center.

1.3 QUESTÕES DE PESQUISA

Como já foi dito, as VNFs são usadas para conseguir um melhor escalonamento entre dispositivos e uma adaptabilidade dinâmica com técnicas de tolerância a falhas. Nessa tese, o escalonamento de VNFs tolerante a falhas, baseado em modelos de aprendizagem profunda, como LTSM tem a vantagem de permitir a análise de datasets em larga escala com características uniformes, ruidosos e não lineares (AHMED et al., 2010; PALIT; POPOVIC,

2006). Em particular, o modelo LTSM (GREFF et al., 2017), resolve o problema de *vanishing gradients*⁵ que aparecem ao utilizar *Recurrent Neural Network* (RNN), onde um novo estado é introduzido e o permite que o erro se propague de volta sem desaparecer. Além disso, conforme as redes se tornam mais extensas e dinâmicas, os requisitos de serviço do usuário mudam com mais frequência.

Neste contexto, os operadores de rede não podem realizar o escalonamento VNFs manualmente, sendo de maneira autônoma um aspecto fundamental para o sucesso da NFV (CHIOSI et al., 2012). Considerar a geração de políticas de afinidade e antiafinidade como um mecanismo adicional de alocação de recursos para NFV pode ajudar a resolver esta questão, uma vez que, no contexto dos modelos LTSM, podem melhorar eficientemente o uso de dispositivos físicos e de rede, permitindo prever o futuro em várias unidades de tempo a frente do escalonamento. Portanto, considerando as hipóteses estabelecidas, a partir desta descrição dos problemas e dos objetivos de pesquisa desta tese, a seguinte *Questão de Pesquisa* (QP) pode ser deduzida:

Como criar uma estratégia inteligente e eficiente para escalonamento de VNFs, considerando aspectos de tolerância a falhas em data center?

Como esta tese explora uma QP primária, outras questões fundamentais que se relacionam com os problemas de pesquisa primária foram produzidas.

QP1: *A partir da H1, qual modelo e estratégia de aprendizagem profunda mais adequada para realizar o escalonamento de VNFs?*

QP2: *A partir da H2, como criar uma estratégia inteligente para o escalonamento de recursos de rede, considerando aspectos de um data center tolerante a falhas em cenário NFV?*

QP3: *A partir da H3, qual é o conjunto de métricas representativas que podem quantificar os impactos na degradação do desempenho do escalonamento de VNFs e servir como input para o operador de rede ajustar o modelo de aprendizagem profunda?*

QP4: *A partir da H4, como identificar ocorrências de falhas, e auxiliar o operador de rede para replanejar a redundância do dispositivo de maneira exata?*

Portanto, buscamos encontrar soluções para essas questões de pesquisa identificadas, como potencialmente relevantes para a solução de problemas de escalonamento de VNFs em data center.

⁵ Problema de *vanishing gradients*, são funções com gradientes, que tem valores absolutos muito altos ou muito próximo de zero, que propagam numericamente ou explodem a cada nova camada do modelo, pelo motivo da aplicação recursiva da regra de cadeias no *backpropagation*

1.4 OBJETIVO E PROPÓSITO DA PESQUISA

Esta tese de doutorado propõe soluções para o problema de escalonamento de VNFs aplicada aos data center. Neste contexto, fornecer novas soluções para a gestão de recursos, com um foco particular na desafiadora e complexa tarefa de lidar com a escassez de recursos alocados às VNFs, torna-se crucial (YI et al., 2018). Embora várias técnicas de inteligência artificial e heurísticas tenham sido propostas na literatura a fim de resolver problemas de escalonamento em data center (YEKKEHKHANY; HOJJATI; HAJIESMAILI, 2018; GILL; JAIN; NAGAPPAN, 2011b). Nessa tese, é utilizado o escalonamento de VNFs para data center, com foco especial em técnicas de tolerância a falhas, bem como levar em conta o escalonamento baseada no paradigma KDN. As estratégias desenvolvidas consideram alguns dos atributos de disponibilidade, desempenho e performabilidade durante a criação de uma solução.

O objetivo é conceber métodos aplicados à NFV em data center para suportar melhor o desempenho, disponibilidade e tolerância a falhas.

De uma perspectiva mais ampla, avaliar se NFV aplicada em data center, melhora os serviços de infraestrutura crítica. Se eles podem ser melhorados, todas as aplicações baseadas nas redes resultantes, podem se beneficiar de ciclos de desenvolvimento mais rápidos. Entre os **objetivos específicos** da pesquisa, podemos listar:

- Implementar o novo paradigma de KDN no escalonamento de VNFs no data center, considerando tolerância a falhas;
- Projetar estratégias para apoiar a avaliação de desempenho, disponibilidade e performabilidade do NFV em data center;
- Projetar e implementar técnicas que permitam um replanejamento eficiente do data center;

1.5 MÉTODOS E METODOLOGIAS

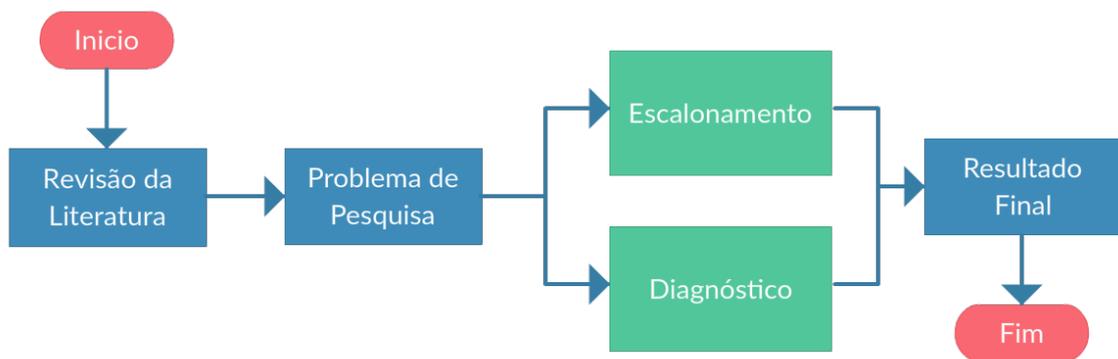
Nesta seção, são discutidos métodos de pesquisa, abordagens de pesquisa, estratégias de pesquisa, coleta de dados e técnicas de análise. Além disso, o método e a metodologia escolhidos para este estudo também são descritos.

Metodologia de Pesquisa

Para alcançar os objetivos acima, esta pesquisa de doutorado seguiu a metodologia descrita na Figura 4. Em primeiro lugar, foi realizada uma revisão sistemática sobre aplicação de NFV em um data center. As principais contribuições dessa tese são conduzidas de duas maneiras: (i) investigar como criar estratégias inteligentes de NFV aplicadas em data

center, utilizando novas ferramentas, no Capítulo 4; (ii) define o experimento e explorar a avaliação em NFV em data center no Capítulo 5.

Figura 4 – Metodologia de Pesquisa



Fonte: O Autor

Método de Pesquisa

O paradigma positivista tem como objetivo da pesquisa, provar ou refutar uma hipótese. Os métodos de pesquisa podem ser categorizados em dois: quantitativos e qualitativos (HÅKANSSON, 2013; TERRELL, 2015). Para desenvolver um mecanismo eficiente de alocação de recursos para um data center, um método de pesquisa quantitativo/experimental é o mais adequado para conduzir o estudo. Na literatura, existem algumas abordagens de pesquisa, como indutiva, dedutiva e abdutiva, que são utilizadas para concluir. Para tanto, a abordagem dedutiva é mais apropriada e escolhida para conduzir o estudo. Existem diversas estratégias de pesquisa associadas ao método de pesquisa quantitativa, como a experimental e a pesquisa (CRESWELL, 2013).

A coleta de dados quantitativos são aplicadas na avaliação preliminar. Neste trabalho, o método experimental é utilizado para coletar grandes datasets de variáveis. Além disso, devido ao número limitado de máquinas disponíveis para testes, vamos explorar e analisar as seguintes lacunas: modelo analítico, simulação através de modelos de LSTM para escalonamento de VNFs em data center, considerando tolerância a falhas. Devido a isso, desenvolvemos modelos de autoaprendizagem no lugar de métodos exatos. Os pesquisadores têm frequentemente aplicado ILP para otimizar a alocação de *Virtual Machine* (VM) e gerenciamento de rede (MENG; PAPPAS; ZHANG, 2010). No entanto, essa abordagem não automatiza o processo de escalonamento de VNFs é muito lenta para atender às demandas e requisitos de tráfego dinâmicos, onde os experimentos mostram que para implementar dez funções de rede otimizando a computação e os recursos de rede, os cálculos da ILP levam mais de 2 horas (RANKOTHGE et al., 2015).

A quantidade de tráfego em data center e a dependência não-linear das variáveis envolvidas indicam que as abordagens de ILP podem não ser a ferramenta apropriada para resolver o problema. Dentro desse contexto, consideramos o uso de LSTM para o

escalonamento de VNFs em situações em que os métodos de otimização são impraticáveis e até mesmo em questões que envolvem um grande número de soluções possíveis. O tempo geralmente necessário para a ILP calcular uma solução é insatisfatório, especialmente considerando que as características do tráfego do cliente podem mudar rapidamente. Os modelos podem ser resolvidos analiticamente ou por simulação. Os modelos analíticos utilizam fórmulas fechadas ou um conjunto de equações do sistema para descrever o comportamento de um sistema. As métricas de interesse podem ser fornecidas pela resolução de fórmulas fechadas ou pela solução exata ou aproximada de um conjunto de equações de sistema fornecidas por algoritmos de matemática numérica (BOLCH et al., 2006). As *Stochastic Petri Net* (SPN) possuem maior poder de representação utilizando simulações e análises numéricas (PUIGJANER; SAVINO; SERRA, 2003). No entanto, o RBD apresenta cálculos mais rápidos utilizando fórmulas fechadas.

Resultados estruturados e conclusões podem ser extraídos com o auxílio da análise de dados. Os métodos de análise de dados mais utilizados na pesquisa quantitativa, são a matemática estatística e a matemática computacional, por isso foram escolhidos para esta tese. A análise de dados inclui inspeção, limpeza, transformação e modelagem dos dados coletados. Nesta tese, a estatística descritiva é utilizada para analisar os dados e avaliar a significância dos resultados. A estatística descritiva sintetiza os dados de forma significativa para que as conclusões possam ser extraídas dos dados.

1.6 CONTRIBUIÇÕES DA TESE

Em resposta aos desafios de pesquisa, o escalonamento de VNFs para o ambiente de data center é desenvolvido com um foco particular em técnicas de tolerância a falhas, bem como levando em conta o fator crucial de prever a tomada de decisão no momento do escalonamento. O framework de escalonamento proposto é um modelo LTSM, juntamente com modelos de afinidade em um processo autônomo de escalonamento de VNFs dentro do ambiente do data center, bem como uma análise holística de performabilidade, desempenho e disponibilidade. Os resultados mostram que o framework de escalonamento proposto permite detectar a falha e o impacto no desempenho da rede antes que o problema ocorra. É essencial compreender o sistema porque a realização de uma análise de falhas pode prevenir e evitar riscos de mal funcionamento de VNFs. Desta forma, a estrutura desenvolvida possibilita um escalonamento sem intervenção humana que é de maneira autônoma e com intervenção humana, onde fornece uma análise preditiva que ajuda o operador de rede a agir antecipadamente, melhorando alocação de recursos e aumentando a disponibilidade, bem como funcionando de maneira autônoma sem a intervenção humana. As principais contribuições são as seguintes:

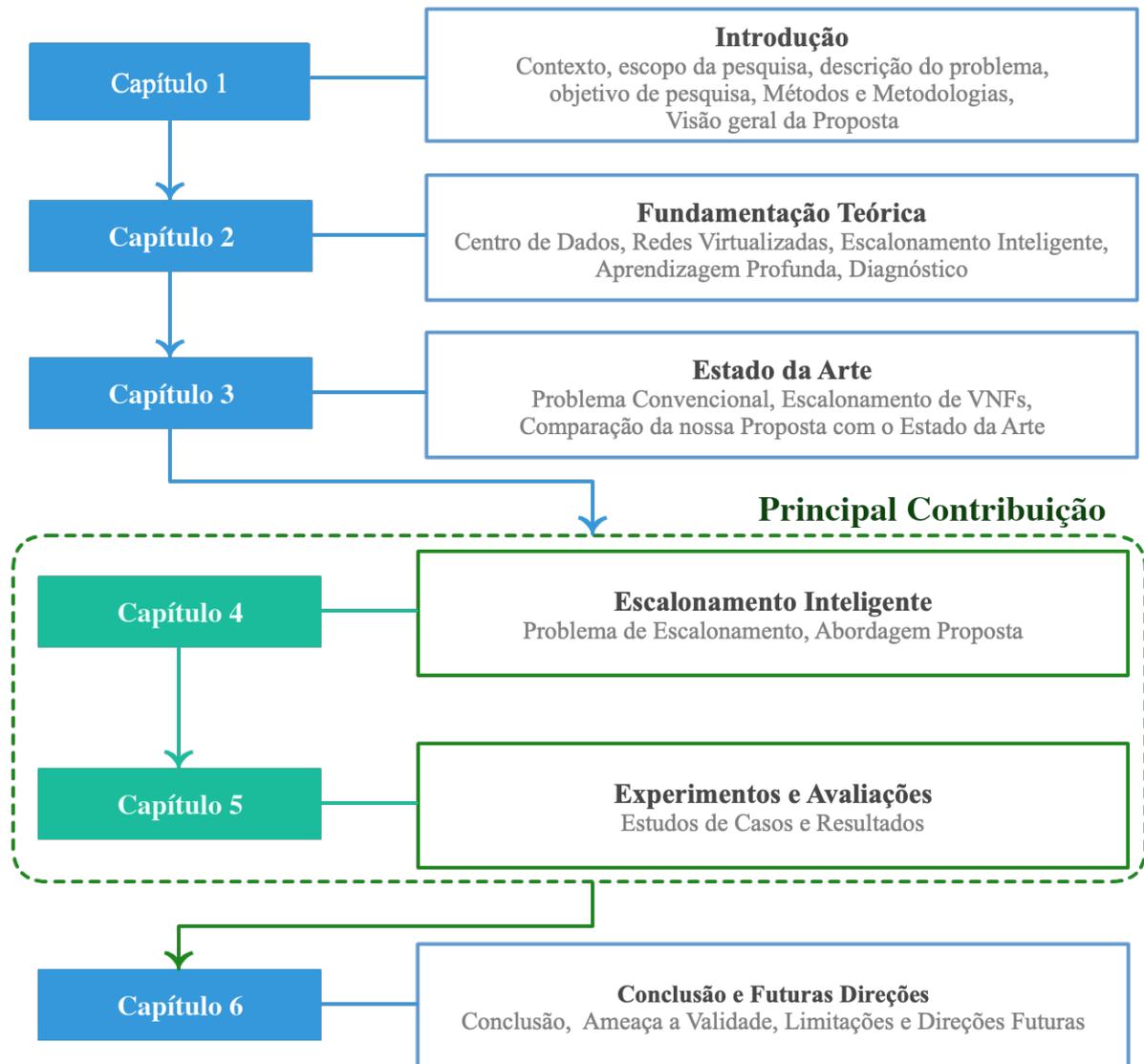
- Framework tolerante a falhas, baseado em KDN para o escalonamento de VNFs em data center

- Ferramentas de diagnóstico de rede, baseadas em novos modelos de performabilidade, EDA, com base em efeitos de temperatura e variação de tráfego.
- Framework inteligente de escalonamento de VNFs que não requer um sistema operacional ou modificações no kernel.
- Demonstração da importância da análise do impacto da temperatura para estimar o desempenho e a disponibilidade após a escalonamento de VNFs nos data center.
- Ajuste automático dos parâmetros de escalonamento de VNFs para fornecer uma alocação justa baseada tanto na taxa de chegada dos pacotes como no custo de computação necessário.

1.7 ESTRUTURA DA TESE

A organização dessa tese deve ser lida sequencialmente, no entanto, são possíveis variações. Um roteiro das conexões entre os capítulos é apresentado na Figura 5. No geral, as contribuições são: arquiteturas e algoritmos de escalonamento inteligente, no Capítulo 4, e os experimentos e resultados, no Capítulo 5.

Figura 5 – Dependências Lógicas entre os Elementos desta Tese.



Fonte: O Autor

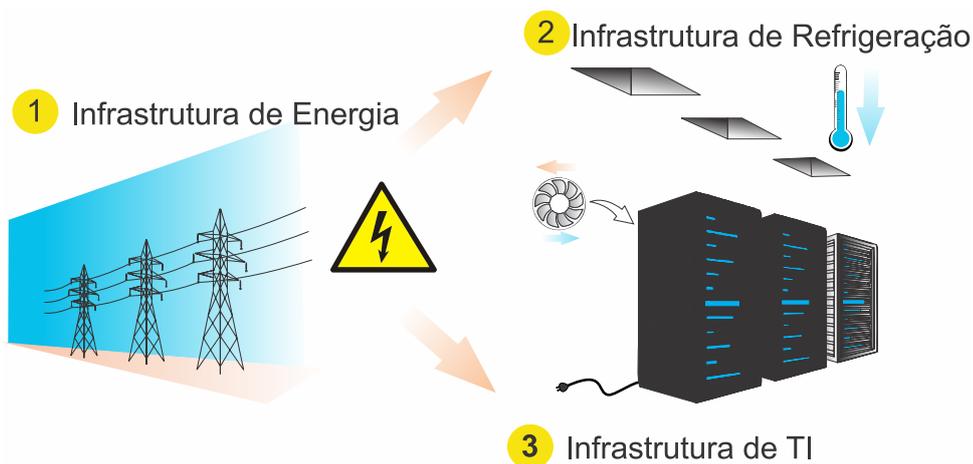
2 CONTEXTO TEÓRICO

Este capítulo apresenta os conceitos necessários para compreender o trabalho. Os tópicos incluem tecnologias de data center, virtualização de rede, aprendizado de máquina, aprendizagem profunda, modelagem de disponibilidade, desempenho de rede e performabilidade. O paradigma NFV é apresentado e seus componentes arquiteturais são discutidos com mais profundidade.

2.1 DATA CENTER

Os data center são conjuntos de recursos computacionais escaláveis para processamento e armazenamento de dados. Os data center são viáveis para organizações de vários tamanhos e de várias naturezas, de instituições educacionais, indústrias, e agências governamentais. No data center atual, o uso de técnicas de virtualização permite a alocação flexível de recursos para máquinas virtuais. No entanto, a virtualização vem a um custo de maior complexidade para sua separação da infraestrutura subjacente. Um sistema de data center genérico (ver Figura 6) consiste essencialmente de três subsistemas: (i) Infraestrutura de TI; (ii) infraestrutura de refrigeração; e (iii) infraestrutura de energia (MAINI; MANZALINI, 2014).

Figura 6 – Infraestruturas de data center.



Fonte: O Autor

A infraestrutura de TI consiste em três componentes de hardware: servidores de processamento, servidores de armazenamento e equipamentos de rede. Os servidores de processamento, em geral usam Ethernet para conectar sistemas cruzados de acesso remoto a arquivos através de um dispositivo de armazenamento conectado a uma rede *Network-Attached Storage* (NAS); os dispositivos de armazenamento gravam dados. Eles são,

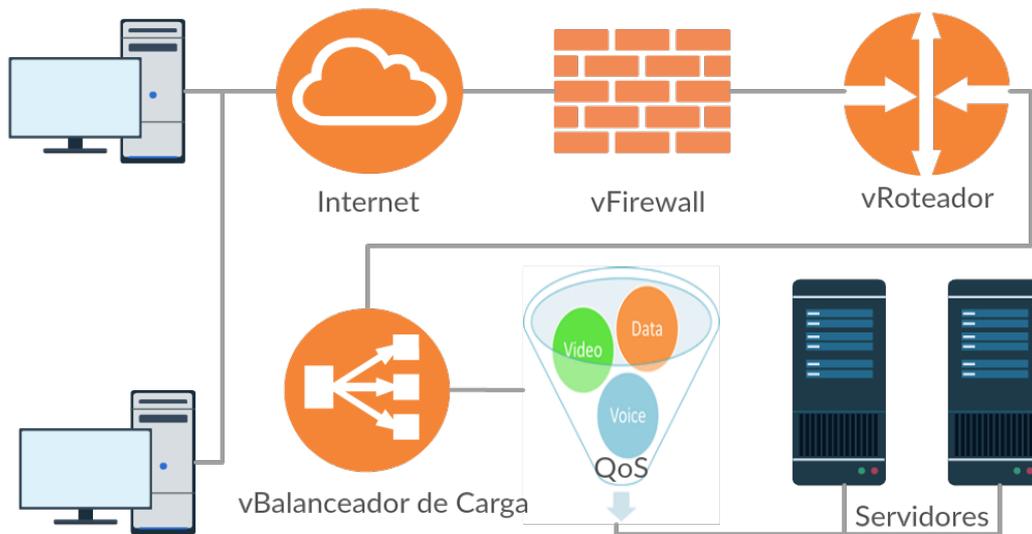
finalmente, conectados através de uma rede de *Storage Area Network* (SAN) a equipamentos de rede, tais como switches e roteadores, intercomunicadores e outros equipamentos.

A Infraestrutura de Energia fornece energia ininterrupta na frequência e tensão especificadas para os dispositivos de TI e refrigeração (FAN; WEBER; BARROSO, 2007). Normalmente, a energia passa por *Step Down Transformer* (SDT), *Power Distribution Unit* (PDU), interruptores estáticos ou automáticos, *Power Supplies* (UPS) e, finalmente, barras de alimentação em rack, que são usadas para conectar dispositivos elétricos. O UPS fornece energia e backup sobre baterias em caso de falha de curto prazo. Em contraste, outro tipo de fonte de alimentação local - como a alimentação de combustível diesel - fornece energia para interrupções de tempo mais prolongadas.

A Infraestrutura de resfriamento, como o nome indica, inclui *Computer Room Air Conditioning* (CRAC), e torres de resfriamento (ARREGOCES; PORTOLANI, 2003). A infraestrutura de resfriamento representa aproximadamente 15%-20% em média do consumo de energia de um data center. As unidades CRAC extraem o calor dissipado dos equipamentos de TI e transferido para os sistemas de distribuição de água refrigerada. Os resfriadores removem o calor do sistema de água gelada e o rejeitam para o meio ambiente por meio de torres de resfriamento. Há perda de água por evaporação para o ambiente durante este processo de rejeição de calor (MILLER, 2011). Uma torre de resfriamento é um equipamento que reduz a temperatura de um fluxo de água, extraindo calor da água e emitindo-o para a atmosfera. Um fluxo de água resfria a infraestrutura de resfriamento, na qual o ar quente retornava ao CRAC em um trocador de calor fechado; quando o calor é transferido do fluxo de ar para o fluxo de água refrigerada, as temperaturas da água aumentam. Além disso, o calor é removido da água por meio de trabalho termodinâmico para melhorar o ciclo de refrigeração do chiller. A água refrigerada é devolvida para a unidade CRAC, enquanto o calor absorvido e seguidamente rejeitado para uma corrente de água secundária. O circuito secundário, finalmente transfere o calor para o ambiente externo em uma torre de resfriamento.

Além dos três componentes significativos acima, os data center requerem redes de alta velocidade com mecanismos de failover. Essas redes evitam gargalos e fornecem disponibilidade com confiabilidade comprovada. Os dispositivos de rede precisam de parâmetros especiais de *Quality of Service* (QoS) para alcançar alta produtividade e baixa latência. Três componentes principais geralmente cobrem a operação do data center: um Firewall, um balanceador de carga e um redirecionamento relacionado à QoS, com sobreposição de rede. Isso é feito, conforme descrito na Figura 7. A regra mostra todos esses componentes e onde eles normalmente estão localizados no caminho de roteamento do data center.

Figura 7 – Semântica de operação do data center



Fonte: O Autor

2.1.1 Virtualização de Rede

A adoção efetiva de tecnologias de virtualização teve origem há aproximadamente 50 anos. Em 1974, o conceito de máquina virtual foi formalizado. Na época, dizia-se que uma máquina virtual é uma réplica eficiente e isolada de uma máquina real (POPEK; GOLDBERG, 1974). A IBM foi pioneira nas primeiras estratégias de virtualização envolvendo processador e RAM (RIXNER, 2008). No entanto, a virtualização em redes de computadores levou muito mais tempo para começar a ser introduzida. Ideias de virtualização para redes de computadores foram introduzidas há cerca de 20 anos com tecnologias como *Virtual Private Networks* (VPN). Em seguida, houve o surgimento de várias outras tecnologias, tais como: *Virtual LAN* (VLAN), *Multiprotocol Label Switching* (MPLS), *Generalized Multi-Protocol Label Switching* (GMPLS), SDN e Network Functions Virtualization (NFV).

O processo de virtualização de redes tem como principal premissa superar barreiras tecnológicas conhecidas, que dificultam a aplicação de mudanças na rede, criando um fenômeno conhecido como a ossificação da Internet, sendo apontado como a solução para a Internet do futuro em diversos trabalhos (PAPADIMITRIOU et al., 2009; VERDOUW et al., 2014). Com a tecnologia de virtualização, é possível ter maior poder para realizar programação em redes. Essa programação permite que a rede seja independente de protocolos como o *Internet Protocol* (IP) ou qualquer tecnologia específica. Torna-se possível criar arquiteturas diferentes e utilizar novos protocolos em um novo ambiente de rede que pode se tornar muito heterogêneo. Por exemplo, os gerentes de nuvem podem contar com novos sistemas para configurar serviços e funções de rede, incluindo

ferramentas: Puppet¹, Chef², Ansible³ Saltstack⁴ e outras. Uma vantagem destas soluções é a possibilidade de integrar uma função de rede sem qualquer modificação na própria função. O uso de módulos de software adicionais, como agentes ou plugin, liberta os desenvolvedores da implementação de uma instância específica de seu papel para cada gerenciador de nuvem utilizado pelo data center.

A virtualização é a capacidade de simular uma plataforma de hardware, serviço ou recurso de rede através de uma camada de software. Neste contexto, uma instância virtual pode operar praticamente da mesma forma que uma solução não virtualizada. Uma plataforma com suporte a virtualização tem hardware de uso geral que pode suportar vários serviços ou dispositivos virtualizados no mesmo hardware, proporcionando maior flexibilidade, capital e custos operacionais reduzidos. Embora uma rede virtualizada ofereça vários benefícios, há vários desafios, como o isolamento entre diferentes redes virtuais, garantia de escalabilidade, disponibilidade e interoperabilidade (CHOWDHURY; BOUTABA, 2010).

2.1.2 Virtualização das Funções de Rede

Usando máquinas de uso geral em uma infraestrutura NFV, podemos executar várias funções de rede diferentes de forma virtualizada (por exemplo, VPN, Firewall, balanceador de carga, etc.). As VPN são instanciadas dentro de redes sob demanda. Os requisitos de desempenho são identificados em aplicações cliente (ETSI, 2014) e as quatro partes da arquitetura de referência NFV são mostradas na Figura 8.

- VNF é uma máquina virtual concebida para executar uma determinada função de trabalho em rede.
- *Network Functions Virtualization Infrastructure* (NFVI) consiste em hardware e software, que é a base para o ambiente virtualizado através de um Hypervisor;
- *Operational Support Systems* (OSS) e *Business Support Systems* (BSS) lidam com o sistema host que precisa ser integrado ao ambiente NFV para se adequar aos processos do operador;
- *Management and Orchestration* (MANO) cria e providenciar as VNFs de forma automatizada, configurar as VNFs e a NFVI, além de controlar o ciclo de vida dos recursos físicos das VNFs e coordena recursos de rede para aplicações baseadas em nuvem.

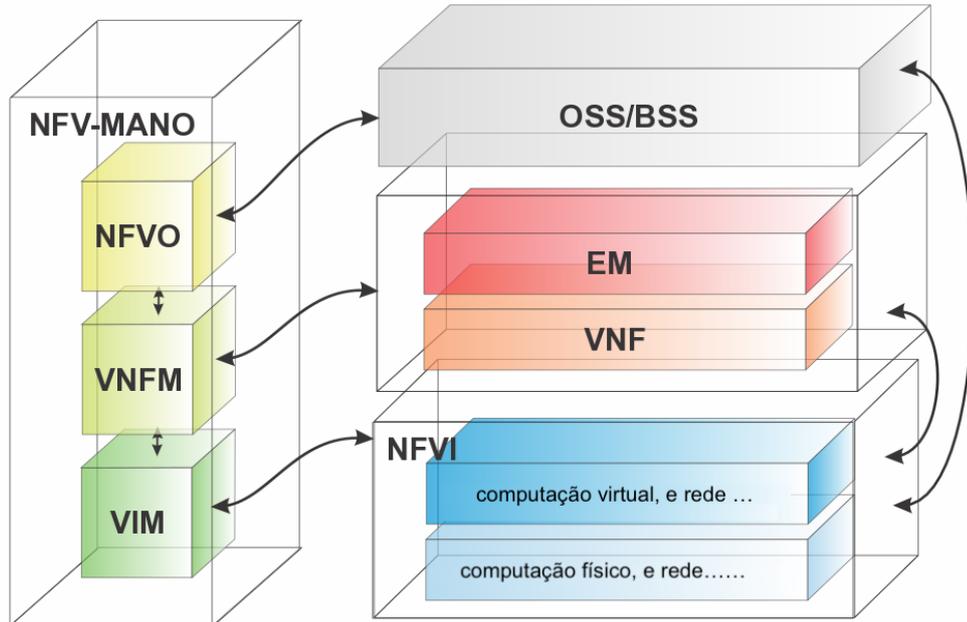
¹ www.puppet.com

² www.chef.io

³ www.ansible.com

⁴ www.saltstack.com

Figura 8 – Arquitetura de Referência NFV



Fonte: Adaptado de (ETSI, 2014)

Outro componente importante na arquitetura do NFV é (i) *VNF Manager* (VNFM), que gerencia o ciclo de vida das VNFs sob controle do *NFV Orchestrator* (NFVO); (ii) *Virtualized Infrastructure Manager* (VIM) que gerencia a infraestrutura de uma solução baseada em NFV; e (iii) NFVO, que concentra a inteligência para a distribuição da VNF pela rede. O NFVO é também responsável pela interligação de ambientes físicos e virtuais, bem como pela integração com OSS e BSS. Dentro do campo da virtualização é importante falar sobre dois conceitos: os nós virtuais e a virtualização de enlace.

Nós virtuais são abstração de rede que não existem em forma física pura, mas se comportam como um equivalente físico. O papel da virtualização é replicar ou estender um comportamento, recurso, serviço ou interface (por exemplo, roteadores, funções de rede, enlace e máquinas virtuais). Cada nó virtual precisa ser alocado a um nó físico. Uma das diferenças de outras abordagens é que um nó virtualizado não é executado em um hardware dedicado. Finalmente, um nó virtual é baseado no isolamento e particionamento de recursos de hardware, de acordo com requisitos predefinidos. Distinguimos dois tipos de dispositivos de rede virtual: (1) agregação de dispositivos, onde múltiplos dispositivos de rede funcionam como uma única entidade lógica; e (2) emulação de dispositivos, onde um equivalente de um dispositivo físico é o software emulado.

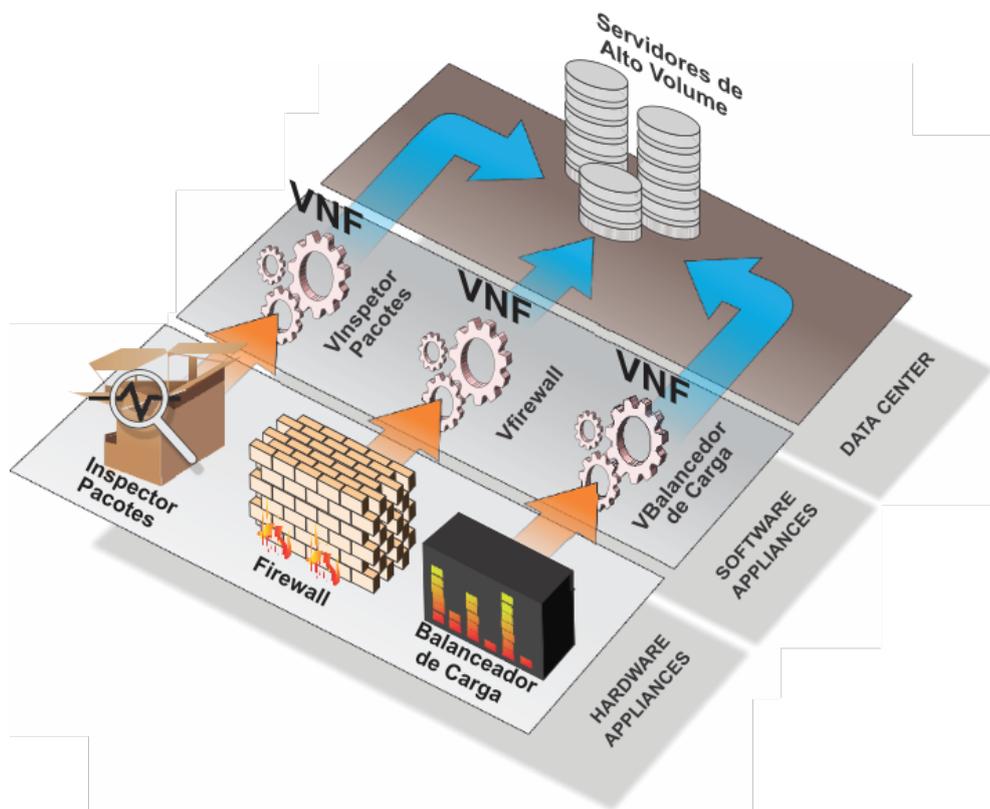
A virtualização de enlace é um conjunto de um ou mais enlaces físicos. Cada conjunto de enlace, ou sequência de enlace, pode transferir múltiplos fluxos de tráfego separados

por um enlace compartilhado (físico ou emulado). Cada fluxo desse tráfego de rede pode estar usando um enlace dedicado chamado enlace virtual. Independentemente de quantos enlaces físicos são usados, um enlace virtual é identificado por um identificador exclusivo. Observe que um enlace físico pode suportar vários enlaces virtuais. Na prática, um enlace virtual é criado por meio de regras de roteamento que definem um caminho específico (por exemplo, MPLS e OpenFlow).

2.1.3 Virtualização das Funções de Rede em data center

A Figura 9 mostra um exemplo de aplicação de NFV a um data center. Assim, diferentes tipos de VNF podem estar localizados em computadores de uso geral no data center. Com NFV, o operador tem maior flexibilidade para definir cadeias de serviço específicas para serviços específicos, de modo que o serviço só siga o caminho diretamente aplicável. O conceito de utilização da virtualização, em combinação com muitos recursos de hardware, também é chamado de computação em nuvem. Na prática, isso pode ser visto como um amplo conjunto de servidores que atuam como hosts, onde cada host pode executar uma ou várias máquinas virtuais.

Figura 9 – NFV data center



Fonte: O Autor

Cada máquina virtual requer um caminho de dados para a rede externa, ou seja, entre a interface de rede e a máquina virtual. Neste processo, alguns dos requisitos em NFV são

importantes (ETSI, 2013):

- Gerenciar redes, tanto físicas como virtuais;
- Realizar migração e provisionamento de VNF em diferentes ambientes;
- Otimizar a alocação de recursos, reduzindo o consumo de energia, distribuindo a carga de trabalho dos serviços de rede;
- Definir a alocação de recursos de forma eficiente para lidar com diferentes tipos de cargas de trabalho para que os contratos de serviço possam ser atendidos;
- Fornecer o mesmo nível de serviço para a função de rede virtual ou física.

Tradicionalmente, a prestação de serviços na área das telecomunicações tem sido realizada com base na implementação de dispositivos de hardware próprios especificamente concebidos para cada serviço de rede. Além disso, os componentes de serviço têm uma sequência e encadeamento restritos que devem ser refletidos, tanto na topologia da rede como na localização dos elementos de serviço (MIJUMBI et al., 2016). Se, por um lado, esta ligação entre hardware, serviços e topologia de rede torna os serviços robustos, estáveis e fortemente compatíveis com os protocolos utilizados, por outro lado, causa fraca agilidade nas mudanças e evolução dos serviços. Além de uma forte dependência de hardware específico, as chamadas middleboxes.

O *Service Function Chain* (SFC) é uma arquitetura proposta pela *Internet Engineering Task Force* (IETF) para a definição e instanciação de um conjunto ordenado de funções de serviço que deve ser aplicado a pacotes, frames ou fluxos selecionados com base em um processo de classificação de tráfego e posterior roteamento de tráfego através de funções de serviço. Um exemplo clássico de uma função de serviço abstrata seria um firewall (QUINN; HALPERN, 2014). A ordem de caminho implica na cadeia de serviços pode não ser uma sequência linear, uma vez que a arquitetura permite que SFCs copiem tráfego para mais de um caminho, bem como definem casos em que a ordem de caminho das funções de serviço é flexível.

O compartilhamento de recursos computacionais funciona relativamente bem na computação em nuvem, porém, o compartilhamento de recursos de rede é mais problemático (ARMBRUST et al., 2010). Existem soluções bem estabelecidas para virtualização de sistemas, permitindo o compartilhamento eficiente e justo de recursos de computação e armazenamento, tais como plataformas de virtualização baseadas em Xen (BARHAM et al., 2003) ou VMware (SUGERMAN; VENKITACHALAM; LIM, 2001). Embora não exista atualmente uma abordagem padrão devidamente adotada para a virtualização de rede em um data center, a NFV procura alocar esses recursos de forma mais eficiente. Os principais benefícios de adotar a NFV em um data center incluem:

- Desacoplamento de hardware e software: O planejamento e a manutenção desses elementos podem ser realizados separadamente;
- Flexibilidade nas operações e na concepção das funções de rede: permite a utilização de múltiplas operações de rede simultâneas. As operadoras de rede podem implantar serviços mais rapidamente para seus clientes ao mesmo tempo;
- Escalabilidade de rede: Esta torna-se mais dinâmica com instâncias NFV em diferentes cenários de tráfego e aplicações. Esta realidade é um dos principais problemas a serem enfrentados pelos operadores.

Neste ponto, deve-se notar que o desacoplamento de software e hardware não significa virtualização obrigatória de recursos de todos os elementos da rede. Os operadores de rede ainda podem desenvolver ou comprar software e executá-lo em estruturas de hardware de commodities atuais. No entanto, o ganho de executar o software em módulos virtualizados é o ponto crítico que leva a melhores resultados de desempenho e lucros de OPEX e CAPEX. Finalmente, são sugeridos cenários híbridos nos quais funções executadas em recursos virtualizados podem operar e coexistir com funções executadas em recursos físicos padrão até que uma transição completa para virtualização ocorra. No entanto, técnicas semelhantes de virtualização de redes já foram estudadas, tais como: VLAN, VPN e *Virtual Network Embedding* (VNE). A virtualização de redes levanta novos desafios a serem enfrentados, como a instanciação, operação e gestão de redes virtuais. No entanto, o problema da escalonamento de VNFs é uma das questões que precisam ser melhor compreendidas. Essa questão está diretamente relacionada à alocação de recursos da infraestrutura física para as redes virtuais solicitadas.

2.2 ESCALONAMENTO INTELIGENTE

Um data center inteligente mantém a estabilidade do sistema em condições de rede inesperadas, como tráfego, software e hardware, e se adapta rapidamente a essas condições. Sistemas e métodos autônomos inteligentes dependem dos parâmetros da rede. Além disso, esses sistemas inteligentes podem lidar facilmente com questões como incerteza, dinamismo e heterogeneidade. Vamos dar uma breve visão geral das técnicas que vamos aplicar ao escalonamento de VNFs. Estas técnicas incluem computação autônoma com aprendizagem profunda. Em outras palavras, a principal diferença entre o software tradicional e as abordagens de auto-aprendizagem é que, nos modelos de auto-aprendizagem, o resultado da execução depende da fase de treinamento do software. Assim, o mesmo algoritmo pode produzir resultados diferentes, dependendo dos dados de formação utilizados. Neste contexto, os métodos de aprendizagem de máquinas se estabeleceram como concorrentes sérios da estatística clássica, no início dos anos 80, tornou-se claro que os modelos lineares não são adequados para a maioria das aplicações reais (GOOIJER; HYNDMAN, 2006).

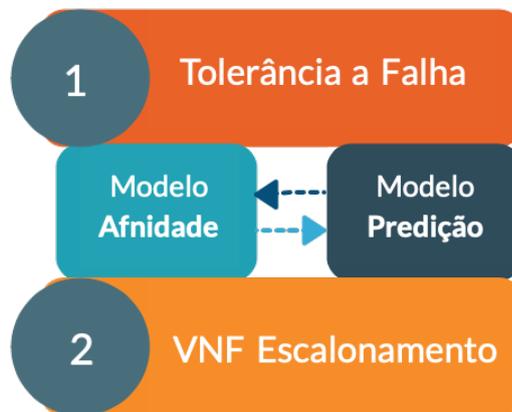
Os modelos de aprendizagem da máquina podem usar o grande volume de dados para criar modelos de previsão mais precisos para os recursos da nuvem (AHMED et al., 2010; PALIT; POPOVIC, 2006). Existem várias abordagens na utilização de aprendizagem de máquinas no contexto de SDN e NFV, por exemplo: classificação de tráfego (AMARAL et al., 2016), balanceador de carga (CHEN-XIAO; YA-BIN, 2016), Detecção de ataque de redes (SWAMI; DAVE; RANGA, 2019), *Quality of Experience* (QoE) (MARTIN et al., 2018), e detecção de fluxo de elefantes (TANG et al., 2017). Trabalho de aprendizagem profundo em posicionamento para alcançar a análise preditiva e a tomada de decisões (GOODFELLOW; BENGIO; COURVILLE, 2016).

Esta tese, aborda a técnica de aprendizagem profunda, porque é resistente a pequenas mudanças e pode generalizar a partir de dados parciais, facilitando a identificação de um objeto parcialmente obstruído, corretamente. O aprendizagem profunda tem a capacidade de avaliar um objeto, aprender adequadamente a informação e adaptar-se a diferentes variantes. (ALOM et al., 2019; LECUN; BENGIO; HINTON, 2015). Aprendizagem profunda proporciona melhorias em relação ao aprendizado de máquina clássico, eliminando a necessidade de feature engineering. Nesta tese, compararmos nossa solução baseado em LSTM com o modelo de *Artificial Neural Networks* (ANN) utilizado no KDN e com o modelo *Gated Recurrent Unit* (GRU).

A escolha de modelos do modelo LSTM, são pelo motivo que é adaptadas às séries temporais, o que corresponde às características dos nossos dados, e porque são muito boas para manter memórias a longo prazo. Ou seja, a previsão por exemplo, em uma sequência de amostras, pode ser influenciada por uma entrada que foi dada em muitas etapas anteriores. A memória pode ou não ser retida pela LSTM, dependendo dos dados. Os seus mecanismos de comportamento fazem a preservação a longo prazo das dependências na LSTM. A LSTM pode armazenar ou libertar memória, passando pelo mecanismo de comutação. Finalmente, será feita uma comparação entre eles para identificar o mais adequado para a nova arquitetura de escalonamento com tolerância a falhas, com base no KDN.

Nossa abordagem considera o trabalho seminal sobre o *knowledge plane*, escrito por David Clark (CLARK et al., 2003), que apresenta um aspecto mais teórico do conhecimento, juntamente com um trabalho recente do Master (MESTRES et al., 2017) que aborda o KDN baseado na aplicação de aprendizagem de máquinas para o controle e operação da rede baseada em SDN e introduz um chamado *knowledge plane* sobre os planos de controle, dados e gerenciamento do SDN. Nossa abordagem estende o paradigma KDN através de uma abordagem integrada de NFV e SDN. Assim, o SDN é mantido como originalmente definido pela KDN para roteamento de tráfego, enquanto o NFV possibilita a virtualização das funções de rede e benefícios da conectividade SDN. A Figura 10 mostra a etapas da nossa proposta para problema de escalonamento de VNFs, com técnicas robustas de tolerância a falhas e diagnóstico de redes integradas.

Figura 10 – Etapas do escalonamento de VNFs



Fonte: O Autor

A nossa abordagem é baseada em aprendizagem profunda e técnicas cognitivas para operar a rede, o que traz muitas vantagens para a rede, como automação (reconhecer-atuar) e recomendação (reconhecer-explicar-sugerir), e tem o potencial de representar uma mudança de paradigma na forma como operamos, otimizamos e resolvemos problemas de rede de dados. As estratégias de construção de sistemas mais confiáveis já eram utilizadas desde a construção dos primeiros computadores (NEUMANN, 1956). Apesar de envolver técnicas e estratégias tão antigas, a tolerância a falhas ainda não é uma preocupação rotineira de projetistas e operadores de rede, sendo sua aplicação quase sempre restrita a sistemas críticos. A técnica de tolerância a falhas são de duas classes disjuntas: mascaramento ou detecção, localização e reconfiguração. Na primeira classe, mascaramento, as falhas não se manifestam como erros porque são mascaradas na fonte. A primeira classe geralmente emprega mais redundância do que a segunda e, por não envolver o tempo gasto em detecção, localização e reconfiguração de tarefas, é preferível para sistemas críticos em tempo real.

A **prevenção de falhas** não é suficiente quando o sistema requer alta confiabilidade ou alta disponibilidade. Nesses casos, o sistema deve ser construído utilizando técnicas de tolerância a falhas. Estas técnicas garantem a operação correta do sistema mesmo em caso de falhas e são todas baseadas em redundância de dispositivos, exigindo componentes adicionais ou algoritmos únicos. Em geral, há diferentes soluções propostas na literatura no contexto das regras de afinidade e antiafinidade para NFV e estão relacionadas a dois aspectos: (i) o desenvolvimento de modelos para descrever regras de afinidade, (ii) desenvolvimento de algoritmos de escalonamento de serviços que podem funcionar sob as restrições dessas regras. Por outro lado, a predição de falhas é outra técnica para aumentar a disponibilidade do sistema. Ela estima a presença, a ocorrência e as consequências das falhas.

A **predição de falhas** torna possível a sua aplicação em várias fases do desenvolvimento de software. Ela prevê a disponibilidade futura da VNF usando métricas e medidas pertencentes a cada domínio. À medida que ocorrem falhas, modelos de disponibilidade podem ser usados para analisá-las. Estas técnicas verificam através de previsões futuras se têm o comportamento desejado em comparação com os requerimentos previamente definidos. A previsão revela, se são necessários mais testes ou medidas/métodos para aumentar a disponibilidade. A previsão não é uma análise de requisitos e não aponta para requisitos que não foram considerados.

2.2.1 Modelos Afinidades

De acordo com a VMware⁵, o conceito de afinidade define as melhores combinações de recursos disponíveis baseado nos requisitos pré-definidos. O grau de coexistência entre as classes de recursos, é caracterizado pelos requisitos que são o grau de compatibilidade entre as máquinas virtuais no mesmo PM ou em diferentes PMs, para evitar que todas as máquinas virtuais falhem ao mesmo tempo no caso de um único PM falhar. Essas relações de afinidade podem ser compartilhamento de memória ou dependência de tráfego. O termo no contexto de NFV surgiu em (JACOBS et al., 2017) considerando requisitos como CPU, memória, armazenamento, largura de banda, vazão, VNF, SFC.

Os serviços de afinidade de data center podem oferecer aos clientes recursos de computação on-demand, ao contrário da infraestrutura tradicional, enquanto novas demandas de recursos permitem que os clientes acessem o que precisam e onde precisam, já que possuem informações sobre o tipo de atividade e recursos requeridos por cada tipo de aplicação, permitindo-lhes pagar pelo que foi utilizado e prevendo necessidades de uso sem recursos excessivos. O uso de afinidade entre aplicações visa contribuir para o escalonamento do ambiente virtual. Isso se baseia nas características do consumo de recursos físicos e de rede entre VNF e PM. Assim, é necessário monitorar e analisar os diversos perfis da requisição de VNF, estabelecidos através do histórico de consumo de recursos, para otimizar o processo de escalonamento de VNFs nos data center para evitar a falha impactada dos recursos computacionais. Uma regra de afinidade coloca um grupo de máquinas virtuais em um host específico para que ele possa facilmente auditar o uso dessas máquinas virtuais. Uma regra antiafinidade define um grupo de máquinas virtuais em diferentes hosts, o que evita que todas as máquinas virtuais falhem ao mesmo tempo se um único host falhar.

As regras podem ser definidas como afinidade, onde são obrigatória o uso de determinados requisitos ou antiafinidade que são requisitos preferenciais. Na caso, as regra obrigatória, não puderem ser cumpridas, as máquinas virtuais adicionadas à regra não serão vinculadas. Regra preferencial, se as regras forem violadas, o cluster ou host ainda estará conectado às máquinas virtuais. Por exemplo, se houver uma regra de antiafinidade entre

⁵ <https://docs.vmware.com/en/vCloud-Director/9.1/com.vmware.vcloud.user.doc/GUID-103BE81A-0762-45C6-915D-19B2B75DEE05.html>

duas máquinas virtuais, mas apenas um host físico estiver disponível, uma regra obrigatória (afinidade forte) não permitirá que ambas as máquinas virtuais sejam vinculadas. Se a regra antiafinidade for preferida (afinidade fraca), ambas as máquinas virtuais poderão ser conectadas. Os modelos de afinidade tornam possível orientar o escalonamento de VNFs.

Essas descrições também incluem parâmetros relacionados ao SLA, como regras de afinidade e antiafinidade e requisitos de latência, a relação de afinidade e o grupo de afinidade das VNFs, respectivamente, e várias regras para agrupar VNFs com base na afinidade. A dependência entre VNFs é definida como uma afinidade de VNFs. Em redes de computadores, também pode ter muitos tipos de afinidade. A afinidade de comunicação é derivada da dependência de comunicação entre máquinas virtuais executando aplicações de comunicação intensiva, enquanto a afinidade de memória é derivada da dependência de memória entre máquinas virtuais quando duas máquinas virtuais se beneficiam da programação para compartilhar memória.

A afinidade de dados é induzida pela dependência entre as máquinas virtuais que executam aplicativos com uso intensivo de dados. Por outro lado, em NFV devem ser amplas e cuidadosamente concebidas para melhorar o desempenho do serviço principal. Os VNF estão encadeados num *Forwarding Graph* (FG) para fornecer um serviço (ou seja, encadeamento de serviços), aumentando substancialmente a complexidade da gestão. Os operadores de rede devem considerar mais do que apenas a atribuição de recursos quando identificam afinidades entre as VNFs. Os critérios de afinidade das VNFs variam muito para diferentes FG de roteamento. Por exemplo, a localização geográfica pode ser tida em conta para minimizar a latência e o atraso de propagação entre VNF em cadeia situadas longe umas das outras, enquanto a perda de pacotes e a utilização da largura de banda podem ser consideradas como um indicador de estrangulamentos nas redes de tráfego elevado. Tudo isso suporta o argumento de que os operadores de rede devem ser capazes de selecionar quais critérios são relevantes ao estabelecer afinidades entre VNFs. Especificamente, para otimizar a automação de escalonamento de VNFs, esses modelos são operados online.

Neste contexto, a afinidade é definida como um indicador da medida em que duas VNFs que funcionam bem, seja quando colocadas na mesma PM ou quando encadeadas à mesma FG, mostrando o gráfico de ligações lógicas, conectando os nós da VNF para descrever o fluxo de tráfego de rede entre eles. Além disso, a afinidade pode ser avaliada com base em dois critérios que são estáticos ou dinâmicas. Os casos estáticos das VNFs não precisam ser avaliados online. Nestes casos, a informação está geralmente disponível em descritores, como os descritos pelo European Telecommunications Standards Institute (ETSI) em (CHIOSI et al., 2012). As regras dinâmicas, por outro lado, estão associadas a casos em que os dados de utilização de recursos em linha são utilizados para medir a afinidade e só devem, por conseguinte, ser considerados quando as VNFs são avaliadas em linha. Em geral, a afinidade é calculada com base em meios harmônicos que combinam

cálculos de baixo nível num único valor numérico que representa a afinidade entre as duas VNFs avaliadas. Ao modelo de previsão são atribuídas regras de afinidade que são compostas pelo tipo de VNF, a classe de escalonamento de VNFs, a afinidade mínima de CPU, a afinidade mínima de memória, a afinidade mínima de armazenamento, os conflitos de afinidade, o mesmo PM e a classe de largura de banda, vazão e FG.

2.2.2 Aprendizado de Máquina

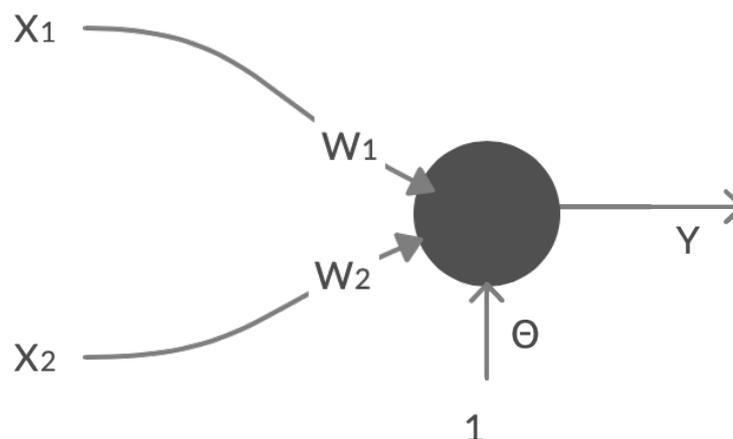
De uma maneira geral, os modelos de aprendizado de máquina, envolvem reconhecimento, diagnóstico, planejamento, previsão, classificação. No software tradicional, a informação necessária para generalizar esse comportamento é codificada permanentemente no programa, enquanto o código dos modelos de autoaprendizagem define a capacidade de aprender, que pode ser usada para generalizar muitos comportamentos. Em outras palavras, a principal diferença entre as abordagens tradicionais de software e autoaprendizagem é que, nos modelos de autoaprendizagem, o resultado da execução depende da fase de treinamento do software. Assim, um mesmo algoritmo pode produzir resultados diferentes dependendo dos dados de treinamento utilizados. Técnicas de aprendizado de máquina são utilizadas em uma ampla gama de aplicações: processamento de imagem, reconhecimento de voz, motores de busca, assistentes pessoais inteligentes, carros autônomos, videogames. No entanto, existem algumas aplicações no domínio das redes. Uma das razões pelas quais podemos dizer algumas formas de autoaprendizagem (MITCHELL, 1998):

- Algumas tarefas, não podem ser bem definidas, exceto através de exemplos. Sistemas capazes de aprender as relações contidas nos casos e de generalizar para outras instâncias do problema são sistemas que têm melhor desempenho;
- Com uma grande quantidade de dados, pode haver relacionamentos desconhecidos que se deseja que sejam explícitos. Os sistemas de aprendizagem podem revelar tais relações;
- Algumas características do ambiente em que o sistema será usado podem ser desconhecidas durante a concepção e implementação do sistema. Um sistema capaz de se adaptar ao ambiente tem uma capacidade potencialmente mais significativa para obter uma melhor eficiência;
- A quantidade de conhecimento disponível para uma determinada tarefa, pode ser demasiado grande para ser explicitamente codificada, onde tornam-se necessários sistemas que aprendem este conhecimento automaticamente;
- Os ambientes mudam ao longo do tempo. Os sistemas que se adaptam à mudança requerem menos esforço de manutenção.

Existem numerosos modelos de autoaprendizagem, que se dividem em quatro categorias: aprendizagem supervisionada, não supervisionada, semi-supervisionada e aprendizagem de reforço. Os principais algoritmos para a aprendizagem supervisionada são Regressão Linear, Árvore de Decisão, *Support Vector Machines* (SVM), Rede Neurais, Bayes Naïve, Algoritmo de Vizinhos *K-Nearest Neighbors* (KNN) e Random forests. Dente os modelos, iremos abordar a explicação sobre ANN, MLP, LSTM.

As **Redes Neurais** são coleções de unidades conectadas chamadas neurônios artificiais. Cada conexão entre neurônios artificiais pode transmitir um sinal de um para outro. Normalmente, o sinal é um número real, e a saída de cada neurônio é calculada por uma função não linear da soma das suas entradas. Essas conexões têm pesos que aumentam ou diminuem a intensidade do sinal e são ajustadas no processo de aprendizagem. Os neurônios são organizados em camadas, onde cada camada pode aplicar um tipo diferente de transformação (função de ativação) às suas entradas. A unidade básica de computação em uma rede neural é o neurônio, muitas vezes chamado de nó ou unidade. Figura 11 mostra que o neurônio recebe a entrada de uma fonte externa e calcula uma saída. Cada entrada tem um peso associado (w), que é atribuído com base na sua importância relativa para outras entradas. A rede aceita entradas numéricas X_1 e X_2 e tem pesos de w_1 e w_2 associados a essas entradas. Além disso, há outra entrada 1 com um peso de θ (chamado Bias) associado a mesma.

Figura 11 – Estrutura de Rede Neural Artificial.

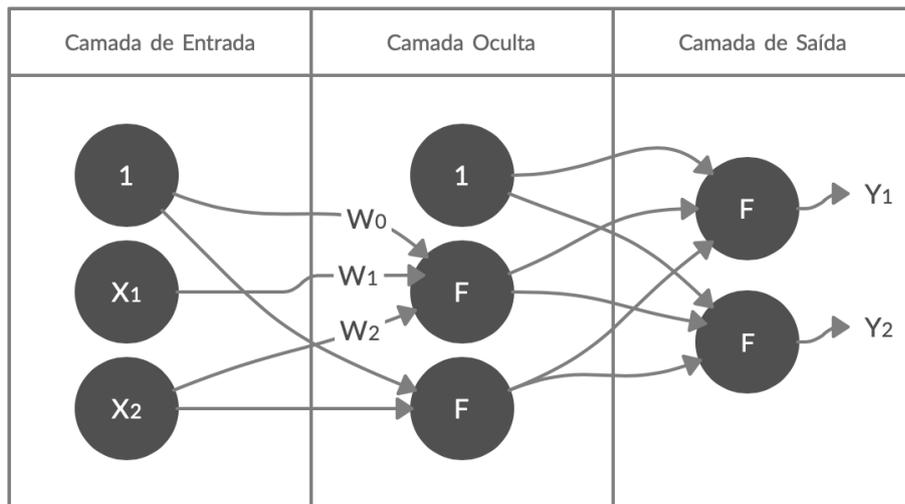


Fonte: O Autor

No entanto, este modelo é incapaz de resolver problemas não-lineares separáveis, o que reduz significativamente a aplicabilidade desse algoritmo (MINSKY; PAPER, 1969). Com isso, foi necessária uma solução para esse problema que resultou no desenvolvimento do algoritmo de treinamento de retropropagação. Um backpropagation contém uma ou mais

camadas ocultas que são conhecida pela presença enquanto a rede neural é treinada, além de uma camada de entrada que pode ser os pacotes de um fluxo de tráfego em uma série temporal e uma camada de saída que fornece principalmente uma previsão da entrada que você inseriu em sua rede. A Figura 12, mostra um perceptron multicamadas com uma única camada oculta. Note que todas as conexões têm pesos associados a elas, mas apenas três pesos (w_0 , w_1 , w_2).

Figura 12 – Modelo Multilayer Perceptron.



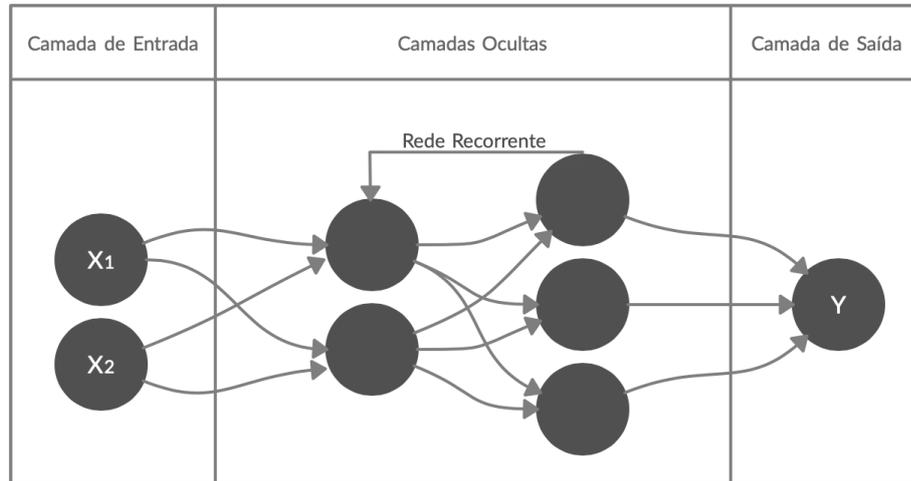
Fonte: O Autor

A camada de entrada tem três nós. Os outros dois nós tomam X_1 e X_2 como entradas externas, que são valores numéricos, dependendo do conjunto de dados de entrada. Nenhum cálculo é executado na camada de entrada, portanto, as saídas do nó na camada de entrada são 1, X_1 , e X_2 , respectivamente, que são entradas na camada oculta. A camada oculta também tem três nós com o nó Bias, tendo uma saída de 1. A saída dos outros dois nós na camada oculta dependem das saídas da camada de entrada (1, X_1 , X_2), bem como a saída dos pesos associados às conexões (bordas). A camada de saída tem dois nós que recebem entradas da camada oculta e realizam cálculos semelhantes. Os valores calculados (Y_1 e Y_2) atuam como saídas do *Multi Layer Perceptron* (MLP). Dado um conjunto de características $X = (x_1, x_2)$ e um alvo Y , um MLP pode aprender a relação entre recursos e alvo para classificação e regressão. As mudanças podem ser melhorias em sistemas já em execução ou a base para novos sistemas. Uma definição mais precisa e formal dada por Mitchell (MITCHELL, 1998).

Um subconjunto de modelos de aprendizado de máquina, chamados modelos de **aprendizagem profunda**. Nessa tese, consideramos o modelo LSTM. Para compreender o modelo LSTM, é essencial o entendimento básico de RNN. Assim, o estado anterior da rede RNN influencia a saída. Para tal, a rede também ter um “senso de tempo”. Esse efeito

é conseguido com um loop na saída da camada para sua entrada, como podemos ver na Figura 13. Isso mostra que os RNNs podem conectar informações anteriores à tarefa atual. O uso das informações anteriores pode ajudá-lo a entender as informações atuais.

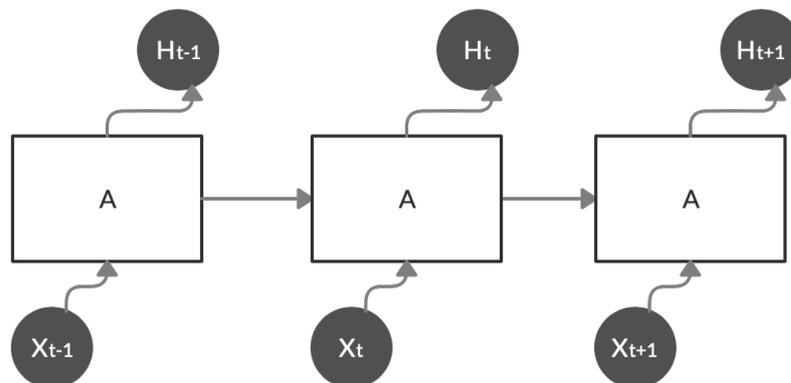
Figura 13 – Representação RNN



Fonte: O Autor

A Figura 14 mostra que cada célula A recebe uma entrada X e a saída da célula anterior. LSTM é um tipo particular de RNN, capaz de armazenar informação a longo prazo. O LSTM fará o mesmo que um RNN, mas com uma memória maior (HOCHREITER; SCHMIDHUBER, 1997).

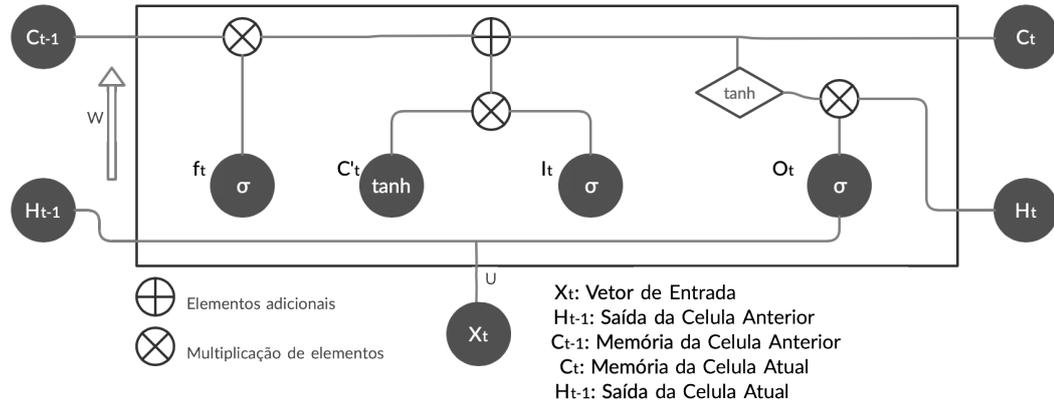
Figura 14 – Representação das células de uma RNN



Fonte: O Autor

Na Figura 15 observamos que a célula (A) contém detalhes dos seguintes componentes: Porta de esquecimento f . Porta do candidato C' , Entrada I , Saída O , Status H , Estado da memória C . U são os vetores de peso para $f_t, C_t, I_t, O_t, \sigma$ (sgimoid), \tanh (hyperbolic tangent) são funções de ativação que possuem valores de 0 até 1 e -1 até 1, respectivamente.

Figura 15 – Célula LSTM em Detalhes



Fonte: O Autor

LSTM usa três portas que são: entrada, esquecimento e saída que são definidas como i_t , f_t e O_t , respectivamente. A entrada X_t e o estado oculto anterior $H_{(t-1)}$ são utilizados pelas portas nas Equações 2.1, 2.2 e 2.3:

$$f_t = \sigma(X_t * U_f + H_{t-1} * W_f) \quad (2.1)$$

$$I_t = \sigma(X_t * U_i + H_{t-1} * W_i) \quad (2.2)$$

$$f_t = \sigma(X_o * U_f + H_{t-1} * W_o) \quad (2.3)$$

Note que as portas são dependentes de H e X. Dito isto, espera-se que o novo estado da memória (saída da célula) seja calculado, como mostrado na Equação 2.4.

$$C = (f * C + I * C') \quad (2.4)$$

A equação, esquece algo do estado a cima, além disso, adiciona alguma entrada, dessa maneira mais alguma informação é necessária para determinar a entrada de célula. Além do I_t , tem que calcular o que poderia entrar no estado da célula (memória). Esta informação do candidato está na Equação 2.5:

$$C' = \tanh(X_t * U_c + H_{t-1} * W_c) \quad (2.5)$$

Finalmente, na Equação 2.6 o novo estado é calculado, este estado oculto é agora usado para calcular o que esquecer, a entrada e a saída da célula no passo seguinte.

$$H_t = O_t * \tanh(C_t) \quad (2.6)$$

Os RNNs são modelos supervisionados de autoaprendizagem, construídos por neurônios artificiais com um ou mais loops de realimentação (HAYKIN; NETWORK, 2004). RNNs são capazes de guarda/armazenar memória, visto que, suas saídas reais dependem de cálculos anteriores. No entanto, o RNN retorna poucos passos devido ao problema do vanishing gradient.

O LTSM foi especialmente projetado para superar esses problemas, introduzindo novas portas que permitem um melhor controle sobre o fluxo, permitindo uma melhor preservação das dependências de longo alcance (HOCHREITER; SCHMIDHUBER, 1997). O termo “memória de longo prazo” surgiu para melhorar as redes neurais simples e recorrentes, porque, tem memória de longo prazo na forma de pesos. Os pesos são alterados lentamente ao longo do treinamento, codificando o conhecimento geral sobre os dados, e é composto de uma memória de curto prazo na forma de atividades efêmeras, que passam de cada nó para nós sucessivos. Os RNN são redes neurais com loops, permitindo que a informação persista e podem ser consideradas cópias múltiplas da mesma rede, cada uma passando sobre uma mensagem a um sucessor. Na Figura 13 ilustrada anteriormente, temos a estrutura de um RNN, onde cada célula “A” é o trecho de uma rede neural, x_t é a entrada da célula, e h_t é a saída da célula.

2.3 MÓDULO DIAGNÓSTICO

Esta seção mostra um resumo dos conceitos necessários para entender esta pesquisa. Os modelos de desempenho são efetivamente usados para combinar questões de desempenho e disponibilidade. A avaliação de desempenho pode especificar a quantidade de trabalho que será realizada em um determinado intervalo, enquanto falhas e reparos afetam o sistema. A avaliação de desempenho é uma atividade essencial em ambientes NFV aplicados em data center. Ela permite que analistas e usuários avaliem os efeitos gerados por diferentes estratégias de gerenciamento de recursos. Além disso, é possível estimar os custos e benefícios correspondentes para cada ambiente disponível. A avaliação de desempenho pode ser realizada através de medições no sistema real ou através de modelos que representam as características e o comportamento do sistema. As três técnicas de avaliação de desempenho são modelagem analítica, simulação e medição (FERNANDES, 2017).

A utilização de medições, são possíveis se existir algo semelhante ao sistema proposto, como o desenho de uma versão melhorada de um produto. Se for um conceito novo, a modelagem analítica e a simulação são as únicas técnicas a serem escolhidas. A modelagem analítica e de simulação pode ser usada em situações onde a medição não é possível. No entanto, em geral, seria mais concludente se a modelagem analítica ou simulação fosse baseada na análise anterior. As técnicas baseadas na medição requerem a construção de um ambiente real e envolvem o monitoramento do sistema enquanto ele está sob a ação de uma carga de trabalho. Antes de aplicar a carga de trabalho ao sistema, deve-se ter um estudo primário da carga a ser utilizada. A escolha da carga de trabalho é tão importante

quanto a definição da estratégia de medição a ser seguida, pois é a partir dela que se deve escolher ferramentas e estratégias de medição (LILJA, 2005).

Por outro lado, os modelos de simulação podem ser utilizados tanto na avaliação de desempenho de sistemas quanto na validação de modelos analíticos. Ao contrário das medições, as simulações são baseadas em modelos de sistemas abstratos, portanto, não requerem que o sistema seja totalmente implementado para ser aplicada. O modelo de desempenho representa o funcionamento de todos os possíveis conjuntos de componentes operacionais. O modelo de desempenho define um nível de desempenho ou taxa de retorno para cada disponibilidade no processo de estado do sistema. O nível de desempenho indica a qualidade de funcionamento do sistema por unidade de tempo. Os níveis incluem a largura de banda de uma comunicação, atraso das fração das tarefas pelo canal de rede em tempo de execução.

2.3.1 Dependabilidade

A dependabilidade de um sistema deve ser entendida como a capacidade de evitar falhas de serviço que são mais frequentes e mais sérias do que aceitáveis. O conceito de dependabilidade engloba diferentes atributos, tais como capacidade de manutenção, confiabilidade e disponibilidade (AVIZIENIS et al., 2004; LAPRIE, 1992). Em particular, no âmbito deste trabalho, é considerada a disponibilidade. A disponibilidade é uma medida estatística que descreve a capacidade de um dispositivo para executar funções sob um conjunto específico de condições e por um tempo específico. A disponibilidade é essencialmente uma medida da probabilidade dessas falhas inesperadas, que são tipicamente aleatórias. Por exemplo, para calcular a disponibilidade (A) de um dispositivo específico, é necessário recorrer ao uptime e downtime ou ao *Time To Failure* (TTF) e *Time To Repair* (TTR) (AVIZIENIS et al., 2004).

Neste contexto, como é o caso típico de que apenas valores médios estão disponíveis, as métricas adotadas são o MTTF é o tempo esperado para a primeira falha do dispositivo em análise, considerando que no tempo zero (referência de tempo inicial), o dispositivo estava em perfeitas condições. Devido à dificuldade de modelagem analítica, este parâmetro é normalmente medido ou estimado. Analiticamente, o MTTF é definido como:

$$MTTF = \int_0^{\infty} R(t)dt \quad (2.7)$$

Assumindo que a distribuição correspondente ao dispositivo é exponencial, tem-se:

$$MTTF = \int_0^{\infty} R(t)dt = \int_0^{\infty} \exp^{-\lambda t} dt = \frac{1}{\lambda} \quad (2.8)$$

onde λ é a taxa de falha do dispositivo. Portanto, uma relação direta é estabelecida entre o MTTF e a taxa de falha λ ; E o MTTR é o tempo médio necessário para reparar um dispositivo defeituoso. Normalmente este parâmetro, tal como o MTTF, é medido ou

estimado. Considerando que $G(t)$, a função de manutenção, representa a probabilidade de que o dispositivo, uma vez falhado, será reparado no intervalo de tempo de $[0, t]$, e que a função de reparo apresenta uma distribuição exponencial, que se tem:

$$MTTR = \int_0^{\infty} (1 - G(t))dt \quad (2.9)$$

onde $G(t) = 1 - \exp^{-\mu t}$ and μ é a taxa de reparação do dispositivo

$$MTTR = \int_0^{\infty} (1 - G(t))dt = \int_0^{\infty} \exp^{-\mu t} dt = \frac{1}{\mu} \quad (2.10)$$

Essas métricas geralmente usam modelos combinatórios, como o RBD e o Fault Trees, ou modelos estocásticos baseados nos estados, como o Markov Chains (AVIZIENIS et al., 2004; LAPRIE, 1992). No entanto, tais abordagens baseado no estado sofrem com o problema da explosão do espaço de estado (GODEFROID et al., 1996). Para evitar este problema, nesta tese, é proposto uma nova ferramenta, chamado diagnóstico (ver Capítulo 4) utiliza os proeminentes modelos analíticos para permitir a análise numérica. Com base neles, é possível criar uma ferramenta de diagnóstico que permite analisar grandes conjuntos de dados.

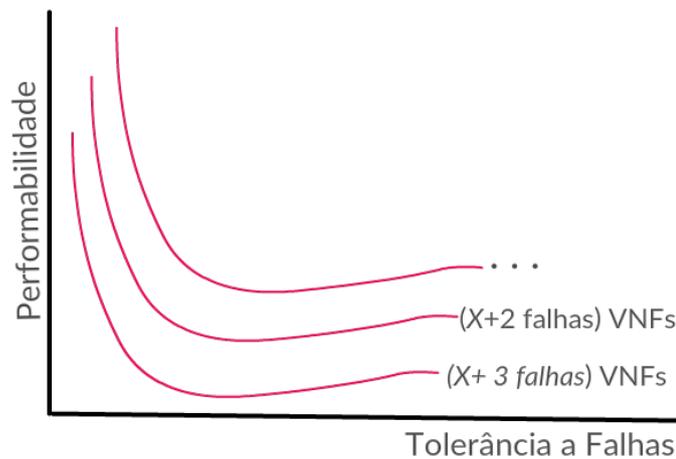
2.3.2 Performabilidade

A performabilidade permite combinar desempenho e disponibilidade. Esta visão conjunta permite uma compreensão e análise suficientes dos cenários de degradação do desempenho. Por exemplo, se um host sobrecarregado sofre degradação de desempenho, isso pode ser um efeito adverso das migrações de VM ou do número de desligamentos, falhas de VM, portanto, pode explicar tal degradação, uma vez que a condição do evento indicaria que a rede está funcionando corretamente (com falhas) ao longo do tempo.

Assim, a variável desempenho não pode explicar tal degradação já que a condição do evento indicaria que a rede está funcionando corretamente (sem falhas) ao longo do tempo. Esta degradação também não é explicada pela variável de confiabilidade, pois ela se refere à qualidade do serviço apenas na medida em que ela é adequada (sem falhas) ao longo do tempo (MEYER, 1992). Desta forma, a modelagem independente é susceptível de proporcionar uma avaliação incompleta ou imprecisa, não sendo aceitável no contexto de sistemas tolerantes a falhas, que visam a prestação contínua de serviços, mesmo com perda de desempenho. Um estudo de performabilidade, que permite combinar as métricas de desempenho e dependabilidade, como a disponibilidade utilizada nesta tese. Figura 16 ilustra uma situação em que a ocorrência de falha, representa a redução de desempenho ao longo do tempo, onde X representa todo o sistema com falhas de VNFs.

Em geral, a modelagem de desempenho e disponibilidade são realizadas separadamente. A modelagem individual não é possível para verificar se a falha de um componente ou subsistema afeta todo o sistema. No entanto, alguns eventos, como o impacto da mudança

Figura 16 – Avaliação de Performabilidade



Fonte: O Autor

de temperatura devido à sobrecarga da CPU, não podem ser detectados nem pelos modelos de desempenho, nem pelos modelos de disponibilidade. Por outro lado, a métrica de performabilidade permite a análise do impacto individual de cada componente no sistema (MEYER, 1992). Por exemplo, suponha que existem falhas na estrutura que reduzem a qualidade do serviço sem causar falhas, ou seja, o desempenho da rede é degradável. A variável desempenho não pode explicar tal degradação, já que a condição do evento indicaria que a rede está funcionando corretamente (sem falhas) ao longo do tempo.

Esta degradação também não é explicada pela variável de disponibilidade, pois se refere à qualidade do serviço apenas na medida em que ela é adequada (sem falhas) ao longo do tempo (MEYER, 1992). Desta forma, a modelagem independente é susceptível de proporcionar uma avaliação incompleta ou imprecisa, não sendo aceitável no contexto de sistemas tolerantes a falhas, que visam a prestação contínua de serviços, mesmo com perda de desempenho. Assim, a métrica de performabilidade é importante porque surge de uma lacuna entre confiabilidade, disponibilidade e desempenho. Duas razões para esta lacuna: (i) a disponibilidade tem apenas uma visão binária do funcionamento do sistema, e (ii) as métricas de desempenho (geralmente) assumem que o sistema está no estado à prova de falhas. Assim, a performabilidade avalia a capacidade de desempenho na presença de falhas, não se sobrepõe às métricas típicas de disponibilidade e desempenho. Embora compartilhe conceitos, ela pode ser considerada como uma classe adicional de métricas para caracterizar a confiabilidade/disponibilidade de um sistema.

2.3.3 Análise Exploratória de Dados

Permitem a análise de dados para uma lógica inicial de modelagem explicativa e preditiva, resumindo os dados de maneira numérica do estado atual ou futuro do data center.

Ao resumir e contabilizar dados, o EDA pode prontamente fornecer informações úteis, encontrar padrões e descobrir relacionamentos gerais que podem orientar análises adicionais e alavancar os resultados. É uma abordagem clássica e estritamente orientada por dados. EDA faz uso de técnicas de investigação de dados para buscar informações e relacionamentos interessantes, a partir de uma abordagem ativamente incisiva, com ênfase real na descoberta do inesperado.

A EDA isola padrões e características dos dados, e os revela vigorosamente ao analista, sem a necessidade de conhecimento prévio ou hipóteses pré-especificadas, ou seja, sem exigir que perguntas antecipadas e bem definidas sejam direcionadas aos dados, onde a EDA não pode levar a conclusões definitivas, mas é um primeiro passo essencial na compreensão dos dados. Os métodos básicos da EDA tentam tornar os dados mais acessíveis e mais eficazmente geridos pelo utilizador, sejam eles estatísticos ou não estatísticos. Na prática, para atingir os seus objetivos, a EDA associa métodos quantitativos e qualitativos da abordagem estatística clássica à análise gráfica.

2.3.4 Medições de Importâncias

A complexidade dos modernos data center em computação virtualizada, se torna um desafio para planejar e replanejar. No entanto, um planejamento consciente do desempenho com políticas de gestão de recursos precisa ser aplicado (MENON et al., 2005; MEI et al., 2013; WANG; NG, 2010; BELOGLAZOV; ABAWAJY; BUYYA, 2012; KANT, 2009). As técnicas de modelagem e replanejamento de desempenho fornecem ferramentas poderosas para analisar o desempenho do data center. No entanto, dada a grande variedade de abordagens de virtualização de rede, não há uma abordagem comum para a modelagem e avaliação do desempenho das redes virtualizadas. Uma análise de disponibilidade é um componente crucial no projeto e ajuste de sistemas no data center. Entretanto, a análise da disponibilidade diz somente a parte da história. Questões de acompanhamento como:

- Como uma mudança em um dispositivo afeta todo o sistema?
- Dados os recursos limitados, como um orçamento fixo, como a disponibilidade de todo o sistema podem ser melhorados?
- Qual a configuração da arquitetura de redes que devemos escolher?

Estas e outras questões semelhantes são melhor respondidas usando os resultados da análise de importância e criticidade para ajudar a identificar quais os componentes que mais contribuem para a disponibilidade do sistema. Portanto, eles serão bons candidatos para os esforços que levam à melhoria da disponibilidade do sistema. Uma análise de significância ajuda a separar os efeitos de vários fatores que podem afetar o desempenho. Eles são amplamente utilizados em uma variedade de aplicações de análise de segurança probabilística *off-line* e *on-line* na tomada de decisão para fins preventivos e corretivos.

Além disso, bem como para classificar os componentes de acordo com a sua contribuição para o risco global, medidas de importância são definidas principalmente para o caso em que o modelo de suporte é uma árvore de falhas coerente, e eventos básicos nesta árvore de falhas representam falhas de componentes.

As propriedades qualitativas para verificação de modelos fornecem uma resposta clara de “sim” ou “não”. Não é apropriado em todos os casos. Assumindo que queremos verificar o Jitter de um data center inteligente, neste caso, não esperamos um simples “sim” ou “não”, mas uma medida quantitativa descrevendo a taxa de transferência do sistema. Incluir propriedades quantitativas na verificação de modelos nos permite verificar uma variedade de medidas de desempenho, disponibilidade e performabilidade. Especialmente para a análise de NFV em um data center, as medições quantitativas são de extrema importância. Perguntas típicas para responder são:

- Qual é a probabilidade de falha do sistema em cinco anos?
- Qual é o tempo esperado para a primeira falha de rede?
- Qual é a disponibilidade de SFC em um data center de longo prazo?

Além disso, alocar ou adicionar um recurso em um data center confiável envolve muitos compromissos:

- O nível de redundância é alto o suficiente para estar disponível em 99,999% do tempo?
- Como a manutenção pode ser planejada para que os custos operacionais sejam minimizados?

Essas questões não apenas exigem métricas quantitativas adicionais para responder, mas também estão sujeitas aos seguintes atributos: (i) tempo estocástico para modelar velocidade ou atraso; (ii) probabilidades discretas para fenômenos de modelos aleatórios; (iii) nenhum determinismo para escolhas de modelos; (iv) recompensas ou custos para medir a qualidade das soluções. Nessa tese, consideramos duas medidas de importância:

- AI são ferramentas matemáticas de análise de medições de importância incluindo o tempo médio até a falha, e o tempo de recuperação dos dispositivos (SOUZA; SANTOS; FERNANDES, 2018).
- CAI é uma extensão natural da métrica Birnbaum (BIRNBAUM, 1968). A métrica CAI inclui a indisponibilidade de dispositivos, enquanto que a medida Birnbaum não inclui (KUO; ZUO, 2003). Um dispositivo menos confiável recebe mais atenção. As medidas críticas são úteis para priorizar decisões de disponibilidade, identificar a classificação dos nós de rede e enlaces fracos. A classificação crítica dos ativos é usada para melhorar as atividades de manutenção.

2.4 CONSIDERAÇÕES FINAIS

Este capítulo apresentou uma introdução à escalonamento de VNFs, considerando aspectos relacionados à tolerância a falhas. Em seguida, apresentou conceitos básicos sobre desempenho e disponibilidade. Em seguida, ele apresentou os principais conceitos de autoaprendizagem. Finalmente, este capítulo mostrou que os conceitos sobre data center e redes virtualizadas são fundamentais para a compreensão do ambiente avaliado pela solução integrada proposta neste trabalho, que será mostrada nos próximos capítulos. Os aspectos avaliados no ambiente de nuvem computacional são compreendidos através dos conceitos de avaliação de desempenho e avaliação de disponibilidade. A LTSM permite aplicar o paradigma KDN no escalonamento de VNFs.

3 ESTADO DA ARTE

Este capítulo discute o trabalho relacionado com a descrição do problema desta proposta de tese. Os trabalhos podem ser classificados em três grupos distintos, cada um dos quais aborda um problema distinto.

3.1 PROBLEMA CONVENCIONAL

O escalonamento de redes virtuais em um substrato físico, consiste em determinar o escalonamento ideal de recursos virtuais em substratos de rede sob demanda. (ANDERSEN, 2002) foi um dos primeiros a discutir ideias sobre VNE. (RAHMAN; AIB; BOUTABA, 2010) foi o primeiro a considerar estratégias de sobrevivência na virtualização de redes. Houve também várias abordagens subsequentes de alocação de recursos para redes virtuais (FAN; AMMAR, 2006; ZHU; AMMAR, 2006; LU; TURNER, 2006). O primeiro trabalho a considerar a análise de dependabilidade em ambientes de redes virtualizadas foi (FERNANDES et al., 2012). Desde então, um grande número de algoritmos VNE tem sido proposto na literatura. Além disso, algumas revisões da literatura são fornecidas por (BELBEKKOUCHE; HASAN; KARMOUCH, 2012; FISCHER et al., 2013). Formulações como esta podem ser encontradas em muitos artigos sobre VNE (ver (YU et al., 2010; CHENG et al., 2011; YU et al., 2008; BELBEKKOUCHE; HASAN; KARMOUCH, 2012; CHOWDHURY; RAHMAN; BOUTABA, 2009)). No entanto, os autores deixam vários conceitos desconectados.

O problema encontrado em VNE no data center é buscar o mapeamento eficiente de redes virtuais nos substratos da rede física. O mapeamento consiste em determinar uma alocação de *Virtual Network* (VN), enlace de rede virtual e enlace de substrato, garantindo a maior quantidade de resposta às requisições. No entanto, o problema da VNE antecede o problema do provisionamento de SFC. Em particular, o problema do mapeamento do nó virtual na VNE é semelhante ao problema de escalonamento de VNFs no provisionamento de SFC. No entanto, estes dois problemas diferem em muitos aspectos. Por exemplo, as requisições de SFC fornecem a origem e o destino, ao passo que as requisições de VN não o fazem. Duas requisições de VNFs da mesma no SFC podem ser implantadas no mesmo nó de substrato, enquanto dois nós virtuais da mesma requisições de VN não podem ser esquematizados para o mesmo nó de substrato. Além disso, a direção do fluxo em SFC é unidirecional, enquanto VN é bidirecional. Portanto, o problema VNE e o problema de escalonamento SFC têm um baixo nível de similaridade.

É importante mencionar os diferentes problemas de mapeamento e escalonamento. Para isto, o mapeamento de VNE é diferente de escalonamento de SFC. Em VNE, os nós virtualizados no mapeamento precisam de alocar recursos de rede. Por outro lado, o escalonamento de SFC aloca recursos de processamento, ou seja, servidores, e não recursos

de rede, além disso, a demanda não é uma requisição para conectar uma rede multiponto a outra rede multiponto, mas a demanda por fluxo ponto-a-ponto. Outro aspecto do NFV, como o comportamento de latência de roteamento, mudanças na taxa de bits de entrada/saída e encadeamento, não é abordado na VNE.

Os problemas envolvendo escalonamento de VNFs são complexos e para derivar um algoritmo de escalonamento realista, é necessário considerar vários aspectos, tais como heterogeneidade dos recursos do substrato, topologia genérica da rede virtual, política de alocação, tempo, buffer, tamanho da fila, e atraso. O escalonamento de VNFs se concentra em como alocar as VNF para a infraestrutura de rede adequadamente, com base em serviços de rede solicitados. Além disso, a otimização dos recursos deve ser realizada com respeito a objetivos específicos, por exemplo, a maximização dos recursos remanescentes da rede, a minimização do consumo de energia do data center. O SFC é composto do conjunto ordenado de VNFs que a rede de substrato executa para atender aos atributos de serviço em questões de desempenho. No entanto, mesmo com conhecimento prévio de todas as requisições de VNF, a maior parte do trabalho realizado nesta área tem se concentrado no desenvolvimento de algoritmos heurísticos ou meta-heurísticos e no uso de redes com complexidade mínima, quando resolvendo modelos de ILP. Para mais informações sobre heurística, o leitor deve consultar (HOOKER, 2012).

Nesse contexto, a aplicação de estratégias de escalonamento de VNFs em um data center é uma lacuna de pesquisa. A mudança na SFC é dinâmica e as topologias do data center são tipicamente homogêneas. Os servidores têm as mesmas capacidades computacionais, de armazenamento e de comunicação. Para este fim, os mecanismos podem ser projetados para melhorar o desempenho do data center. O desempenho da rede de substrato - trabalho será afetado pelo comportamento das diferentes funções de composição, e a ordem em que as funções são processadas. É fundamental alcançar um SFC eficiente em relação aos objetivos do projeto de rede. As definições dos problemas da VNF diferem quanto às restrições impostas. A solução para o pedido deve garantir que o maior número possível de VNFs seja alocado e deve ser feito em tempo hábil. Em um ambiente real, as requisições de rede virtual não são conhecidas previamente, aumentando ainda mais a complexidade do problema, porém, mesmo que todos as requisições sejam conhecidas anteriormente, o problema é NP-difícil (AMALDI et al., 2016; HOUIDI; LOUATI; ZEGHLACHE, 2008) e, portanto, intratável. Observamos que a maioria dos pesquisadores adotou uma abordagem relacionada a algoritmos exatos ou heurísticos para comprovar suas hipóteses. No entanto, não há uma lista comum de quais outros poderiam ser utilizados na pesquisa de escalonamento de VNFs, assim realizamos uma revisão sistemática da literatura para obter orientações importantes nesse sentido (SOUZA; DIAS; FERNANDES, 2020).

3.2 PROBLEMA DE ESCALONAMENTO DE VNFS

A fim de compreender mais profundamente o problema de pesquisa, foi realizada uma revisão sistemática da literatura sobre NFV aplicados no data center, a fim de obter um conhecimento profundo e amplo. Um estudo de revisão sistemática da literatura é um tipo de pesquisa que possui natureza baseada em evidências, aplicada de forma a fornecer uma visão geral de uma área de pesquisa, caracterizando-a (HERRERA; BOTERO, 2016). Antes de apresentar qualquer estratégia de escalonamento que pudesse resolver o objetivo perseguido por esta tese, nos realizamos uma revisão sistemática da literatura (SOUZA; DIAS; FERNANDES, 2020), com o objetivo de identificar problemas, técnicas, métodos, métricas e ferramentas utilizadas na alocação de recursos de NFV aos centros de dados. Em linhas gerais, usamos a seguinte string de busca: (“DC” **OR** “Data Center”) **AND** (“NFV” **OR** “Network Function Virtualization”).

Seguindo o processo descrito acima, os resultados obtidos são os seguintes: o processo de pesquisa automática (Passo 1) retornou 1408 estudos primários. A partir deles, identificamos 733 duplicados e os removemos. Em seguida, os pesquisadores revisaram os demais estudos com base em seu título e resumo (Passo 2), resultando em um total de 275 estudos restantes. Finalmente, após a Etapa 3, o número total de estudos resultou em 65 que abordam problemas de alocação de recursos de NFV em um data center dividido em subproblemas de alocação de recursos, dentre esses, 7 artigos estão relacionados ao problema de escalonamento de VNFs. Na Tabela 1 resumimos as implicações para a prática, identificando tendências de pesquisa, questões em aberto e áreas de melhoria. Dado o estado atual da pesquisa escalonamento de VNFs, acreditamos que existem poucos estudos com experimentos usando aplicações reais. A marcação “√” indica que o trabalho relacionado atende ao critério estabelecido, enquanto a ausência de marcação indica que não foi atendido o critério. As Siglas na tabela, foram adicionadas, por motivos de espaço, onde significam: P: performabilidade, D: desempenho; CAI: Disponibilidade Crítica, AI: Disponibilidade Importante, KDN: Knowledge Define Networking, EAD: Análise Exploratória de Dados, TF: Tolerância a Falha.

Tabela 1 – Estado da Arte do Problema de Escalonamento

Artigos	Objetivos				Método de Interesse			Métricas de Interesse				
	VNF	Escalonamento	KDN	TF	Predição	Diagnostico	Afinidade	D	P	CAI	AI	EAD
ID01	√	√						√				
ID02	√	√						√				
ID03	√	√						√				
ID04	√	√						√				
ID05	√	√						√				
ID06	√	√						√				
ID07	√	√						√				
Tese	√	√	√	√	√	√	√	√	√	√	√	√

Fonte: O Autor

Objetivos: Investigamos as diferentes lacunas relacionadas com o problema de escalonamento de VNFs e as soluções que podem ajudar os operadores de rede e provedores de serviço no gerenciamento NFV em um data center. O primeiro trabalho (RIERA et al., 2014) tratava da definição de uma formalização para o problema de escalonamento de VNFs. No ano seguinte Mijumbi et al. (2015) (ID05) foram propostas soluções com heurística para otimizar o mapeamento e o escalonamento de VNFs para um determinado serviço numa rede física. Eles focaram apenas na melhoria das funções de escalonamento e mapeamento virtual online, e foram semelhantes à formulação do problema da VNE, pois visa maximizar a taxa de aceitação de requisições virtuais, minimizar os custos de integração ou maximizar as receitas globais. Poucos trabalhos foram encontrados com problema de escalonamento de VNFs (QU; ASSI; SHABAN, 2016) (ID06), (RODRIGUEZ; GUILLEMIN, 2016) (ID01).

Os primeiros foram: (ASSI et al., 2018; CHOWDHURY et al., 2018; CAO; ZHANG; YANG, 2019; YANG et al., 2019), mas todos eles enfrentaram as restrições de uma abordagem de tolerância a falhas considerando prevenção e predição, bem como automação do escalonamento através de uma abordagem como a KDN. Como os benefícios da utilização do aprendizado de máquina no contexto do KDN, ele se apresenta como uma nova tendência de pesquisa que inicialmente surgiu com (CLARK et al., 2003), e mas recentemente com (MESTRES et al., 2017). O objetivo deste campo é conseguir uma utilização inteligente e autônoma de recursos limitados do data center até o aprendizado de máquina. O uso do aprendizado de máquina tem sido aplicado em diversas áreas da computação, avaliando a classificação e predição de problemas. A maioria deles tem se dedicado a avaliar quais problemas de rede de computadores no aspecto de cognição dos modelos de aprendizado de máquina. O escalonamento de VNFs apresentou a necessidade de aplicar o aprendizado de máquina que exige alcançar níveis de qualidade de disponibilidade, desempenho de tolerante a falha. Por esta razão, o presente trabalho centra-se em uma nova abordagem para o escalonamento de VNFs no data center, o conhecimento aplicado com métodos de tolerância a falhas.

Método de Interesse: Refere-se aos métodos utilizados para o problema de escalonamento de VNFs no data center. Os métodos de escalonamento investigados estão relacionados à otimização com ILP, *Mixed Integer Linear Programming* (MILP), *Genetic Algorithm* (GA) na otimização do processo de escalonamento. Dentre os trabalhos que aplicaram o aprendizado de máquina considerando afinidade e predição e não consideraram a aplicação de modelagem analítica como parte da estratégia de solução, ou seja, SPN, RBD ou *Markov Chain* (MC) não foram incluídos.

Portanto, os trabalhos encontrados na revisão sistemática da literatura não resolvem o problema, considerando o escalonamento de VNFs para data center, nem utilizam um método de modelagem analítica, aprendizado de máquina para resolver o problema. No

entanto, esta comparação é essencial para medir o quão negligenciados são esses métodos durante o processo de escalonamento. Nosso trabalho, é o primeiro, ao melhor de nosso conhecimento, que fornece um framework inteligente para o escalonamento de VNFs e diagnósticos, considerando aspectos de tolerância a falhas. O framework concebe o uso do aprendizado profundo com métodos de prevenção que chamamos de afinidade e uma ferramenta de diagnóstico para avaliar o NFV em data center.

Métricas de Interesse Uma das decisões mais importantes na área de avaliação de desempenho é a escolha da métrica de interesse. Os artigos relacionado ao tema de otimização de processos das VNFs têm explorado (RODRIGUEZ; GUILLEMIN, 2016) (ID01) Tempo de Resposta, (CHOWDHURY et al., 2018) (ID02) CPU, Throughput, Latência, (YANG et al., 2019) (ID03) disponibilidade, atraso, taxa de aceitação, número médio de nós usados, (CAO; ZHANG; YANG, 2019) (ID04) CPU, largura de banda, taxa de aceitação, (MIJUMBI et al., 2015)(ID05) taxa de aceitação, receitas, custo, tempo médio de fluxo, (QU; ASSI; SHABAN, 2016) (ID06) largura de banda, tempo de escalonamento, (ASSI et al., 2018) (ID07) consumo de energia, CPU, fluxo médio, tempo de execução, taxa de admissão.

Disponibilidade é definida como a probabilidade de que um dispositivo funcione conforme necessário durante um período, sob condições operacionais especificadas. Como o desempenho de um sistema geralmente depende do desempenho de seus componentes, a disponibilidade de todo o sistema é uma função da disponibilidade de seus componentes (WAY; MING, 2003). Além disso, avaliar a rede de um data center é muito importante, pois ajuda a responder a questões de pesquisa definidas no Capítulo ??, bem proporcionar quais são os principais fatores de sucesso na construção de uma rede. A partir da definição do nosso problema de pesquisa, definimos métricas de disponibilidade, desempenho, performabilidade, análise de importância, análise crítica de falha e EDA que nos ajudariam a encontrar a melhor gestão e (re)planejamento da rede. No entanto, como podemos ver na Tabela 1, não encontramos trabalhos que abordem disponibilidade, performabilidade e desempenho aplicados à escalonamento de VNFs . O problema do escalonamento de VNFs é NP-difícil, portanto, há uma forte demanda por algoritmos eficientes para resolver o problema. Assim, resumimos a pesquisa disponível com base no trabalho existente e apresentamos os desafios para pesquisas avançadas que influenciam a disponibilidade, o desempenho e a performabilidade dos recursos da rede de data center. Assim, acreditamos que também são excelentes oportunidades de pesquisa na *Internet Research Task Force* (IRTF), IETF para esta área.

3.3 CONSIDERAÇÕES FINAIS

Este capítulo mostrou os principais trabalhos relacionados com o estudo proposto. Embora existam vários trabalhos na literatura que forneceu a avaliação de desempenho ou a avaliação de dependabilidade do data center por meio de modelos, nenhum desses trabalhos foca

na avaliação de aspectos de desempenho, dependabilidade, performabilidade, medições de importância através de aprendizado de máquina, modelos analíticos e expressões matemáticas. Alguns trabalhos apresentam uma solução para a avaliação de desempenho ou custo do escalonamento de VNFs, mas trabalho fornece uma ferramenta para avaliação de desempenho, dependabilidade e performabilidade.

4 ESCALONAMENTO INTELIGENTE

Este capítulo apresenta uma estratégia de escalonamento autônomo, considerando modelos de aprendizagem profunda e modelos de afinidade para NFV aplicado em data center. A abordagem tem considerado uma estratégia inovadora de parâmetros de infraestrutura equilibrada e tolerante a falhas. O gerenciamento inteligente de VNFs com circuito fechado (Closed-Loop) de gerenciamento não é uma tarefa trivial. Ao longo da última década, uma quantidade significativa de esforços têm sido investido na arquitetura de soluções de gerenciamento ágeis e adaptáveis em suporte a redes autogerenciadas e autônomicas. O escalonamento inteligente requer decisões automatizadas para ações de gerenciamento para evitar falhas e aumentar o desempenho do data center. A falha de rede é uma norma, não uma exceção, e seu impacto pode levar a perdas financeiras¹. A complexidade, heterogeneidade e escala das redes têm crescido muito além dos limites da administração manual.

4.1 DESCRIÇÃO DO PROBLEMA DE ESCALONAMENTO

Esta seção introduz o problema de escalonamento de VNFs, considerando regras de precedência e encaminhamento de SFC que realiza o escalonamento de diferentes VNFs em uma única CPU seguindo as diretrizes (HERRERA; BOTERO, 2016; BRUCKER; KNUST, 2006). Em um problema de escalonamento, é necessário encontrar intervalos de tempo nos quais as atividades devem ser processadas sob certas restrições, tais como restrições de recursos, restrições de precedência entre essas atividades, executar cada função de forma a minimizar o tempo total de execução sem degradar o desempenho do serviço e respeitando todos os precedentes. As dependências entre as VNFs que compõem o serviço de rede de acordo com sua ordem predefinida e rotear o tráfego entre essas VNFs sem violar as restrições de SLA, tais como largura de banda, taxa de transferência, consumo de CPU, memória, entre outras (CHOWDHURY et al., 2018; ZHANG et al., 2016). O problema de escalonamento de VNFs pode ser considerado análogo ao problema de escalonamento tradicional no contexto da computação em nuvem (ARUNARANI; MANJULA; SUGUMARAN, 2019). Entretanto, o escalonamento tradicional não considera que múltiplas funções de rede compartilham recursos computacionais, como escaloná-los dinamicamente e de acordo com sua ordem pré-definida. Além disso, a VNF apresenta inúmeras vantagens para a infraestrutura do data center:

- Primeiro, reduz os gastos operacionais, evitando a constante manutenção e atualização dos dispositivo

¹ <https://www.networkcomputing.com/networking/high-price-it-downtime>

- Segundos, permite uma gestão dinâmica da qualidade do serviço
- Terceiro, o compartilhamento controlado entre os fluxos de serviços de rede para utilizar os recursos disponíveis de forma eficiente.

O escalonamento de VNFs desempenha um papel essencial no data center, atribuindo e intercalando as requisições VNF a cada máquinas virtuais de forma eficiente e eficaz (YI et al., 2018). Figura 17 mostra o problema de escalonamento abordado nessa tese para compreender a importância crucial da tolerância a falhas. Suponha que existam três SFCs diferentes distribuídos em cinco racks de servidor de alto volume (Rack1-Rack5) e o objetivo é minimizar o tempo total de execução do conjunto de serviços para garantir alta disponibilidade e evitar falhas de repetição.

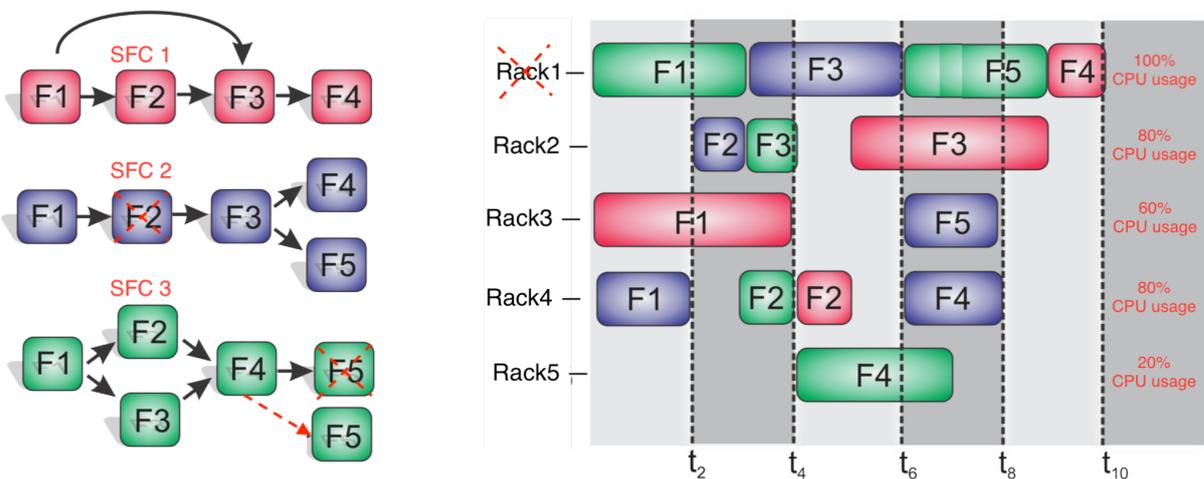


Figura 17 – Problema de Pesquisa no Escalonamento de VNFs

Neste contexto, há alguns casos em que as solicitações podem levar a problemas críticos de disponibilidade e desempenho no escalonamento de VNFs em data center, quando surgem durante Situações Anormais. Entre os quais, podem ser mencionados:

- 1): Suponha que não haja falhas e, por exemplo, o SFC3 consiste em cinco funções de rede F1-F5, rodando no Rack2, Rack3, Rack4, Rack5, respectivamente. Se estas funções de rede levam 3, 1, 1, 3 e 2 unidades de tempo, respectivamente, então o tempo total de execução do serviço SFC3 é de 9 unidades de tempo. No entanto, considere que uma falha no link, que por sua vez leva a falha ou um atraso maior do que 9 unidades de tempo.
- 2): Suponha que os serviços SFC1 estejam funcionando corretamente, mas o Rack1 falha ao longo do tempo, devido à sobrecarga máquinas virtuais, levando à redução da disponibilidade e degradação do dispositivo no data center.

- 3): Suponha que ocorra uma falha no SFC2 (F2) que cause a falha de todo o SFC. Neste caso, como não há redundância, o serviço não estará disponível. Falhas no serviço devido a falhas de uma cadeia têm um impacto severo nos clientes e podem resultar em perdas financeiras significativas.
- 4): Suponha que haja redundância na VNF (F5) no SFC3. Neste caso, a VNF falha, mas permanece operacional, e assim permanece todo o sistema, tendo a mesma disponibilidade, supondo que não haja falha no Rack1 do dispositivo físico.
- 5): Suponha que dois tipos de falhas ocorrem no Rack1: (i) o rack está operando a 100% de sua capacidade, o superaquecimento ocorreu e o rack falha inesperadamente; (ii) o pedido de escalonamento de recursos sobre a capacidade do Rack1 gera sua falha, uma vez que já está operando a 100%, e não há espaço para manutenção ou realocação de VNF.
- 6): Suponha uma alocação de recursos ineficiente de VNF para cada máquinas virtuais nos Racks. Neste caso, alguns racks irão operar na sua capacidade total e outros na sua capacidade mínima podem levar a falhas ou degradação do desempenho.
- 7): Suponha que a quantidade de recursos da CPU alocada não é suficiente. Neste caso, a VNF pode falhar. Por outro lado, se a quantidade de recursos for superprovisionada, isso pode causar um desperdício de recursos.

Vale ressaltar que uma falha em uma única VNF proporcionar falhas em toda a SFC, o que resultaria em desperdício de recursos, atrasos, degradação do desempenho e perdas significativas de dados. Uma solução possível para tal situação seria confiar em estratégias de redundância. No entanto, onde colocar a redundância continuaria a ser uma tarefa complicada a ser resolvida. Além disso, é difícil escalar eficientemente para otimizar a gestão de serviços sem violar restrições de capacidade, atrasos e disponibilidade. Finalmente, como o controle e gerenciamento centralizados são muito mais vulneráveis à propagação de erros entre diferentes VNFs, a falha em operar um módulo de software pode afetar outras funcionalidades relacionadas que dependem de informações compartilhadas. Neste contexto, apesar de depender de uma plataforma de hardware distribuída, o sistema fica muito mais exposto a falhas de causa comum no projeto e operação. Neste sentido, o problema de escalonamento abordado nessa tese, consiste em quatro partes que podem ser associadas a cada uma das QPs introduzidas no Capítulo ??, em termos de:

- Qual é a estratégia de aprendizagem profunda mais adequada para realizar o escalonamento?
- Como criar uma estratégia inteligente e automatizada para o escalonamento de recursos de rede considerando aspectos de um data center tolerante a falhas em cenários NFV?

- Qual é o conjunto de métricas representativas que podem quantificar os impactos na degradação do desempenho do escalonamento de VNFs e servir como input para o operador de rede ajustar o modelo de aprendizagem profunda ou redesenhar a redundância do dispositivo?

4.2 ABORDAGEM PROPOSTA

Nesta seção, é apresentada uma solução para o problema de escalonamento de VNFs. A solução é baseada no paradigma KDN (MESTRES et al., 2017). A nossa abordagem estende-a ao englobar também o ecossistema VNFs, juntamente com novos modelos e funcionalidades, conforme listados abaixo:

- Aplicado ao Contexto da NFV com SDN
- Modelos de Previsão com Vários Passos à Frente
- Modelos de Afinidade
- Escalonamento de VNFs Online e Offline
- Modelos EDA
- Modelos de Performabilidade
- Modelo de Falha
- Modelo de Temperatura
- Modelos de Índice de Importância de Dispositivos
- Modelos de Índice de Dispositivos Críticos

A Figura 18 mostra a Framework inteligente neste trabalho para a realizar escalonamento de VNFs tolerante a falhas, com base em um modelo de prevenção e um modelo de previsão de falhas. A arquitetura é composta por três planos:

- Plano de conhecimento: pode tomar decisões automaticamente em nome do operador de rede. Assim, ele pode ser usado para a otimização da configuração de rede existente, já que o modelo de rede aprendido pode ser explorado através de técnicas convencionais de otimização para configurações ideais. Por outro lado, o operador de rede ainda está encarregado de tomar as decisões em circuito aberto (Open-Loop);
- Plano de Controle/Gestão: permite efetivamente que aplicações utilizadas pelos administradores de rede controlem e monitorem as funções da rede sem a necessidade de ajustes. Além disso, tem interface entre um controlador SDN/NFV e o operador de rede. Através dele, define a topologia da rede e trata do fornecimento e configuração de dispositivos de rede para realizar diversas funções;

- Plano de Dados: responsável pelo armazenamento, processamento e encaminhamento de dados na rede. Consiste em dispositivos de encaminhamento, que são elementos de hardware (roteadores ou switches) ou software (máquinas virtuais) especializados em encaminhamento de pacotes.

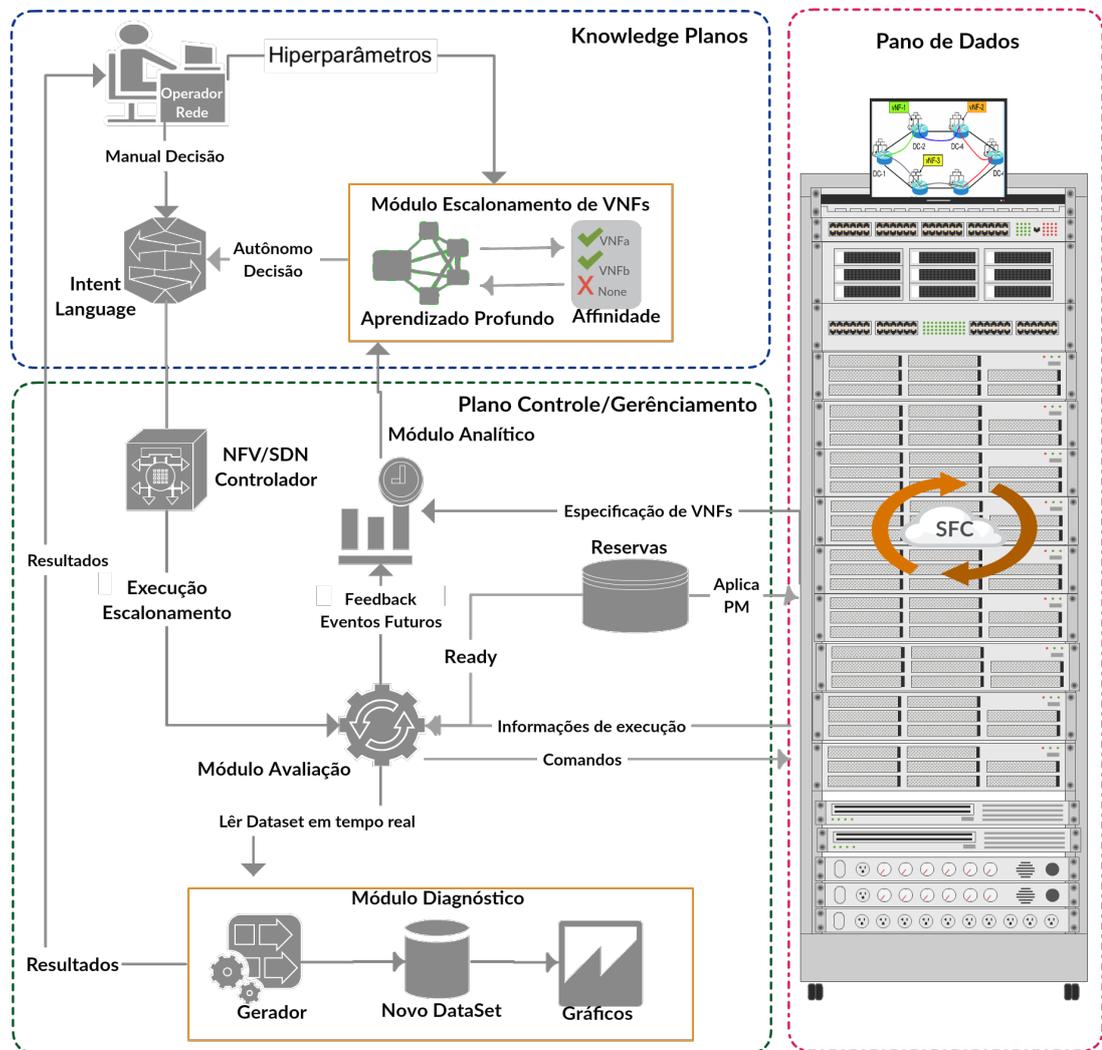


Figura 18 – Framework Inteligente para Escalonamento de VNFs em Data Center

Na Figura 18, a arquitetura na linha tracejada em azul contém a abordagem de escalonamento baseada em KDN, que é o foco da pesquisa conduzida nesta tese. A arquitetura na linha tracejada em vermelho, por outro lado, representa o data center onde as VNFs estão instaladas. Em particular, a arquitetura proposta representada na Figura 18 consiste em cinco módulos, o analítico, a escalonamento tolerante a falhas, a reserva, o desempenho e os módulos de diagnóstico, descritos a seguir.

- **Módulo Analítico:**

- O módulo analítico visa reunir informações suficientes para fornecer uma visão completa das VNFs (CLEMM; CHANDRAMOULI; KRISHNAMURTHY, 2015) ao

Módulo de Tolerância a Falhas. O módulo analítico monitora os elementos da rede em tempo real enquanto eles encaminham pacotes para acessar informações de tráfego de diferentes níveis de granularidade e exigências de QoS.

- As informações coletadas, sobre futuros eventos de *feedback* para evitar falhas, evita a alocação ineficiente de recursos e permite um melhor treinamento de algoritmos. Assim, os computadores aprendem automaticamente sem a intervenção humana.
- O módulo analítico recebe as especificações de VNF, que contém informações do plano de dados, relacionadas com a definição de como um recurso é tratado. Além disso, recebe informações de status, uso da CPU, memória e outros fatores relacionados ao desempenho da máquina virtual, na qual a VNF está implantada e pode precisar de *self-healing* de VNF e escalar o serviço de rede.

- **Módulo Escalonamento:**

- Inicialmente, o Módulo Analítico destina-se a estimar com precisão quantas VNFs devem ser solicitadas, e quando devem ser disponibilizadas, e que tarefa deve ser atribuída a cada uma delas.
- O módulo de escalonamento, que toma a matriz de tempo de execução, fará um plano de tempo de execução completo sobre quantas VNFs devem ser locadas, e devem ser disponibilizadas, e qual tarefa deve ser atribuída a quais VNFs.
- A seguir, oferece uma interface comum para configuração com linguagem de intenção (*intent language*) para descrever a necessidade de alterar a configuração da rede e permite expressar uma decisão autônoma ou com configurações manuais ao atualizar a configuração da política nos controladores NFV/SDN. Em seguida, analisa o escalonamento e gera automaticamente planos de atualização que satisfaçam a intenção expressa.
- Ajuste de hiperparâmetros: otimização de hiperparâmetros, o que maximiza a precisão preditiva do modelo com regras de afinidade para avaliar até que ponto (e como) o escalonamento pode atender às restrições do sistema:(i) modelos de afinidade (ii) modelos aprendizagem profunda
- Neste módulo são implementadas as bibliotecas da linguagem python: *scikit-learn*², e *Keras*³. Assim, é possível estender facilmente este framework para utilizar mais 60 algoritmos de aprendizado de máquinas e de aprendizagem profunda para diferentes abordagens de escalonamento.

- **Módulo de Reserva:**

² scikit-learn.org

³ keras.io

- Ready: O módulo de reserva, obtém as VNFs do provedor de serviços e preparas. Uma vez obtido o escalonamento, o módulo de escalonamento está em condições de informar ao módulo de reserva sobre o montante e os tipos de VNFs necessários,
- Aplicar os máquinas virtuais: fornecer o plano de execução para o módulo de execução. Desta forma, o módulo de reserva pode aplicar os máquinas virtuais para atender às regras de afinidade.

- **Módulo de Execução:**

- Comando e Informações de Execução: envia os comandos ao data center para executar a alocação de recursos, informando ao data center qual VNF deve ser alocada a qual máquinas virtuais, bem como quando iniciar ou encerrar a alocação. Além disso, informações sobre o tempo real de execução das tarefas, largura de banda, latência e perda de pacotes, entre outras, são coletadas e retornadas ao módulo de estimativa como *feedback*. Desta forma, se um fluxo de trabalho similar estiver em execução, as informações de *feedback* podem ser usadas para melhorar a precisão da estimativa do tempo de execução. Finalmente, uma vez executado todo o fluxo de trabalho, o módulo de execução deve retornar o resultado final ao operador de rede.
- Execução do Escalonamento: fornece ao controlador SDN o comportamento do encaminhamento de pacotes basicamente, transformando os dispositivos no comportamento que a política dita pela *intent language* a ser transformada pelo controlador SDN em ações que serão enviadas aos roteadores para se comportarem como firewalls, perda de pacotes, ou alguma QoS fazendo com que o fluxo tenha uma taxa constante, ou direção de equilíbrio/tráfico. O controlador NFV lida com a orquestração e o ciclo de vida das NFV para os seus propósitos de escalonamento e outros da arquitetura NFV.
- Durante a execução, recebe informações sobre a execução das tarefas em tempo real e retorna essas informações para o módulo de estimativa como *feedback*. Neste ponto, se houver outro SFC similar a ser executado, a informação de *feedback* pode ajudar a melhorar a eficiência do tempo de execução. Após executar todo o SFC, o módulo de execução também é responsável por devolver o resultado final ao operador de rede.

- **Módulo de Diagnóstico:**

- Lêr Dataset: utiliza diferentes modelos analíticos para detectar gargalos em relação à performabilidade, temperatura e problemas de disponibilidade para fornecer um diagnóstico completo da rede.

-
- Resultados: desta forma, o operador de rede recebe informações úteis para atualizar e ajustar o modelo de previsão, bem como para replanejar a redundância do dispositivo para atender às demandas necessárias dos recursos da rede.

Subseções 4.2.2 e 4.2.3 apresenta uma descrição detalhada dos dois módulos importantes ou centrais da nossa proposta, a saber, a programação tolerante a falhas e os módulos de diagnóstico, respectivamente. Os módulos restantes visam a interação com os prestadores de serviços e, portanto, não serão detalhados nessa tese, bem como não é a contribuição principal da tese. Uma descrição detalhada dos mesmos está fora do escopo desta tese.

4.2.1 Módulo Analítico

Este módulo recebe informações das especificações VNF, requisitos de QoS, obtém *feedback* de eventos futuros do módulo de Diagnóstico. Além de funcionar como uma interface para configuração manual do módulo escalonamento tolerância a falhas. Utilizamos um modelo de afinidade no qual é possível medir matematicamente a afinidade de um par de VNFs com base num conjunto de critérios propostos no trabalho (JACOBS et al., 2017). Os serviços de afinidade de centros de dados podem oferecer aos clientes capacidades de computação *on-demand*, ao contrário da infraestrutura tradicional. Ao mesmo tempo, novas demandas de recursos permitem aos clientes acessar o que eles precisam e onde eles precisam, pois eles têm informações sobre o tipo de atividade e recursos exigidos por cada tipo de aplicativo, permitindo que eles paguem pelo que foi usado e antecipando as necessidades de uso sem recursos excessivos. O uso da afinidade entre aplicações tem como objetivo contribuir para o escalonamento do ambiente virtual. Baseia-se nas características do consumo de recursos físicos e redes entre VNF e máquinas virtuais.

Assim, o monitoramento ativo do desempenho atual da rede é um passo crítico para identificar problemas existentes e futuros para garantir a dependabilidade durante a operação e evitar falhas. Uma regra de afinidade define um grupo de VNFs em diferentes máquinas virtuais, e evita que todas as máquinas virtuais falhem ao mesmo tempo se uma única máquinas virtuais falhar. As regras incluem parâmetros relacionados ao SLA, como regras de afinidade e antiafinidade e requisitos de latência, a relação de afinidade e o grupo de afinidade das VNFs, respectivamente, e várias regras de agrupamento de VNFs baseadas em afinidade. A dependência entre as VNFs é definida como uma afinidade das VNFs (JACOBS et al., 2017).

- Coletar estatísticas
- Manter e examinar os registros dos históricos do estado do sistema
- Determinar o desempenho do sistema em condições naturais e artificiais
- Alterar os modos de operação do sistema a fim de conduzir atividades de gerenciamento de desempenho

Se um operador de rede tomar uma decisão manual, ele pode usar configurações de modelo de aprendizagem profunda e alterar os requisitos de afinidade de acordo com os critérios FG e máquinas virtuais que podem ser estáticos ou dinâmicos e variam entre 0.001 até 0.9. Os parâmetros são utilizados como entrada para o modelo LTSM. Por exemplo, se duas VNFs têm um requisito mínimo de 50% dos recursos computacionais e as VMs que hospedam as VNFs têm recursos 100% disponíveis, a afinidade resultante para esta função será de 0.50. Outro exemplo está na situação de conflito padrão que afirma que uma VNF de balanceamento de carga não deve ser encadeada antes de uma VNF de firewall, e as duas VNFs sendo avaliadas são uma DPI que é encadeada antes de um firewall na mesma FG, de modo que esta característica resultará no valor 0.0001.

1. Escopo Estático

- a) Critério máquinas virtuais: mínimo de máquinas virtuais, mínimo de memória, mínimo de armazenamento
- b) Critério FG: Conflitos VNFs

2. Escopo Dinâmico

- a) Critério máquinas virtuais: Uso de CPU, uso de memória, uso de armazenamento
- b) Critério FG: Uso de largura de banda, perda de pacotes, latência

4.2.2 Módulo Escalonamento de VNFs

Figura 19 mostra o fluxograma para a abordagem de escalonamento proposto tolerante a falhas, que se baseia na integração do modelo de previsão e das regras de afinidade para alocação dos pares VNFs aplicados a um determinado máquinas virtuais, destacados em linhas tracejadas azuis e vermelhas na Figura 19, respectivamente. Em geral, a abordagem de escalonamento tolerante a falhas proposta flui da seguinte forma. O modelo de previsão baseado em LTSM recebe a matriz de entrada do escalonamento de VNFs e a passa através das camadas LTSM. Em seguida, o processo de abandono (dropout) para evitar o sobreajuste dos dados de treinamento e abandono de unidades na rede neural (SRIVASTAVA et al., 2014) é aplicado à saída da rede LTSM, e o resultado é enviado para a camada densa, a rede neural totalmente conectada. Finalmente, uma função de ativação (i.e., um *Rectified Linear Unit* (ReLU) é usada no fim de uma unidade escondida para introduzir complexidades não lineares ao modelo, assim, processar as regras de afinidade tanto para abordagens estáticas como dinâmicas. Dentro do contexto do modelo LTSM proposto, é considerada a correlação de diferentes entidades em eventos futuros que são inicialmente definidos pelo operador de rede. Em particular, para otimizar o escalonamento, esses modelos são operados on-line.

Neste contexto, a afinidade é definida como um indicador de até que ponto dois VNFs operam bem, seja quando colocados no mesmo máquinas virtuais ou quando encadeados

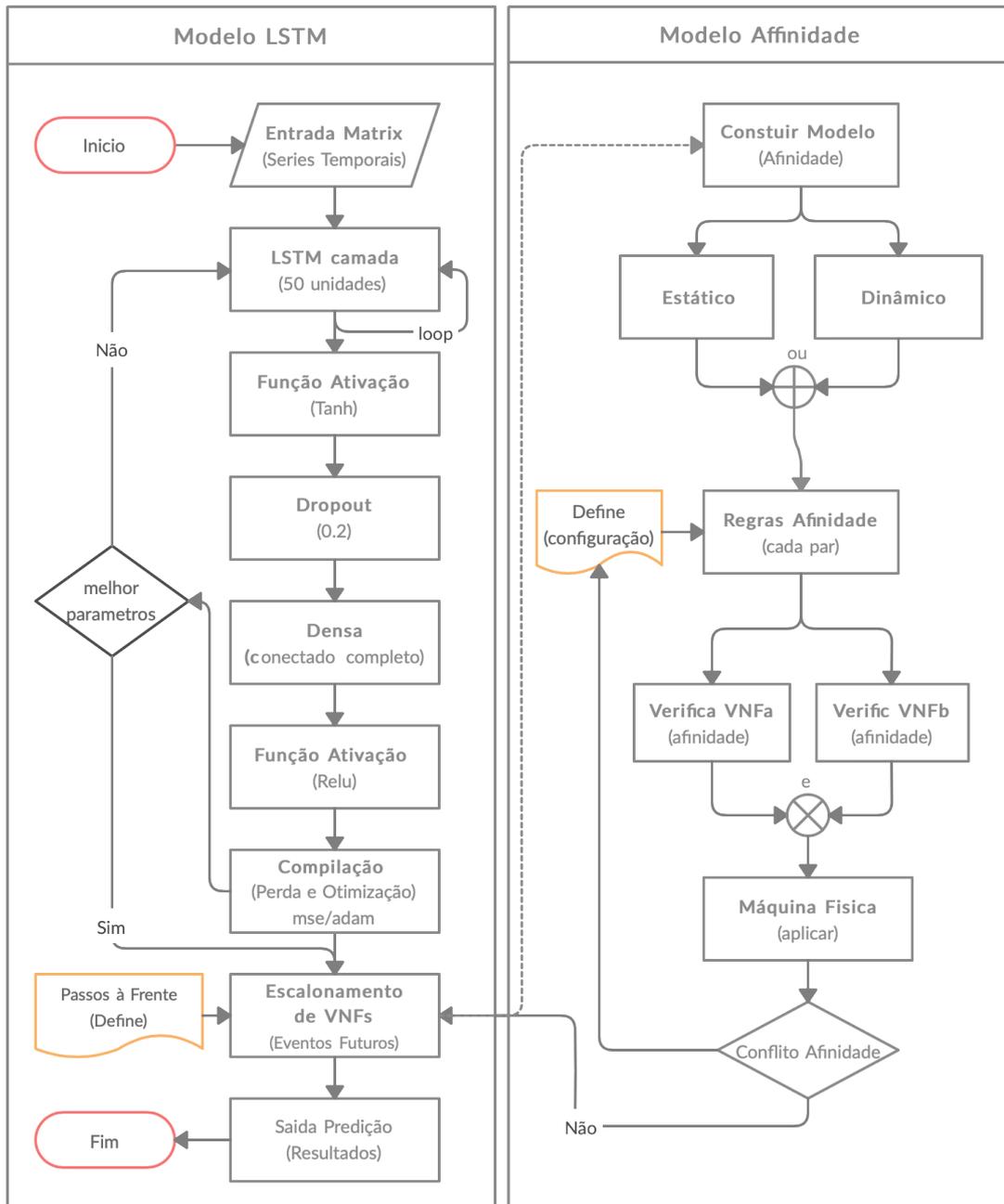


Figura 19 – Fluxograma do Framework Inteligente de Escalonamento de VNFs

no mesmo FG, mostrando os enlaces lógicos conectando os nós VNFs para descrever o fluxo de tráfego entre eles. Além disso, a afinidade pode ser avaliada com base em dois critérios diferentes, regras estáticas ou dinâmicas. Os casos estáticos que as VNFs não precisam ser avaliadas online. Nesses casos, as informações estão geralmente disponíveis em descritores, como os descritos pelo *European Telecommunications Standards Institute* (ETSI) em (CHIOSI et al., 2012). As regras dinâmicas, por outro lado, estão associadas aos casos, em que os dados de utilização de recursos on-line são usados para medir afinidade e, portanto, só devem ser consideradas quando as VNFs são avaliadas on-line.

Em geral, a afinidade é calculada com base em média harmônica combinando cálculos de baixo nível num único valor numérico que representa a afinidade entre as duas VNFs avaliadas. Neste trabalho, cada regra de afinidade, como CPU, memória, armazenamento, largura de banda, perda de pacotes, latência, conflitos VNF é definida dentro de um determinado intervalo e é classificada dentro de um determinado tipo a fim de calculá-la recorrendo a uma equação particular, resultando em valor dentro do intervalo 0.001 e 1, indicando o quão bem, em termos de afinidade, os VNFs analisados operam em conjunto. Equação 4.1 apresenta o modelo de afinidade, correspondendo a uma média harmônica entre afinidade estática (α_s) e afinidade dinâmica (α_d), onde Forwarding Graph (FG) é usado para orquestrar e gerenciar o tráfego através de VNFs. Além disso, como os dados monitorados podem variar para diferentes cadeias, a afinidade resultante, representada na Equação 4.1 por $\alpha(VNF_a, VNF_b)$, terá um valor diferente para cada FG em que ambas as VNFs estão encadeadas (JACOBS et al., 2017).

$$\alpha_{(VNF_a, VNF_b)} = \frac{1 + p}{\frac{1}{\alpha_s} + \frac{p}{\alpha_d}} \quad (4.1)$$

$$p = \begin{cases} 1 & \text{se dois VNFs estiverem funcionando,} \\ 0 & \text{caso contrário} \end{cases} \quad (4.2)$$

Uma vez que a Equação 4.1 já foi calculada, o modelo de previsão é construído com base em modelos sensíveis à afinidade para prever as necessidades esperadas de recursos SFC para o escalonamento de VNFs, com base no cálculo da afinidade dinâmica de máquinas virtuais e afinidade de rede. Entretanto, a consideração ou não de máquinas virtuais e afinidades de rede depende das seguintes condições: 1 se as duas VNFs estiverem alojadas no mesmo máquinas virtuais, e 0 caso contrário, da mesma forma para encadeadas em FG, onde será 1 se as duas VNFs estiverem diretamente encadeadas em FG e 0 caso contrário. Se essas condições não forem satisfeitas, então x ou y serão zero, desconsiderando máquinas virtuais ou a afinidade de rede da equação.

4.2.3 Abordagem do Módulo Diagnóstico

O módulo de diagnóstico, que visa executar um diagnóstico holístico levando em conta o Modelo de Falha, EDA e Performabilidade da escalonamento de VNFs no data center,

está incluído na abordagem proposta para fornecer ao operador de rede ferramentas úteis para melhor definir as regras de afinidade de escalonamento. Em particular, ele fornece métricas baseadas na observação da utilização de recursos contabilizando a ocupação de recursos abstraídos, permitindo que eles sejam utilizados internamente na VNF.

Assim, ferramentas úteis ao operador de rede para planejar ou ajustar as regras de afinidade para escalonamento autônomo ou verificar se é necessário adicionar redundância de dispositivos físicos ou virtuais. Desta forma, espera-se que o módulo de diagnóstico permita ao operador de rede possa definir com maior precisão as regras de afinidade de escalonamento. Em seguida, sempre que o mecanismo de tomada de decisão (seja manual ou autônomo) tiver que lidar com algumas situações não controladas. Por exemplo, se houver uma sequência entre diagnósticos de sub-provisionamento e de sobre-provisionamento sem uma indicação adequada de recursos, procura-se minimizar os custos de operação para manter a alocação de recursos com uma capacidade adequada tanto no aprovisionamento como privilegiando a qualidade do serviço.

Figura 20 ilustra o módulo de diagnóstico proposto com base em modelos analíticos para detectar e quantificar a disponibilidade e performabilidade, bem como para realizar análises de falhas. A fim de fornecer um diagnóstico holístico da escalonamento de VNFs no data center, o módulo de diagnóstico proposto é mostrado na Figura 20 e depende de cinco passos, (i) Conversor de frequência de CPU para utilização de CPU, (ii) Cálculo do MTTF, (iii) Gerador, (iv) Avaliação, e (v) Gráficos, realizados nos módulos descritos abaixo.

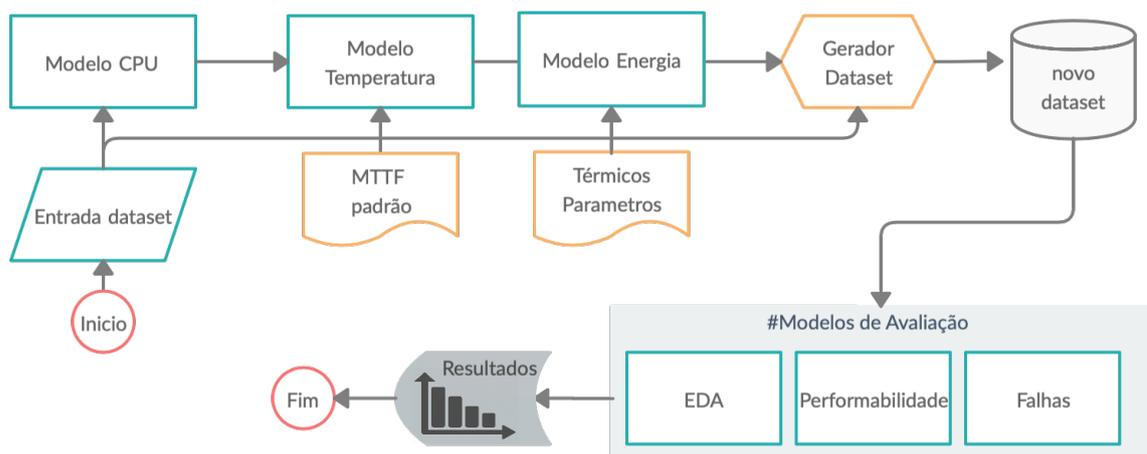


Figura 20 – Módulo Diagnóstico

4.2.3.1 Modelo de CPU

As unidades dos dados de consumo da CPU no dataset⁴ são verificadas e convertidas em MHz. A frequência da CPU está relacionada com o uso da CPU de acordo com a Equação

⁴ github.com/rafaroques/nfvdatacenter

4.3:

$$Freq = N_{cores} * F_{(m_c)} * CPU_{uso}, \quad (4.3)$$

onde: $Freq$ é a frequência média de utilização da CPU em MHz, $N_{núcleos}$ é o número de núcleos, $F_{(m_c)}$ é a frequência máxima de cada núcleo e CPU_{uso} é a porcentagem de uso da CPU.

4.2.3.2 Modelo de Temperatura

A partir de nosso trabalho anterior (SOUZA et al., 2013), estendemos e definimos um novo modelo que é a base para a criação do modelo de performabilidade. Assim, definir uma equação que contabilize as mudanças de temperatura relativas ao tempo de processamento, certas suposições, tais como tratar o processador da CPU como um sistema lumped ($Bi < 0.1$) e considerando as propriedades materiais constantes para a temperatura, são levadas em conta. Então, conforme a primeira lei da termodinâmica, a equação é a seguinte.

$$\sum Q_{in} - \sum Q_{out} + \sum Q_{gen} = m * C * \frac{dT}{dt}, \quad (4.4)$$

onde, Q_{in} é a taxa de transferência de calor de entrada, Q_{out} é a taxa de transferência de calor residual, Q_{gen} é a taxa de transferência de calor gerada, m é a massa (kg), C é o calor específico (J/kgK), T é a temperatura graus Celsius e t é o tempo (em horas). Em particular, no caso aqui abordado, a taxa de transferência de calor que está entrando no processador é igual a zero. Por outro lado, a taxa de transferência de calor que sai do processador é dada por um mecanismo de transferência de calor chamado convecção. Finalmente, a taxa de transferência de calor gerada é a que é produzida pela dissipação total de energia do processador (SOUZA et al., 2013). Em tal contexto, a Equação 4.4 para:

$$\begin{aligned} -Q_{in} + P_{total} &= m * C * \frac{dT}{dt} - h * A_s * (T_f - T_\infty) I * V + \alpha * C_p * V^2 * f \\ &= m * C * \frac{dT}{dt}, \end{aligned} \quad (4.5)$$

onde: T_t é a temperatura no tempo t , T_∞ é a temperatura ambiente em Celsius, V é a Voltagem (V), α é o fator ativação C_p é a capacitância (μF), h é o coeficiente convectivo (W/m^2K), f é a frequência (MHz) e A_s é a área de superfície da placa-mãe (m^2).

A resolução da Equação 4.5 leva à solução geral para a temperatura dada pela equação 4.6 (o passo a passo para alcançar a Equação 4.6 pode ser encontrado no apêndice ??).

$$T_{(t)} = T_\infty \frac{I * V + \alpha * C_p * V^2 * f}{m * C} + A_1 * e^{-\frac{h * A_s}{m * C} t} \quad (4.6)$$

A constante A_1 na equação 4.6 deve ser calculado resolvendo a solução geral para uma condição inicial. Assumimos que o processador CPU está em equilíbrio térmico com o

ambiente, conduzindo a temperatura inicial (T_i) ser igual à temperatura ambiente (T_∞). Este equilíbrio térmico só ocorre antes de o processador começar a funcionar. Levando isto em conta os rendimentos para:

$$T_{(t=0)} = T_\infty = T_\infty + \frac{I * V + \alpha * C_p * V^2 * f}{m * C} + A_1 * e^{-\frac{h * A_s}{m * C} t}. \quad (4.7)$$

Então, resolvendo para o valor constante A_1 , Equação 4.8 é obtida:

$$A_1 = \frac{I * V + \alpha * C_p * V^2 * f}{m * C}. \quad (4.8)$$

A solução particular para a temperatura do processador da CPU é então dada por:

$$T_{(t=0)} = T_\infty = T_\infty + \frac{I * V + \alpha * C_p * V^2 * f}{m * C} + \left(1 - e^{-\frac{3600 * h * A_s}{m * C} t}\right). \quad (4.9)$$

4.2.3.3 Gerador do Dataset

O módulo gerador de dataset cria um novo dataset baseado nos parâmetros térmicos recebidos da sala do data center, modelo de CPU, modelo de temperatura e tempo de reparo do dispositivo, além de considerar o dataset anterior, como largura de banda e perda de pacotes, entre outros. Além disso, este módulo permite definir vários formatos de arquivos para salvar o dataset.

4.2.3.4 Módulo de Avaliação

O módulo de avaliação permite escolher e aplicar diferentes estratégias de análise, tais como (i) Modelo de Falha, (ii) Modelo de Performabilidade e Análise EDA, com base nos dados do novo dataset.

O **Modelo de Falha** calcula a taxa de falhas, que é a frequência com que um sistema ou componente falhou, expressa em falhas por unidade de tempo. A relação número de componentes que falharam pelo número de total de componentes. Por exemplo, para um único equipamento em operação, pode ser estimado de acordo com a seguinte expressão:

$$\lambda = \frac{1}{MTTF}, \quad (4.10)$$

onde, λ é a taxa de falha e MTTF é o período de tempo que se espera que um dispositivo esteja funcionando. Por exemplo, um sistema com uma taxa de falha constante, o MTTF pode ser calculado pela taxa de falha inversa, $1/\lambda$. Assumindo a taxa de falha, λ , seja em termos de falha/horas, $MTTF = 1.000/\text{taxa de falha}$, λ , para componente.

Nessa tese, são considerados dois novos modelos de performabilidade, o baseado em falha e o baseado na temperatura, sendo este último uma das principais contribuições desta tese. Dentro do contexto do modelo de performabilidade baseado na falha, a Equação

4.10 pode ser usado para mostrar que a falha de uma VNF pode levar à falha do serviço, tornando o sistema inoperante:

$$P_f = 1 - P(\lambda_{PM}). \quad (4.11)$$

Este modelo de performabilidade concentra-se na avaliação do desempenho durante a ocorrência de falha no sistema. Neste caso, a probabilidade de reparo das falhas não é analisada. O modelo de performabilidade baseado na temperatura é usado para calcular a influência da temperatura na taxa de falha, que normalmente é chamado de "Fator de Aceleração" (ALLEN; ROESCH, 1990).

$$A_T = \frac{\lambda_T}{\lambda_R} = \frac{\frac{1}{MTTF_T}}{\frac{1}{MTTF_R}} = \frac{MTTF_T}{MTTF_R}, \quad (4.12)$$

onde: λ_T é a taxa de falha prevista (falhas/horas), λ_R é a taxa de falhas tendo em conta os efeitos da temperatura (falhas/horas), $MTTF_T$ é o MTTF previsto (horas) e $MTTF_R$ é o MTTF, tendo em conta os efeitos da temperatura (horas). Além disso, o fator de aceleração satisfaz a seguinte equação.

$$A_T = e^{\frac{E_a}{k} \left(\frac{1}{T_p} - \frac{1}{T_{(p0)}} \right)}, \quad (4.13)$$

onde: E_a é a energia de ativação (eV), K é a constante de Boltzmann (eV/K), $T_{(f)}$ é a temperatura do processador (K) e T_p é a temperatura ambiente (K). A energia de ativação para a quebra dielétrica de semicondutores está normalmente dentro da faixa 0.3eV - 0.7eV. Esta tese adotou um valor de 0.642eV (SOUZA et al., 2013).

Duas condições principais de temperatura afetam a disponibilidade em um processador da CPU, a temperatura ambiente e a temperatura do processador (que varia em função da frequência do processador). Neste contexto, para estudar a disponibilidade num processador de uma CPU, tendo em conta as condições de temperatura mencionadas, Equação 4.13 pode ser reescrito da seguinte forma:

$$A_T = e^{\frac{E_a}{k} \left(\frac{1}{T_f} - \frac{1}{T_\infty} \right)}, \quad (4.14)$$

onde: T_f é a temperatura calculada em função da frequência e T_∞ é a temperatura ambiente. De acordo com a Equação 4.13 e 4.14, pode ser dito que:

$$\frac{MTTF_R}{MTTF_T} = e^{\frac{E_a}{k} \left(\frac{1}{T_{(f)}} - \frac{1}{T_\infty} \right)}, \quad (4.15)$$

Então, com base na Equação 4.15, a influência das mudanças na temperatura do processador no MTTF é dada por:

$$MTTF_R = MTTF_T * e^{\frac{E_a}{k} \left(\frac{1}{T_{(f)}} - \frac{1}{T_\infty} \right)}. \quad (4.16)$$

Neste contexto, a performabilidade inerente do processador pode ser estimada após a escalonamento de VNFs da seguinte forma:

$$P_f = 1 - P(\lambda_{MTTF_R}). \quad (4.17)$$

Finalmente, criamos um novo modelo de performabilidade, que se baseia nos aspectos do impacto da temperatura sobre o desempenho. As novas métricas captam a degradação do desempenho em função da variação da temperatura. A variação de temperatura ocorre por razões como: alocação ineficiente de recursos, arquitetura de resfriamento ineficiente, falha em dispositivos virtuais ou físicos. Em particular, este modelo é útil para analisar os casos em que o sistema não falha, mas ainda há uma degradação do desempenho devido a problemas de disponibilidade.

4.2.4 Cenários Realistas

A solução desenvolvida na tese, pode ser aplicada a vários cenários realistas no data center/nuvem. Por exemplo, suponha que as cargas de trabalho intensivas do plano de dados com uma faixa de 10% a 100% do consumo de CPU para transportar o tráfego da rede do *Network Interface Card* (NIC) físico para VNFs, essa alta variação de tráfego pode levar a uma falha de sobrecarga da CPU ou a uma falha de escalonamento de VNFs ineficiente. Ambos os problemas podem levar a períodos de inatividade.

Segundo Gartner⁵, o custo financeiro estimado do tempo parado sugere que uma perda de receita na faixa de \$250,000 para \$500,000, sendo que 60% empresas por uma hora de inatividade e um sexto das organizações sofreu uma perda de \$1 milhão ou mais. Além de, 98% das organizações dizem que uma única hora de inatividade custa mais de \$100,000 e pode prejudicar permanentemente a reputação de uma empresa. As operações realizadas pelas VNFs também consomem recursos da CPU para implementar com sucesso e eficiência as cargas de trabalho do plano de dados. Entretanto, surge um problema de como escalar e evitar falhas de forma eficiente. Tradicionalmente, o operador de rede terá que configurar manualmente todas as operações de rede para cada cenário e reconfigurar para mudanças baseadas na dinâmica de tráfego.

Por outro lado, nossa solução permite o escalonamento automático da rede, considerando tanto o aspecto preventivo quanto o preditivo das falhas, de forma que é possível obter o máximo desempenho da infraestrutura NFV que realiza intensas cargas de trabalho de planos de dados, selecionando dispositivos com políticas de afinidade de rede e físicos para a gestão do tráfego de planos de dados. Com base na linguagem python, para a qual a estrutura foi desenvolvida, podemos usar o RestAPI usando DjangoRestPlus⁶ ou FlaskRestPlus⁷. Assim, é possível definir estratégias de treinamento, testes e transferência

⁵ <https://blogs.gartner.com/andrew-lerner/2014/07/16/the-cost-of-downtime>

⁶ www.django-rest-framework.org

⁷ <https://flask-restplus.readthedocs.io>

de aprendizagem (usando o modelo pré-treinado para treinamento). Em seguida, é possível implementá-lo no servidor usando vários métodos. No entanto, o mais proeminente é o emprego de container como: Docker⁸, Kubernetes⁹ na Microsoft Azure Cloud¹⁰, Google Cloud¹¹, ou Amazon Cloud¹² ou outros provedores para modelo de deploy, onde o software é executado em um cenário realista para o escalonamento de VNFs no data center da nuvem.

Um provedor de infraestrutura, pode oferecer recursos de infraestrutura como máquinas virtuais, redes, armazenamento, que podem ser usados pelos provedores de serviços para fornecer serviços ao usuário final, como *Software as a Service* (SaaS), a seus clientes, serviços que podem ser desenvolvidos usando ferramentas *Platform as a Service* (PaaS). Três tipos principais de cenários de nuvem podem ser listados abaixo (TOOSI; CALHEIROS; BUYYA, 2014).

- Nuvem híbrida: É a combinação do ambiente público e privado, que normalmente é configurado e desenvolvido sob demanda em um data center.
- Nuvem Federada: Nuvens Federadas são colaborações de nuvens baseadas em acordos de partilha de carga que lhes permitem descarregar capacidade umas das outras de forma semelhante à forma como os fornecedores de eletricidade trocam capacidade.
- Múltiplas nuvens: Em cenários multi-nuvem, o serviço prestado é responsável por lidar com a complexidade adicional de coordenar o serviço através de múltiplos fornecedores de infraestrutura.

O modelo ANN foi selecionado, porque, é o mesmo modelo utilizado no trabalho de (MESTRES et al., 2017; JACOBS et al., 2018), mas com aplicação na escalonamento de VNFs e outros aprendizados profundos, foram escolhidos modelos, por ser o que apresenta os melhores resultados (ver Subseção 5.4.1) e por serem aplicadas às nossas características de séries temporais dataset¹³. Assim, foi escolhida porque a LSTM apresentou o melhor resultado e tem a vantagem da sua propriedade de recordar seletivamente padrões por longos períodos de tempo. A abordagem proposta proporciona um escalonamento inteligente da alocação de recursos, permitindo ao operador de rede, o gerenciamento preventivo de falhas, ao ajustar os parâmetros de afinidade com base na análise de predição de eventos, evitando, desta forma, possíveis falhas. Além disso, os modelos são construídos utilizando um grande conjunto de dados de um cenário realista. A utilização destes modelos de predição supervisionado pode ser utilizada em aplicações que utilizam séries temporais no contexto de NFV.

⁸ docker.com

⁹ kubernetes.io

¹⁰ azure.microsoft.com

¹¹ cloud.google.com

¹² aws.amazon.com

¹³ github.com/rafaroques/nfvdatacenter

O operador de rede pode ou não estar envolvido no processo de tomada de decisão, que é: circuito fechado e circuito aberto. O circuito fechado são os modelos de rede obtidos que podem ser utilizados em primeiro lugar para a automação, uma vez que o *Plano conhecimento* pode tomar decisões automaticamente em nome do operador de rede. Assim, ele pode ser usado para a otimização da configuração de rede existente, uma vez que o modelo de rede aprendido pode ser explorado através de técnicas convencionais de otimização para configurações ideais. Por outro lado, o operador de rede ainda está encarregado de tomar as decisões em circuito aberto. No entanto, ele pode contar com o Plano conhecimento para facilitar esta tarefa. Através da aprendizagem supervisionada, o modelo de aprendizagem profunda em múltiplos passos pode ser usado para validação, onde é possível verificar antes de aplicar mudanças provisórias ao sistema. O modelo também pode ser usado como uma ferramenta para estimativa de desempenho e análise de variações hipotéticas, pois o operador pode ajustar as variáveis consideradas no modelo e obter uma avaliação do desempenho da rede.

4.3 CONSIDERAÇÕES FINAIS

Nesse capítulo, apresentamos um framework de escalonamento autônomo de VNFs tolerante a falhas, baseada em conjunto com os modelos de previsão LSTM com regras de afinidade, bem como na análise dos aspectos de performabilidade, desempenho e disponibilidade, dentro do contexto de um data center.

5 EXPERIMENTOS E AVALIAÇÕES

Este capítulo apresenta os experimentos e avaliações de escalonamento de VNFs, diagnóstico e replanejamento do NFV no data center. A tese foi implementada usando Scikit-Learn¹ e keras² para desenvolver a aprendizagem da máquina e modelos de aprendizagem profunda; para os testes estáticos foram utilizados: Orange³, Scikit-PostHocs⁴, ScipyStats⁵; Numpy⁶ para arrays e matriz multidimensional; Pandas, Matplotlib para análise de dados e gráficos, que são bibliotecas *open-source* na linguagem Python. O dataset está disponível em Github⁷

5.1 DESCRIÇÃO DA BASE DE DADOS

Foram realizados experimentos com um dataset de referência apresentado em (JACOBS et al., 2018), onde o consumo da CPU das VNFs do mundo real é medido quando se opera sob tráfego real. De forma precisa, o dataset NFV selecionado representa um dos 50 slots disponíveis pelo dataset Google com um tamanho original de 300GB (com 11,5 milhões de VNFs). A estrutura do dataset NFV é uma série temporal contendo 1694 FGs, 170510 VNFs, e 12478 máquina físicas distribuídas numa janela temporal de 55 horas. As principais características são tempo, VNF_ *id*, FG_ *id*, tempo de execução, SLA, CPU mínima, memória mínima, armazenamento mínimo, fluxo de tráfego, latência, uso de largura de banda, perda de pacotes. Neste trabalho, todos os pares VNFs possíveis no dataset NFV são considerados, sejam os hospedados na mesma máquina físicas ou encadeados na mesma FG, resultando em 1.302.524 pares de VNFs. Uma vez que as VNFs já estão pareadas, medidas estáticas e dinâmicas de afinidade são consideradas para treinar e avaliar o modelo LTSM.

A Tabela 2 mostra os parâmetros térmicos utilizados no módulo de diagnóstico. A informação contida nesta tabela reflete as características reais dos dispositivos físicos, tais como MTTF, MTTR do Google (GARRAGHAN et al., 2014), e mesmo valores térmicos referentes às mesmas características físicas do dispositivo (TRAVERS, 2015), (COUNCIL, 2002) (MILLER, 2008). Para realizar os experimentos, a série temporal do NFV dataset é dividido em três subconjuntos, usando 70% das amostras para treinar o modelo, 10% das amostras para fins de validação e 20% das amostras para realizar testes independentes. É essencial notar que os conjuntos de treinamento e testes são estatisticamente independentes,

¹ scikit-learn.org

² keras.io

³ orange.biolab.si

⁴ scikit-posthocs.readthedocs.io

⁵ docs.scipy.org

⁶ numpy.org

⁷ github.com/rafaroques/nfvdatacenter

e as amostras no conjunto de testes nunca são usadas para nenhuma das instâncias de treinamento. Naturalmente, ao realizar os experimentos de teste, todos os atributos do conjunto de testes foram levados em conta. Finalmente, a fim de comparar o desempenho do modelo de previsão proposto baseado em LSTM, bem como usando alguns outros modelos de aprendizagem amplamente conhecidos, também são realizados experimentos usando uma rede neural totalmente conectada e modelos de ReLU. Em particular, esses experimentos são realizados considerando tamanhos de camadas ocultas de [10, 5]. Todos os experimentos são conduzidos no Linux com CPU Intel Core i7-2600 em 3.40 GHz.

Tabela 2 – Parâmetros Térmicos Utilizados

Símbolo	Parâmetros	Magnitude
T_{∞}	Temperatura Ambiente (°C)	26.7
I	Corrente Elétrica (A)	6
V	Voltage (V)	1
α	Fator de Ativação	0.1
C_p	Capacitância (μ F)	0.1
h	Coeficiente Convectivo (W/m ² K)	50
A_s	Área de superfície da prancha Mãe (m ²)	$60 \cdot 10^{-4}$
m	Massa(kg)	$50 \cdot 10^{-3}$
C	Calor Específico (J/kgK)	900
f	Frequência (MHz)	samples
t	Time (Hours)	43

5.2 NORMALIZAÇÃO DE DADOS

Como os dados da amostra estão dispersos e o intervalo da amostra é grande, os dados precisam ser normalizados, a fim de melhorar a precisão do modelo de previsão e suavizar o procedimento de treinamento. Neste trabalho, toda a amostra de dados é normalizada para que todos eles se encaixem no intervalo [0, 1], recorrendo à seguinte fórmula de mapeamento linear:

$$X' = \frac{x_i - x_{min}}{(x_{max} - x_{min})}, i = 1, 2, 3...n \quad (5.1)$$

onde X' é o valor mapeado, x é o valor inicial dos dados experimentais, N denota o número total, x_i são os dados de entrada, e x_{max} and x_{min} indicar os valores máximo e mínimo dos dados iniciais, respectivamente.

5.3 VALIDAÇÃO DO MODELO

A fim de avaliar a capacidade de previsão do modelo LSTM proposto, os parâmetros do modelo são calibrados no subconjunto de dados previamente separado para fins de validação (10% de todas as amostras disponíveis no dataset). Uma vez que o modelo e seus parâmetros já tenham sido calibrados, seu desempenho é testado em um conjunto de testes estatisticamente independente. Em particular, o modelo de previsão proposto é avaliado considerando 50 épocas com um tamanho de lote 1, uma função de ativação ReLU. Utilizamos um otimizador durante o treinamento para ajustar a taxa de aprendizagem definida para 0,01 e o peso-dia para $9e-5$, pois é computacionalmente eficiente (pequena necessidade de memória) e adequado para otimizar os problemas que são grandes em termos de dados e para resolver problemas com ruído denso (ou gradientes esparsos) (KINGMA; BA, 2014). Para o problema da regressão, e previsão de série temporal, não há uma métrica única universalmente adotada pelos pesquisadores para avaliar a adequação preditiva do modelo.

Então, nesta tese, diferentes métricas são empregadas para medir o desempenho do modelo LSTM proposto. Exatamente, sete métricas são consideradas para permitir uma melhor apreciação do desempenho do sistema de previsão. A metodologia experimental é a seguinte (NETO et al., 2015): o *Absolute Percentage Error* (APE), *Mean Squared Error* (MSE), *Mean Absolute Error* (MAE), *Average Relative Variance* (ARV), *Theil's U statistic* (Theil's) e *Prediction of Change in Direction* (POCID). Para todas as medidas descritas abaixo, N é o número total de observações, a saída é a previsão do modelo, a meta é o valor atual da série, e i correspondendo ao ponto na amostra.

- Para a primeira iteração i , o Erro de Predição (PE_i) é calculado da seguinte forma:

$$PE_i = \frac{1}{N} \sum_{j=1}^N (target_j - output_j), \quad (5.2)$$

onde N é o número de amostras, $target_j$ é o valor real (verdadeiro) VNF da j th amostra de teste, e o $output_j$ é o valor previsto. O APE é utilizada como métrica principal para comparar as diferentes técnicas de previsão apresentadas.

$$APE = \frac{1}{100} \sum_{i=1}^{100} PE_i. \quad (5.3)$$

- MAE é uma métrica de interpretação direta que calcula a diferença média absoluta entre duas sequências em um conjunto de previsões, independentemente de sua direção. Em outras palavras, é a média na amostra de teste das diferenças absolutas

entre a previsão e a observação real. As diferenças individuais têm o mesmo peso em todos os casos. O MAE é definido como:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_j - \hat{y}_j|. \quad (5.4)$$

- *Mean Absolute Percentage Error* (MAPE) é definido como:

$$MAPE = \frac{1}{N} \sum_{j=1}^N \left| \frac{target_j - output_j}{target_j} \right| \quad (5.5)$$

- Outra medida de avaliação relevante é dada pela ARV, definido como:

$$ARV = \frac{1}{N} \frac{\sum_{j=1}^N (output_j - target_j)^2}{\sum_{j=1}^N (output_j - \overline{target})^2} \quad (5.6)$$

O ARV compara o desempenho do modelo com a média temporal da série, que é representada pelo termo \overline{target} . Se $ARV = 1$, o desempenho do modelo é semelhante à média da série temporal, se $ARV > 1$ o desempenho do modelo é pior do que a média da série temporal, e se $ARV < 1$ o desempenho do modelo é melhor do que a média da série temporal.

- Theil's, compara o desempenho do modelo preditivo com o modelo *Random Walks* (RW). Um modelo RW assume que o valor justo para a previsão no momento $t + 1$ é o valor obtido no momento, mais um termo de ruído. Se o valor Theil's é inferior a 1, então o modelo é melhor que o RW; caso contrário, o desempenho do modelo é igual ou pior que o do RW (valor Theil's igual ou superior a 1, respectivamente). O Theil's é definido como:

$$Theil's = \frac{\sum_{j=1}^N (target_j - output_j)^2}{\sum_{j=1}^N (output_j - output_{j-1})^2} \quad (5.7)$$

- POCID mapeia a tendência da série temporal. Ou seja, determina se a capacidade do modelo de prever valores futuros irá aumentar ou diminuir. A medida é calculada da seguinte forma:

$$POCID = 100 \frac{\sum_{j=1}^N D_j}{N}, \quad (5.8)$$

onde:

$$D_j = \begin{cases} 1 & \text{if } (target_j - target_{j-1}) \\ & (output_j - output_{j-1}) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (5.9)$$

Em específico, a função de aptidão considera o MSE. No entanto, o algoritmo também é avaliado em termos das outras seis métricas aqui discutidas. Em um caso ideal, todas as medidas de desempenho devem se aproximar de zero, exceto o POCID, que deve se aproximar de 100.

5.4 AVALIAÇÃO DE DESEMPENHO

Nesta seção, é realizada a avaliação de desempenho do modelo proposto de escalonamento VNFs baseado em LTSM tolerante a falhas. Em particular, é apresentada a precisão de cada um dos modelos de previsão para previsões multi-passo adiante, a fim de analisar sua capacidade preditiva. A fim de responder as perguntas de QP1 a QP3, os detalhes das pesquisas relacionadas a cada grupo serão discutidos em seções separadas.

5.4.1 Análise de Acurácia de Previsões

Esta seção tem como objetivo responder à **QP1: Qual modelo e estratégia de aprendizagem profunda mais adequada para realizar o escalonamento de VNFs?**. Esta avaliação de desempenho utiliza dois testes estatísticos não paramétricos e considera dois estudos básicos que utilizaram a ANN. Portanto, adotamos a mesma configuração do modelo de redes neurais (ANN) em (JACOBS et al., 2018), trabalho que foi a base para o desenvolvimento dos nossos modelos de afinidade e as mesmas configurações também do trabalho (MESTRES et al., 2017), trabalho que foi a base da KDN. No entanto, os trabalhos não abordaram o escalonamento, mas utilizamos o mesmo para comparar com os últimos modelos de aprendizagem profunda para os dados das séries temporais. A Figura 21 ilustra o avaliação de desempenho do modelos preditivos ANN, GRU e LSTM considerados no dataset⁸. Em particular, o mapa de calor na Figura 21 permite identificar o modelo em que um método (linha) tem um desempenho significativamente melhor que outro (coluna).

Em particular, o teste Conover foi utilizado para comparar as taxas de erro correspondentes de todas as combinações do modelos preditivos. Aqui, vale ressaltar que o teste Conover é um teste estático não paramétrico e também foi usado para comparar a precisão dos modelos (CONOVER; IMAN, 1979). A significância estatística foi determinada por meio de um teste de p -value limiar (*i.e.*, 0.05 dividido pelo número de testes). As diferentes cores no mapa de calor apresentado na Figura 21, representam até que ponto a diferença de desempenho entre os modelos comparados é estatisticamente significativa. Em particular, diferentes tons verdes representam diferentes níveis de significância estatística, desde $p < 0.05$ para $p < 0.001$, enquanto o vermelho representa casos não significativos. A partir da Figura 21, pode-se ver que o modelo proposto baseado em LTSM tem um desempenho significativamente melhor que outros modelos de aprendizagem de máquinas propostos na literatura, tais como ANN e GRU que, de acordo com a Figura 21, não

⁸ github.com/rafaroques/nfvdatacenter

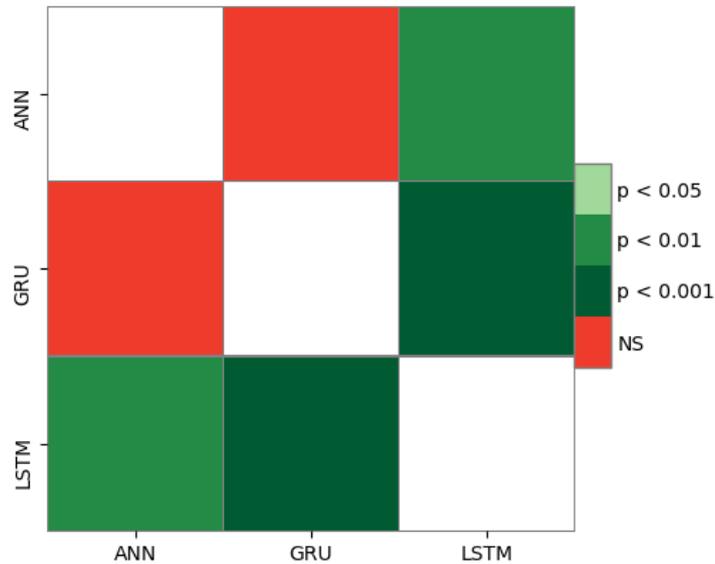


Figura 21 – Avaliação baseado no Teste Conover

são estatisticamente diferentes uns dos outros. Além disso, é importante destacar que a superioridade do modelo LTSM proposto é altamente significativa estatisticamente, sendo $p < 0.001$.

Figura 22 apresenta o teste Friedman para os métodos considerados. O teste de Friedman é um teste não paramétrico, utilizado com um nível de significância $\alpha = 0.05$ (DEMŠAR, 2006). Portanto, se a hipótese nula - que afirma que não há diferença estatística entre os métodos - for rejeitada, então o teste Nemenyi com nível de significância $\alpha = 0.05$ é usado. A razão para escolher tais testes é que o teste Friedman classifica os algoritmos de acordo com seu desempenho para o dataset, enquanto o teste Nemenyi compara todos os algoritmos uns com os outros. Assim, usando esses testes, é possível identificar se a diferença entre os métodos é estatisticamente significativa ou não.

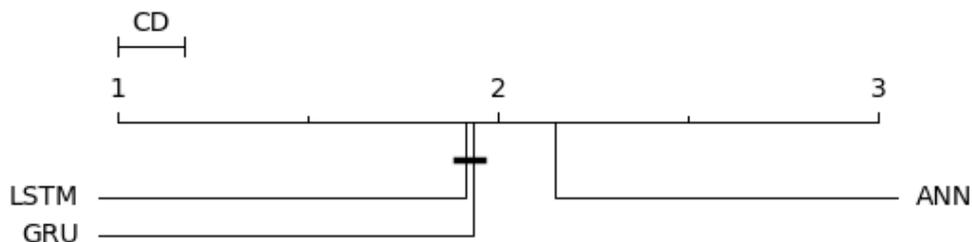


Figura 22 – Avaliação baseado no Teste de Friedman

Para isso, a Figura 22 mostra a classificação dos métodos (começando da esquerda para a direita). O tamanho da *Critical Distance* (CD), que mede a diferença entre os algoritmos, é mostrado no início da classificação (esquerda). Os algoritmos serão então estatisticamente

equivalentes e, portanto, comparáveis se uma linha horizontal os interligar. A partir da Figura 22, pode-se então concluir que os modelos LSTM e GRU são estatisticamente equivalentes. Nesta linha, levando em conta os resultados mostrados na Figura 22, a LSTM proposta e os modelos GRU são estatisticamente comparáveis, no sentido em que a sua comparação tem valor estatístico, tendo em conta os resultados apresentados na Figura 21, esta comparação estatisticamente significativa mostra que o modelo LSTM proposto tem um desempenho superior ao modelo GRU com $p < 0.001$.

5.4.2 LSTM com Previsão Multi-passo à Frente

Esta seção tem como objetivo responder à **QP2: Como criar uma estratégia inteligente para o escalonamento de recursos de rede considerando aspectos de um data center tolerante a falhas em cenário NFV?**. Um dos principais objetivos dos experimentos conduzidas nesta seção é avaliar até que ponto o modelo desenvolvido pode prever o futuro uso da afinidade, bem como avaliar até que ponto a acurácia da previsão diminui. Uma vez que, como mostrado na Subseção 5.4.1, o modelo LSTM proposto mostrou o melhor desempenho de previsão, este foi o modelo implementado para prever o uso de afinidade no escalonamento para várias passos(ou unidades de tempo) no futuro para o experimento conduzido nessa seção. Em particular, o experimento conduzido nesta seção envolve a previsão do uso do escalonamento em 6 passos no futuro, cada um deles correspondendo a 1 minuto.

Tabela 3 mostra o desempenho da abordagem proposta do modelo LSTM para o conjunto de teste em termos de APE e MAE apresentaram os erros próximo de zero, o que mostram ser um ótimo resultado. MAPE mostrou que obtivemos um bom resultado, visto que teve uma pequena variação no erro percentual a medida que se vai prevendo o futuro em passos a frente. ARV mostrou que o desempenho do modelo é melhor do que a média da série temporal. Theil's mostoru que o modelo é melhor que o modelo random walk e com POCID identificamos que LSTM é mais acurados com relação a previsão de tendência da série. O modelo LSTM, foi o que obteve os melhores resultados como apresentado na Seção 5.4.1, assim, as métrica são em função do modelo LSTM.

Tabela 3 – Classificação do Modelo LSTM.

N° de Passos	APE	MAE	MAPE	ARV	U	POCID
1 Passos à frente	0.035	0.526	52.59	0.272	145.9	47.98
2 Passos à frente	0.036	0.528	52.77	0.274	196.1	50.64
3 Passos à frente	0.037	0.529	52.80	0.2903	244.8	53.65
4 Passos à frente	0.038	0.531	52.95	0.300	255.5	56.81
5 Passos à frente	0.039	0.533	52.98	0.338	316.01	59.82
6 Passos à frente	0.04	0.535	53.00	0.374	386.23	62.83

Na Tabela 3, as seguintes observações podem ser extraídas. Como é razoável esperar, a acurácia das previsões futuras diminui constantemente, no sentido de que quanto mais no futuro, o modelo tentar prever, menor será a acurácia da previsão. No entanto, para a maioria dos erros mostrados na Tabela 3 foram baixos, como APE e MAE que pode-se observar bons resultados, pois mostram pequenos erros que estão próximo de zero. MAPE mostrou bons resultados, e permite uma visão sobre a diferença percentual entre os dados reais e predito, onde observa-se que variou muito pouco o erro percentual ao longo dos passos a frente de predição futura. O que significa que a cada minuto de predição a frente, teve-se pouco erro percentual, o que valida a eficiência do modelo. POCID apresentou bons resultados, visto que quanto maior o valor, melhor o desempenho do modelo em função da previsão de múltiplos passos a frente, assim, os erros estão aumentado pouco a cada passo a frente. Por outro lado, nos casos dos erros ARV e Theil's, o aumento do erro em cada passo à frente é elevado. . Esta tese demonstra a eficiência do modelo LTSM na resolução do problema de estimativa considerado neste trabalho.

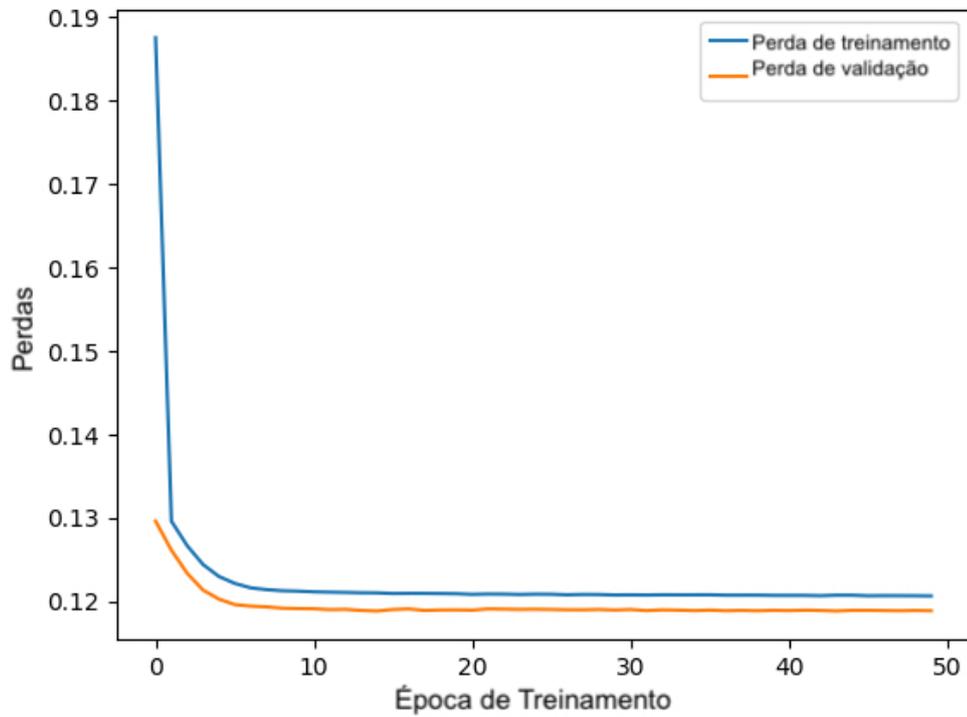
A Figura 23 mostra a perda através do processo de aprendizagem. Nossa configuração LTSM pode aprender com precisão como modelar o escalonamento de VNFs. Observamos que a perda de treinamento é maior do que a perda de validação, ao contrário da intuição, uma vez que a função de perda para validação não utiliza regularização, e o processo de aprendizagem utiliza três vezes mais dados, resultando em maiores valores médios de perda. Após cada lote de treinamento, os pesos de rede são ajustados e a perda média de todos os lotes é relacionada. O LTSM pode aprender, em poucas épocas, uma representação precisa da normalidade, como pode ser percebido a partir da baixa perda nos dados de validação.

A Figura 24 mostra a demanda real de escalonamento de VNFs entre todas as máquina físicas. As linhas azuis exibem o tráfego real, e seu gráfico scatters, e a linha laranja representa a previsão nos 6 passos à frente nos dados de teste.

O objetivo deste gráfico é ilustrar em que pontos dos dados de afinidade no escalonamento de VNFs, a precisão da rede aumenta e diminui. Além disso, ele revela que os erros de previsão mais significativos ocorrem quando há grandes e instantâneas mudanças no escalonamento. Os valores esperados são muito próximos dos valores reais. Isso mostra que o modelo LTSM de treinado online é preciso o suficiente para o processo de escalonamento. O atraso de escalonamento de VNFs tradicional é o intervalo de tempo desde a tarefa enviada até o escalonamento e recebido. Em tal contexto, não é possível avaliar corretamente o desempenho do escalonamento se a tarefa for eliminada ou descartada antes da conclusão. Neste trabalho, o atraso na escalonamento de VNFs é computado como o intervalo de tempo desde a primeira tarefa enviada até a última antes de ser finalizada com sucesso.

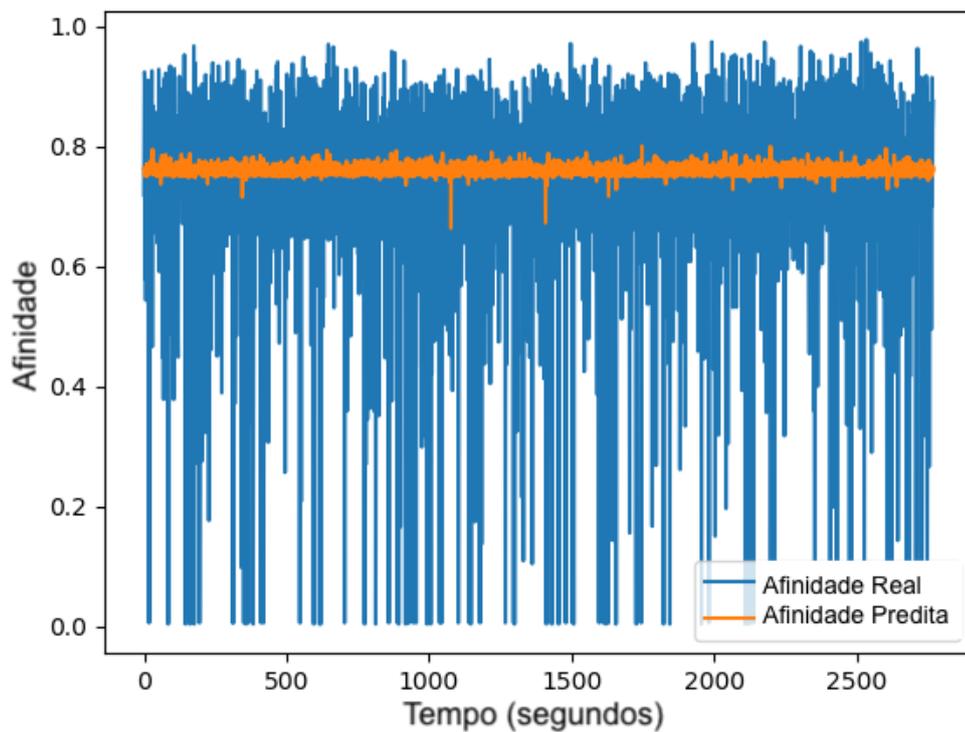
Figura 25 mostra um gráfico *Cumulative Distribution Function* (CDF) para o delay do escalonamento. Inicialmente, a fração de VNFs era muito alta, o que pode ser devido ao tráfego da rede, que é um processo rápido. Entretanto, mais adiante, a fração de VNFs aumentou com o tempo, o que pode ser atribuído a reações de tráfego de rede.

Figura 23 – Treinamento e validação durante o processo de aprendizagem de um LSTM.



Fonte: O Autor

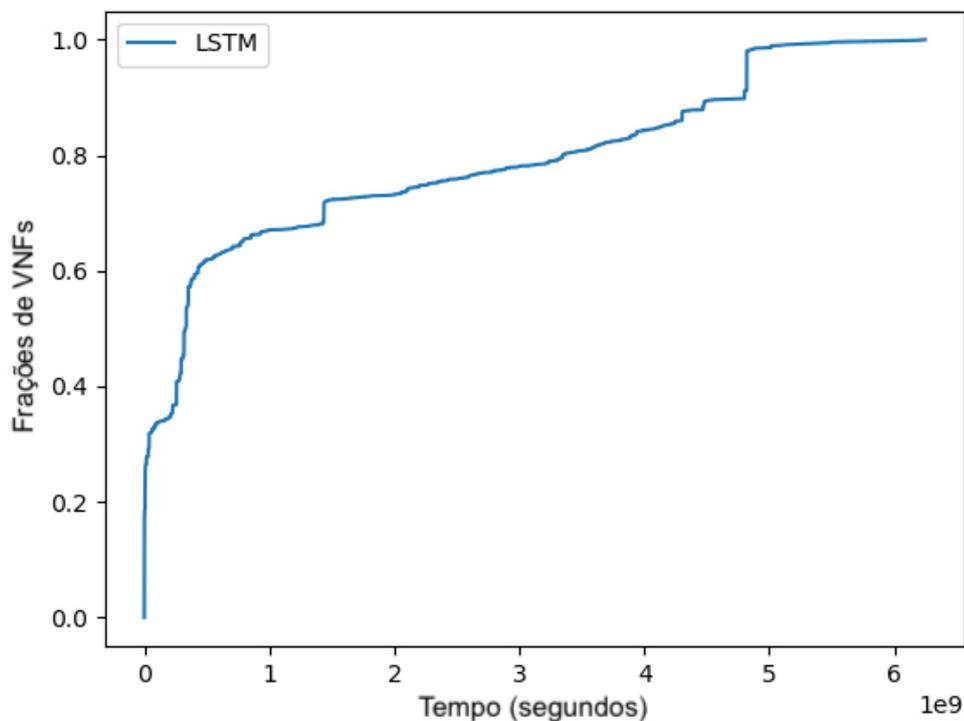
Figura 24 – Predição de Escalonamento de VNFs com Afinidade



Fonte: O Autor

Como mostra o CDF, a metodologia proposta funciona notavelmente bem, partindo de baixos valores de atraso que estão significativamente relacionados ao tráfego extremo. Isto é causado pelo melhor desempenho do módulo LSTM para detecção de tráfego de alto impacto. Obtivemos duas observações nesta experiência. Primeiro, o atraso total do escalonamento normalizado diminui com o aumento do tamanho do trabalho até 0,5 dos 95% de distribuição cumulativa. É porque um maior tamanho de job leva a um maior tempo de execução. Entretanto, o atraso aumenta quando a fração de VNF excede 80% e quase 6 segundos no pior dos casos. Em segundo lugar, observamos uma tendência de que um maior tamanho de entrada leva a um atraso mais significativo no escalonamento total.

Figura 25 – Delay no Escalonamento de VNFs



Fonte: O Autor

5.4.3 Ferramenta de Diagnóstico

Esta seção visa responder à **QP3: Qual é o conjunto de métricas representativas que podem quantificar os impactos na degradação do desempenho do escalonamento de VNFs e servir como input para o operador de rede ajustar o modelo de aprendizagem profunda?**, em que avaliamos o impacto da alocação de recursos com base na performabilidade. A performabilidade é calculada com base no consumo da CPU, o que é um aspecto crítico uma vez que, à medida que os níveis de temperatura aumentam, a disponibilidade diminui.

5.4.3.1 Validação do Impacto da Temperatura

A presença de altas temperaturas no data center pode levar a graves problemas de hardware. Neste contexto, é crucial estar atento ao aumento de temperatura induzido ao consumo da CPU devido ao escalonamento de VNFs. Nesta seção, são apresentados resultados de validação mostrando este aumento de temperatura com base em dados reais. Por este fim, são analisados diferentes gráficos de correlação comparando a temperatura sobre o escalonamento de VNFs ao longo do tempo. Para validar estes resultados, utilizamos um dataset representando a temperatura real no cenário real de NFV aplicado em data center. A Figura 26 mostra os limites e variações de altas temperaturas no data center ao longo do tempo, devido à escalonamento de VNFs.

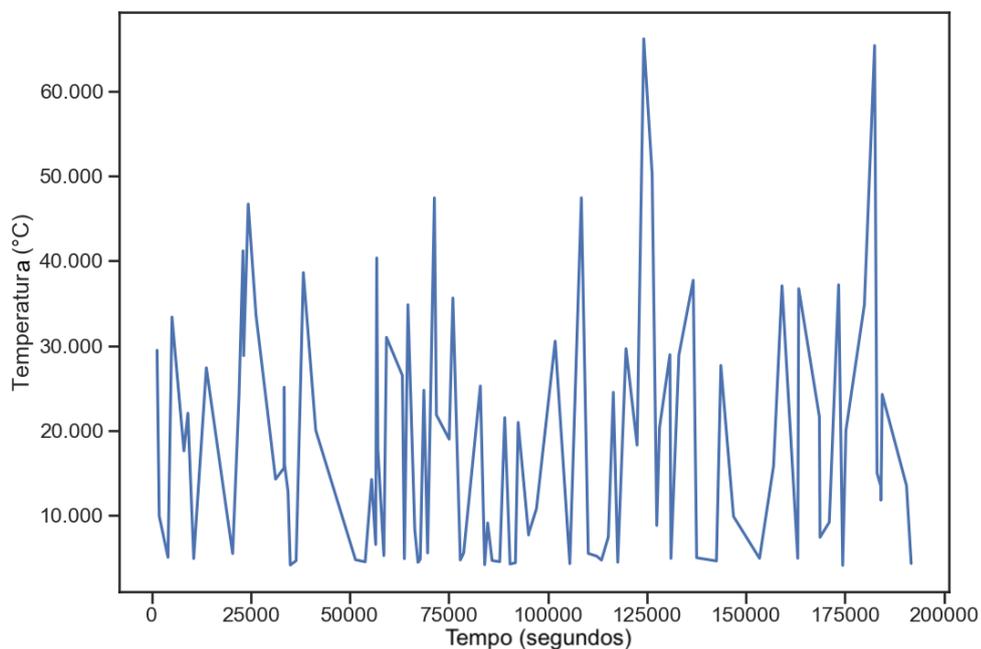


Figura 26 – Variação de Temperatura ao longo Tempo

Em particular, os resultados em Figura 26 mostram que o escalonamento de VNFs tem um impacto forte e direto na variação de PM, uma vez que a temperatura varia continuamente ao longo do tempo. O tráfego de rede não produz necessariamente uma distribuição uniforme da temperatura. Enquanto a temperatura é desequilibrada, pontos quentes de magnitude e gravidade diferentes são visíveis nas extremidades das linhas, acima de 30 graus Celsius. A temperatura máxima no data center era de 60 graus Celsius. Além disso, conforme o consumo de energia aumenta, a dissipação de calor aumenta e, conseqüentemente, a temperatura, em função da escalonamento de VNFs, aumenta.

Figura 27 mostra o aumento da temperatura de máquina físicas como consequência do referido aumento do consumo de energia. Para este fim, afirmamos que os desequilíbrios térmicos interferem na operação do sistema de refrigeração, e os pontos quentes criam o risco dos servidores excederem a temperatura máxima de entrada de ar especificada, danificando os componentes eletrônicos e causando a sua falha prematura. O operador

de rede, pode prever a carga térmica monitorando o uso do servidor e os componentes de armazenamento. Os resultados na Figura 27 mostra uma relação altamente linear entre o aumento da temperatura e a dissipação de energia. A temperatura das CPUs está diretamente relacionada com a velocidade de processamento, pois quanto maior for a velocidade de processamento, maior será a dissipação da energia utilizada e, posteriormente, mais calor é gerado.

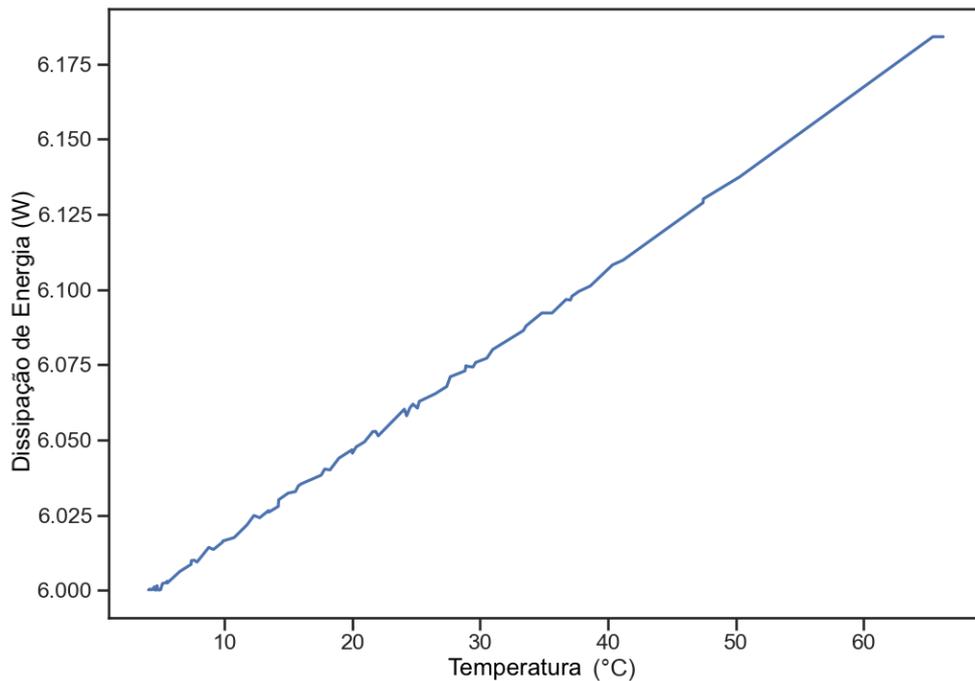


Figura 27 – Aumento de Temperatura Devido à Dissipação de Energia

A Figura 28 mostra a degradação do desempenho máquina físicas em função do aumento da temperatura devido à escalonamento de VNFs. Esta forte dependência evidencia o impacto da temperatura na degradação do desempenho em data center.

Assim, na Figura 28 pode ser observado que existe uma diminuição contínua do desempenho à medida que a temperatura aumenta, refletindo, como esperado, uma relação linear negativa entre a performabilidade e a temperatura. Finalmente, a análise dos aspectos de temperatura apresentados, permite tirar novas conclusões, bem como confirmar o aumento da temperatura tem forte (e negativo) impacto no desempenho do sistema, reduzindo a performabilidade em função do aumento da temperatura.

5.4.3.2 Avaliação de Performabilidade

Nesta seção, é estudado um novo modelo de performabilidade, que foi analisado após o escalonamento de VNFs. A partir da construção de ferramenta de diagnostico adicionamos os modelos de performabilidade que faz uma avaliação analítica. Na Figura 29 mostra a performabilidade do sistema em termos da taxa de falhas.

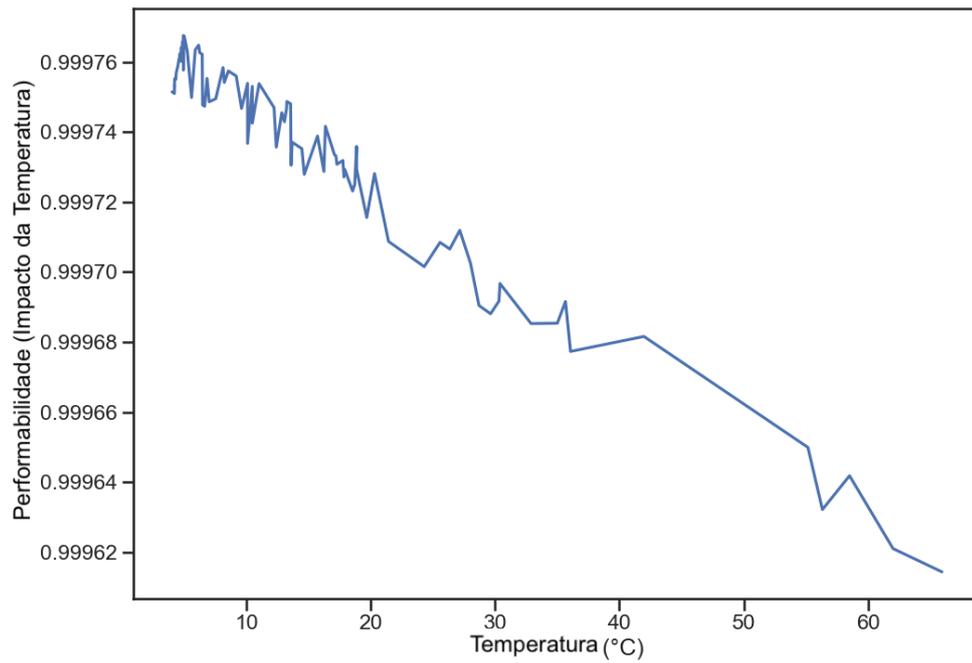


Figura 28 – Degradação do Desempenho em Relação a Temperatura

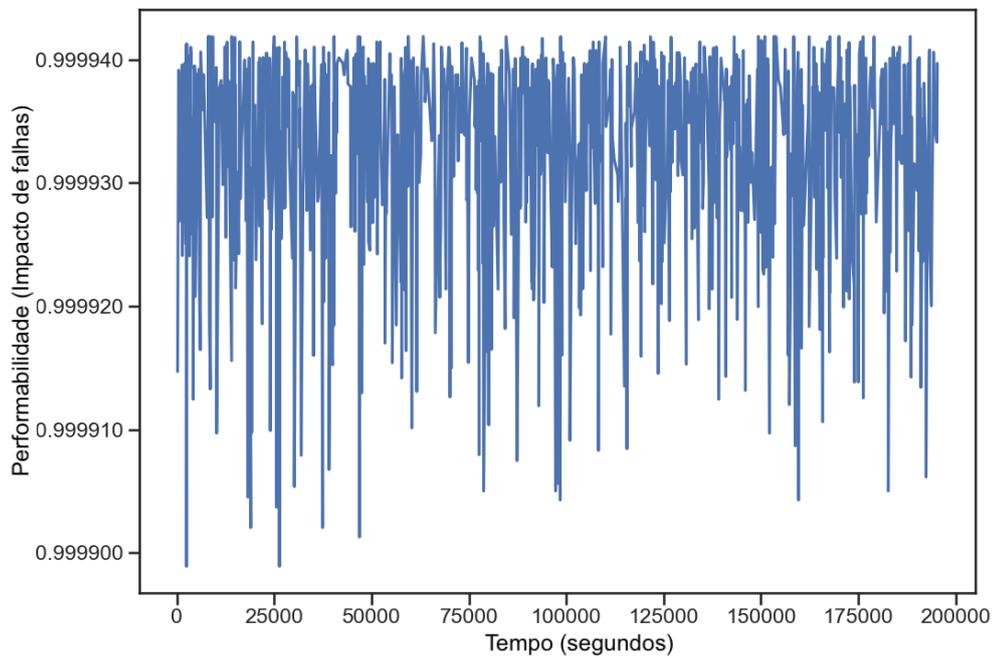


Figura 29 – Performabilidade baseada na Taxa de Falhas

Analisando os resultados após o escalonamento inteligente, pode-se ver que a ocorrência de eventos de falha foram poucas, a partir dos momentos em que se tem redução de desempenho em função de falha, o que resulta na degradação do desempenho em termos da performabilidade do sistema. Por outro lado, a performabilidade permanece aproximadamente 0.9999 em todos os intervalos de avaliação ao longo de toda a janela da série temporal. Neste sentido, embora a ocorrência de eventos de falha resulta na variação da performabilidade do sistema, o modelo de previsão proposto detecta estes eventos de falha, sendo a performabilidade média de 0.9999.

Finalmente, Figura 30 mostra que mudanças significativas na temperatura relativa ao tempo resultam não apenas na performabilidade do sistema, como já discutido na Subseção 5.4.3.1. Então, os resultados obtidos neste job podem ser usados para identificar possíveis causas de performabilidade, já que há um grande número de componentes de NFV do data center, podemos esperar uma taxa de falha significativa, mesmo que os tempos médios de falha dos componentes individuais sejam altos. Assim, negligenciar o impacto das falhas nos estudos de desempenho de tais sistemas pode levar a resultados enganosos. Por exemplo, um NFV pode tolerar várias falhas de NFV, mas o desempenho geral será afetado, uma vez que alguns NFV podem estar sujeitos a despesas gerais. Da mesma forma, falhas em servidores em ambientes de data center podem diminuir o número de recursos de processamento disponíveis e, portanto, aumentar os tempos de fila e diminuir a capacidade de job.

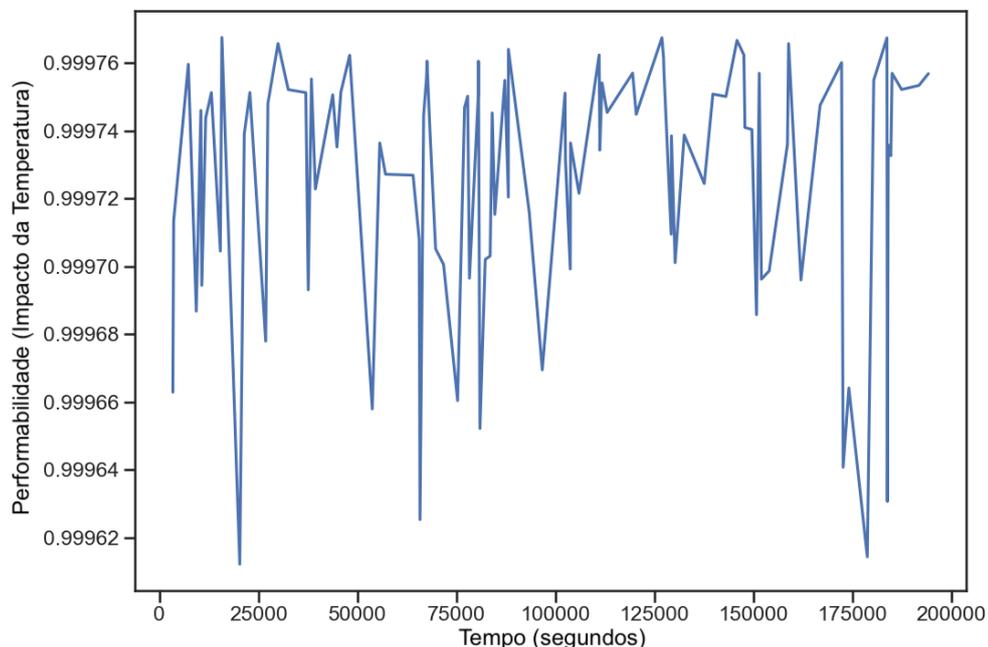


Figura 30 – Performabilidade baseada no Impacto da Temperatura.

Portanto, como validado na (Seção 5.4.3.1), a falha da máquina ocorre em uma situação real em função do impacto da variação de temperatura. Evitar pontos únicos de falha sempre que possível é difícil, e os modelos de performabilidade permitem obter degradação

no desempenho devido a mau planejamento de NFV no data center e detectar gargalos na escalonamento de VNFs, além de evitar falhas devido a sobrecarga de recursos, bem como devido a altas temperaturas.

5.4.3.3 Análise Exploratória de Dados

O EDA foi adicionado na nova ferramenta de diagnostico de rede, para permite a análise de dataset reais, após o escalonamento de VNFs. Para tal, tem-se uma modelagem explicativa e preditiva, resumindo os dados numéricos e gráficos do estado atual ou futuro do data center. Ao resumir e contabilizar os dados, o EDA pode prontamente fornecer informações úteis, encontrar padrões e descobrir relações gerais que podem orientar análises posteriores e alavancar seus resultados. A abordagem clássica da EDA , orientada por dados, faz uso de técnicas de investigação de dados para buscar novas informações e relacionamentos, a partir de uma abordagem ativamente incisiva com ênfase real na descoberta do inesperado. A EDA isola padrões e características dos dados. Revela-os ao analista sem a necessidade de conhecimento prévio ou hipóteses pré-especificadas, ou seja, sem exigir que perguntas bem definidas e antecipadas sejam dirigidas aos dados, onde a EDA não pode levar a conclusões definitivas, mas é um primeiro passo essencial para a compreensão dos dados.

Os métodos primários da EDA tentam tornar os dados mais acessíveis e mais eficazmente geridos pelo utilizador, sejam eles estatísticos ou não estatísticos. Na prática, para alcançar seus objetivos, a EDA combina métodos quantitativos e qualitativos da abordagem estatística clássica com a análise gráfica. O modelo de previsão proposto auxilia os gestores/operadores da rede na tomada de decisões para melhor calibrar o modelo de escalonamento de VNFs. A ideia é permitir que o operador de rede realize pesquisas iniciais de dados para entender a relação entre as variáveis, descobrir padrões, identificar anomalias, testar hipóteses e verificar suposições, com o auxílio de estatísticas resumidas e representações gráficas analisando diferentes métricas críticas, tais como as apresentadas nesta seção. Assim, a Figura 31 permite analisar a correlação de diferentes características que associam e conectam o consumo da CPU e o MTTF.

Contribui para a compreensão do comportamento do escalonamento, e ajuda na localização de variáveis criticamente importantes. Na Figura 31, pode ser observado que a correlação entre o consumo da CPU destas características e o MTTF é não só prejudicial mas também uma forte correlação linear negativa. Além disso, a força da correlação é denotada na Figura 31 pelo fato de que os pontos do gráfico não estão dispersos. Os resultados são mostrados na Figura 31, onde a correlação entre a utilização da CPU e o MTTF é demonstrada, evidenciam a importância do estudo do impacto de ambas as variáveis no escalonamento do VNFs. A análise da tráfego é um dos requisitos mais críticos no provisionamento de recursos do data center. Este resultado pode ser usado para ajudar as operadoras de rede a melhorar as previsões. Se duas variáveis foram conhecidas no

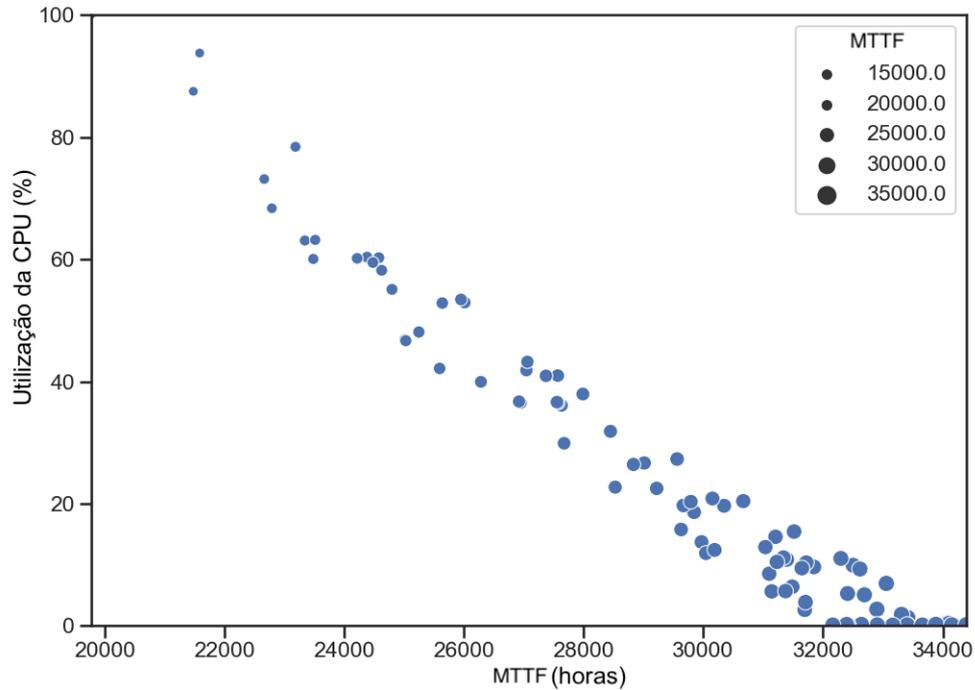


Figura 31 – Correlação entre o utilização de MTTF e CPU.

passado para correlacionar, então podemos assumir que elas continuarão a se correlacionar no futuro. Podemos usar o valor de uma variável que é conhecida agora para prever o valor que a outra variável assumirá no futuro.

A fim de ajudar o operador de rede a melhor planejar e operar o data center, a Figura 32 mostra a distribuição do consumo de recursos após a escalonamento. O recurso de consumo da CPU atinge valores de até 50% sob diferentes condições de tráfego na rede. Isso é crucial porque se a utilização da CPU fornece informações sobre vários problemas diferentes. Se uma VNF está consumindo o processador inteiro, há uma boa chance de que ela não esteja se comportando corretamente. Uma CPU operando no máximo de utilização da CPU é também um sinal de má alocação de recursos ou falha de dispositivo, que deve ser tratada imediatamente.

A Figura 33 mostra um histograma em que os recursos de consumo de memória atingem valores em torno de 15% e 20%, respectivamente. É importante ressaltar que há uma grande prevalência de afinidades no dataset e mostrar que a afinidade calculada para cada par de VNFs do dataset que foram colocados no mesmo máquina físicas ou encadeados no mesmo FG de acordo com o percentual definido.

Figuras 34 mostra o histograma em que o consumo de recursos de armazenamento atinge valores em torno de 1% e 18%, respectivamente. Estes valores são semelhantes aos definidos nas regras de afinidade, o que significa que, após a execução, o consumo de recursos permanece próximo do padrão.

Ao analisar os valores apresentados, a operadora de rede pode fazer novos ajustes para

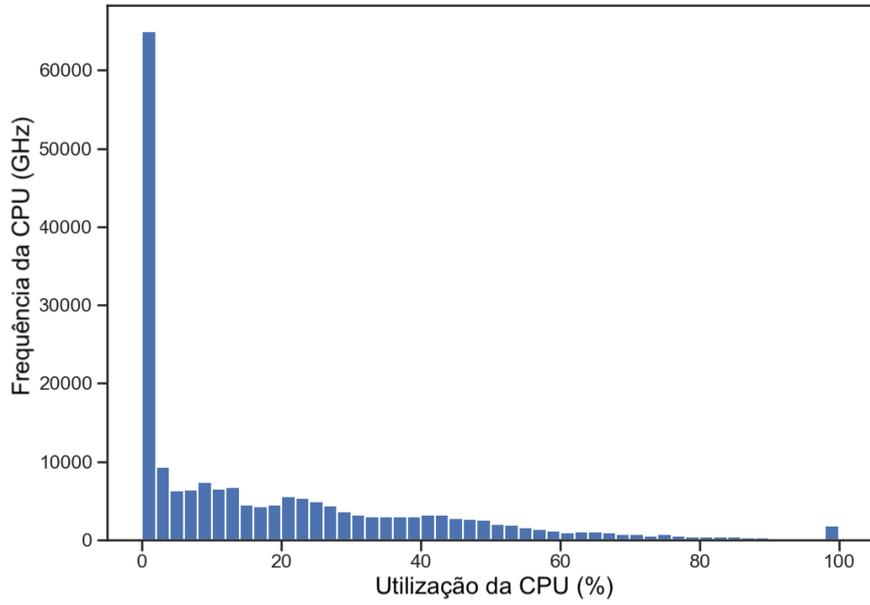


Figura 32 – Distribuição do consumo da CPU.

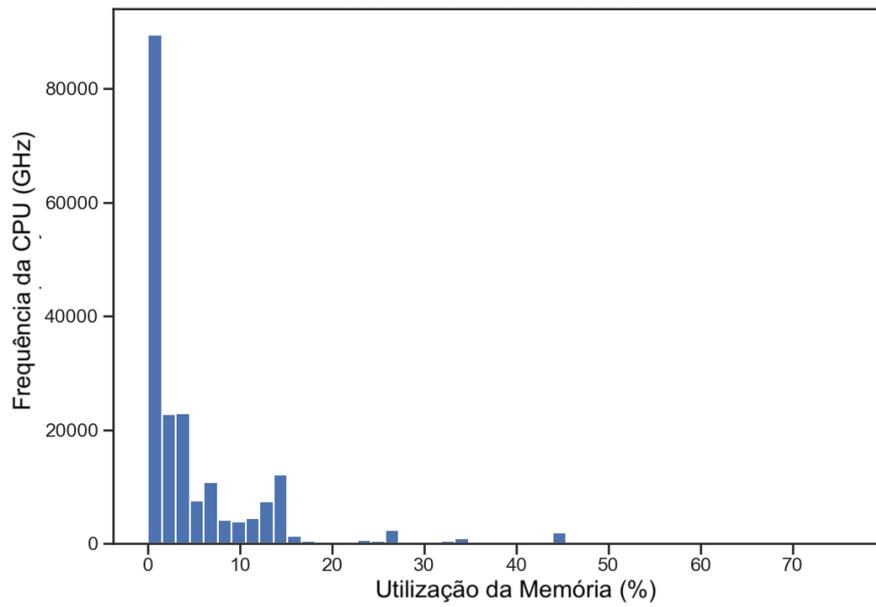


Figura 33 – Distribuição do Consumo de Memória.

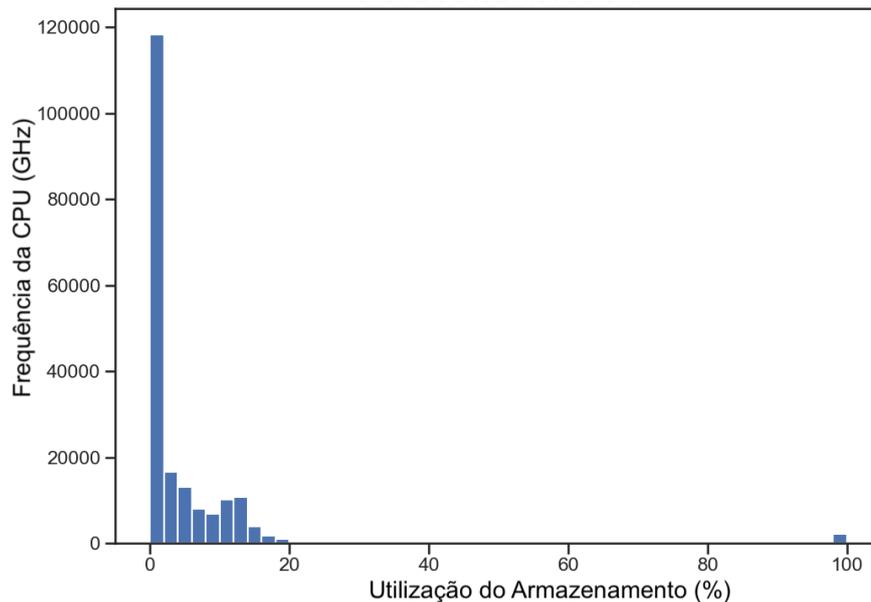


Figura 34 – Distribuição de Consumo de Armazenamento.

obter uma melhor estratégia de consumo de recursos, em termos de CPU, memória e armazenamento, para a escalonamento da VNFs. Para isso, a estratégia ideal é permitir o ajuste oportuno da afinidade de escalonamento de acordo com a demanda atual. Em particular, os resultados aqui apresentados são uma pré-condição para poder alocar a carga futura. É importante para o operador de rede porque pode utilizar o EDA para analisar dados após a realização do escalonamento VNFs para entender cinco princípios essenciais de análise de dataset: exibição, re-expressão, resíduos, resistência, iteração, agrupamento e técnicas de redução de dimensão, que ajudam a criar exibições gráficas de dados de alta dimensão contendo muitas variáveis; Univariadas, Bivariadas, Multivariadas para mapear e entender as interações entre os diferentes características do dataset.

5.4.4 Análise de Importância de Disponibilidade e Criticidade de Dispositivos

Esta seção tem como objetivo responder à **QP4: Como identificar ocorrências de falhas, e auxiliar o operador de rede para replanejar a redundância do dispositivo de maneira exata?**. Os aspectos da disponibilidade requerem uma atenção especial à qualidade do serviço da rede. A disponibilidade pode ser instantânea ou estacionária. A disponibilidade instantânea representa a probabilidade de a rede não falhar dentro de um determinado período de tempo. A disponibilidade em estado estacionário é a porcentagem de tempo que um dispositivo ou rede pode desempenhar a sua função (AVIZIENIS et al., 2004).

Os seguintes benefícios para o operador de rede no NFV do data center que utilizam as duas medidas são: (i) AI: é relevante no estabelecimento da direção e priorização de ações

relacionadas a um esforço de atualização (melhoria da disponibilidade) no projeto do NFV data center. AI também recomenda a maneira mais eficiente de operar e manter o status do sistema; (ii) CAI: prioriza uptime de melhoria de disponibilidade, identifica cadeias de VNFs fracas no sistema, probabilidade de falha de cada dispositivo e muitos outros usos.

Este experimento utilizou inicialmente, o AI de diferentes dispositivos de rede, construindo um ranking. Seguidamente, a análise crítica baseada em falhas, que foi testada para redes compostas de múltiplas cadeias de VNFs, onde pode ser usada para construir redundância em seu processo de implantação. Realizamos uma segunda avaliação para entender quais são os dispositivos mais críticos. A Equação 5.10 foi usada para calcular o AI dos dispositivos.

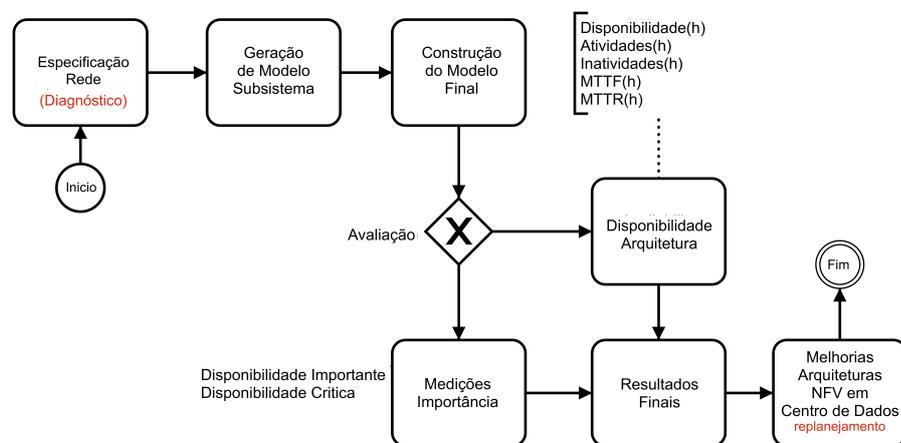
$$I_{ai}(i; p) = \frac{pi(1_i, p) - A(0_i, p))}{A(p)} \quad (5.10)$$

Os dispositivos de rede i ao operar, são indicados por $I_{ai}(i; p)$, é, portanto, definida como a probabilidade em que o dispositivo i funciona. O dispositivo i é crítico para o funcionamento da rede, uma vez que a rede está funcionando. Quando i falha, é denotado por $I_{cai}(i; p)$, sendo definido como a probabilidade do dispositivo i falhar. Quando a rede apresenta uma falha, ela é expressa pela Equação 5.11.

$$I_{cai}(i; p) = \frac{qi(1_i, p) - A(0_i, p)}{1 - A(p)} \quad (5.11)$$

A Figura 35 descreve a metodologia proposta de modelagem analítica do replanejamento do data center, através de uma ferramenta complementar de diagnóstico. Esta metodologia emprega uma abordagem integrada baseada na disponibilidade e na criticidade do NFV data center.

Figura 35 – Metodologia para Análise de Medições de Importância



Fonte: O Autor

As seis etapas da metodologia são descritas a seguir: (i) Especificação de Rede: Inicialmente, a rede foi definida como um conjunto de nós, switches, servidores, VM e VNF:

- A especificação pode ser gerada automaticamente por uma ferramenta, como um algoritmo inteligente ou manualmente fornecida por um operador de rede. Nesta etapa, é essencial esclarecer o problema a ser analisado.
- Geração de Modelos de Subsistemas: Nesta fase, são gerados modelos de sub-rede NFV em data center considerando seu modo operacional. Nesta fase, não há interações de dependência entre os dispositivos. O modelo fornece uma iteração lógica entre os componentes do sistema, definindo quais combinações ativas determinam a funcionalidade da rede por meio de modelos RBD.
- Construção do Modelo Final: inclui a geração de modelos de dependabilidade baseado em medições de importância para definir qual dispositivo deverá ser feito a redundância;
- Avaliação da Disponibilidade da Arquitetura e Medições de Importância: nesta fase, é quantificado o impacto da redundância de dispositivos na disponibilidade do data center. Além disso, são identificados os dispositivos mais importantes;
- Resultados Finais: são apresentados os resultados de disponibilidade(h), uptime(h), downtime(h), MTTF(h), MTTR(h), disponibilidade importante (h) e disponibilidade crítica(h) para auxiliar no redesenho da arquitetura de NFV em data center.
- Melhorar o NFV no Data Center: os novos componentes são adicionados ao NFV em data center, baseado nas medições de importância para melhorar a disponibilidade.

Assim, é possível saber exatamente quais são as importâncias e os componentes críticos na operação. A Tabela 4 detalha os valores de MTTF e MTTR para cada dispositivo no NFV no data center (GILL; JAIN; NAGAPPAN, 2011b; POTHARAJU; JAIN, 2013; FERNANDES et al., 2012). O tempo necessário para realizar a análise de disponibilidade foi de 8760 horas (1 ano), o que foi calculado para o estado estacionário.

Uma pequena topologia fat-tree (ver Figura 37) é considerada para interconectar o NFV em data center. A arquitetura de baseline (A1) consiste em dispositivos físicos conectados em sequência por 1 Core, 1 Aggregation, 1 Tor switch, 2 Servidores. Os dispositivos virtuais, são Orquestrador, VNFM, VIM, 3 SFC são SFP₁ (Firewall -> Balanceamento de carga, Gateway), SFP₂ (Cache -> Firewall -> IDS), SFP₃ DPI, Proxy, *Network Address Translation* (NAT). Cada servidor consiste em um CPU, HD, NIC, Memória, e VM. O

Tabela 4 – Parâmetros de Entrada do RBD

Dispositivos	MTTF _{horas}	MTTR _{horas}
CPU	2500000	1
<i>Hard Disk</i> (HD)	2000000	1
NIC	6200000	1
Memória RAM	48000	1
VM	2880	2
VNF, VIM, VNF Manager, Orquestrador	2893	0,25
TorSwitch	175200.0	2.9
Agregation	87600.0	2.1
Core Switch	600000.0	2.1
enlace	19996	12

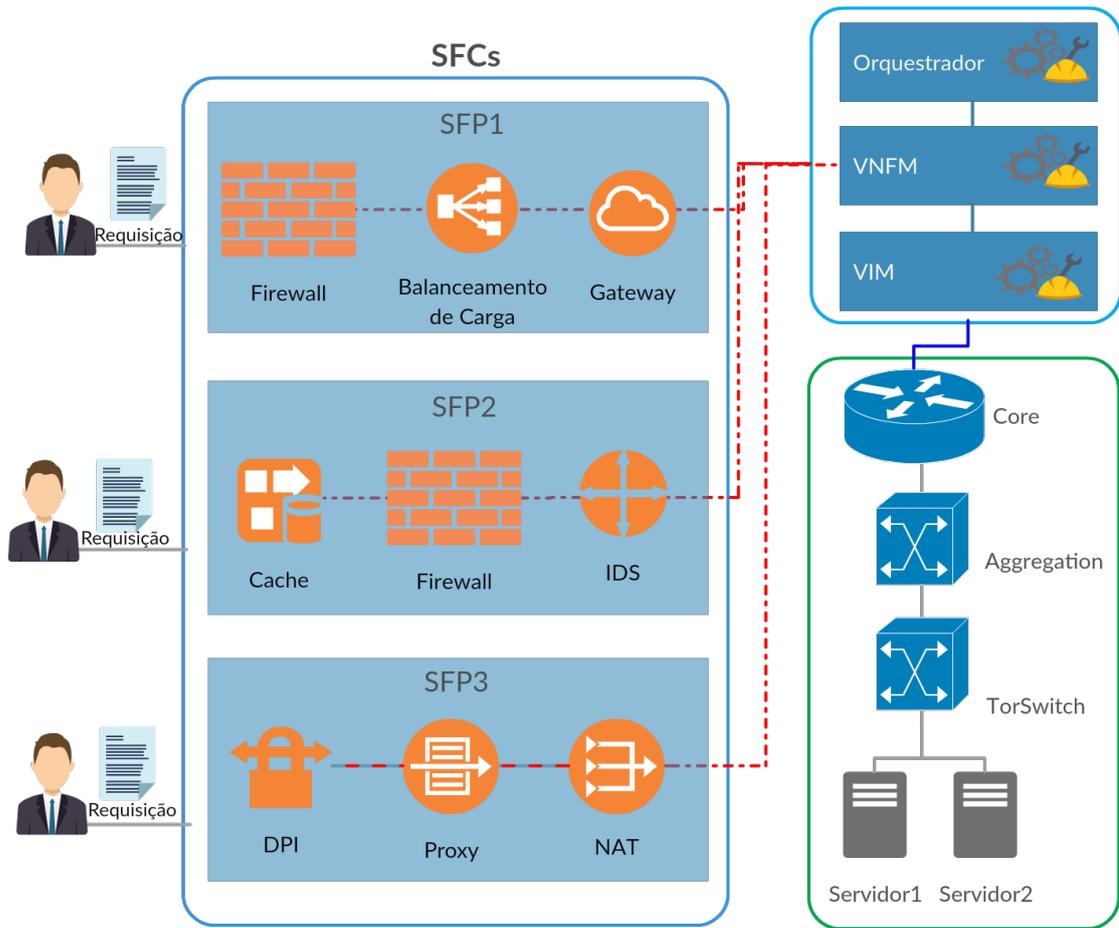
próximo objetivo é extrair o MTTF e o MTTR dos servidores utilizando a seguinte Equação 5.12.

$$\lambda\mu_{cpu} * \lambda\mu_{hd} * \lambda\mu_{memory} * \lambda\mu_{nic} * \lambda\mu_{vm} \quad (5.12)$$

onde o λ é o MTTF e μ representa o MTTR. A falha de um componente (CPU ou memória ou HD ou NIC ou VIM) em executar suas funções necessárias pode levar um servidor a falhar e, portanto, negar a requisição da VNF. Primeiro, definimos a estrutura fixa de NFV composta por quatro SFC independentes que compartilham os mesmos recursos físicos. Assumimos que os SFC são geridos por um orquestrador, um gestor de VNF e um VIM. Uma rede completa de data center consiste em geração de tráfego, protocolos de rede, arquitetura de energia e resfriamento. Consideramos a arquitetura de energia e arquitetura de refrigeração confiáveis. Consequentemente, não serão considerados nos modelos de disponibilidade, nem serão considerados pelos protocolos de tráfego e rede. Figura 37 mostra uma pequena arquitetura que definimos para servidor de base para construção do modelo RBD que representa o NFV no data center.

É importante mencionar que esta arquitetura foi modelada usando a ferramenta Mercury (SILVA et al., 2015). No entanto, trata-se de demonstrar uma nova aplicação dos modelos de medida de Birnbaum com foco na análise crítica de falhas dos dispositivos (SOUZA; SANTOS; FERNANDES, 2018). Em trabalhos futuros, iremos incorporar esta abordagem na ferramenta de diagnóstico para computar em grandes dataset, visto que a ferramenta de diagnóstico desenvolvida não sofre do problema de explosão de espaço de estado. Atualmente, está acoplada como uma ferramenta externa. Então, um modelo analítico, semelhante ao cenário real, foi feito para extrair a equação das fórmulas fechadas para a disponibilidade computacional. Esta equação recebe um novo MTTF, baseado na variação da temperatura e calcula a disponibilidade impactada. A equação de fórmula fechada

Figura 36 – Arquitetura para o modelo RBD

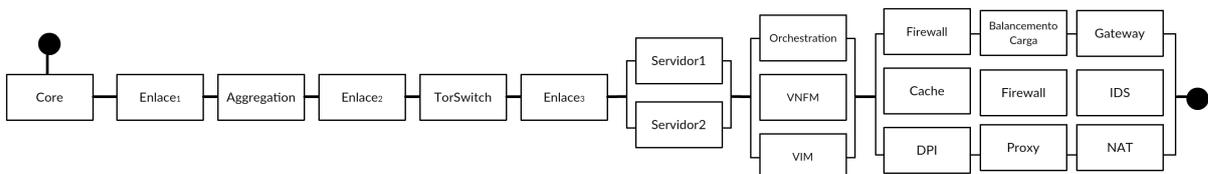


Fonte: O Autor

serve para criar um novo recurso, chamado disponibilidade. Este processo possibilitou transformar dados brutos em recursos que melhor representem a previsão de disponibilidade resultando em melhor precisão do modelo.

Esta tese define as arquiteturas de base de dados (A1) e arquiteturas alternativas como criadas com e sem medições de importância. A Figura 37 como mostrado na Arquitetura NFV (A1), assume que não há redundância na topologia do data center Fat-Tree (k=1). O modelo RBD para representar a Arquitetura A1 mostrado na Figura 37.

Figura 37 – RBD da Arquiteturas (A1)



Fonte: O Autor

No **Cenário 1: sem medições de importância**. O objetivo é estimar a disponibili-

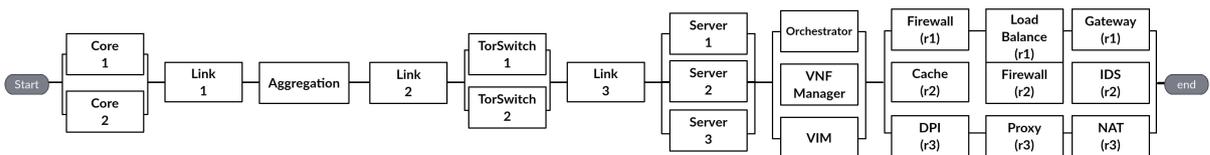
dade do sistema quando não há nenhum tipo de evidência que indique qual dispositivo deve ser replicado. A Tabela 5 retrata a topologia e os nós adotados. Mais especificamente, o experimento considera: dispositivo físico (topologia), dispositivo virtual (topologia), e número de nós (nó). Para cada tipo de tratamento, são gerados modelos de RBD.

Tabela 5 – Experimento I - sem medições de importância

Topologia	Nodes
Físico	Core, enlace, Agregation, Servidor
Virtual	Firewall, Balanceamento de carga, Proxy

Sem qualquer critério, é possível definir como implantar dispositivos adicionais (redundância) com base na experiência da operadora de rede. No entanto, pode ser uma escolha errada. Para mostrar isso, a Figura 38, apresenta uma topologia Fat-tree com redundâncias nos dispositivos Core, TorSwitch e Servidor, mas sem medições de importância.

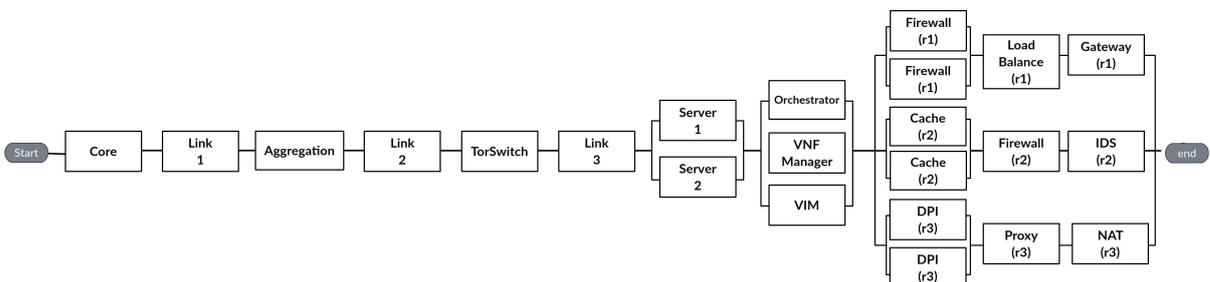
Figura 38 – RBD da Arquitetura (A2)



Fonte: O Autor

Figura 39 mostra o modelo RBD que considera redundâncias nos dispositivos virtuais (Firewall, Cache, DPI), mas sem medições de importância.

Figura 39 – RBD da Arquitetura (A3)

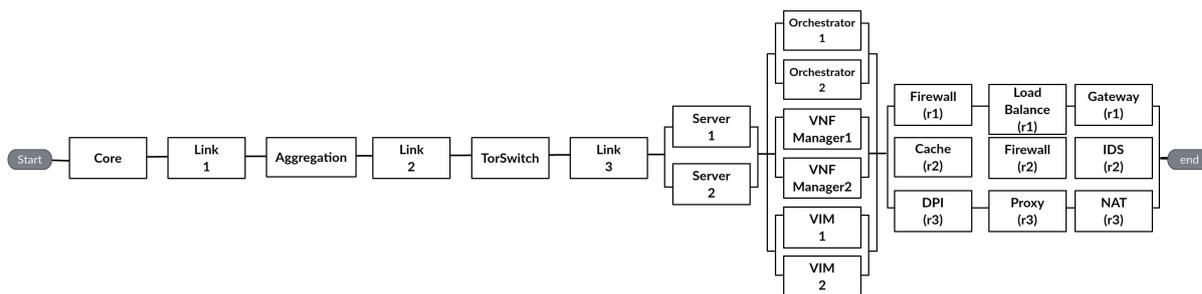


Fonte: O Autor

A Figura 40 exibe o modelo que considera redundâncias nos dispositivos virtuais (Orquestrador, VNF Manager, VIM), mas sem medições de importância.

No **Cenário 2: com medições de importância**. Os medições de importância foram testados para redes compostas de múltiplas cadeias. Assim, primeiro estudamos o índice de importância de diferentes dispositivos de rede, construindo um ranking. Em seguida,

Figura 40 – RBD da Arquitetura (A4)



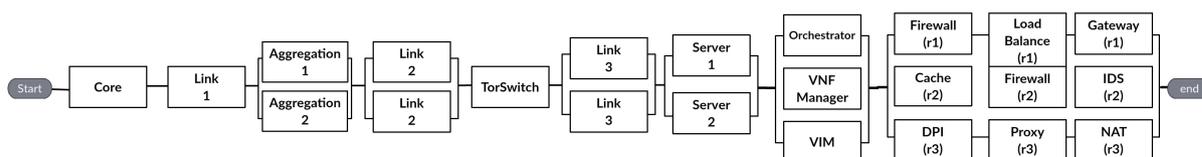
Fonte: O Autor

determinamos como os medições de importância podem ser usados para construir redundância em seu processo de implantação. Tabela 6 descreve a classificação de importância da disponibilidade e criticidade seguindo a ordem descrita na arquitetura A1. Além disso, o ranking varia entre 1 até 5, sendo 1 o mais propenso a falha. O símbolo “-” significa que não há necessidade de redundância ou não tem importância nessa análise. Os parâmetros são apresentados permitindo o operador de rede analisar de duas maneiras quais são os dispositivos para serem realizadas redundância da rede.

Essas métricas são essenciais pelo motivo de evitar aleatoriedade na escolha de qual dispositivo substituir ou adicionar redundância. Os dispositivos mais bem classificados são Agregation, enlace2, e enlace3. Esta técnica nos ajuda a identificar os parâmetros com impacto significativo na disponibilidade em estado estável do sistema. A classificação de MI obtida também nos permitirá, justificadamente, ignorar parâmetros que tenham menos impacto na medida de interesse. Os modelos indicaram que três dispositivos causam o maior impacto na disponibilidade do data center: Agregation, enlace2 e enlace3. Foi definida uma nova Arquitetura 5 (A5).

Figura 41 apresenta o modelo RBD relacionado com o A5. Uma consideração importante é o modo de falha do dispositivo. É possível identificar os dispositivos com maior impacto na disponibilidade do data center.

Figura 41 – RBD da Arquitetura (A5)



Fonte: O Autor

A disponibilidade é calculada pela Equação 5.13. A disponibilidade de um sistema é

Tabela 6 – Medições de Importância de Disponibilidade (A1)

Dispositivo	Ranking de Importância	Ranking de Falhas
Core1	5	2
enlace1	2	2
Agregation	3	1
Enlace2	1	1
TorSwitch1	4	2
Enlace3	1	1
Servidor1	-	-
Servidor2	-	-
Orquestrador	-	-
VNF Manager	-	-
VIM	-	-
SFP1	-	-
SFP2	-	-
SFP3	-	-

expressa através da relação entre o MTTF e o MTTR (EBELING, 2004).

$$A \lim_{t \rightarrow \infty} A = t = \frac{MTTF}{MTTF + MTTR} \quad (5.13)$$

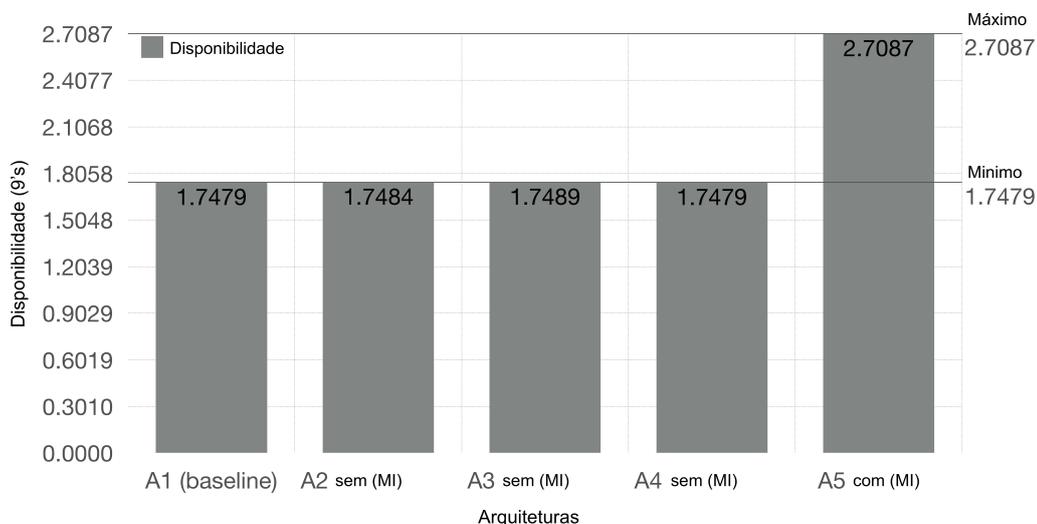
O número de noves (9's) é uma forma de representar a disponibilidade e pode ser calculado de acordo com a Equação 5.14. O número 100 representa o nível máximo de disponibilidade que o sistema de rede pode alcançar. A variável A significa a disponibilidade do sistema de rede.

$$N = 2 - \log(100 - A) \quad (5.14)$$

Figura 42 apresenta os resultados de disponibilidade das cinco arquiteturas. Os limites de disponibilidade são apresentados em número de noves. Como podemos ver, houve poucas melhorias usando a estratégia de disponibilidade sem medições de importância para planejar a melhor aplicação medições de importância no data center. As arquiteturas A2, A3, e A4 tiveram um pequeno aumento percentual na disponibilidade em relação a arquitetura A1.

Entretanto, nossa proposta é baseada em medições da importância da disponibilidade e de forma crítica. Esta solução mostra aumentos significativos na disponibilidade, quando comparado com A1, A5 aumentou em 96%. No entanto, é importante notar que mesmo após todas as melhorias, o sistema não atingiu os cinco 9's por causa da quantidade do

Figura 42 – Análise de Disponibilidade das arquiteturas A1 a A5

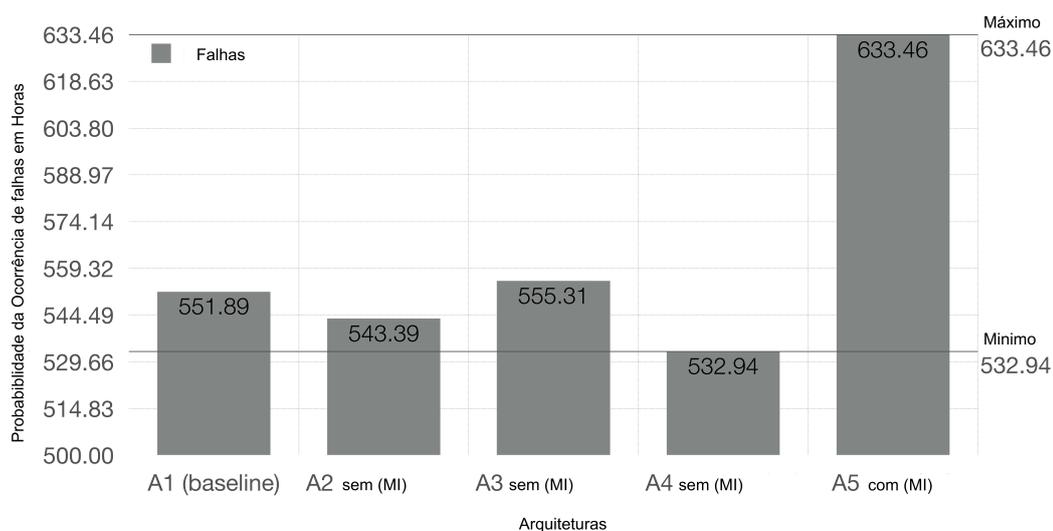


Fonte: O Autor

dispositivo para a redução. Nossa busca foi limitada a um máximo de 24 dispositivos em ambos os cenários.

A Figura 43 são mostrados os resultados das cinco arquiteturas sob a probabilidade de uma falha. Observamos, em comparação com a arquitetura A1, que a arquitetura A5 é mais tolerante a falhas do que A2, A3 e A4. Portanto, o uso de medições de importância é essencial para o replanejamento tolerante a falhas do NFV em data center.

Figura 43 – Análise de Falhas das arquiteturas A1 a A5

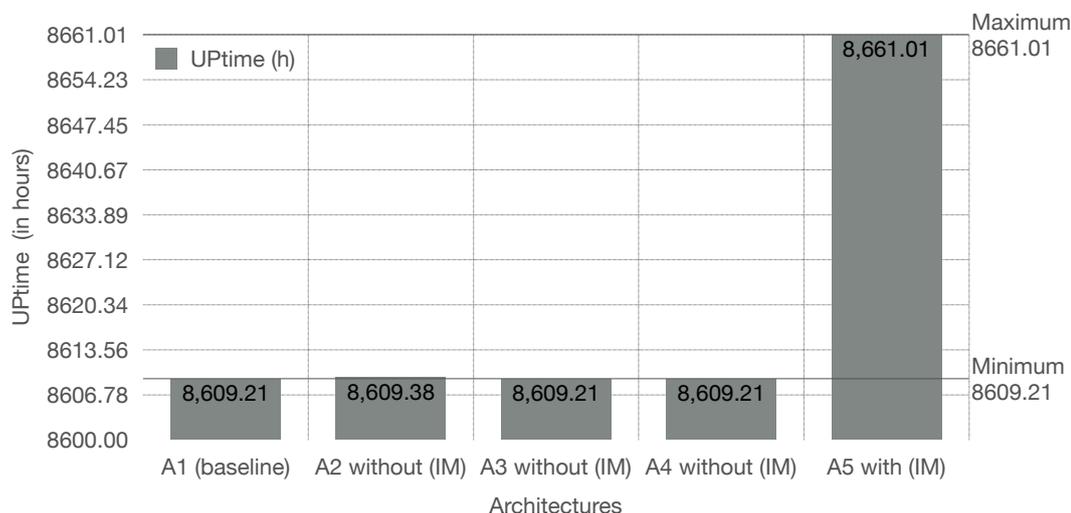


Fonte: O Autor

A Figura 44 resume o tempo de uptime das cinco arquiteturas. Como podemos observa, representa o tempo operacional disponível durante o tempo espacial em 8760 horas (um

ano). Os resultados mostram que o tempo de uptime habitual por um período superior ao normal ocorre entre as arquiteturas quando se utiliza a metodologia medições de importância. Em comparação com A1, o nível de uptime é alto na arquitetura A5.

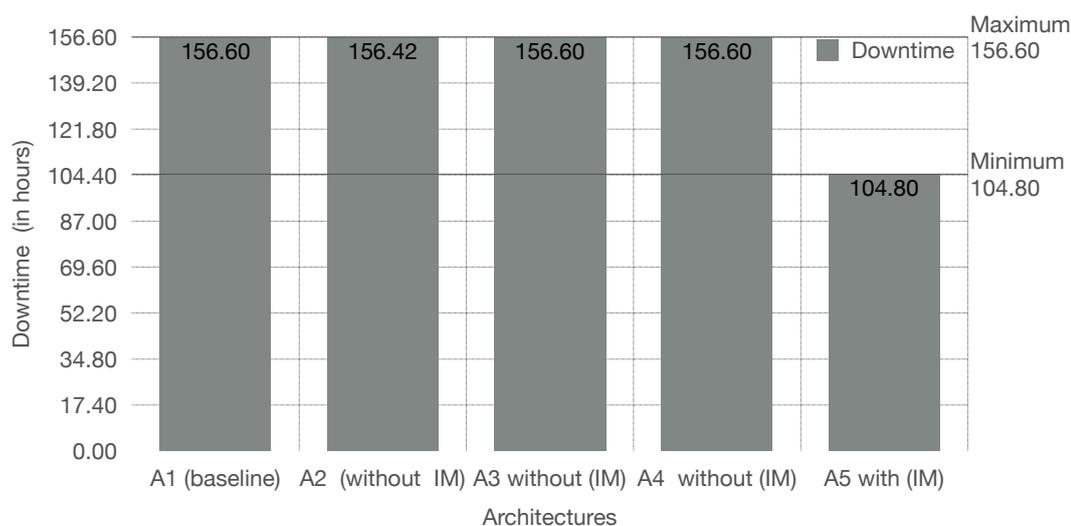
Figura 44 – Análise do Tempo de Uptime das Arquiteturas A1 a A5



Fonte: O Autor

Já a Figura 45 apresenta os resultados do tempo de downtime no NFV em data center. Os resultados mostram a eficiência na redução do período de downtime entre as arquiteturas quando se utiliza a metodologia medições de importância. O tempo de downtime é de 104,80 horas para a arquitetura A5 em comparação com a A1 (156.600 horas).

Figura 45 – Análise de Downtime para as Arquiteturas A1 a A5



Fonte: O Autor

A Tabela 7 sintetiza os resultados das cinco arquiteturas. A partir da avaliação desta pesquisa, as perguntas a seguir foram respondidas: Como decidir que dispositivo ou VNF

deve ser replicado? Como melhorar a disponibilidade ao mesmo tempo em que reduz o custo? Como melhorar as operações do NFV em data center? Então nossa resposta é que a medições de importância tem o objetivo principal de avaliar os pontos críticos para melhorias no NFV em data center. Esta técnica nos ajuda a identificar os parâmetros com impacto significativo na disponibilidade do sistema em estado estável.

Tabela 7 – Resultados das Arquiteturas A1 até A5

Arquiteturas	MTTF(h)	MTTR(h)	Disponibilidade(h)	Uptime(h)	Downtime(h)
A1	528.037680	9.605030985	0.9821349182	8609.21	156.60
A2	543.397214	9.873055170	0.9821550954	8609.38	156.42
A3	555.312656	10.10116412	0.9821349182	8609.21	156.50
A4	532.944419	9.694284812	0.9821349182	8609.21	156.50
A5	633.460555	7.665113223	0.9980442889	8661.01	104.80

As seguintes observações podem ser feitas com base nos resultados do experimento. A arquitetura A5 tem maior disponibilidade em comparação com a arquitetura A1. Isto ocorre porque eles possuem dispositivos redundantes como Aggregation, enlace2 e enlace3, o que reduz o gargalo da disponibilidade da rede. Em geral, os resultados demonstram o impacto significativo de dispositivos distintos sobre a disponibilidade do respectivo dispositivo físico. Ambos os modelos RBD são ferramentas valiosas para os operadores de rede de NFV em data center durante o replanejamento arquitetônico. A introdução de redundância nos dispositivos físicos torna os dispositivos virtuais confiáveis, com base nos dois cenários avaliados. Se o objetivo é auxiliar os projetistas de rede na redução de gargalos, a medições de importância de disponibilidade é adequada. O replanejamento da infraestrutura de rede consiste em muitos dispositivos. É necessário explorar todas as combinações possíveis para maximizar a disponibilidade do NFV em data center. Entretanto, com a aplicação dos parâmetros importantes e críticos, as evidências mostram um processo otimizado com indicadores matemáticos. O processo é baseado no índice para realizar redundância para cada elemento do NFV em data center.

5.4.5 Discussão

A alocação de recursos em cenários reais de NFV é uma questão complexa, uma vez que o escalonamento das VNFs pode ter um impacto significativo no desempenho geral do sistema. O framework inteligente, baseado em KDN aplicada ao escalonamento de VNFs aborda muitos dos desafios relacionados ao problema do escalonamento de VNFs, obtendo vantagens significativas. O modelo LSTM proposto baseado na afinidade, juntamente com o novo módulo de diagnóstico, permite gerenciar recursos VNFs como uma função da análise dos dados coletados do processo, tais como o tráfego processado pela VNFs. A abordagem proposta tem a vantagem de permitir a ajuda na automação da rede para

reduzir erros e o tempo gasto na configuração e gestão da rede. Portanto, é crucial para otimizar a escalonamento de uma VNF com alta disponibilidade, tornando possível otimizar o desempenho geral da rede.

Finalmente, a análise dos resultados apresentados na Subsecção 5.4.3 mostrando as novas métricas propostas associadas à performabilidade, falhas e aspectos térmicos, permite responder à QP3 relativa ao módulo de diagnóstico. A integração da nova ferramenta de diagnóstico desenvolvida, permite aos operadores de rede detectar rapidamente gargalos e quantificar a degradação do desempenho, monitorando recursos virtuais e físicos, assim como modelar analiticamente os diversos componentes envolvidos na comunicação em rede. Além disso, nossa proposta pode ser aplicada a outros cenários, por exemplo, cenários reais com falhas simultâneas de várias VNFs e estabelece caminhos recuperados com consciência de QoS. Além disso, é aplicável a outros cenários onde os fluxos de tráfego bruto precisam ser direcionados através do ambiente data center/nuvem. Além de cenários de cadeia de serviços VNF únicos ou múltiplos, pode ser usado como base para prever o desempenho de múltiplas cadeias VNFs.

5.5 CONSIDERAÇÕES FINAIS

Nesse capítulo, são apresentados os resultados experimentais, que mostram o framework autônomo proposto para escalonamento de VNFs em data center, baseado em LSTM e modelos de afinidades. Além disso, pode prever com precisão os efeitos térmicos de uma carga de trabalho, tornando-se essencial para auxiliar o operador de rede no replanejamento do escalonamento futuro de VNFs. Os resultados de previsão obtidos com o modelo proposto de LSTM, superam os obtidos com outros métodos de previsão propostos na literatura.

6 CONCLUSÃO E DIREÇÕES FUTURAS

Nesta tese, desenvolvemos uma nova abordagem de escalonamento de VNFs tolerante a falhas, baseada num modelo de previsão LSTM, juntamente com regras de afinidade, bem como a análise dos aspectos de performabilidade, desempenho e disponibilidade, dentro do contexto de um data center.

6.1 CONCLUSÃO

Embora várias abordagens tenham sido propostas na literatura para a realização de escalonamentos de VNFs tolerantes a falhas, na medida do conhecimento dos autores, nenhuma delas abordou o problema a partir da perspectiva apresentada neste trabalho. Neste trabalho, várias experiências foram realizadas para avaliar os efeitos da temperatura sobre o desempenho do sistema, confirmando o alto impacto da temperatura na degradação do desempenho, em termos de performabilidade e disponibilidade, portanto, foi confirmando a importância de levar em conta o aumento da temperatura na alocação de recursos. Um modelo LSTM é proposto para ajudar o escalonamento. Os resultados experimentais mostram que o modelo proposto baseado em LSTM pode prever com precisão os efeitos térmicos de uma carga de trabalho, tornando-os essenciais para prever a disponibilidade e auxiliar o operador de rede no planejamento do escalonamento futuro dos recursos. Além disso, os resultados da previsão obtidos com o modelo LSTM proposto superam os obtidos com outros métodos de previsão propostos na literatura. Os benefícios derivados da análise podem ser utilizados como mecanismo auxiliar para uma melhor alocação de recursos.

6.2 AMEAÇAS À VALIDADE

Nesta seção, são discutidas algumas das ameaças identificadas à validade da nossa solução desenvolvida. Desta forma, pretende-se destacar as principais questões que podem ser encontradas ao implementar o framework de escalonamento na tese e estabelecer a base para o trabalho futuro. Em particular, são consideradas diferentes ameaças. Ameaças a conclusões, validade construtiva, interna e externa, assim como as limitações da abordagem proposta.

A **validade de conclusão** tem uma natureza estocástica da abordagem proposta para a escalonamento de VNFs, torna a validade dos resultados fortemente dependente da execução das repetições dos experimentos. Neste trabalho, uma quantidade razoável de repetições experimentais, bem como testes estatísticos, são considerados na realização dos experimentos propostos, a fim de garantir a significância estatística dos resultados. No entanto, ainda pode ser que as repetições experimentais não reflitam os resultados alcançados.

A **Validade Construtiva** avalia até que ponto os experimentos projetados representam o sistema real que está sendo estudado. Se os experimentos de simulação representam sistemas reais de centros de dados, é uma das ameaças mais críticas à validade construtiva. Além disso, a escolha de um número justo de níveis, fatores e parâmetros são ameaças à validade construtiva. Finalmente, são consideradas diferentes suposições para poder lidar com instâncias de redes de comunicação do mundo real contendo múltiplos atributos, tais como latência física ou interferência, bem como técnicas, tais como largura de banda ou taxas de dados. Essas suposições, como a que diz respeito ao NFV dataset convertidos, permitem que o modelo proposto se concentre em detalhes topológicos relevantes. Além disso, é também possível que estas suposições possam ter um impacto negativo no processo de aprendizagem dos modelos de aprendizagem da máquina, uma vez que podem esconder informações importantes. Neste contexto, é provável que, a fim de lidar com aplicações do mundo real, estes pressupostos devam ser diminuídos, e mais esforços devem ser feitos para lidar com as limitações impostas pelos modelos.

A **Validade Interna** analisa até que ponto as conclusões do estudo são corretas, enquanto a validade externa analisa até que ponto os resultados obtidos podem ser generalizados. Modelos de aprendizagem de máquinas são altamente parametrizáveis e as pesquisas heurísticas são configuráveis. A otimização dos parâmetros é uma tarefa computacional complexa e onerosa. Além disso, mesmo quando se realiza uma busca exaustiva testando um extenso conjunto de diferentes instâncias e parâmetros de problemas, podem ocorrer casos em que o conjunto otimizado de parâmetros não funcione suficientemente bem para uma instância de problema em particular. Nesta tese, os parâmetros padrão sugeridos em (GREFF et al., 2017) são utilizados desde que se provou que têm um desempenho suficientemente bom para o modelo de aprendizagem profunda proposto.

A **Validade Externa** é um problema particular estudado no contexto deste trabalho e foi escolhido com base nas diferentes classes de instâncias que refletem a situação real da NFV no mundo real. Embora, nesta tese, uma ampla gama de classes de instâncias problemáticas tenha sido coberta, ainda pode ser o caso de que a abordagem proposta não poderia ser generalizada para atender aos requisitos de outras (novas) instâncias, como, por exemplo, as relacionadas com o posicionamento da VNF ou as relacionadas com redes que têm instâncias de topologia excepcional.

As **limitações** estão associadas com a formulação matemática para problemas NP-difícil. O problema de programação VNF em um data center é semelhante aos problemas do tipo VNE, que são NP-difícil. A prova matemática de tal similaridade está fora do escopo deste trabalho. A formulação matemática particular usada neste trabalho, bem como a sua demonstração, pode ser encontrada em (GU et al., 2016).

6.3 TRABALHOS FUTUROS

Para trabalhos futuros, visa obter uma melhor caracterização das interferências sob diferentes variações de carga de trabalho. Para isso, o modelo de custos, bem como o algoritmo genético, devem ser melhorados no sentido de uma melhor integração da afinidade com o modelos de aprendizado por reforço em conjunto com LSTM para escalonamento autônomo de VNFs. Desta forma, espera-se conseguir uma estratégia dinâmica de tráfego durante o tempo de execução e recomendar novos escalonamento por meio de sistemas de recomendação, e ampliar o módulo diagnóstico para incorporar os modelos de AI e CAI. O nosso objetivo é contribuir para a comunidade do IETF e a IRTF, discutindo os resultados dessa tese, além de criar documentos, descrevendo os desafios e a terminologia dos problemas abordados. Neste contexto, tendo em conta os últimos esforços realizados pela IETF/IRTF, as futuras atividades incluem a busca de contribuições para a padronização ao longo do caminho do escalonamento de VNFs de uma forma autônoma, eficiente e escalável. Além disso, o módulo de diagnóstico, pode ser utilizado como ferramenta independente para outros cenários de SFC, considero podem ser contribuições para o IETF/IRTF.

REFERÊNCIAS

- ABUJODA, A.; KOUCHAKSARAEI, H. R.; PAPADIMITRIOU, P. SDN-based source routing for scalable service chaining in datacenters. In: SPRINGER. *International Conference on Wired/Wireless Internet Communication*. [S.l.], 2016. p. 66–77.
- AHMED, N. K.; ATIYA, A. F.; GAYAR, N. E.; EL-SHISHINY, H. An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews*, Taylor & Francis, v. 29, n. 5-6, p. 594–621, 2010.
- ALLEN, R.; ROESCH, W. Reliability prediction: The applicability of high temperature testing. 1990.
- ALOM, M. Z.; TAHA, T. M.; YAKOPCIC, C.; WESTBERG, S.; SIDIKE, P.; NASRIN, M. S.; HASAN, M.; ESSEN, B. C. V.; AWWAL, A. A.; ASARI, V. K. A state-of-the-art survey on deep learning theory and architectures. *Electronics*, Multidisciplinary Digital Publishing Institute, v. 8, n. 3, p. 292, 2019.
- AMALDI, E.; CONIGLIO, S.; KOSTER, A. M.; TIEVES, M. On the computational complexity of the virtual network embedding problem. *Electronic Notes in Discrete Mathematics*, Elsevier, v. 52, p. 213–220, 2016.
- AMARAL, P.; DINIS, J.; PINTO, P.; BERNARDO, L.; TAVARES, J.; MAMEDE, H. S. Machine learning in software defined networks: Data collection and traffic classification. In: IEEE. *2016 IEEE 24th International Conference on Network Protocols (ICNP)*. [S.l.], 2016. p. 1–5.
- ANDERSEN, D. G. Theoretical approaches to node assignment. *Computer Science Department*, p. 86, 2002.
- ARMBRUST, M.; FOX, A.; GRIFFITH, R.; JOSEPH, A. D.; KATZ, R.; KONWINSKI, A.; LEE, G.; PATTERSON, D.; RABKIN, A.; STOICA, I. et al. A view of cloud computing. *Communications of the ACM*, ACM, v. 53, n. 4, p. 50–58, 2010.
- ARREGOCES, M.; PORTOLANI, M. *Data center fundamentals*. [S.l.]: Cisco Press, 2003.
- ARUNARANI, A.; MANJULA, D.; SUGUMARAN, V. Task scheduling techniques in cloud computing: A literature survey. *Future Generation Computer Systems*, Elsevier, v. 91, p. 407–415, 2019.
- ASSI, C.; AYOUBI, S.; KHOURY, N. E.; QU, L. Energy-aware mapping and scheduling of network flows with deadlines on vnfs. *IEEE Transactions on Green Communications and Networking*, IEEE, v. 3, n. 1, p. 192–204, 2018.
- AVIZIENIS, A.; LAPRIE, J.-C.; RANDELL, B.; LANDWEHR, C. Basic concepts and taxonomy of dependable and secure computing. *IEEE transactions on dependable and secure computing*, IEEE, v. 1, n. 1, p. 11–33, 2004.
- BARHAM, P.; DRAGOVIC, B.; FRASER, K.; HAND, S.; HARRIS, T.; HO, A.; NEUGEBAUER, R.; PRATT, I.; WARFIELD, A. Xen and the art of virtualization. In: ACM. *ACM SIGOPS operating systems review*. [S.l.], 2003. v. 37, n. 5, p. 164–177.

- BARI, M. F.; BOUTABA, R.; ESTEVES, R.; GRANVILLE, L. Z.; PODLESNY, M.; RABBANI, M. G.; ZHANG, Q.; ZHANI, M. F. Data center network virtualization: A survey. *IEEE Communications Surveys & Tutorials*, IEEE, v. 15, n. 2, p. 909–928, 2013.
- BECK, M. T.; BOTERO, J. F. Scalable and coordinated allocation of service function chains. *Computer Communications*, Elsevier, v. 102, p. 78–88, 2017.
- BELBEKKOUCHE, A.; HASAN, M. M.; KARMOUCH, A. Resource discovery and allocation in network virtualization. *IEEE Communications Surveys & Tutorials*, IEEE, v. 14, n. 4, p. 1114–1128, 2012.
- BELOGLAZOV, A.; ABAWAJY, J.; BUYYA, R. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future generation computer systems*, Elsevier, v. 28, n. 5, p. 755–768, 2012.
- BIRNBAUM, Z. W. *On the importance of different components in a multicomponent system*. [S.l.], 1968.
- BOLCH, G.; GREINER, S.; MEER, H. de; TRIVEDI, K. S. *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. [S.l.]: John Wiley & Sons, 2006.
- BRUCKER, P.; KNUST, S. *Complex Scheduling (GOR-Publications)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN 3540295453.
- CAO, H.; ZHANG, Y.; YANG, L. Dynamic embedding and scheduling of virtual network service for future networks. In: IEEE. *2019 IEEE International Conference on Communications Workshops (ICC)*. [S.l.], 2019. p. 1–6.
- CHEN-XIAO, C.; YA-BIN, X. Research on load balance method in sdn. *International Journal of Grid and Distributed Computing*, v. 9, n. 1, p. 25–36, 2016.
- CHENG, X.; SU, S.; ZHANG, Z.; WANG, H.; YANG, F.; LUO, Y.; WANG, J. Virtual network embedding through topology-aware node ranking. *ACM SIGCOMM Computer Communication Review*, ACM, v. 41, n. 2, p. 38–47, 2011.
- CHIOSI, M.; CLARKE, D.; WILLIS, P.; REID, A.; FEGER, J.; BUGENHAGEN, M.; KHAN, W.; FARGANO, M.; CUI, C.; DENG, H. et al. Network functions virtualisation: An introduction, benefits, enablers, challenges and call for action. In: *SDN and OpenFlow World Congress*. [S.l.: s.n.], 2012. p. 22–24.
- CHOWDHURY, N. M. K.; BOUTABA, R. Network virtualization: state of the art and research challenges. *IEEE Communications magazine*, IEEE, v. 47, n. 7, 2009.
- CHOWDHURY, N. M. K.; BOUTABA, R. A survey of network virtualization. *Computer Networks*, Elsevier, v. 54, n. 5, p. 862–876, 2010.
- CHOWDHURY, N. M. K.; RAHMAN, M. R.; BOUTABA, R. Virtual network embedding with coordinated node and link mapping. In: IEEE. *INFOCOM 2009, IEEE*. [S.l.], 2009. p. 783–791.
- CHOWDHURY, S. R.; BAI, T.; BOUTABA, R.; FRANÇOIS, J. et al. Unis: A user-space non-intrusive workflow-aware virtual network function scheduler. In: IEEE. *2018 14th International Conference on Network and Service Management (CNSM)*. [S.l.], 2018. p. 152–160.

CHUN, B.-G.; IANNACCONE, G.; IANNACCONE, G.; KATZ, R.; LEE, G.; NICCOLINI, L. An energy case for hybrid datacenters. *ACM SIGOPS Operating Systems Review*, ACM, v. 44, n. 1, p. 76–80, 2010.

CISCO. *Cisco Global Cloud Index: Forecast and Methodology, 2015–2020 (White Paper)*. 2016. Disponível em: www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.pdf.

CLARK, D. D.; PARTRIDGE, C.; RAMMING, J. C.; WROCLAWSKI, J. T. A knowledge plane for the internet. In: ACM. *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*. [S.l.], 2003. p. 3–10.

CLAYMAN, S.; MAINI, E.; GALIS, A.; MANZALINI, A.; MAZZOCCA, N. The dynamic placement of virtual network functions. In: IEEE. *Network Operations and Management Symposium (NOMS), 2014 IEEE*. [S.l.], 2014. p. 1–9.

CLEMM, A.; CHANDRAMOULI, M.; KRISHNAMURTHY, S. Dna: An sdn framework for distributed network analytics. In: IEEE. *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. [S.l.], 2015. p. 9–17.

CONOVER, W. J.; IMAN, R. L. On multiple-comparisons procedures. *Los Alamos Sci. Lab. Tech. Rep. LA-7677-MS*, p. 1–14, 1979.

COUNCIL, J. Failure mechanisms and models for semiconductor devices. *JEDEC Publication JEP122-A*, 2002.

CRESWELL, J. W. *Research design: Qualitative, quantitative, and mixed methods approaches*. [S.l.]: Sage publications, 2013.

DEMŠAR, J. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, v. 7, n. Jan, p. 1–30, 2006.

EBELING, C. E. *An introduction to reliability and maintainability engineering*. [S.l.]: Tata McGraw-Hill Education, 2004.

ETSI. *Network Functions Virtualisation (NFV)-Virtual Network Functions Architecture*. [S.l.], 2013.

ETSI. NFV performance & portability best practises. *ETSI Standard GS NFV-PER*, v. 1, 2014.

FAN, J.; AMMAR, M. H. Dynamic topology configuration in service overlay networks: A study of reconfiguration policies. In: *INFOCOM*. [S.l.: s.n.], 2006. v. 2, n. 9, p. 1–12.

FAN, X.; WEBER, W.-D.; BARROSO, L. A. Power provisioning for a warehouse-sized computer. *SIGARCH Comput. Archit. News*, ACM, New York, NY, USA, v. 35, n. 2, p. 13–23, jun. 2007. ISSN 0163-5964. Disponível em: <http://doi.acm.org/10.1145/1273440.1250665>.

FEAMSTER, N.; REXFORD, J.; ZEGURA, E. The road to sdn. *Queue*, ACM, v. 11, n. 12, p. 20, 2013.

-
- FERNANDES, S. *Performance Evaluation for Network Services, Systems and Protocols*. [S.l.]: Springer, 2017.
- FERNANDES, S.; TAVARES, E.; SANTOS, M.; LIRA, V.; MACIEL, P. Dependability assessment of virtualized networks. In: IEEE. *Communications (ICC), 2012 IEEE International Conference on*. [S.l.], 2012. p. 2711–2716.
- FISCHER, A.; BOTERO, J. F.; BECK, M. T.; MEER, H. D.; HESSELBACH, X. Virtual network embedding: A survey. *IEEE Communications Surveys & Tutorials*, IEEE, v. 15, n. 4, p. 1888–1906, 2013.
- GARRAGHAN, P.; MORENO, I. S.; TOWNEND, P.; XU, J. An analysis of failure-related energy waste in a large-scale cloud environment. *IEEE Transactions on Emerging topics in Computing*, IEEE, v. 2, n. 2, p. 166–180, 2014.
- GILL, P.; JAIN, N.; NAGAPPAN, N. Understanding network failures in data centers: measurement, analysis, and implications. In: ACM. *ACM SIGCOMM Computer Communication Review*. [S.l.], 2011. v. 41, n. 4, p. 350–361.
- GILL, P.; JAIN, N.; NAGAPPAN, N. Understanding network failures in data centers: measurement, analysis, and implications. In: ACM. *ACM SIGCOMM Computer Communication Review*. [S.l.], 2011. v. 41, n. 4, p. 350–361.
- GODEFROID, P.; LEEUWEN, J. V.; HARTMANIS, J.; GOOS, G.; WOLPER, P. *Partial-order methods for the verification of concurrent systems: an approach to the state-explosion problem*. [S.l.]: Springer Heidelberg, 1996. v. 1032.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep learning*. [S.l.]: MIT press, 2016.
- GOOIJER, J. G. D.; HYNDMAN, R. J. 25 years of time series forecasting. *International journal of forecasting*, Elsevier, v. 22, n. 3, p. 443–473, 2006.
- GREFF, K.; SRIVASTAVA, R. K.; KOUTNÍK, J.; STEUNEBRINK, B. R.; SCHMIDHUBER, J. Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, IEEE, v. 28, n. 10, p. 2222–2232, 2017.
- GU, S.; LI, Z.; WU, C.; HUANG, C. An efficient auction mechanism for service chains in the nfv market. In: IEEE. *Computer Communications, IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on*. [S.l.], 2016. p. 1–9.
- HÅKANSSON, A. Portal of research methods and methodologies for research projects and degree projects. In: THE STEERING COMMITTEE OF THE WORLD CONGRESS IN COMPUTER SCIENCE, COMPUTER ENGINEERING AND APPLIED COMPUTING (WORLDCOMP). *Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS)*. [S.l.], 2013. p. 1.
- HAYKIN, S.; NETWORK, N. A comprehensive foundation. *Neural networks*, v. 2, n. 2004, p. 41, 2004.
- HERRERA, J. G.; BOTERO, J. F. Resource allocation in nfv: A comprehensive survey. *IEEE Transactions on Network and Service Management*, IEEE, v. 13, n. 3, p. 518–532, 2016.

- HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural computation*, MIT Press, v. 9, n. 8, p. 1735–1780, 1997.
- HOOKER, J. N. *Integrated methods for optimization*. Springer, 2012.
- HOUIDI, I.; LOUATI, W.; ZEGHLACHE, D. A distributed and autonomic virtual network mapping framework. In: IEEE. *Autonomic and Autonomous Systems, 2008. ICAS 2008. Fourth International Conference on*. [S.l.], 2008. p. 241–247.
- INDEX, C. V. N. *The zettabyte era—trends and analysis*. *Cisco white paper*, 2013.
- ISLAM, S.; LIU, A. How a consumer can measure elasticity for cloud platforms. In: *3rd ACM/SPEC International Conference on Performance Engineering*. [S.l.: s.n.], 2012. p. 85–96.
- JACOBS, A. S.; PFITSCHER, R. J.; SANTOS, R. L. dos; FRANCO, M. F.; SCHEID, E. J.; GRANVILLE, L. Z. Artificial neural network model to predict affinity for virtual network functions. In: IEEE. *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*. [S.l.], 2018. p. 1–9.
- JACOBS, A. S.; SANTOS, R. L. do; FRANCO, M. F.; SCHEID, E. J.; PFITSCHER, R. J.; GRANVILLE, L. Z. Affinity measurement for nfv-enabled networks: A criteria-based approach. In: IEEE. *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. [S.l.], 2017. p. 125–133.
- KANT, K. *Data center evolution: A tutorial on state of the art, issues, and challenges*. *Computer Networks*, Elsevier, v. 53, n. 17, p. 2939–2965, 2009.
- KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- KUO, W.; ZUO, M. J. *Optimal reliability modeling: principles and applications*. [S.l.]: John Wiley & Sons, 2003.
- LAPRIE, J.-C. Dependability: Basic concepts and terminology. In: *Dependability: Basic Concepts and Terminology*. [S.l.]: Springer, 1992. p. 3–245.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015.
- LILJA, D. J. *Measuring computer performance: a practitioner's guide*. [S.l.]: Cambridge university press, 2005.
- LU, J.; TURNER, J. Efficient mapping of virtual networks onto a shared substrate. 2006.
- MAINI, E.; MANZALINI, A. Management and orchestration of virtualized network functions. In: SPRINGER. *IFIP International Conference on Autonomous Infrastructure, Management and Security*. [S.l.], 2014. p. 52–56.
- MARTIN, A.; EGAÑA, J.; FLÓREZ, J.; MONTALBÁN, J.; OLAIZOLA, I. G.; QUARTULLI, M.; VIOLA, R.; ZORRILLA, M. Network resource allocation system for qoe-aware delivery of media services in 5g networks. *IEEE Transactions on Broadcasting*, IEEE, v. 64, n. 2, p. 561–574, 2018.

- MCKEOWN, N.; ANDERSON, T.; BALAKRISHNAN, H.; PARULKAR, G.; PETERSON, L.; REXFORD, J.; SHENKER, S.; TURNER, J. Openflow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, ACM, v. 38, n. 2, p. 69–74, 2008.
- MEI, Y.; LIU, L.; PU, X.; SIVATHANU, S.; DONG, X. Performance analysis of network i/o workloads in virtualized data centers. *IEEE Transactions on Services Computing*, IEEE, v. 6, n. 1, p. 48–63, 2013.
- MENG, X.; PAPPAS, V.; ZHANG, L. Improving the scalability of data center networks with traffic-aware virtual machine placement. In: IEEE. *INFOCOM, 2010 Proceedings IEEE*. [S.l.], 2010. p. 1–9.
- MENON, A.; SANTOS, J. R.; TURNER, Y.; JANAKIRAMAN, G. J.; ZWAENEPOEL, W. Diagnosing performance overheads in the xen virtual machine environment. In: ACM. *Proceedings of the 1st ACM/USENIX international conference on Virtual execution environments*. [S.l.], 2005. p. 13–23.
- MESTRES, A.; RODRIGUEZ-NATAL, A.; CARNER, J.; BARLET-ROS, P.; ALARCÓN, E.; SOLÉ, M.; MUNTÉS-MULERO, V.; MEYER, D.; BARKAI, S.; HIBBETT, M. J. et al. Knowledge-defined networking. *ACM SIGCOMM Computer Communication Review*, ACM, v. 47, n. 3, p. 2–10, 2017.
- MEYER, J. F. Performability: a retrospective and some pointers to the future. *Performance evaluation*, Elsevier, v. 14, n. 3-4, p. 139–156, 1992.
- MIJUMBI, R.; SERRAT, J.; GORRICO, J.-L.; BOUTEN, N.; TURCK, F. D.; DAVY, S. Design and evaluation of algorithms for mapping and scheduling of virtual network functions. In: IEEE. *Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft)*. [S.l.], 2015. p. 1–9.
- MIJUMBI, R.; SERRAT, J.; GORRICO, J.-L.; BOUTEN, N.; TURCK, F. D.; BOUTABA, R. Network function virtualization: State-of-the-art and research challenges. *IEEE Communications Surveys & Tutorials*, IEEE, v. 18, n. 1, p. 236–262, 2016.
- MILLER, R. *Google: Raise your data center temperature*. 2008.
- MILLER, R. A look inside amazon’s data centers. *Data Center Knowledge*, 2011.
- MINSKY, M.; PAPERT, S. An introduction to computational geometry. *Cambridge tiass., HIT*, 1969.
- MITCHELL, M. *An introduction to genetic algorithms*. [S.l.]: MIT press, 1998.
- NARTEN, T.; GRAY, E.; BLACK, D.; FANG, L.; KREEGER, L.; NAPIERALA, M. *Problem statement: Overlays for network virtualization*. [S.l.], 2014.
- NETO, P. S. de M.; CAVALCANTI, G. D.; MADEIRO, F.; FERREIRA, T. A. An approach to improve the performance of pm forecasters. *PloS one*, Public Library of Science, v. 10, n. 9, p. e0138507, 2015.
- NEUMANN, J. V. Probabilistic logics and the synthesis of reliable organisms from unreliable components. *Automata studies*, v. 34, p. 43–98, 1956.

-
- PALIT, A. K.; POPOVIC, D. *Computational intelligence in time series forecasting: theory and engineering applications*. [S.l.]: Springer Science & Business Media, 2006.
- PAPADIMITRIOU, P.; MAENNEL, O.; GREENHALGH, A.; FELDMANN, A.; MATHY, L. Implementing network virtualization for a future internet. In: *20th ITC specialist seminar*. [S.l.: s.n.], 2009.
- PONEMON. *Cost of Data Center Outages*. [S.l.], 2016.
- POPEK, G. J.; GOLDBERG, R. P. Formal requirements for virtualizable third generation architectures. *Communications of the ACM*, ACM, v. 17, n. 7, p. 412–421, 1974.
- POTHARAJU, R.; JAIN, N. When the network crumbles: An empirical study of cloud network failures and their impact on services. In: ACM. *Proceedings of the 4th annual Symposium on Cloud Computing*. [S.l.], 2013. p. 15.
- PUIGJANER, R.; SAVINO, N. N.; SERRA, B. *Computer Performance Evaluation: Modelling Techniques and Tools*. [S.l.]: Springer, 2003.
- QU, L.; ASSI, C.; SHABAN, K. Network function virtualization scheduling with transmission delay optimization. In: IEEE. *NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium*. [S.l.], 2016. p. 638–644.
- QUINN, P.; HALPERN, J. M. *Service Function Chaining (SFC) Architecture*. [S.l.], 2014. Work in Progress. Disponível em: <<https://datatracker.ietf.org/doc/html/draft-quinn-sfc-arch-05>>.
- RAHMAN, M. R.; AIB, I.; BOUTABA, R. Survivable virtual network embedding. In: SPRINGER. *International Conference on Research in Networking*. [S.l.], 2010. p. 40–52.
- RANKOTHGE, W.; MA, J.; LE, F.; RUSSO, A.; LOBO, J. Towards making network function virtualization a cloud computing service. In: IEEE. *Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on*. [S.l.], 2015. p. 89–97.
- RICCOBENE, V.; LOMBARDO, A.; MANZALINI, A.; SCHEMBRA, G. Network functions at the edge (netfate): design and implementation issues. *National Telecommunications and Information Theory Group (GTTI)*, 2014.
- RIERA, J. F.; HESSELBACH, X.; ESCALONA, E.; GARCÍA-ESPÍN, J. A.; GRASA, E. On the complex scheduling formulation of virtual network functions over optical networks. In: IEEE. *2014 16th International Conference on Transparent Optical Networks (ICTON)*. [S.l.], 2014. p. 1–5.
- RIXNER, S. Network virtualization: Breaking the performance barrier. *Queue*, ACM, v. 6, n. 1, p. 37, 2008.
- RODRIGUEZ, V. K. Q.; GUILLEMIN, F. Performance analysis of resource pooling for network function virtualization. In: IEEE. *2016 17th International Telecommunications Network Strategy and Planning Symposium (Networks)*. [S.l.], 2016. p. 158–163.
- SCHAFFRATH, G.; WERLE, C.; PAPADIMITRIOU, P.; FELDMANN, A.; BLESS, R.; GREENHALGH, A.; WUNDSAM, A.; KIND, M.; MAENNEL, O.; MATHY, L. Network virtualization architecture: proposal and initial prototype. In: ACM. *Proceedings of the 1st ACM workshop on Virtualized infrastructure systems and architectures*. [S.l.], 2009. p. 63–72.

-
- SHEN, W.; YOSHIDA, M.; KAWABATA, T.; MINATO, K.; IMAJUKU, W. vConductor: An NFV management solution for realizing end-to-end virtual network services. In: IEEE. *Network Operations and Management Symposium (APNOMS), 2014 16th Asia-Pacific*. [S.l.], 2014. p. 1–6.
- SILVA, B.; MATOS, R.; CALLOU, G.; FIGUEIREDO, J.; OLIVEIRA, D.; FERREIRA, J.; DANTAS, J.; LOBO, A.; ALVES, V.; MACIEL, P. Mercury: An integrated environment for performance and dependability evaluation of general systems. In: *Proceedings of Industrial Track at 45th Dependable Systems and Networks Conference, DSN*. [S.l.: s.n.], 2015.
- SOUZA, R.; CALLOU, G.; CAMBOIN, K.; FERREIRA, J.; MACIEL, P. The effects of temperature variation on data center it systems. In: IEEE. *2013 IEEE International Conference on Systems, Man, and Cybernetics*. [S.l.], 2013. p. 2354–2359.
- SOUZA, R.; DIAS, K.; FERNANDES, S. Nfv data centers: A systematic review. *IEEE Access*, p. 1–1, 2020. ISSN 2169-3536.
- SOUZA, R.; SANTOS, M.; FERNANDES, S. Importance measures for nfv data center: An availability evaluation. In: SBC. *Workshop Pré-IETF*. [S.l.], 2018.
- SRIVASTAVA, N.; HINTON, G.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, JMLR. org, v. 15, n. 1, p. 1929–1958, 2014.
- SUGERMAN, J.; VENKITACHALAM, G.; LIM, B.-H. Virtualizing i/o devices on vmware workstation’s hosted virtual machine monitor. In: *USENIX Annual Technical Conference, General Track*. [S.l.: s.n.], 2001. p. 1–14.
- SWAMI, R.; DAVE, M.; RANGA, V. Software-defined networking-based ddos defense mechanisms. *ACM Computing Surveys (CSUR)*, ACM, v. 52, n. 2, p. 28, 2019.
- TANG, F.; LI, L.; BAROLLI, L.; TANG, C. An efficient sampling and classification approach for flow detection in sdn-based big data centers. In: IEEE. *2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)*. [S.l.], 2017. p. 1106–1115.
- TERRELL, S. R. *Writing a proposal for your dissertation: guidelines and examples*. [S.l.]: Guilford Publications, 2015.
- TOOSI, A. N.; CALHEIROS, R. N.; BUYYA, R. Interconnected cloud computing environments: Challenges, taxonomy, and survey. *ACM Computing Surveys (CSUR)*, ACM New York, NY, USA, v. 47, n. 1, p. 1–47, 2014.
- TRAVERS, M. Cpu power consumption experiments and results analysis of intel i7-4820k. *School of Electrical and Electronic Engineering, Newcastle University, Technical Report Series*, 2015.
- URIARTE, R. B.; TIEZZI, F.; NICOLA, R. D. Slac: A formal service-level-agreement language for cloud computing. In: IEEE COMPUTER SOCIETY. *Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*. [S.l.], 2014. p. 419–426.

-
- VERDOUW, C. N.; VUCIC, N.; SUNDMAEKER, H.; BEULENS, A. Future internet as a driver for virtualization, connectivity and intelligence of agri-food supply chain networks. *International Journal on Food System Dynamics*, v. 4, n. 4, p. 261–272, 2014.
- WANG, G.; NG, T. E. The impact of virtualization on network performance of amazon ec2 data center. In: IEEE. *Infocom, 2010 proceedings ieee*. [S.l.], 2010. p. 1–9.
- WANG, R.; HU, H.; YANG, X. Potentials and challenges of C-RAN supporting multi-rats toward 5G mobile networks. *IEEE Access*, IEEE, v. 2, p. 1187–1195, 2014.
- WAY, K.; MING, J. Z. *Optimal Reliability Modeling - Principles and Applications*. [S.l.]: Wiley, 2003. v. 1.
- YANG, S.; LI, F.; YAHYAPOUR, R.; FU, X. Delay-sensitive and availability-aware virtual network function scheduling for nfv. *IEEE Transactions on Services Computing*, IEEE, 2019.
- YEKKEHKHANY, A.; HOJJATI, A.; HAJIESMAILI, M. H. Gb-pandas:: Throughput and heavy-traffic optimality analysis for affinity scheduling. *ACM SIGMETRICS Performance Evaluation Review*, ACM, v. 45, n. 2, p. 2–14, 2018.
- YI, B.; WANG, X.; LI, K.; DAS, S. k.; HUANG, M. A comprehensive survey of network function virtualization. *Comput. Networks*, v. 133, p. 212—262, 2018.
- YU, H.; QIAO, C.; ANAND, V.; LIU, X.; DI, H.; SUN, G. Survivable virtual infrastructure mapping in a federated computing and networking system under single regional failures. In: IEEE. *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. [S.l.], 2010. p. 1–6.
- YU, M.; YI, Y.; REXFORD, J.; CHIANG, M. Rethinking virtual network embedding: substrate support for path splitting and migration. *ACM SIGCOMM Computer Communication Review*, ACM, v. 38, n. 2, p. 17–29, 2008.
- ZHANG, J.; LI, K.; GUO, D.; QI, H.; TAO, X.; JIN, Y. Data rate guarantee for coflow scheduling in network function virtualization. In: IEEE. *2016 IEEE/ACM 24th International Symposium on Quality of Service (IWQoS)*. [S.l.], 2016. p. 1–6.
- ZHU, Y.; AMMAR, M. H. Algorithms for assigning substrate network resources to virtual network components. In: *INFOCOM*. [S.l.: s.n.], 2006. v. 1200, n. 2006, p. 1–12.

APÊNDICE A – APPENDIX

Neste apêndix, apresenta novas equações, que são uma extensão do nosso trabalho anterior (SOUZA et al., 2013). O passo a passo de um novo procedimento de equação para chegar à solução final proposta para o modelo de equação de temperatura 4.6 da equação principal Equação 4.4 com base na primeira lei da termodinâmica, é fornecida.

Rearranjo da Equação 4.5 na Subseção 4.2.3.2 para que a equação diferencial possa ser facilmente resolvida, obtém-se a seguinte expressão:

$$\frac{dT}{dt} + \frac{h * A_s}{m * C} * T_{\infty} + \frac{I * V + \alpha * C_p * V^2 * f}{m * C} \quad (\text{A.1})$$

A equação diferencial de primeira ordem na Equação A.1 é resolvido através da aplicação do conhecido Método do Fator de Integração. O Método dos Fatores Integrantes afirma que, a solução para a seguinte equação diferencial:

$$\frac{dT}{dt} + P(t) * T(t) = g(t), \quad (\text{A.2})$$

é dado por:

$$T(t) = e^{-\int P(t)dt} * \int e^{\int P(t)dt} * g(t) * dt + A_1 * e^{-\int P(t)dt}. \quad (\text{A.3})$$

, cedendo a:

$$T(t) = e^{-\int \frac{h * A_s}{m * C} dt} * \int e^{\frac{h * A_s}{m * C} dt} * \left(\frac{h * A_s}{m * C} * T_{\infty} \frac{I * V + \alpha * C_p * V^2 * f}{m * C} \right) * dt + A_1 * e^{-\int \frac{h * A_s}{m * C} dt}. \quad (\text{A.4})$$

Retirando constantes de cada integrante, e resolvendo os integrais simples em Equação A.4, Equação A.5 é obtida:

$$T(t) = e^{-\frac{h * A_s}{m * C} t} * \left(\frac{h * A_s}{m * C} * T_{\infty} \frac{I * V + \alpha * C_p * V^2 * f}{m * C} \right) * \int e^{\frac{h * A_s}{m * C} t} * dt + A_1 * e^{-\int \frac{h * A_s}{m * C} t}. \quad (\text{A.5})$$

Solucionando o restante integral na Equação A.5 cede a:

$$T(t) = e^{-\frac{h * A_s}{m * C} t} * \left(\frac{h * A_s}{m * C} * T_{\infty} \frac{I * V + \alpha * C_p * V^2 * f}{m * C} \right) * \frac{h * A_s}{m * C} t * A_1 * e^{-\frac{h * A_s}{m * C} t}. \quad (\text{A.6})$$

Finalmente, simplificando a expressão na Equação A.6, a solução geral para a temperatura dada na Equação 4.6 na Subseção 4.2.3.2 é obtido