

Mapeamento Sistemático sobre Gerenciamento de Recursos em Ambiente Fog Computing

Danilo Souza Silva¹, José dos Santos Machado², Admilson de Ribamar Lima Ribeiro¹,
Edward David Moreno Ordonez¹

¹Departamento de Computação - Universidade Federal de Sergipe, UFS - São Cristóvão, Brasil

²Departamento de Computação - Instituto Federal de Sergipe, IFS - Estância, Brasil

danilo.silva@dcomp.ufs.br, jsmac18@hotmail.com, admilson@ufs.br, edwdavid@gmail.com

Resumo—Este trabalho investiga o paradigma *Fog Computing* como uma alternativa ao ambiente centralizado da nuvem para aplicações IoT sensíveis à latência, apresenta seu conceito, características e a taxonomia das suas aplicações. Neste artigo, o método de mapeamento sistemático da literatura é usado para encontrar lacunas e futuras direções relacionadas ao gerenciamento de recursos em um ambiente *Fog Computing*. O mapeamento identificou sete desafios principais para a *Fog Computing*. As descobertas deste trabalho de pesquisa podem ajudar profissionais e pesquisadores a entender o contexto operacional do gerenciamento de recursos da *Fog Computing*, para aplicações IoT e fornece uma série de informações úteis para futuros trabalhos de pesquisa e desenvolvimento no meio acadêmico.

1. Introdução

A Internet das Coisas (*Internet of Things - IoT*) refere-se a interconexão de milhões de objetos inteligentes suportando um imenso número de aplicações [1]. Em 2020 estima-se que existirão mais de 24 bilhões de dispositivos inteligentes conectados, causando um impacto econômico de trilhões de dólares [2].

Devido à capacidade limitada de recursos dos dispositivos IoT, tais como, processamento e armazenamento, a computação em nuvem é tradicionalmente referenciada como umas das preferidas plataformas para agregar, processar e analisar a grande massa de dados gerada por esses dispositivos [3].

Contudo, com a explosão prevista no número de dispositivos conectados e serviços IoT, é esperado que inevitavelmente os *data centers* sofram com uma carga de rede excessiva, altas taxas de latência e aumento no consumo geral de energia, isto afeta diretamente a qualidade de serviço (QoS) em aplicações que necessitem de respostas rápidas e processamento imediato tais como, aplicações na área da saúde [4], sistemas de emergência [5], e veículos autônomos [6].

Com o objetivos de superar estas limitações, o conceito de *Fog Computing* foi introduzido como um caminho para melhorar a interação entre dispositivos IoT e a computação em nuvem. De acordo com Bonomi *et al.* [7], *Fog Computing* é definida como uma plataforma altamente virtualizada

que fornece serviços de processamento, armazenamento e rede entre dispositivos e *data centers* tradicionais, normalmente, mas não exclusivamente na borda da rede.

No entanto, de acordo com [8], os desafios particulares da adoção de recursos computacionais baseados em *Fog Computing* incluem a adesão à distribuição geográfica das fontes de dados IoT, a sensibilidade ao atraso dos serviços IoT e as quantidades potencialmente muito grandes de dados emitidos e consumidos pelos dispositivos IoT.

Outro fator a ser considerado é a necessidade da existência de técnicas de gerenciamento que permitam de forma eficiente determinar quais módulos de serviço ou aplicação serão alocados na borda da rede para cada dispositivo de acordo com sua necessidade e requisitos de QoS [9]. Além disso, soluções apresentadas nesse paradigma ainda estão em seu estágio inicial de desenvolvimento.

Portanto, várias questões de pesquisa permanecem abertas em termos de gerenciamento eficiente de recursos. É digno de nota observar que, a identificação da melhor abordagem para a *Fog Computing* em termos de aspectos estruturais, de serviço e de segurança ainda é uma tarefa difícil.

Neste contexto, o objetivo deste mapeamento sistemático é:

- Obter um compreensivo entendimento acerca das abordagens desenvolvidas para otimizar a qualidade de serviço de aplicações IoT sensíveis à latência no contexto da *Fog Computing*.
- Prover uma visão geral do estado da arte nessa área de pesquisa.
- Identificar importantes lacunas nas abordagens existentes, bem como, promissoras direções de pesquisa.

O restante deste trabalho está estruturado conforme apresentado a seguir. Na seção 2 apresentamos seu conceito, características e a taxonomia das aplicações *Fog Computing*. Na seção 3 elencamos a metodologia do mapeamento sistemático, questões de pesquisa, estratégia de busca, critérios de seleção e método de extração de dados. Na seção 4 apresentamos a análise dos resultados. Na seção 5 apresentamos os desafios e direções futuras de pesquisa. Por fim, na seção 6, apresentamos a conclusão e considerações finais do trabalho.

2. Fog Computing

Nesta seção são apresentados e explicados o conceito, as características e a taxonomia das aplicações do novo paradigma da computação distribuídas que é a *Fog Computing*.

2.1. Conceito da Fog Computing

Grande parte dos trabalhos tem focado em como efetivamente descarregar tarefas computacionais intensivas de dispositivos com recursos limitados para a nuvem e obter os resultados desejados [10]. No entanto, devido à latência de rede frequentemente imprevisível, especialmente em um ambiente móvel, muitas vezes a computação em nuvem não pode atender aos requisitos rigorosos de latência, segurança e privacidade dos aplicativos em área restrita geograficamente [11]. Por outro lado, a crescente quantidade de dados gerados por dispositivos e sistemas, com poucos recursos pode se tornar impraticável para transportar dados através de redes para nuvens remotas [12].

Para isso, surgiu um novo paradigma, a *Fog Computing*. O conceito de computação em Fog foi adotado pela *Cisco Systems* como um novo paradigma em 2012 [13]. A *Fog Computing* é a computação em nuvem que distribuirá serviços avançados de computação, armazenamento, rede e gerenciamento mais próximos dos usuários finais, enviando informações dos dispositivos IoT para *Cloud Computing*, formando assim uma plataforma distribuída e virtualizada [14]. Assim, também é referido como computação de borda [15]. A figura 1 mostra a localização entre *Fog Computing* e *Cloud Computing*.

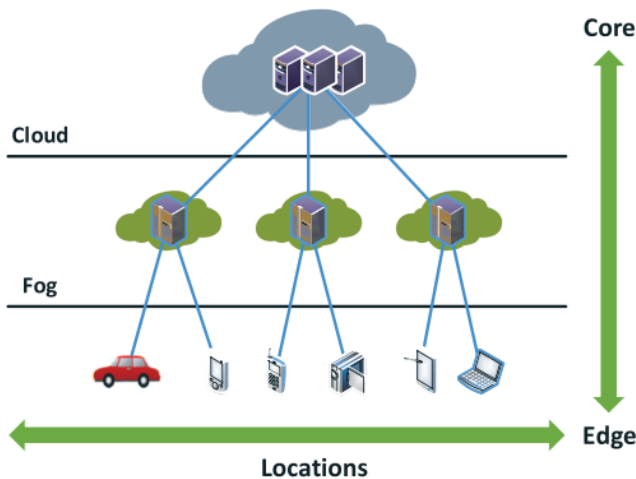


Figura 1. Localização da Fog Computing [10]

A computação em *Fog* oferece muitas vantagens desejadas pelas aplicações de hoje, como processamento em tempo real, rápida escalabilidade e compartilhamento de recursos e armazenamento local. Como tal, a computação em *Fog* rapidamente conquistou muita atenção da indústria e da academia. É naturalmente a ponte entre a Internet das Coisas (IoT) e a computação em nuvem, com a infraestrutura de computação existente na Internet [16].

2.2. Características da Fog Computing

A computação *Fog* é um paradigma inovador que realiza computação distribuída, serviços de rede e armazenamento, além da comunicação entre *Cloud Computing Data Centers* até os dispositivos ao longo da borda da rede [14]. Essa comunicação amplia as operações e serviços inerentes à computação em nuvem, permitindo assim uma nova geração de aplicativos [12]. A principal função é filtrar e agregar dados para os centros de dados da *Cloud* e aplicar inteligência lógica a dispositivos finais. A figura 2 apresenta arquitetura da *Fog Computing*.

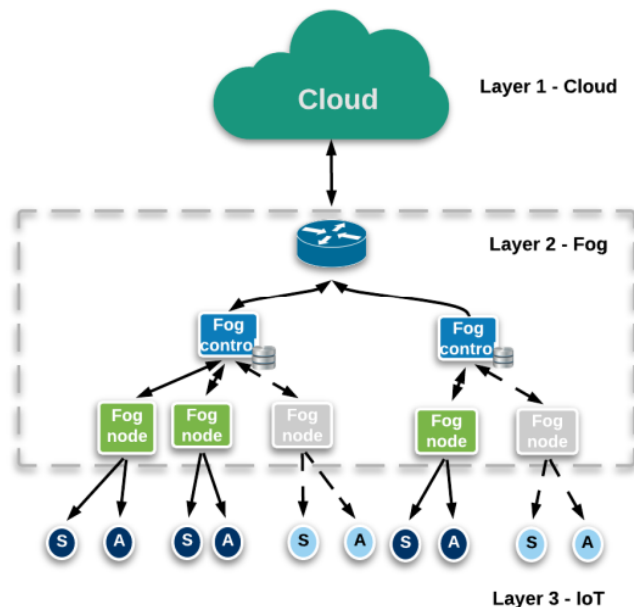


Figura 2. Arquitetura Fog Computing

Devido à sua recente introdução e emergência, não há nenhuma arquitetura padrão disponível em relação ao gerenciamento de recursos baseado em *Fog*, e por esse motivo ainda existe um modelo simples para esse propósito, considerando a previsão de recursos, a alocação de recursos e os custos de forma realista e dinâmica, considerando também o tipo, e características dos clientes, este modelo é adaptável às exigências de diferentes clientes.

A *Cisco Systems* tenta padronizar a computação em *Fog* e criou a *OpenFog Consortium* para promover sua visão [14]. A chave é reduzir a latência para aplicativos críticos, movendo a transferência de dados e o processamento para a borda da rede, mais perto do dispositivo IoT [17].

É necessária uma arquitetura aberta para desenvolver um ambiente de vida inteligente com potencialmente milhares de fornecedores [18]. O Consórcio *OpenFog* é formado por várias grandes empresas do setor tecnológico para acelerar a adoção da computação em *Fog* (www.openfogconsortium.org) e está tomando medidas para criar uma arquitetura de referência, comum que cobre plataforma de *hardware* e *software* com recursos altamente

sofisticados. A *OpenFog* pode beneficiar em muitos fatores a adesão da *Fog Computing*.

2.3. Taxonomia das Aplicações Fog Computing

Diferentes aplicações de computação em *Fog* foram sugeridas na literatura. Segundo OSANAIYE *et al.* (2017) [12], as categorias das aplicações de computação em *Fog* são divididas em: (i) Aplicações em tempo real; (ii) Aplicações quase em tempo real; (iii) Aplicações introduzidas em redes.

- (i) As aplicações em tempo real são aplicações de baixa latência, que funcionam dentro de um período de tempo pré-definido, sendo classificada pelo usuário como imediata ou urgente.
- (ii) Quase em tempo real, por outro lado, são aplicações que estão sujeitas a atraso de tempo introduzido pelo processamento de dados ou transmissão de rede, entre o momento em que ocorre um evento e o uso dos dados para processamento.
- (iii) A computação em *Fog* também pode ser introduzida em uma rede (para aplicações que não necessitem de processamento e transmissão em tempo real) para reduzir a quantidade de tráfego no núcleo de processamento.

3. Método de Pesquisa

Este artigo aplicou as diretrizes da (SLM, *Systematic Literature Mapping*), abordadas em [19] para pesquisar, selecionar, revisar e sintetizar os desafios da *Fog Computing* de publicações acadêmicas relevantes entre os anos de 2012 e 2018. Seguindo a SLM, esta seção traz os passos necessários para condução deste estudo. Para tanto faz-se necessário a apresentação do objetivo, questões de pesquisa, da estratégia de busca e da metodologia (critérios) utilizada para a seleção dos artigos de estudos primários. Além disso, cada subseção seguinte detalha cada um dos passos no processo de mapeamento sistemático realizado no presente estudo.

3.1. Objetivo do Mapeamento

Esta pesquisa tem como objetivo analisar publicações científicas com o propósito de identificar quais contribuições foram propostas e suas limitações no âmbito do paradigma *Fog Computing* em relação a abordagens para o gerenciamento de recursos que beneficiem a utilização de aplicações sensíveis à latência, do ponto de vista de pesquisadores, no contexto da IoT.

3.2. Questão de Pesquisa

De acordo com [19], a questão de pesquisa é primeira etapa da fase de planejamento, onde deve-se determinar o que se está procurando e definir quais resultados se deve alcançar com o mapeamento sistemático.

As questões apresentadas a seguir foram definidas para nortear a pesquisa e traçar um perfil das publicações existentes na literatura especializada, a saber:

(Q1) Quais as técnicas mais utilizadas para a alocação de recursos em um ambiente de *Fog Computing*?

(Q2) Quais as ferramentas de simulação são mais utilizadas?

(Q3) Quais as plataformas de embarcado são mais utilizadas?

(Q4) Quais são as métricas utilizadas para avaliar a abordagem?

(Q5) Quais os tipos de domínio de aplicações podem se beneficiar com a implantação da *Fog computing*?

Essas perguntas são a base para a construção da *string* de busca e para a definição dos critérios de inclusão e exclusão, servindo para especificar os pontos a serem observados e ponderados na condução da pesquisa, visando a obtenção de resultados que sejam como um resposta a essas questões.

3.3. Escopo e Restrições da Pesquisa

Com o objetivo de assegurar a viabilidade da pesquisa, foi definido um escopo para a mesma, que pode ser descrito por meio da definição de critérios de seleção de fontes e algumas restrições. Para a seleção das fontes de pesquisa, foram definidos os seguintes critérios:

- Disponibilidade para consultas web;
- Disponibilidade através do portal de periódicos da Capes (<https://www.periodicos.capes.gov.br>).
- Disponibilidade de artigos na íntegra por meio do domínio da UFS/IFS ou a partir da utilização da engine de busca *Google e/ou Google Scholar*;
- Disponibilidade de artigos em inglês, uma vez que é o idioma adotado pela grande maioria das conferências e periódicos nacionais e internacionais relacionados com tema de pesquisa.

3.4. Seleção de Fontes

Para a realização da busca por estudos relevantes, as bases eletrônicas alvo desse estudo foram as descritas a seguir na tabela I:

Tabela 1. BASES ELETRÔNICAS

Base	URL
Scopus	http://www.scopus.com
IEEEXplore	http://ieeexplore.ieee.org
ScienceDirect	http://www.sciencedirect.com
Springer	https://www.springer.com/
ACM	https://dl.acm.org/

3.5. Identificação de Palavras-Chave e Sinônimos

Baseado na questão de pesquisa, três principais palavras-chave são inicialmente identificadas, a saber, *Internet of Things*, *Fog Computing* e *Resource Management*. Além

disso, possíveis variações como sinônimos ou formas do singular/plural são consideradas.

Como forma de abranger sinônimos (e com base em alguns artigos encontrados previamente) foram adicionadas as palavras: *edge computing*, como forma de sinônimos para *Fog Computing*; *orchestration*, *service placement*, *resource provisioning* e *optimization* como forma de sinônimos para *resource management*.

3.6. Criação da String de Busca

A string de busca é um procedimento que deve ser adaptado para os motores de busca específicos, com o objetivo de produzir um retorno mais aproximado do ideal para a pesquisa. Uma *string* ineficaz pode trazer um grande número de falsos positivos [19]. O resultado da *string* é apresentado a seguir:

((fog AND computing OR fog OR edge AND computing OR edge) AND (internet AND of AND things OR iot) AND (resource AND management OR placement AND service OR quality AND of AND service OR qos OR orchestration OR application OR resource AND provisioning OR optimization OR optimisation))

As principais palavras-chave foram conectadas usando o operador lógico AND. Por sua vez, as possíveis variações e sinônimos foram conectados usando o operador lógico OR.

3.7. Seleção dos Estudos Primários

De acordo com [20], devem ser seguidos critérios de inclusão e exclusão para os artigos que são retornados pela *string* de busca. Além disso, os critérios de inclusão e exclusão dos estudos primários são os que vão nortear os pesquisadores na seleção dos estudos que foram coletados das fontes de pesquisas, além do que determina o rigor da pesquisa e impossibilita os vieses dos pesquisadores no momento da seleção. Foram definidos os seguintes critérios de inclusão:

- CI1 - Selecionar publicações que apresentam modelos de arquitetura, técnicas ou métodos aplicados a gerenciamento de serviços em ambientes *Fog Computing*;
- CI2 - Selecionar publicações que descrevem a implementação de técnicas ou métodos aplicados a gerenciamento de recursos em ambientes *Fog Computing*;

Em paralelo aos critérios de inclusão, foram definidos critérios de exclusão, descritos a seguir:

- CE1 - Não selecionar publicações que não satisfaçam a nenhum critério de inclusão;
- CE2 - Não selecionar publicações em que o idioma seja diferente do exigido;
- CE3 - Não selecionar publicações de artigos duplicados;

- CE4 - Não selecionar publicações em que abordagem voltada para o gerenciamento de recursos não seja em um ambiente *Fog Computing*.
- CE5 - Não selecionar publicações em que o conteúdo disponha apenas conceitos.

3.8. Processo de Seleção Preliminar (1° Filtro)

Foram selecionados artigos que apresentaram informações no título e no abstract relacionadas à questão de pesquisa principal.

3.9. Processo de Seleção Final (2° Filtro)

Como a leitura de duas informações (título e abstract) não são suficientes para identificar se o estudo é realmente relevante para a pesquisa realizada, torna-se necessário realizar a leitura completa dos estudos que restaram do 1° filtro. Dessa forma, esta fase do mapeamento, tem como objetivo fazer uma análise mais apurada dos estudos, identificando e extraíndo dados também de acordo com os critérios de inclusão e exclusão descritos anteriormente.

A tabela 2 apresenta o resultado do critério de seleção para cada base de pesquisa utilizada.

Tabela 2. TOTAL DE ARTIGOS SELECIONADOS COM CRITÉRIOS APLICADOS EM CADA BASE

Base	Resultado da pesquisa	Resultado do 1° filtro	Resultado do 2° filtro
Scopus	393	129	5
IEEEExplore	227	88	11
ScienceDirect	247	97	5
Springer	189	135	8
ACM	365	102	4
Total	1.421	551	33

Utilizando os 33 trabalhos selecionados após aplicados os critérios de inclusão e exclusão, foi feita a análise detalhada de cada artigo para posterior extração de informações que possam responder às questões de pesquisa propostas e consequentemente, alcançar o objetivo do presente trabalho.

4. Análise dos resultados

Nesta seção é apresentada a análise dos resultados obtidos a partir da extração de informações dos 33 trabalhos para responder às questões de pesquisa do presente estudo. A seguir são apresentados os resultados relacionando-os com as questões de pesquisa apresentadas na seção II deste mapeamento.

4.1. Quais as técnicas mais utilizadas para a alocação de recursos em um ambiente de *Fog Computing* (Q1)?

Esta questão visa apontar lacunas de pesquisa e como analisar tendências da utilização de técnicas de alocação de recursos em *Fog Computing*. Classificamos os trabalhos analisados em duas perspectivas, provisionamento e escalonamento de recursos. A primeira perspectiva está relacionada

a onde alocar e a segunda corresponde à quando e como alocar os resultados são apresentados a seguir.

4.1.1. Provisionamento de Recursos. De acordo [21], provisionamento de recursos é um processo que busca identificar quais recursos são adequados para uma determinada demanda de carga de trabalho, baseada nas necessidades de QoS descritas pelo usuário. Podemos considerar que um dos seus objetivos é tentar solucionar o problema de como e onde alocar recursos entre os dispositivos disponíveis no ambiente. Neste contexto, a eficiência está vinculada a vários objetivos, por exemplo, otimização de custo, energia, tempo de execução e utilização de recursos. Assim, definição de melhor execução pode variar de acordo com o modelo do sistema e o objetivo de cada pesquisa. A seguir apresentamos relevantes trabalhos que apresentam soluções para resolver o problema de provisionamento em um ambiente *Fog Computing*.

Skarlat *et al.* [22] propuseram um algoritmo genético como solução para alocação de serviços entre ambientes de nuvem e *Fog Computing*. Além disso, os autores apresentam o conceito de *fog colonies*, que subdivide o ambiente em pequenos grupos que são orquestrados por dispositivos *fog* com mais poder computacional e permite que tarefas sejam alocadas em colônias vizinhas caso necessitem.

Cardellini *et al.* [23], desenvolveram um modelo matemático para o problema de distribuição de aplicações DSP que considera atributos de aplicações e recursos disponíveis para flexibilizar parâmetros de qualidade de serviço. Além disso, os autores propuseram um protótipo baseado no *Apache Storm* como uma ferramenta para a comparação de diferentes políticas de alocação de recursos.

Taneja *et al.* [24] apresentaram uma heurística para mapear e alocar módulos de aplicações IoT entre o ambiente de nuvem e dispositivos na borda da rede. O algoritmo utiliza um estratégia gulosa que ordena de forma ascendente nós e módulos, e os aloca de acordo com as restrições de recursos computacionais.

Aazam *et al.* [25] apresentaram um modelo de gerenciamento de recursos orientado a serviços. A abordagem considera características do usuário para estimar valores para o provimento de serviços.

4.1.2. Escalonamento de Recursos. De acordo com Singh *et al.* [21], escalonamento de recursos é o processo de mapear e executar a carga de trabalho solicitada de acordo com os recursos selecionados pelo provisionador de recursos. Nesta direção, considerando técnicas de como os serviços e aplicações, podem ser movidos durante a execução e qual o melhor local para eles serem executados.

Velasques *et al.* [26], propuseram uma arquitetura para alocação de serviços de aplicações IoT entre a nuvem e os dispositivos na rede. O principal objetivo desta arquitetura é localizar em quais nós os serviços estão alocados e convenientemente migrá-los para outros nós de acordo com as condições da rede.

Tarneberg *et al.* [27] investigaram o posicionamento orientado a aplicações e apresentam um modelo de sistema

para rede de nuvens móveis com uma heurística de posicionamento dinâmico com o objetivo de garantir o desempenho de aplicações, minimizar o custo e resolver problemas de assimetria de recursos.

Urgaonkar *et al.* [28] discutiram uma abordagem baseada na técnica de otimização Lyapunov com o objetivo de resolver o problema de migração de serviços, atendendo aos desafios de mobilidade dos usuários e variação de demanda, preservando qualidade de serviço e performance.

Plachy *et al.* [29] apresentam um algoritmo de colocação de VM cooperativo e dinâmico, associado a outro algoritmo cooperativo para selecionar um caminho de comunicação adequado. Eles usam a migração de VMs para resolver problemas de mobilidade do usuário.

Em relação ao gerenciamento de recursos para tarefas de migração, [30] apresentaram um estudo com foco na migração de VMs, mas com diferentes objetivos de otimização tais como, priorizar o uso de energia verde e minimizar tempos de resposta.

Por fim, Yousaf e Taleb [31] propuseram um sistema de migração e gerenciamento de VMs que leva em conta a relação entre as unidades de recursos ao tomar decisões de migração. Eles apresentam esse trabalho no contexto de redes 5G, mas a técnica pode ser aplicada em outros domínios.

Kaur *et al.* [32] propuseram uma arquitetura que utiliza métodos de seleção e escalonamento de tarefas, com o objetivo de reduzir o consumo de energia ao mesmo tempo em que mantém níveis aceitáveis de SLA (*Service Level Agreement*) [33].

No trabalho proposto do Nassife *et al.* [34], apresentaram um mecanismo utilizando métodos de ponto interno como meio para otimizar a QoS de sistemas de tempo real sujeitos a restrições de energia e escalonamento.

O trabalho apresentado por [35], propôs um esquema que controla o tempo de desligamento das antenas e a conectividade de rede para reduzir o consumo de energia do sistema, enquanto satisfazem níveis aceitáveis de SLA.

Outro aspecto importante a ser considerado são os algoritmos aplicados para este tipo de problema. Os principais trabalhos apresentam soluções baseadas em heurísticas, programação linear e abordagens evolucionárias.

Xu *et al.* [5], propuseram um modelo para o problema de alocação de tarefas na borda da rede. Baseado neste modelo, os autores apresentam um mecanismo que maximiza a utilização de recursos entre a borda da rede e os provedores de serviço.

Bajpai *et al.* [36] propuseram um mecanismo de otimização para alocar serviços entre o núcleo e borda da rede, de tal forma que o custo de fornecer um serviço ao consumidor seja minimizado sem violar os requisitos de QoS dos consumidores. Além disso os autores, modelaram o custo do consumo de serviços para a nuvem em termos de disponibilidade do serviço, em diferentes níveis e custos de replicação nos níveis correspondentes. A estratégia apresentada permite tomar uma decisão sobre quando e onde um determinado serviço deve ser replicado.

Mennes *et al.* [37], apresentaram GRECO (*Genetic Algorithm for Reliable Application Placement in Hybrid Clouds*), um algoritmo genético como solução para um problema de alocação de recursos e o compararam com uma solução de programação de inteiro linear.

Wang *et al.* [38], propuseram um algoritmo online baseado em uma heurística gulosa para solucionar o problema de alocação de aplicações, com o objetivo de minimizar a utilização máxima de recursos em nós e links físicos entre dispositivos na borda da rede.

Em He *et al.* [39], os autores propuseram um algoritmo heurístico de otimização baseado no enxame de partículas MPSO-CO. Que realiza o balanceamento de carga entre os nós na borda da rede utilizando redes SDN.

Dentre os principais abordagens utilizadas nos trabalhos elencados deste mapeamento, podemos agrupar os principais algoritmos como solução baseada em heurísticas, abordagens evolucionárias, soluções baseadas em programação linear e soluções para problemas conhecidos.

Heurísticas: Heurísticas são algoritmos próximos a problemas do mundo real, que buscam soluções quase ótimas. Para melhorar a solução de heurística, muitas vezes o cenário e o ambiente do problema precisam ser refinados e descritos com mais detalhes. Exemplos de trabalhos que abordam algoritmos heurísticos são [24], [8], [40], [27], [41], [38].

Programação Linear: Outro grupo de trabalhos resolvem esse problema através de uma abordagem analítica, a programação linear. Por exemplo, [23], [26], [42], [43].

Abordagens Evolucionárias: Outra abordagem para este problema é usar um algoritmo genético, técnica bem consolidada em ambientes de nuvens tradicionais. Contudo, no contexto da *Fog Computing* poucos trabalhos, um exemplo é o de [22], [37].

A figura 3 ilustra os tipos de algoritmos em relação ao número de trabalhos por tipo de algoritmo, seguido por um quadro que apresenta as referências dos trabalhos onde estes algoritmos são utilizados.

Também podemos observar através do resumo apresentado na tabela 3, que a maioria dos trabalhos utilizam programação linear para a solução dos problemas de otimização, seguido de heurísticas e abordagens evolucionárias. O uso de algoritmos genéticos (GA) [44], já é um tema consolidado na área de computação em nuvem, contudo, em ambientes *Fog Computing* ainda são poucos explorados.

No tópico a seguir, apresentamos os resultados para a segunda questão de pesquisa.

4.2. Quais as ferramentas de simulação são mais utilizadas (Q2)?

Desenvolver um cenário real para avaliar o desempenho de políticas baseadas na *Fog* é em muitos casos inviável financeiramente e de difícil escalabilidade. Assim, pesquisadores precisam se apoiar em ferramentas eficientes para a simulação de um ambiente *Fog*. Nesta direção, a resposta para esta questão é ilustrada na Fig. 4, que apresenta um gráfico de frequência, representando as ferramenta utilizadas

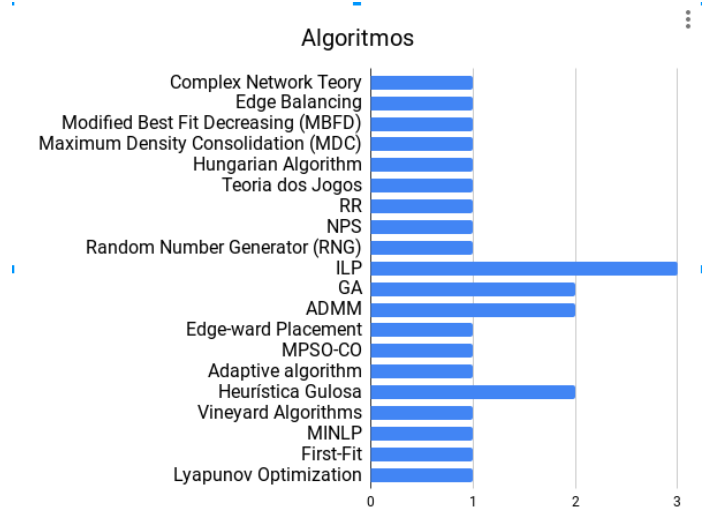


Figura 3. Algoritmos

Tabela 3. ALGORITMOS ANALISADOS NOS TRABALHOS SELECIONADOS

Trabalho	Algoritmos
[35]	Complex Network Theory
[45]	Edge Balancing
[46]	Modified Best Fit Decreasing (MBFD)
[46]	Maximum Density Consolidation (MDC)
[47]	Hungarian Algorithm
[48]	Teoria dos Jogos
[49]	RR
[49]	NPS
[49]	Random Number Generator (RNG)
[50], [23], [26], [43]	ILP
[22], [37]	GA
[51], [52]	Alternating Direction Method of Multipliers (ADMM)
[9]	Edge-ward Placement
[39]	MPSO-CO
[36]	Adaptive Algorithm
[38], [24]	Heurística Gulosa
[38]	Vineyard Algorithms
[53]	MINLP
[22]	First-Fit
[28]	Lyapunov Optimization

para validação dos experimentos gerenciamento de recursos em *Fog Computing*.

Conforme resultado apresentado na tabela 4, a ferramenta IBM CPLEX [54], é a mais utilizada nestas pesquisas. Esta informação reforça os resultados apresentados na questão 1, onde apresenta que as técnicas mais utilizadas neste tipo de problema é a programação linear. Outro aspecto importante a se observar é que, algumas ferramentas adotadas são extensões do CloudSim [55]. A exemplo podemos citar, iFogsim [9].

4.3. Quais as plataformas de embarcado são mais utilizadas (Q3)?

A resposta para esta pergunta é ilustrada na Fig. 5, que apresenta um gráfico de pizza representando os tipos de

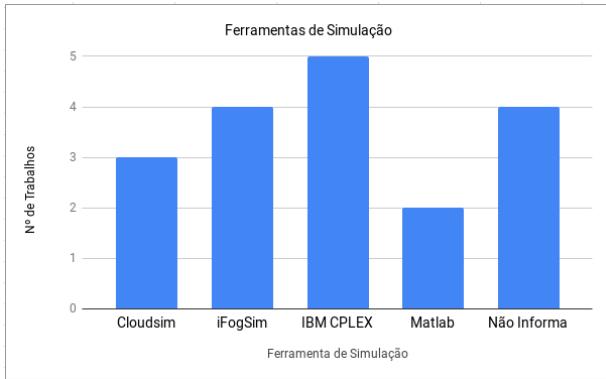


Figura 4. Ferramentas de Simulação

Tabela 4. FERRAMENTAS DE SIMULAÇÃO

Trabalhos	Ferramenta de Simulação
[25], [56], [45]	Cloudsim
[50], [36], [38], [53]	iFogSim
[46], [22], [24], [9], [57]	IBM CEPLEX
[47], [48]	Matlab
[58], [52], [37], [28]	Não Informa

plataforma utilizadas em experimentos e na implementação da *Fog Computing*.

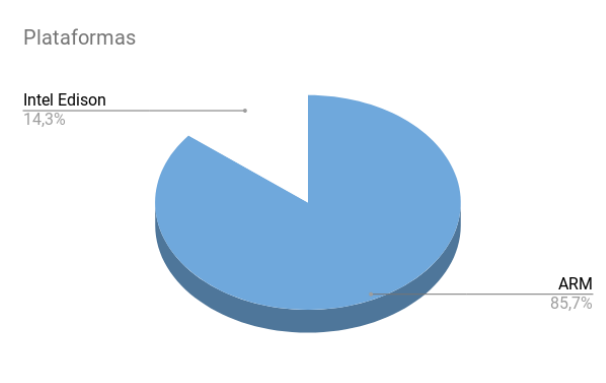


Figura 5. Plataformas

O quadro 5 apresenta os artigos que utilizaram cada uma das plataformas apresentadas no gráfico da Fig. 5. A partir dos resultados apresentados, pode-se observar que os pesquisadores trabalham, em sua maioria, com a plataforma ARM. Um dos principais motivos para que isto ocorra pode estar relacionado ao custo e facilidade de obtenção destes dispositivos, grande oferta de dispositivos embarcados no mercado, também denominados SBC (*Single-Board Computers*). Como exemplo, é possível citar Raspberry Pi Zero, Raspberry 2 Model B, Raspberry Pi 3 Model B, entre outros. É digno de nota observar que apenas um dos trabalhos analisados utilizou a plataforma Intel Edison.

Tabela 5. PLATAFORMAS

Autores	Plataformas
[59], [60], [61], [62], [63], [64]	ARM
[65]	Intel Edison

4.4. Quais são as métricas utilizadas na avaliação das abordagens (Q4)?

Métricas são indicadores de desempenho utilizados para transformar em números questões como desempenho, qualidade de serviço, entre outros aspectos. Quando bem utilizadas as métricas ajudam a melhorar a qualidade dos serviços e servem como indicadores de prevenção de problemas no ambiente.

Neste trabalho, as métricas estão relacionadas ao tipo de recursos gerenciado na *Fog*. Assim, os trabalhos de pesquisa relacionados são analisados com base no cumprimento de uma seleção de critérios importantes. A fim de agrupar e comparar adequadamente os trabalhos de pesquisa, os critérios incluem o tipo recurso do ambiente gerenciado, por exemplo, recursos computacionais, comunicação, energia e custo, bem como tópicos específicos abordados pelos autores.

Comunicação: As métricas relacionadas às questões de comunicação no ambiente *Fog* incluem aspectos como, latência e largura de banda. Em [59], os autores propuseram e avaliaram um esquema de alocação de recursos baseado no uso de contêineres, com o objetivo de reduzir o tráfego de dados gerado na rede por dispositivos inteligentes.

Computação: Esta categoria agrupa os trabalhos que tem por objetivo otimizar aspectos como a utilização de recursos computacionais tais como CPU, Memória, etc. No trabalho [50], os autores apresentaram um modelo de provisionamento de serviços IoT, formulado como um problema inteiro misto com o objetivo de otimizar a utilização de recursos em nós físicos e o consumo de largura de banda na rede.

Energia: tais como, consumo de energia e a sua relação entre consumo de desempenho. Em [35], os autores propuseram um esquema de operação de um sistema de computação sem fio (WCS), com o objetivo de controlar o tempo de desligamento das antenas e a conectividade de rede para reduzir o consumo de energia enquanto satisfaz níveis aceitáveis de SLA.

Custo Em certos casos, os fatores relacionados aos custos envolvem tanto os provedores de serviços quanto os usuários, tornando-se muito influentes no provisionamento de serviços na *Fog*. Isto inclui, adotar estratégias para reduzir o custo com a utilização de serviços providos pela nuvem. Em [25], os autores apresentam um modelo de gerenciamento de recursos orientado a serviços. A abordagem considera características do usuário para estimar valores para o provimento de serviços.

Segurança Outro aspecto importante a ser considerado é o da segurança, uma vez que a *Fog* é composta por uma rede de dispositivos heterogêneos, novos desafios precisam

ser superados. No trabalho de Dsouza *et al.* [61], os autores apresentam uma política de gerenciamento de recursos com suporte a colaboração segura e interoperabilidade entre diferentes níveis de QoS, baseado em um modelo de arquitetura *Fog Computing*.

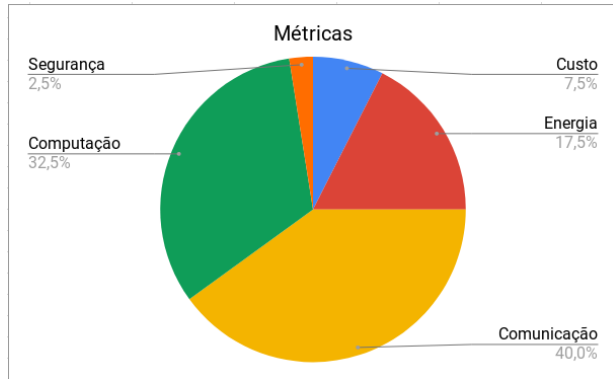


Figura 6. Métricas

A tabela 6 representa os critérios selecionados para a classificação dos trabalhos relacionados. É digno de nota observar que, embora existem esforços de otimização diferentes tipos de recursos, os resultados evidenciam que a maioria dos trabalhos analisados concentram-se em otimizar dois principais aspectos, comunicação e recursos computacionais, representando um total de 32,5% e 40% respectivamente do total de trabalhos analisados.

Tabela 6. MÉTRICAS

Trabalho	Métricas
[25], [49], [66]	Custo
[35], [60], [46], [47], [9], [63], [66]	Energia
[35], [23], [60], [22], [24], [47], [47], [9], [39], [52], [36], [62], [63], [28], [67], [66]	Comunicação
[50], [45], [22], [24], [65] [53] [9], [36], [37], [63], [38], [53], [66]	Computação
[47]	Segurança

4.5. Quais os domínios de aplicações podem se beneficiar com a implantação da *Fog computing* (Q5) ?

A resposta para esta pergunta é apresentada na figura 7. Os domínios de aplicação são descritos de acordo com a área de atuação exploradas nos trabalho selecionados. A figura 7, representada por um gráfico de pizza, ilustra os resultados obtidos.

Multimídia: As aplicações multimídias podem ser consideradas como programas e sistemas em que a comunicação entre homem e o computador se dá através de múltiplos meios de representação da informação, como som e imagem animada. No âmbito da IoT, estas aplicações podem ser representadas como por exemplo, com a interação entre câmeras IP e um sistema de vigilância. Gupta *et al.* [9] apresentaram uma estratégia para diminuir a latência de informações entre o sistema de detecção movimento e o ambiente de monitoramento.

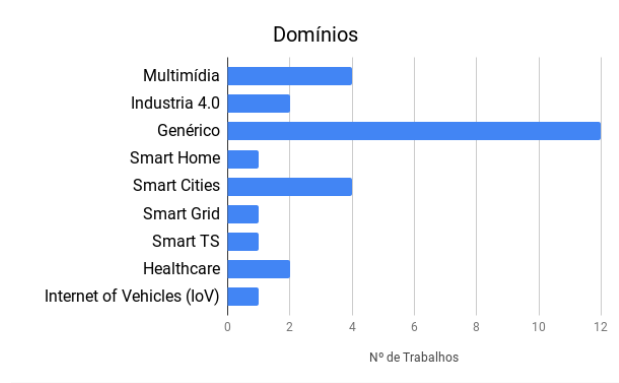


Figura 7. Domínios de Aplicações

Smart Cities: O conceito de *Smart Cities* engloba a forma com que serviços essenciais tais como, energia, água, transporte, etc, são geridos em conjunto para proporcionar um ambiente limpo, econômico e seguro para o convívio das pessoas [68]. Informações oportunas de logística são coletadas e fornecidas ao público por todos os meios disponíveis ao usuário, por exemplo, por meio de redes de mídia social.

Smart Grid: Este conceito define uma rede elétrica flexível, resiliente, performática segura e permitindo a exploração de recursos de rede em tempo real, de modo a otimizar a produtividade. A *Fog Computing* pode beneficiar esta área através do intermédio da implantação de milhares de dispositivos de controle, sensores e medidores inteligentes.

Smart Home: O conceito de *Smart Home* pode ser descrito como um conjunto de dispositivos e sensores conectados a ambientes domésticos. Além disso, para algumas tarefas, como por exemplo, análise de vídeo em tempo real, o ambiente necessita de recursos computacionais além do que é encontrado na maioria dos dispositivos IoT.

Smart Transportation: Neste conceito sensores são embutidos na infraestrutura para monitorar informações de trânsito em tempo real e otimizar caminhos. Outras questões também podem ser analisadas, tais como, fadiga estrutural, monitoramento de acidentes para gerenciamento de incidentes e coordenação de respostas de emergência.

Genérico: Diversos trabalhos apresentam soluções para o gerenciamento de recursos na *Fog*, contudo, não estão associados diretamente a nenhum domínio em particular. Para esta situação, agrupamos estes trabalhos em um domínio genérico, uma vez que, o método proposto pode beneficiar deferentes domínios de aplicações. Entre eles, podemos citar os trabalhos de [23], [50] e [63].

Healthcare: Aplicações Healthcare são destinadas a dar apoio a questões ligadas à saúde e bem estar das pessoas. Atividades, como triagem, monitoramento de pacientes, monitoramento de pessoal são algumas das que podem colaborar com informações preditivas, para ajudar a tomada de decisão de profissionais médicos ou decisões políticas em cenários de pandemia.

Internet of Vehicles (IoV): Em aplicações IoV, a *Fog Computing* pode ser integrada em redes veiculares, permi-

tindo a criação autônoma de uma comunicação sem fio entre veículos para troca de dados e aumento de recursos.

Como pode-se analisar, a grande maioria dos trabalhos não apresentam um domínio de aplicação específico. No quadro 7 é apresentado o resultado completo, com os artigos onde são apresentados os domínios de cada aplicação.

Tabela 7. ÁREAS DE DOMÍNIO POR APLICAÇÃO

Trabalhos	Domínio
[25], [56], [51], [9], [38]	Multimídia
[35], [22]	Industria 4.0
[23], [50], [45], [24], [36], [37], [62], [63], [28], [65], [52], [48]	Genérico
[59]	Smart Home
[60], [47], [26], [69]	Smart Cities
[46]	Smart Grid
[61]	Smart Transportation System
[9], [66]	Healthcare
[58]	Internet of Vehicles (IoV)

5. Desafios e Direções Futuras

Apesar dos esforços recentes em pesquisa delineados na sessão 2, muitos desafios ainda permanecem em aberto. Em particular, identificamos os seguintes desafios para pesquisas futuras.

5.1. Segurança

Manter segurança em ambientes de computação em *Fog* é um desafio consideravelmente difícil, uma vez que seus componentes localizam-se principalmente na borda da rede próximo ao dispositivos e sensores. Não obstante, embora seja possível utilizar técnicas semelhantes em um ambiente *Fog*, as pesquisas ainda se encontram em seu estágio inicial. No entanto, na literatura existente, as preocupações de segurança na computação *Fog* foram discutidas em termos de autenticação de usuários, privacidade, troca segura de dados, ataque DoS, etc. Outros aspectos são igualmente importantes, tais como, segurança e privacidade são abordados no trabalhos de [70] e [71].

5.2. Abordagens Autônomicas

É crescente o surgimento de soluções da IoT que buscam resolver problemas de alocação de recursos. Além disso uma promissora direção de pesquisa parte do princípio de explorar interações diretas entre coisas, com o objetivo de monitorar e controlar a si mesmos e o ambiente circundante com a mínima intervenção humana. Nesta direção, abordagens autônomicas são amplamente exploradas na literatura em contextos de computação em nuvem, como solução para problemas de gerenciamento de recursos [72], [21], [73], porém no âmbito da *Fog Computing*, ao melhor do nosso conhecimento, não encontramos trabalhos que discutem este tipo de abordagem.

5.3. Migração

Um dos desafios que a mobilidade impõe às plataformas de IoT é a capacidade de fornecer e manter os recursos

de computação e armazenamento próximos das “coisas” e, no caso de uma mudança repentina no local, a plataforma deve ser capaz de reconfigurar o contexto de comutação e orquestrar a alocação de recursos e a migração do estado para um novo local. As técnicas de virtualização de recursos baseadas em contêineres são candidatos em potencial para permitir migrações de serviços de forma rápida, movendo apenas o estado do contêiner. Tecnologias como, *Docker* [74] e *CRIU* [75] são alguns exemplos representativos, mas definir técnicas para a migração de serviços entre dispositivos da *Fog* e a nuvem ainda é um desafio a ser superado.

5.4. Virtualização

Os modelo de virtualização de serviços disponibilizados através de nós *Fog Computing* também é um desafio a ser superado. Rede Definidas por Software (SDN) é considerada como um dos principais facilitadores da rede virtualizada. SDN resume-se em uma arquitetura que permite que o controle de rede se torne diretamente programável e a infraestrutura subjacente seja abstraída para as aplicações e implementadas em servidores separados. Um dos aspectos importantes do SDN é fornecer suporte para NFV. NFV é um conceito de arquitetura que virtualiza as funções tradicionais de rede tais como, conversão de endereços de rede (NAT), *firewall*, detecção de intrusões, serviço de nomes de domínio (DNS), etc.

5.5. Ferramentas de Benchmark

Definir métodos para avaliar ambientes IoT através de ferramentas de *benchmark* é uma tarefa importante, principalmente no que se diz respeito à avaliação de técnicas de gerenciamento. Ao melhor do nosso conhecimento, são poucos os trabalhos nesta área. Como exemplo podemos citar, RIOTBench [76], um conjunto de micro *benchmarks* desenvolvidos para avaliar sistemas de processamento de fluxo distribuído para aplicações IoT.

5.6. Algoritmos para solução de problemas com múltiplos objetivos

A maioria dos estudos sobre otimização de recursos abordam um único critério. No entanto, muitos problemas do mundo real, exigem a consideração de vários critérios. Por esta razão, pesquisas recentes tendem a olhar para situações de múltiplo objetivo.

5.7. Gerenciamento de Energia

Os nós da *Fog* precisam lidar com um grande número de solicitações simultâneas de serviço provenientes dos dispositivos IoT. Uma das soluções triviais é implantar os nós *Fog* no ambiente de acordo com a demanda. No entanto, esta abordagem aumentará em grande medida o número de nós *Fog* ativos, afetando o consumo total de energia do sistema. Portanto, ao responder a solicitações de grande número de

serviços, o gerenciamento de energia adequado dentro da rede da *Fog* é muito necessário. No entanto, na literatura existente, um dos objetivos da adoção da *Fog Computing* é minimizar o consumo de energia nos *Data Centers* da nuvem. Além disso, para gerenciar o consumo de energia, faz-se necessário a consolidação de estratégias de migração de tarefas de um nó para outro. A investigação para as soluções de migração de tarefas também pode ser um campo potencial de pesquisa baseada em *Fog Computing*.

6. Conclusão

Neste trabalho foi apresentado o processo de um estudo de mapeamento sistemático, desde seu planejamento até a extração de dados, cujo objetivo foi a análise e identificação de abordagens de gerenciamento de Recursos utilizados em pesquisas relacionadas à *Fog Computing*, apresentamos também seu conceito, características e a taxonomia das suas aplicações.

O processo seguido para realização do mapeamento consistiu da criação de questões de pesquisa que nortearam o desenvolvimento da busca e seleção de artigos para análise mais aprofundada. De modo que foram encontrados 1.421 artigos em cinco bases de dados distintas (Scopus, IEEE Xplorer, ScienceDirect, Springer e ACM), sendo que destes, ao serem aplicados os critérios de seleção, reduziram-se a 33 artigos considerados relevantes para o objetivo proposto.

À partir da análise dos estudos primários pode-se encontrar as diferentes estratégias de gerenciamento de recursos a serem desenvolvidos na área da *Fog Computing*. Descobriu-se ainda que a maioria dos trabalhos selecionados utilizam algoritmos bastante diversificados, tendo destaque para os algoritmos genético e ILP, que já são bem consolidados em trabalhos relacionados à ambientes de nuvem (Q1).

Além disso, embora os resultados das ferramentas de simulação não apresentem uma ferramenta com maior destaque. Observamos que, iFogsim é uma extensão do Cloudsim, o que nos permite concluir à partir destas informações, que Cloudsim é a ferramenta mais utilizada para simulação de ambientes *Fog Computing* (Q2). Por outro lado, no que se diz respeito ao tipo de plataforma utilizada nos trabalhos, concluímos que os pesquisadores trabalham, em sua maioria, com plataforma ARM (Q3). Não obstante, a maioria das abordagens analisadas deram ênfase a otimização de aspectos de comunicação e computação (Q4).

Existem diversas áreas de domínio de aplicação que podem se beneficiar da *Fog Computing*, contudo, a maioria dos trabalhos optaram desenvolver as estratégias sem associá-las a um determinado domínio (Q5).

Por fim, apresentamos vários desafios e futuras direções para pesquisas em ambiente *Fog Computing*. Percebemos que faltam muitas questões a serem investigadas no meio acadêmico para que sua implementação seja uma realidade, porém ficou evidente que sua adesão é inevitável para a internet do futuro.

Desta forma, acredita-se que o presente trabalho contribui de maneira relevante para a academia, fornecendo suporte para pesquisadores da área de *Fog Computing*, como

uma fonte de consulta para estratégias, algoritmos, tipos de ferramentas, métricas e domínios de aplicação para garantir uma resposta adequada aos problemas no gerenciamentos de recursos.

Referências

- [1] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of things (iot): A vision, architectural elements, and future directions, *Future generation computer systems* 29 (7) (2013) 1645–1660.
- [2] J. Rivera, R. v. D. MEULEN, Gartner says the internet of things installed base will grow to 26 billion units by 2020. 2013, URL: <http://www.gartner.com/newsroom/id/2636073> (visited on 04/05/2016).
- [3] D. Lohin, B. M. Tudor, H. Zhang, B. C. Ooi, Y. M. Teo, A performance study of big data on small nodes, *Proceedings of the VLDB Endowment* 8 (7) (2015) 762–773.
- [4] W.-T. Sung, Y.-C. Chiang, Improved particle swarm optimization algorithm for android medical care iot using modified parameters, *Journal of medical systems* 36 (6) (2012) 3755–3763.
- [5] B. Xu, L. Da Xu, H. Cai, C. Xie, J. Hu, F. Bu, Ubiquitous data accessing method in iot-based information system for emergency medical services, *IEEE Transactions on Industrial Informatics* 10 (2) (2014) 1578–1586.
- [6] M. Gerla, E.-K. Lee, G. Pau, U. Lee, Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds, in: *Internet of Things (WF-IoT)*, 2014 IEEE World Forum on, IEEE, 2014, pp. 241–246.
- [7] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog computing and its role in the internet of things, in: *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, ACM, 2012, pp. 13–16.
- [8] O. Skarlat, M. Nardelli, S. Schulte, S. Dustdar, Towards qos-aware fog service placement, in: *Fog and Edge Computing (ICFEC)*, 2017 IEEE 1st International Conference on, IEEE, 2017, pp. 89–96.
- [9] H. Gupta, A. V. Dastjerdi, S. K. Ghosh, R. Buyya, ifogsim: A toolkit for modeling and simulation of resource management techniques in internet of things, edge and fog computing environments, *arXiv preprint arXiv:1606.02007*.
- [10] S. Chen, T. Zhang, W. Shi, Fog computing, *IEEE Internet Computing* (2) (2017) 4–6.
- [11] J. Zhu, D. S. Chan, M. S. Prabhu, P. Natarajan, H. Hu, F. Bonomi, Improving web sites performance using edge servers in fog computing architecture, in: *2013 IEEE Seventh International Symposium on Service-Oriented System Engineering*, IEEE, 2013, pp. 320–323.
- [12] O. Osanaiye, S. Chen, Z. Yan, R. Lu, K.-K. R. Choo, M. Dlodlo, From cloud to fog computing: A review and a conceptual live vm migration framework, *IEEE Access* 5 (2017) 8284–8300.
- [13] M. Hajibaba, S. Gorgin, A review on modern distributed computing paradigms: Cloud computing, jungle computing and fog computing, *Journal of computing and information technology* 22 (2) (2014) 69–84.
- [14] J. Machado, E. Moreno, A. Ribeiro, A review of computing fog and its research challenges, *Journal on Advances in Theoretical and Applied Informatics* 3 (2) (2017) 32–39. doi:10.26729/jadi.v3i2.2469. URL <http://revista.univem.edu.br/jadi/article/view/2469>
- [15] A. Jain, P. Singhal, Fog computing: Driving force behind the emergence of edge computing, in: *System Modeling & Advancement in Research Trends (SMART)*, International Conference, IEEE, 2016, pp. 294–297.
- [16] S. Li, M. A. Maddah-Ali, A. S. Avestimehr, Coding for distributed fog computing, *IEEE Communications Magazine* 55 (4) (2017) 34–40.

- [17] D. S. Linthicum, Connecting fog and cloud computing, *IEEE Cloud Computing* 4 (2) (2017) 18–20.
- [18] B. McMillin, T. Zhang, Fog computing for smart living, *Computer* 50 (2) (2017) 5–5.
- [19] K. Petersen, S. Vakkalanka, L. Kuzniarz, Guidelines for conducting systematic mapping studies in software engineering: An update, *Information and Software Technology* 64 (2015) 1–18.
- [20] B. Kitchenham, Procedures for performing systematic reviews, Keele, UK, Keele University 33 (2004) (2004) 1–26.
- [21] S. Singh, I. Chana, M. Singh, The journey of qos-aware autonomic cloud computing, *IT Professional* 19 (2) (2017) 42–49.
- [22] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski, P. Leitner, Optimized iot service placement in the fog, *Service Oriented Computing and Applications* (2017) 1–17.
- [23] V. Cardellini, V. Grassi, F. L. Presti, M. Nardelli, On qos-aware scheduling of data stream applications over fog computing infrastructures, in: *Computers and Communication (ISCC), 2015 IEEE Symposium on*, IEEE, 2015, pp. 271–276.
- [24] M. Taneja, A. Davy, Resource aware placement of iot application modules in fog-cloud computing paradigm, in: *Integrated Network and Service Management (IM), 2017 IFIP/IEEE Symposium on*, IEEE, 2017, pp. 1222–1228.
- [25] M. Aazam, E.-N. Huh, Fog computing micro datacenter based dynamic resource estimation and pricing model for iot, in: *Advanced Information Networking and Applications (AINA), 2015 IEEE 29th International Conference on*, IEEE, 2015, pp. 687–694.
- [26] K. Velasquez, D. P. Abreu, M. Curado, E. Monteiro, Service placement for latency reduction in the internet of things, *Annals of Telecommunications* 72 (1-2) (2017) 105–115.
- [27] W. Tärneberg, A. Mehta, E. Wadbro, J. Tordsson, J. Eker, M. Kihl, E. Elmroth, Dynamic application placement in the mobile cloud network, *Future Generation Computer Systems* 70 (2017) 163–177.
- [28] R. Urgaonkar, S. Wang, T. He, M. Zafer, K. Chan, K. K. Leung, Dynamic service migration and workload scheduling in edge-clouds, *Performance Evaluation* 91 (2015) 205–228.
- [29] J. Plachy, Z. Becvar, E. C. Strinati, Dynamic resource allocation exploiting mobility prediction in mobile edge computing, in: *Personal, Indoor, and Mobile Radio Communications (PIMRC), 2016 IEEE 27th Annual International Symposium on*, IEEE, 2016, pp. 1–6.
- [30] T. G. Rodrigues, K. Suto, H. Nishiyama, N. Kato, Hybrid method for minimizing service delay in edge cloud computing through vm migration and transmission power control, *IEEE Transactions on Computers* 66 (5) (2017) 810–819.
- [31] F. Z. Yousaf, T. Taleb, Fine-grained resource-aware virtual network function management for 5g carrier cloud, *IEEE Network* 30 (2) (2016) 110–115.
- [32] K. Kaur, T. Dhand, N. Kumar, S. Zeadally, Container-as-a-service at the edge: Trade-off between energy efficiency and service availability at fog nano data centers, *IEEE Wireless Communications* 24 (3) (2017) 48–56.
- [33] P. Leitner, W. Hummer, B. Satzger, C. Inzinger, S. Dustdar, Cost-efficient and application sla-aware client side request scheduling in an infrastructure-as-a-service cloud, in: *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on*, IEEE, 2012, pp. 213–220.
- [34] R. Nassiffe, E. Camponogara, G. Lima, D. Mossé, Optimising qos in adaptive real-time systems with energy constraint varying cpu frequency, *International Journal of Embedded Systems* 8 (5-6) (2016) 368–379.
- [35] K. Suto, H. Nishiyama, N. Kato, C.-W. Huang, An energy-efficient and delay-aware wireless computing system for industrial wireless sensor networks, *IEEE Access* 3 (2015) 1026–1035.
- [36] A. Bajpai, B. Choudhury, S. Choudhury, An adaptive and elastic cloud based framework for service oriented computing in internet of things, in: *Communication Systems and Networks (COMSNETS), 2017 9th International Conference on*, IEEE, 2017, pp. 460–463.
- [37] R. Mennes, B. Spinnewyn, S. Latré, J. F. Botero, Greco: A distributed genetic algorithm for reliable application placement in hybrid clouds, in: *2016 5th IEEE International Conference on Cloud Networking (Cloudnet)*, IEEE, 2016, pp. 14–20.
- [38] S. Wang, R. Urgaonkar, T. He, K. Chan, M. Zafer, K. K. Leung, Dynamic service placement for mobile micro-clouds with predicted future costs, *IEEE Transactions on Parallel and Distributed Systems* 28 (4) (2017) 1002–1016.
- [39] X. He, Z. Ren, C. Shi, J. Fang, A novel load balancing strategy of software-defined cloud/fog networking in the internet of vehicles, *China Communications* 13 (Supplement2) (2016) 140–149.
- [40] K. Habak, E. W. Zegura, M. Ammar, K. A. Harras, Workload management for dynamic mobile device clusters in edge femtoclouds, in: *Proceedings of the Second ACM/IEEE Symposium on Edge Computing*, ACM, 2017, p. 6.
- [41] A. R. Zamani, M. Zou, J. Diaz-Montes, I. Petri, O. Rana, A. Anjum, M. Parashar, Deadline constrained video analysis via in-transit computational environments, *IEEE Transactions on Services Computing*.
- [42] W. Liu, R. Shinkuma, T. Takahashi, Opportunistic resource sharing in mobile cloud computing: The single-copy case, in: *Network Operations and Management Symposium (APNOMS), 2014 16th Asia-Pacific*, IEEE, 2014, pp. 1–6.
- [43] Y. Liu, M. J. Lee, Y. Zheng, Adaptive multi-resource allocation for cloudlet-based mobile cloud computing system, *IEEE Transactions on Mobile Computing* 15 (10) (2016) 2398–2410.
- [44] J. Á. Morell, E. Alba, Running genetic algorithms in the edge: A first analysis, in: *Conference of the Spanish Association for Artificial Intelligence*, Springer, 2018, pp. 251–261.
- [45] S. Verma, A. K. Yadav, D. Motwani, R. Raw, H. K. Singh, An efficient data replication and load balancing technique for fog computing environment, in: *Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on*, IEEE, 2016, pp. 2888–2895.
- [46] Z. Pooranian, M. Shojafar, P. G. V. Naranjo, L. Chiaraviglio, M. Conti, A novel distributed fog-based networked architecture to preserve energy in fog data centers, in: *Mobile Ad Hoc and Sensor Systems (MASS), 2017 IEEE 14th International Conference on*, IEEE, 2017, pp. 604–609.
- [47] R. Deng, R. Lu, C. Lai, T. H. Luan, H. Liang, Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption, *IEEE Internet of Things Journal* 3 (6) (2016) 1171–1181.
- [48] I. Farris, L. Militano, M. Nitti, L. Atzori, A. Iera, Mifaas: A mobile-iot-federation-as-a-service model for dynamic cooperation of iot cloud providers, *Future Generation Computer Systems* 70 (2017) 126–137.
- [49] M. Aazam, M. St-Hilaire, C.-H. Lung, I. Lambadaris, Mefore: Qoe based resource estimation at fog to enhance qos in iot, in: *Telecommunications (ICT), 2016 23rd International Conference on*, IEEE, 2016, pp. 1–5.
- [50] E. El Rachkidi, N. Agoulmine, D. Belaid, N. Chendeb, Towards an efficient service provisioning in cloud of things (cot), in: *Global Communications Conference (GLOBECOM), 2016 IEEE*, IEEE, 2016, pp. 1–6.
- [51] C. T. Do, N. H. Tran, C. Pham, M. G. R. Alam, J. H. Son, C. S. Hong, A proximal algorithm for joint resource allocation and minimizing carbon footprint in geo-distributed fog computing, in: *Information Networking (ICOIN), 2015 International Conference on*, IEEE, 2015, pp. 324–329.

- [52] J. Xu, B. Palanisamy, H. Ludwig, Q. Wang, Zenith: Utility-aware resource allocation for edge computing, in: *Edge Computing (EDGE), 2017 IEEE International Conference on*, IEEE, 2017, pp. 47–54.
- [53] D. Zeng, L. Gu, S. Guo, Z. Cheng, S. Yu, Joint optimization of task scheduling and image placement in fog computing supported software-defined embedded system, *IEEE Transactions on Computers* 65 (12) (2016) 3702–3712.
- [54] I. C. Optimizer, {Available Online} <http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer>, Accessed May.
- [55] A. Beloglazov, et al., Cloudsim: A framework for modeling and simulation of cloud computing infrastructures and services, *Cloud Computing and Distributed Systems (CLOUDS) Laboratory, Department of Computer Science and Software Engineering, the University of Melbourne, Australia*.
- [56] S. Agarwal, S. Yadav, A. K. Yadav, An efficient architecture and algorithm for resource provisioning in fog computing, *International Journal of Information Engineering and Electronic Business* 8 (1) (2016) 48.
- [57] R. Mahmud, F. L. Koch, R. Buyya, Cloud-fog interoperability in iot-enabled healthcare solutions, in: *Proceedings of the 19th International Conference on Distributed Computing and Networking, ACM*, 2018, p. 32.
- [58] J. He, J. Wei, K. Chen, Z. Tang, Y. Zhou, Y. Zhang, Multi-tier fog computing with large-scale iot data analytics for smart cities, *IEEE Internet of Things Journal*.
- [59] T. Renner, M. Meldau, A. Kliem, Towards container-based resource management for the internet of things, in: *Software Networking (ICSN), 2016 International Conference on*, IEEE, 2016, pp. 1–5.
- [60] P. Charalampidis, E. Tragos, A. Fragkiadakis, A fog-enabled iot platform for efficient management and data collection, in: *Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2017 IEEE 22nd International Workshop on*, IEEE, 2017, pp. 1–6.
- [61] C. Dsouza, G.-J. Ahn, M. Taguinod, Policy-driven security management for fog computing: Preliminary framework and a case study, in: *Information Reuse and Integration (IRI), 2014 IEEE 15th International Conference on*, IEEE, 2014, pp. 16–23.
- [62] S. Hoque, M. S. de Brito, A. Willner, O. Keil, T. Magedanz, Towards container orchestration in fog computing infrastructures, in: *Computer Software and Applications Conference (COMPSAC), 2017 IEEE 41st Annual, Vol. 2*, IEEE, 2017, pp. 294–299.
- [63] R. Morabito, I. Farris, A. Iera, T. Taleb, Evaluating performance of containerized iot services for clustered devices at the network edge, *IEEE Internet of Things Journal* 4 (4) (2017) 1019–1030.
- [64] C. Pahl, B. Lee, Containers and clusters for edge cloud architectures—a technology review, in: *Future Internet of Things and Cloud (FiCloud), 2015 3rd International Conference on*, IEEE, 2015, pp. 379–386.
- [65] M. Samaniego, R. Deters, Hosting virtual iot resources on edge-hosts with blockchain, in: *Computer and Information Technology (CIT), 2016 IEEE International Conference on*, IEEE, 2016, pp. 116–119.
- [66] R. Mahmud, R. Kotagiri, R. Buyya, Fog computing: A taxonomy, survey and future directions, in: *Internet of Everything*, Springer, 2018, pp. 103–130.
- [67] V. Cardellini, V. Grassi, F. Lo Presti, M. Nardelli, Optimal operator replication and placement for distributed stream processing systems, *ACM SIGMETRICS Performance Evaluation Review* 44 (4) (2017) 11–22.
- [68] K. Geisler, The relationship between smart grids and smart cities, *IEEE Smart Grid Newsletter*.
- [69] X. Masip-Bruin, E. Marin-Tordera, A. Jukan, G.-J. Ren, Managing resources continuity from the edge to the cloud: Architecture and performance, *Future Generation Computer Systems* 79 (2018) 777–785.
- [70] S. Yi, Z. Qin, Q. Li, Security and privacy issues of fog computing: A survey, in: *International conference on wireless algorithms, systems, and applications*, Springer, 2015, pp. 685–695.
- [71] I. Stojmenovic, S. Wen, The fog computing paradigm: Scenarios and security issues, in: *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*, IEEE, 2014, pp. 1–8.
- [72] S. Singh, I. Chana, Earth: Energy-aware autonomic resource scheduling in cloud computing, *Journal of Intelligent & Fuzzy Systems* 30 (3) (2016) 1581–1600.
- [73] T. Guérout, M. B. Alaya, Autonomic energy-aware tasks scheduling, in: *Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2013 IEEE 22nd International Workshop on*, IEEE, 2013, pp. 119–124.
- [74] D. Merkel, Docker: lightweight linux containers for consistent development and deployment, *Linux Journal* 2014 (239) (2014) 2.
- [75] C. Yang, Checkpoint and restoration of micro-service in docker containers, in: *Proc. 3rd Int. Conf. Mechatron. Ind. Informat*, 2015, pp. 915–918.
- [76] A. Shukla, S. Chaturvedi, Y. Simmhan, Riotbench: An iot benchmark for distributed stream processing systems, *Concurrency and Computation: Practice and Experience* 29 (21) (2017) e4257.