



UNIVERSIDADE FEDERAL DA BAHIA

DISSERTAÇÃO DE MESTRADO

**iSim: Uma Estratégia de Agrupamento de Usuários para
Descarregamento de Dados em Redes Móveis Centradas na
Informação**

Adriana Viriato Ribeiro

Programa de Pós-Graduação em Ciência da Computação

Salvador
21 de Agosto de 2017

PGCOMP-Msc-2017

ADRIANA VIRIATO RIBEIRO

**ISIM: UMA ESTRATÉGIA DE AGRUPAMENTO DE USUÁRIOS
PARA DESCARREGAMENTO DE DADOS EM REDES MÓVEIS
CENTRADAS NA INFORMAÇÃO**

Esta Dissertação de Mestrado foi apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal da Bahia, como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Leobino Nascimento Sampaio

Salvador
21 de Agosto de 2017

Ficha catalográfica.

Ribeiro, Adriana Viriato

iSim: Uma Estratégia de Agrupamento de Usuários para Descarregamento de Dados em Redes Móveis Centradas na Informação/ Adriana Viriato Ribeiro– Salvador, 21 de Agosto de 2017.

105p.: il.

Orientador: Prof. Dr. Leobino Nascimento Sampaio.
Dissertação (Mestrado)– Universidade Federal da Bahia, Instituto de Matemática, 21 de Agosto de 2017.

1. Redes Centradas no Conteúdo. 2. Agrupamento de usuários. 3. Redes Móveis. 4. Descarregamento de Dados. .
I. Sampaio, Leobino Nascimento. II. Universidade Federal da Bahia. Instituto de Matemática. III Título.

TERMO DE APROVAÇÃO

ADRIANA VIRIATO RIBEIRO

ISIM: UMA ESTRATÉGIA DE AGRUPAMENTO DE USUÁRIOS PARA DESCARREGAMENTO DE DADOS EM REDES MÓVEIS CENTRADAS NA INFORMAÇÃO

Esta Dissertação de Mestrado foi julgada adequada à obtenção do título de Mestre em Ciência da Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação da Universidade Federal da Bahia.

Salvador, 21 de Agosto de 2017

Prof. Dr. Leobino N. Sampaio
Universidade Federal da Bahia - UFBA

Profa. Dra. Fabíola Greve
Universidade Federal da Bahia - UFBA

Prof. Dr. Artur Ziviani
Laboratório Nacional de Computação Científica -
LNCC

Prof. Dr. Italo Fernando Scotá Cunha
Universidade Federal de Minas Gerais - UFMG

*Aos que me fazem querer ir em frente sem esquecer minhas
origens: família.*

AGRADECIMENTOS

Agradeço à Deus, por sempre me colocar no caminho certo e sempre me cercar de pessoas que me apóiam, me ajudam, me motivam e contribuem com meu crescimento. É Nele que confio e sempre peço forças para continuar encarando todos os desafios que surgem em minha vida.

Agradeço à minha família, em especial à minha mãe, que nunca mediu/mede esforços quando se trata da minha educação. Também a minha vó Matilde, minhas tias Bel e Céu, Fillipe, Bia, Tiúscia, Digo, Leilinha, por me darem amor, por sempre esperarem o melhor de mim e acreditar em tudo que faço.

Agradeço à meu namorado, Iuri, por todas as conversas motivacionais, por sempre torcer e acreditar no meu sucesso, por me emprestar o notebook para executar as simulações, por me emprestar seus ouvidos para escutar minhas lamúrias e por sempre me dar força e não me deixar desistir em nenhum momento.

Agradeço ao meu orientador, Leobino, primeiramente por ter me acolhido como orientanda sem me conhecer, pela humildade e pela humanidade ao tratar seus alunos, pelas tantas conversas e conselhos, pelo tempo disponibilizado (quase 25 horas por dia), por carregar sempre um sorriso no rosto, por se preocupar com o meu futuro e por ser, para mim, um exemplo de pessoa e de profissional a ser seguido.

Agradeço ao professor Artur Ziviani por ter contribuído tanto com esse trabalho e por sempre fazer sugestões extremamente pertinentes, a ajuda foi essencial para a construção do meu trabalho.

Agradeço aos meus companheiros do grupo INSERT que contribuíram muito com meu crescimento durante todas as nossas reuniões. Agradeço em especial a Eliseu, que suportou meus muitos questionamentos e me salvou muitas vezes.

Agradeço à minha amiga Jhenni, por sofrer e se divertir junto comigo todos esses anos, a Luiz, por não exitar em ajudar sempre que necessário, à Elienai, por me acompanhar desde a graduação e sempre torcer pelo meu sucesso, a George Caique, pela paciência e disponibilidade para ajudar sempre que preciso e a todos que, de alguma forma, contribuíram com meu trabalho.

Agradeço também às minhas companheiras diárias, Bel e Rafa, por me acolherem aqui em Salvador e por me proporcionarem um lar além do que eu já tinha em Jequié.

A todos vocês, muito obrigada!

(Não pensem que estão livres de mim, pois, em breve, tem doutorado!)

RESUMO

A densificação de redes, através da adição de estações secundárias, tem sido vislumbrada como uma das alternativas adotadas pelas operadoras de telefonia para aumentar a cobertura do serviço e atender às demandas de tráfego locais, com o intuito de melhorar o tempo de resposta aos usuários e alcançar descarregamento de dados (*do inglês, data offloading*). Tais propósitos tem sido associados ao uso de Redes Centradas na Informação, cujo foco no conteúdo permite potencializar o serviço através do uso de cache nos equipamentos de rede para manter os conteúdos mais populares mais próximos dos consumidores finais. Trabalhos mais recentes têm sugerido o uso de técnicas cientes de contexto e de agrupamento com o objetivo de reunir usuários com base em características similares e associá-los ao cache mais apropriado. Esta dissertação apresenta a iSim, uma estratégia de cálculo de similaridade e agrupamento entre usuários baseada na diversidade e na frequência dos conteúdos requisitados. A estratégia foi aplicada a um cenário de Redes Celulares de Pequeno Porte, com uma arquitetura baseada em Redes Centradas na Informação, com o objetivo de prover descarregamento de dados. Para alcançar esse objetivo, faz-se uso do Índice de Jaccard e da similaridade do cosseno para determinar a afinidade entre diferentes usuários a partir da diversidade entre seus interesses por conteúdo e da frequência dos interesses em comum. Após determinar a afinidade entre os usuários, eles são agrupados e associados a um cache. A intenção é aumentar a taxa de acerto no cache local e diminuir a quantidade de requisições encaminhadas à estação central, alcançando o descarregamento. Resultados experimentais demonstram a capacidade de adaptação da estratégia proposta a diferentes cenários e evidenciam melhoras na taxa de acerto e no descarregamento de dados quando a estratégia é utilizada. Além disso, foi possível observar que quando a similaridade é baseada tanto na quantidade de conteúdos em comum quanto na frequência, as taxas de acerto e de descarregamento de dados são superiores a quando apenas um fator é utilizado. As demais contribuições trazidas por esta dissertação são: i) desenvolvimento do ambiente simulado envolvendo rede sem fio centrada na informação; e ii) avaliação experimental sobre os impactos da similaridade e agrupamento dos usuários e da forma de comunicação estabelecida na rede.

Palavras-chave: Descarregamento de dados. Similaridade. Cache. Redes Centradas na Informação. Redes Móveis Agrupamento.

ABSTRACT

Network densification through small-cell addition have become an alternative adopted by mobile operators to increase service coverage and deal with local traffic. The goal is to use this solution to improve user response time and data offloading. The Information Centric Networks can help in this process by adding cache on network devices to keep the most popular content closest to final consumers. The most recent works in the area suggest the use of context-aware and clustering approaches to put similar users together and associate them to the best cache. This thesis presents iSim, a strategy to calculate similarity and group users based on frequency and diversity of user requests. We applied the solution in a Small Cell Information Centric Network to provide data offloading in a macrocell. So, we use the Jaccard Index and the Cosine Similarity in order to evaluate user similarity according to frequency and diversity of requests. Then, we group the users and associate them to a cache to increase hit ratio in local caches and decrease the requests on the macrocell. Experimental results show iSim's adaptability to several scenarios and show the improving in data offloading and hit ratio when the strategy is used. Furthermore, we have better results when the similarity is based not only in frequency of common requests but also in the relation between content requests in common and distinct. Other contributions of this thesis are: i) development of a simulation environment involving Mobile Information Centric Network; and ii) experimental evaluation on similarity and clustering users and the communication method.

Keywords: Data Offloading. Similarity. Cache. Information Centric Networks. Mobile Networks. Clustering.

SUMÁRIO

Capítulo 1—Introdução	1
1.1 Contextualização/Motivação	1
1.2 Problema	4
1.3 Objetivos	6
1.4 Aspectos Metodológicos	6
1.5 Contribuições	7
1.6 Organização	8
Capítulo 2—Redes Centradas na Informação	9
2.1 Fundamentos de ICN	9
2.2 Arquitetura ICN	12
2.2.1 Tipos de Pacotes	13
2.2.2 Modelo Publica/Assina	14
2.2.3 Sistemas de Nomeação	15
2.2.4 Roteamento	17
2.2.5 Equipamentos de Rede	20
2.2.5.1 <i>Cache</i>	20
2.2.5.2 <i>Forwarding Information Base</i>	22
2.2.5.3 Pending Interest Table	23
2.2.6 Encaminhamento e Aquisição de Conteúdos	24
2.3 Projetos de ICN	26
2.4 Considerações Finais	28
Capítulo 3—Trabalhos Relacionados	29
3.1 Descarregamento de Dados utilizando Redes Wi-Fi	29
3.2 Descarregamento de Dados Ciente de Contexto	31
3.3 Descarregamento de Dados Ciente de Contexto em SCN	32
3.4 Considerações Finais	34
Capítulo 4—iSim: Estratégia de Agrupamento por Interesses Similares	35
4.1 Descrição Geral	35
4.2 Cenário de Aplicação	36
4.2.1 Cenário Técnico	37
4.3 Estrutura da ICN	39

4.3.1	Equipamentos de Rede	39
4.3.2	Tipos de Pacote	42
4.3.3	Encaminhamento de Pacotes	42
4.4	Função de Similaridade	47
4.5	Algoritmo de Similaridade e Agrupamento	48
4.6	Formas de Comunicação	51
4.7	Considerações Finais	52
Capítulo 5—Avaliação Experimental		55
5.1	Simulação	55
5.2	Cenário de Aplicação	55
5.3	Arquitetura ICN	56
5.4	Cenários, Métricas, Fatores e Parâmetros	57
5.5	Ambiente de Simulação	58
5.6	Análise dos Resultados	59
5.6.1	Grupo 1: Relação entre Tamanho do <i>Cache</i> , β e Taxa de Acerto	59
5.6.2	Grupo 2: Relação entre o parâmetro α da Lei de Zipf e a Taxa de Acerto	61
5.6.3	Grupo 3: Relação entre Similaridade e Comunicação na Taxa de Acerto e no Descarregamento de Dados	63
5.6.4	Avaliação da variação na quantidade de dispositivos móveis e conteúdo	64
5.6.5	Avaliação do recálculo dos grupos	65
5.7	Considerações Finais	67
Capítulo 6—Conclusões e Trabalhos Futuros		69
6.1	Contribuições	70
6.2	Trabalhos Futuros	71
Apêndice A—Ambiente de Simulação		79

LISTA DE FIGURAS

1.1	Principais evoluções de arquitetura e infraestrutura da Internet ao longo dos anos em decorrência da mudança no objetivo da rede, da adição de novas tecnologias e do perfil dos usuários, associada à uma prospecção futura levando em consideração os aspectos abordados nesta dissertação para solucionar o problema de descarregamento de dados.	4
2.1	Comparativo entre requisição orientada à localização e requisição orientada ao conteúdo.	10
2.2	Relação entre o paradigma ICN e a arquitetura atual decorrente da mudança de pacotes IP para pacotes nomeados. Traduzida e adaptada de (JACOBSON et al., 2009)	11
2.3	Modelo de pacote de interesse. Traduzido de (JACOBSON et al., 2009).	13
2.4	Pacote de Interesse CCN composto pelos campos tipo de pacote, tamanho do pacote, nome, metadados opcionais e <i>payload</i> opcional. Traduzida e adaptada de (IETF, 2014)	14
2.5	Pacote de Dados CCN composto pelos campos tipo, tamanho, nome, metadados opcionais e <i>payload</i> opcional. Traduzida e adaptada de (IETF, 2014)	14
2.6	Encaminhamento de pacotes de interesse e de dados. Adaptado de (GUIMARAES, 2013)	15
2.7	Exemplo de nomeação hierárquica composta por prefixo, sufixo e parte do conteúdo.	16
2.8	Representação do roteamento baseado em “migalhas de pão”.	18
2.9	Roteamento baseado no modelo proposto no projeto PSIRP (AHLGREN et al., 2012)	19
2.10	Modelo de roteamento em árvore. Fonte: Adaptado de (JACOBSON et al., 2009)	19
2.11	Uso do <i>cache</i> para deixar o conteúdo mais próximo do usuário. Adaptado de (GUIMARAES, 2013)	21
2.12	Comparação entre a Base de Informações de Encaminhamento no modelo IP e no modelo ICN.	22
2.13	Representação da Tabela de Interesses Pendentes.	23
2.14	Funcionamento das estruturas básicas na arquitetura ICN. Adaptado de (JACOBSON et al., 2009)	24
2.15	Fluxograma representando as ações realizadas por um roteador ao receber um pacote de interesse.	25

4.1	Representação de um cenário real em que a iSim pode ser utilizada. . . .	37
4.2	Representação dos tipos de densificação: distribuída ou como complemento à estrutura principal.	38
4.3	Estrutura genérica de uma SCN na qual a iSim foi baseada.	39
4.4	Estrutura do <i>cache</i> em uma rede na qual a iSim é adotada.	40
4.5	Estrutura da PIT em uma rede na qual a iSim é adotada.	41
4.6	Estrutura da RT em uma rede na qual a iSim é adotada.	41
4.7	Estrutura dos pacotes de interesse e de dados definidos para dar suporte à iSim.	42
4.8	Estrutura dos pacotes de controle definidos para dar suporte à iSim. . . .	43
4.9	Processamento de pacotes de interesse na MBS.	44
4.10	Processamento de pacotes de interesse e de dados na SCBS	45
4.11	Diagrama de Sequência que ilustra o encaminhamento de pacotes em diferentes momentos.	46
4.12	Demonstração da formação da matriz $N \times C$	49
4.13	Demonstração da formação da matriz $N \times N$	50
4.14	Demonstração da ordenação da matriz $N \times N$ e da utilização de um tipo abstrato de dados para manutenção dos identificadores.	51
4.15	Encaminhamento de mensagens utilizando comunicação direta.	52
4.16	Encaminhamento de mensagens utilizando comunicação indireta.	53
5.1	Cenário genérico utilizado para validação da iSim.	56
5.2	Relação entre o Tamanho do <i>Cache</i> e a Taxa de Acerto.	60
5.3	Efeito do tamanho do cache na Taxa de Descarregamento de Dados. . . .	61
5.4	Efeito do parâmetro Zipf na Taxa de Acerto.	62
5.5	Efeito do parâmetro Zipf na Taxa de Acerto.	62
5.6	Influência da Similaridade e da Hierarquia no Descarregamento de Dados.	63
5.7	Influência da Similaridade e da Hierarquia na Taxa de Acerto.	64
5.8	Efeito da variação da quantidade de dispositivos na Taxa de Descarregamento de Dados.	65
5.9	Efeito da quantidade de conteúdos na Taxa de Descarregamento de Dados.	66
5.10	Efeito do reagrupamento de Usuários.	67
A.1	Composição do módulo <i>wirelessHost</i> que representa os dispositivos móveis.	80
A.2	Diagrama de classes das aplicações criadas.	81

LISTA DE TABELAS

2.1	Comparação entre os conceitos da arquitetura atual e da arquitetura centrada na informação. Adaptado de (OEHLMANN, 2013)	12
2.2	Principais características dos projetos de ICN. Traduzido e adaptado de (AHLGREN et al., 2012)	27
3.1	Comparação entre a iSim e os trabalhos relacionados ((ELBAMBY et al., 2014) e (PANTISANO et al., 2015)	33
4.1	Comparação entre macro-células e diferentes tipos de células pequenas . .	38
4.2	Relação entre os dispositivos e suas funcionalidades e pacotes trafegados.	43
5.1	Fatores e Cenários de Avaliação.	58
5.2	Parâmetros da Simulação.	59

LISTA DE SIGLAS

AA	Approximation Algorithm
CCN	Content-Centric Networks
CDN	Content Delivery Networks
CS	Content Store
DM	Dispositivos Móveis
DNS	Domain Name System
DONA	Data Oriented Network Architecture
DTN	Delay-Tolerant Networks
FIB	Forwarding Information Base
FIFO	First In First Out
GA	Greedy Algorithm
HA	Homogeneous Algorithm
ICN	Information-Centric Networks
IRTF	Internet Research Task Force
iSim	interests' Similarity
kNN	k Nearest Neighbor
LCE	Leave Copy Everywhere
LFRU	Least Frequently Recently Used
LFU	Least Frequently Used
LRF	Least Recently Forwarded
LRU	Least Recently Used
MACACO	Mobile context-Adaptative Caching for COntent-centric networking

MBS	Macro Base Station
MOFO	Most Forwarded
NAT	Network Address Translation
NDN	Named Data Networks
NDO	Named Data Object
NetInf	Network of Information
PIT	Pending Interest Table
PSIRP	Publish-Subscribe Internet Routing Paradigm
RH	Resolution Handler
RINA	Recursive InterNetwork Architecture
RT	Requisition Table
RWI	Real-World Identity
SAIL	Scalable Adaptive Internet Solution
SCBS	Small Cell Base Stations
SCN	Small Cell Networks
SDN	Software Defined Networks
TLV	Type, Length, Value
TTL	Time-To-Live
UCN	User-Centric Networks
VNI	Visual Networking Index

INTRODUÇÃO

Neste capítulo são expostas a contextualização e motivação para o uso de Redes Centradas na Informação associadas às infraestruturas de Redes Celulares de Pequeno Porte, bem como os principais problemas e desafios enfrentados para obter descarregamento de dados nesse cenário. Além disso, também são explanados o problema de pesquisa, os objetivos, a metodologia utilizada para a construção e desenvolvimento da proposta e a organização da dissertação.

1.1 CONTEXTUALIZAÇÃO/MOTIVAÇÃO

A Internet atual teve sua estrutura definida por volta dos anos 70 e 80 e se baseava na ideia de que a rede deveria ser composta por um núcleo minimalista, cujo objetivo era prover a entrega de pacotes, e que a inteligência deveria estar localizada nos sistemas finais, nos quais qualquer pessoa que soubesse programar esses computadores seria capaz de criar novos aplicativos e funcionalidades (CLARK, 1988) (REXFORD; DOVROLIS, 2010) (STUCKMANN; ZIMMERMANN, 2009). Desta forma, a Internet foi construída com o intuito de atender requisitos que supriam as necessidades da época, como conectividade, robustez e heterogeneidade (MOREIRA et al., 2009). Esses requisitos focavam em estabelecer o compartilhamento de recursos através da conexão entre os computadores e, por isso, as premissas básicas da rede são atendidas de maneira satisfatória até os dias atuais.

Os principais usuários, na época em que a Internet foi criada, eram cientistas, indústrias e grandes empresas governamentais. Nessa época, era necessário ter conhecimento técnico para manusear os computadores e estabelecer as conexões de rede. Com a globalização da computação, a Internet passou a ser utilizada por diferentes pessoas, de diferentes faixas etárias e que não tinham, necessariamente, conhecimento técnico. Além disso, a popularização de novas tecnologias, como dispositivos móveis, e a mudança no perfil dos usuários e de seus interesses fizeram com que novas necessidades fossem surgindo e a Internet, do jeito que foi concebida, não atendia mais aos requisitos necessários (FELDMANN, 2007).

A simplicidade do núcleo da rede e dos protocolos TCP/IP foi o fator de sucesso da Internet e está sendo seu fator de mudança. Ao mesmo tempo em que a simplicidade tornou a rede relativamente fácil de ser operada, permitindo seu crescimento e a evolução de aplicações, a falta de inteligência no núcleo da rede tornou-a ossificada, visto que impõe restrições para o desenvolvimento de novos serviços. Os serviços considerados mais urgentes ou importantes foram resolvidos através da execução de alguns “remendos” na rede, como a criação do NAT (do inglês, *Network Address Translation*) para solucionar o problema de esgotamento de endereços IP, criação de sub-redes para aumentar a escalabilidade, criação de IPs móveis devido a característica de mobilidade dos dispositivos, entre outros. No entanto, essas adequações não conseguem prover soluções gerais para os desafios que a Internet atual enfrenta (MOREIRA et al., 2009). Assim, começou-se a pensar em criar uma nova arquitetura para a Internet que fosse capaz de sanar os desafios de escalabilidade, gerenciamento, segurança, mobilidade, *multihoming*, múltiplos caminhos, caminhos programáveis e outros desafios encontrados atualmente e que provavelmente serão encontrados no futuro (FELDMANN, 2007) (MOREIRA et al., 2009) (CAMPISTA et al., 2014).

Para atender a esses desafios, (CAMPISTA et al., 2014) organizam direções de pesquisa em algumas classes, como separação de localizador e identificador, mobilidade da rede e roteamento baseado em conteúdo. Esses novos requisitos exigem que seja estabelecido um novo relacionamento entre nomeação, endereçamento e roteamento, de forma que problemas de segurança, escalabilidade, mobilidade e gerenciamento sejam sanados. Todo o esforço para sanar desafios da Internet atual de forma evolutiva ou substitutiva foram organizados em uma área denominada de Internet do Futuro, cujo principal objetivo é estabelecer os requisitos atuais e futuros da rede e determinar novas arquiteturas, modelos e serviços para resolvê-los. Entre algumas alternativas estão as Redes Definidas por Software (do inglês, *Software Defined Networks – SDN*) (FOUNDATION, 2012), as Redes Centradas no Usuário (do inglês, *User Centric Networks – UCN*) (SOFIA, 2011), a Arquitetura Inter Rede Recursiva (do inglês, *Recursive InterNetwork Architecture*)¹ e as Redes Centradas na Informação (do inglês, *Information Centric Networks – ICN*) (JACOBSON et al., 2009) (KUTSCHER et al., 2016).

Enquanto a Internet atual foi projetada com a intenção de fazer compartilhamento de recursos, e para alcançar esse objetivo foi estabelecida a comunicação entre máquinas (modelo de comunicação *host centric*), as ICN substituem o foco do compartilhamento de recursos pelo compartilhamento do conteúdo. Assim, a rede deixaria de ser *host centric* e passaria a ser *content centric*, o importante agora é saber *qual* conteúdo a rede está oferecendo, e não *onde* esse conteúdo está armazenado. Partindo desse pressuposto, as ICN tem como principais características o roteamento baseado em conteúdo e a separação de localizador e de identificador. A importância dada ao conteúdo nessa arquitetura é consequência da mudança no perfil dos usuários, que passou de consumidor para produtor e consumidor de conteúdos, e da utilização de aplicações como *Whatsapp*, *Instagram* e *Netflix* que geraram um crescimento exponencial no tráfego.

De acordo com Relatório Anual Cisco® Visual Networking Index™ (VNI) (CISCO,

¹<http://irati.eu/>

2016), o tráfego IP anual irá triplicar entre os anos de 2015 e 2020. Enquanto a Internet fixa terá um crescimento três vezes maior, estima-se que os dados móveis passem de uma marca de 3.685 Petabytes (PB) por mês em 2015 para 30.564 PB por mês em 2020, evidenciando um volume de tráfego aproximadamente oito vezes maior. Sendo que os principais tipos de serviços solicitados pelos consumidores são *streaming* de vídeo, Web, e-mail, dados e compartilhamento de arquivos. Isso evidencia que houve não apenas uma mudança do ponto de vista de perfil dos usuários e aplicações, mas também da tecnologia e infraestrutura utilizadas, decorrente de uma migração dos usuários das redes fixas para as redes móveis.

No que diz respeito às redes móveis, o aumento exponencial do tráfego acabou causando desafios para as operadoras de telefonia, visto que a infraestrutura existente não foi criada para suportar a nova demanda. Os novos desafios envolvem, principalmente, alternativas para aumentar capacidade de cobertura do serviço e para diminuir a sobrecarga no *backhaul*² da rede. O conjunto de técnicas e tecnologias com o objetivo de diminuir o congestionamento no núcleo da rede e otimizar o uso dos recursos disponíveis ficou conhecido como estratégias de descarregamento de dados (do inglês, *data offloading*). Essas estratégias envolvem a utilização de redes Wi-Fi (DIMATTEO et al., 2011), adição de pequenas células como *femtocells* (ROH et al., 2011), criação de redes oportunistas (HAN et al., 2010), densificação de redes e criação de *cache*.

A densificação de redes (BHUSHAN et al., 2014) trata-se de uma maior oferta do serviço de telefonia a partir do aumento do número de antenas de transmissão na mesma área, que geralmente são instaladas em pequenas estações-base (*Small-Cell Base Station* – SCBS). O conjunto de antenas de uma SCBS é constituído por meio da implantação de equipamentos e tecnologias de transmissão de curto alcance, oferecendo, desta forma, redes celulares de pequeno porte (tradução adaptada do termo em inglês, *do inglês, Small Cell Networks* – SCNs³). As SCBS oferecem serviços complementares aos disponibilizados pelas macro estações-base (do inglês, *Macro Base Station* – MBS), com o propósito de aumentar a cobertura do serviço e atender às demandas de tráfego locais, melhorando, assim, os tempos de resposta ao usuário.

Uma das estratégias utilizadas para diminuir o tráfego no *backbone*, melhorar o tempo de resposta aos usuários e otimizar o uso dos recursos é a inserção de *cache* nos equipamentos da rede e em servidores mais próximos dos usuários (ZHANG; LI; LIN, 2013). O Netflix, por exemplo, utiliza uma Rede de Distribuição de Conteúdo (do inglês, *Content Delivery Networks* – CDN) com mais de mil servidores locais para manter cópias dos conteúdos mais próximas dos usuários, diminuindo o tráfego na Internet, aumentando a velocidade de transmissão e aumentando a disponibilidade do serviço em diferentes partes do mundo (NETFLIX, 2016). Essa mesma alternativa tem sido adotada pelas redes de telefonia móvel com o intuito de potencializar o efeito da densificação de rede, provendo descarregamento de dados nas estações principais, aumentando a cobertura do serviço e melhorando o tempo de resposta ao usuário.

²Parcela da rede responsável por fazer a ligação entre o *backbone* (núcleo da rede) e as sub-redes

³Na literatura, a expressão em inglês *Small cell* refere-se às chamadas “células pequenas” que engloba femto-células, pico-células e microcélulas. Essas redes aumentam a cobertura do serviço prestado pelas operadoras e são utilizadas pelos usuários na redução do congestionamento das macro-células.

Devido aos desafios enfrentados pelo aumento do tráfego, adição de novas tecnologias, mudança do perfil dos usuários e dos novos objetivos associados à rede, é possível observar na Figura 1.1 que há uma relação direta entre a necessidade da evolução tanto da arquitetura quanto da infraestrutura ao longo do tempo. Enquanto o estudo acadêmico sobre a arquitetura e a Internet do Futuro lidam com esses desafios do ponto de vista mais científico, as operadoras de telefonia móvel se depararam com o desafio de adequar a infraestrutura existente para lidar com a grande sobrecarga de dados que foi migrada da rede fixa para as redes móveis. Em decorrência da inadequação da arquitetura atual, dos desafios em lidar com mobilidade e da sobrecarga gerada pelo aumento do tráfego nas redes móveis, propostas descritas na literatura sugerem o uso de ICN em cenários de SCN. Assim, trabalhos tem sido desenvolvidos levando em consideração o uso de *cache* nas SCBS para armazenar conteúdos e deixá-los mais próximos dos consumidores finais, reduzindo a quantidade de requisições na MBS e fornecendo descarregamento de dados (ELBAMBY et al., 2014; PANTISANO et al., 2015; CHEN; KOUNTOURIS, 2015; HAJRI; ASSAAD, 2016).

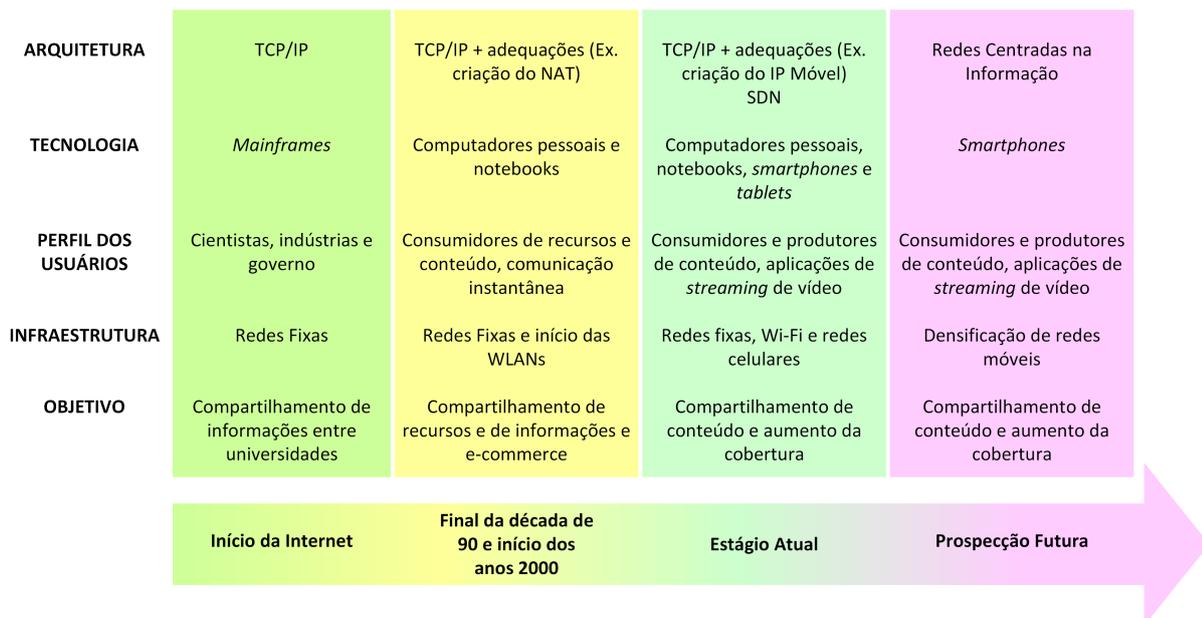


Figura 1.1 Principais evoluções de arquitetura e infraestrutura da Internet ao longo dos anos em decorrência da mudança no objetivo da rede, da adição de novas tecnologias e do perfil dos usuários, associada à uma prospecção futura levando em consideração os aspectos abordados nesta dissertação para solucionar o problema de descarregamento de dados.

1.2 PROBLEMA

Embora a inclusão de *cache* nos equipamentos de rede seja uma alternativa para deixar o conteúdo mais próximo do usuário, é preciso definir estratégias que potencializem o uso do *cache*, visto que a rotatividade de conteúdos é grande e a probabilidade de o usuário requisitar um conteúdo que esteja armazenado em *cache* pode ser pequena. Existem al-

gumas políticas clássicas de gerenciamento de *cache*, como *Least Frequently Used* (LFU) e *Least Recently Used* (LRU), que alcançam melhores resultados quando comparadas aos cenários em que há descarte aleatório ou do tipo *First In First Out* (FIFO), por exemplo. No entanto, ambas estratégias de descarte não levam em consideração as preferências individuais dos usuários. As preferências dos usuários de acordo com faixa etária, localização geográfica, entre outros atributos, tem sido um dos aspectos observados por empresas como Netflix e Google para fazer direcionamento de anúncios, recomendações e para tomar decisões sobre dados armazenados nos servidores de suas Redes de Distribuição de Conteúdo, isso porque as preferências dos usuários acabam interferindo nas requisições e podem ser utilizadas como insumo para melhorar o tempo de resposta e fazer sistemas de recomendação mais eficientes.

Trabalhos mais recentes têm sugerido o uso de técnicas cientes de contexto (CHANG et al., 2016) e técnicas de agrupamento com o objetivo de reunir usuários com base em características similares e associá-los às SCBS mais apropriadas (IOANNOU; WEBER, 2016). Existem diferentes alternativas para determinar a similaridade dos usuários, tanto do ponto de vista de qual característica utilizar: mobilidade (CHEN et al., 2016), conteúdo (ELBAMBY et al., 2014; PANTISANO et al., 2015), topologia (KIM; SHIN; KO, 2013), entre outros, quanto do ponto de vista de qual função utilizar para determinar a similaridade. A alternativa utilizada deve adaptar-se ao objetivo e ao cenário que está sendo estudado, sendo possível fazer utilização de agrupamentos com a intenção de aumentar QoS, diminuir tempo de resposta ao usuário, gerar descarregamento de dados, entre outros.

Em cenários que visam utilizar agrupamentos e *caches* para prover descarregamento de dados, espera-se que, através de agrupamentos, haja melhoras nas taxas de acerto das redes de *cache* durante a busca por conteúdos. Apesar do agrupamento apresentar ganhos nas taxas de acerto, os trabalhos relatados na literatura (ELBAMBY et al., 2014; PANTISANO et al., 2015) que recorrem a esta estratégia levam em consideração apenas a frequência dos conteúdos requisitados na definição dos grupos de usuários que serão associados às SCBS. O problema, contudo, é que a frequência dos conteúdos não relata de forma precisa o nível de afinidade entre os usuários. A afinidade deve corresponder à aproximação entre os usuários do ponto de vista das suas preferências em relação aos conteúdos disponíveis na rede, como a avaliação da frequência leva em consideração apenas os conteúdos solicitados em comum, muitas vezes ela pode se tornar ineficiente ao determinar usuários afins. Assim, dois consumidores podem ter interesse apenas por um único conteúdo em comum e, ainda assim, serem considerados similares em decorrência da frequência de solicitações e da função de similaridade utilizada. Por exemplo, tendo dois consumidores A e B cujos conjuntos de conteúdos solicitados são $C_a = \{x, y, r, t, s, z\}$ e $C_b = \{a, b, c, l, w, z\}$ e as frequências de solicitação são $F_a = \{1, 2, 3, 1, 4, 1\}$ e $F_b = \{7, 3, 1, 2, 5, 1\}$, levando em consideração apenas a frequência dos conteúdos solicitados em comum (conteúdo z) e aplicando a similaridade do cosseno para definir a afinidade entre os dois consumidores, eles teriam similaridade máxima.

Assim, esta dissertação visa resolver o seguinte problema: **como melhorar o descarregamento de dados em redes de *cache* considerando a afinidade entre os usuários para formação dos agrupamentos.** Para isso, foi proposta a iSim (inte-

rests' Similarity), uma estratégia ciente de contexto que explora a formação de grupos de usuários similares em uma SCN centrada na informação com o intuito de melhorar o descarregamento de dados na MBS, levando em consideração a diversidade dos conteúdos requisitados pelos usuários e a frequência de solicitações. A iSim tem as seguintes características:

- Utiliza as requisições/interesses dos usuários como insumo para determinar suas preferências por conteúdo.
- Explora a afinidade dos usuários de acordo a diversidade e frequência de seus conteúdos.
- Dá suporte à flexibilidade e adaptatividade através do uso das funções do Índice de Jaccard e da Similaridade do Cosseno para calcular a afinidade entre usuários.
- Utiliza a técnica de agrupamento dos k vizinhos mais próximos (do inglês, *k Nearest Neighbor* - kNN) para associar usuários afins.
- Aumenta a taxa de acerto nas SCBS para melhorar o descarregamento de dados na MBS.

1.3 OBJETIVOS

O principal objetivo desta dissertação foi realizar descarregamento de dados em uma rede móvel centrada na informação através da utilização de uma estratégia de similaridade e agrupamento de usuários afins.

Para atingir esse objetivo geral, foram estabelecidos alguns objetivos específicos:

1. Compreender os desafios de prover descarregamento de dados em redes móveis.
2. Determinar como a arquitetura de Redes Centradas na Informação poderia ser aplicada ao cenário móvel para contribuir na resolução do problema.
3. Implementar o cenário da redes celulares de pequeno porte centrado na informação.
4. Propor e implementar uma estratégia de similaridade e agrupamentos de usuários de acordo com seus interesses por conteúdos.
5. Avaliar a estratégia proposta, através de simulação, considerando métricas, fatores e parâmetros apropriados.

1.4 ASPECTOS METODOLÓGICOS

O interesse em trabalhar com Internet do Futuro guiou o início desse trabalho e a definição do problema de pesquisa. Uma vez definida que a abordagem de Internet do Futuro a ser trabalhada seria a ICN, foi feito um levantamento bibliográfico para identificar os principais problemas de pesquisa da área. Desta forma, foi possível observar que os problemas estavam agrupados em diferentes categorias: mobilidade, segurança,

escalabilidade, nomeação, roteamento, entre outros. Dentro da área de mobilidade, um problema que não era intrínseco das ICN, mas causado pelo grande volume do tráfego, foi o descarregamento de dados. Desta forma, pensou-se em utilizar as características das ICN em uma estratégia que pudesse solucionar esse problema em redes móveis.

O cenário móvel escolhido para validação da proposta foi o de redes celulares de pequeno porte. Esse cenário permite a utilização de um novo modelo de arquitetura associado a um novo modelo de infraestrutura e tem sido utilizado em trabalhos relacionados. Assim, na primeira etapa da pesquisa, foram feitos levantamentos bibliográficos envolvendo ICN, SCN e descarregamento de dados. Definido o problema, o cenário e os mecanismos que seriam utilizados na solução, iniciou-se o processo de decisão do método de validação.

De acordo com (JAIN, 1991), existem três técnicas básicas de validação da pesquisa científica: modelagem analítica, simulação e medição. Para escolher a técnica adequada à pesquisa, é necessário levar em consideração alguns critérios como tempo requerido, acurácia e custo. A partir desses critérios, foi analisada a possibilidade de utilização de simulações de redes de eventos discretos, que são caracterizadas por garantir que o estado do sistema mudará apenas quando ocorrer um evento em um instante de tempo pré determinado. Esse tipo de simulação contém elementos que foram essenciais na confecção desse trabalho, como variáveis de estado, rotinas de evento e contadores estatísticos, e por isso foi considerada a técnica que mais se adequava.

Durante o período de qualificação, o simulador OMNeT++⁴ foi recomendado para a construção do trabalho. Após recomendação, foram avaliados aspectos como documentação, capacidade de expansão, citações em artigos científicos, entre outros. Ao confirmar a adequação do simulador, iniciou-se o processo de implementação do cenário.

Com o cenário definido, também foi inicializado o desenvolvimento da solução proposta, que consiste em um algoritmo de agrupamento utilizando uma função de similaridade que leva em consideração os interesses dos usuários por conteúdo.

1.5 CONTRIBUIÇÕES

As principais contribuições desta dissertação foram:

- Ambiente de simulação para redes celulares de pequeno porte centrada na informação.
- Avaliação de uma estrutura ICN dentro de um cenário de SCN.
- Nova medida de similaridade que é resultado da soma ponderada entre o Índice de Jaccard e a lei de similaridade do cosseno.
- Avaliação do algoritmo de similaridade e agrupamento de usuários.

⁴<http://www.omnetpp.org/>

1.6 ORGANIZAÇÃO

Esta dissertação está organizada da seguinte maneira: No Capítulo 2 é apresentada a fundamentação teórica sobre Redes Centradas na Informação. São explorados conceitos e características das ICN, como conteúdo nomeado, utilização de *cache*, formas de roteamento, entre outros. No Capítulo 3 é feito um sumário dos trabalhos relacionados, apresentando as principais abordagens utilizadas para lidar com descarregamento de dados. No Capítulo 4 são apresentadas as características da iSim, como determinação da função de similaridade e do algoritmo de agrupamento, formas de comunicação entre os equipamentos de rede, entre outras definições. No Capítulo 5 são exibidas as características da implementação e as configurações dos experimentos, bem como a análise sobre os resultados obtidos. Por fim, no Capítulo 6, apresenta-se as considerações finais desta dissertação.

REDES CENTRADAS NA INFORMAÇÃO

A abordagem de ICN (JACOBSON et al., 2009)(OEHLMANN, 2013)(BRITO; VELLOSO; MORAES, 2012) foi criada com o intuito de superar os desafios encontrados na arquitetura atual através da desvinculação entre identificação e localização de conteúdos. Assim, os dados se tornam independentes de localização, aplicação, armazenamento e meios de transporte, tornando possível superar os problemas de persistência, escalabilidade e segurança presentes na rede atual por meio de uma nova arquitetura que engloba conteúdo nomeado, roteamento baseado em conteúdo, segurança aplicada diretamente ao conteúdo e armazenamento de dados nos equipamentos de rede (KUTSCHER et al., 2016).

Neste capítulo são abordados os principais conceitos relacionados às ICN, como tipos de pacotes trafegados na rede, definição dos tipos de nomeação para conteúdo e seus impactos em aspectos como escalabilidade e segurança, tipos de roteamento, arquitetura dos equipamentos e suas principais funcionalidades, encaminhamento de pacotes e um breve sumário das principais propostas e projetos baseados em ICN.

2.1 FUNDAMENTOS DE ICN

A arquitetura atual da Internet é baseada no modelo TCP/IP. Esse modelo relaciona diretamente os serviços e conteúdos disponíveis na rede à sua localização. Desta forma, se um usuário deseja solicitar um determinado conteúdo, é necessário que ele saiba o endereço do *host* que o contém. O princípio das Redes Centradas na Informação é substituir essa necessidade de conhecer a localização e permitir que a solicitação seja feita pelo próprio nome do conteúdo. Assim, o usuário não precisa saber onde o conteúdo está (por exemplo, o endereço IP do servidor de conteúdos), ele deve poder requisitar um conteúdo diretamente à rede e ela deve ser capaz de provê-lo de maneira eficiente (OEHLMANN, 2013) (JACOBSON et al., 2009).

Essa mudança de paradigma pode ser observada na Figura 2.1. A Figura 2.1(a) mostra como é realizada uma solicitação para um conteúdo na Internet atual: o usuário precisa indicar não apenas o conteúdo que deseja mas também o local onde esse conteúdo

está armazenado. Ainda que exista um servidor mais próximo (nesse caso, o servidor 10.10.1.20) que tenha o conteúdo, a requisição é enviada para o servidor cuja localização é conhecida. Caso o usuário não saiba nenhuma localização do conteúdo na rede, ele não consegue realizar a solicitação. Já no paradigma centrado na informação, conforme Figura 2.1(b), o usuário solicita o próprio conteúdo à rede e qualquer equipamento de rede ou publicador que tenha o conteúdo armazenado pode servi-lo.

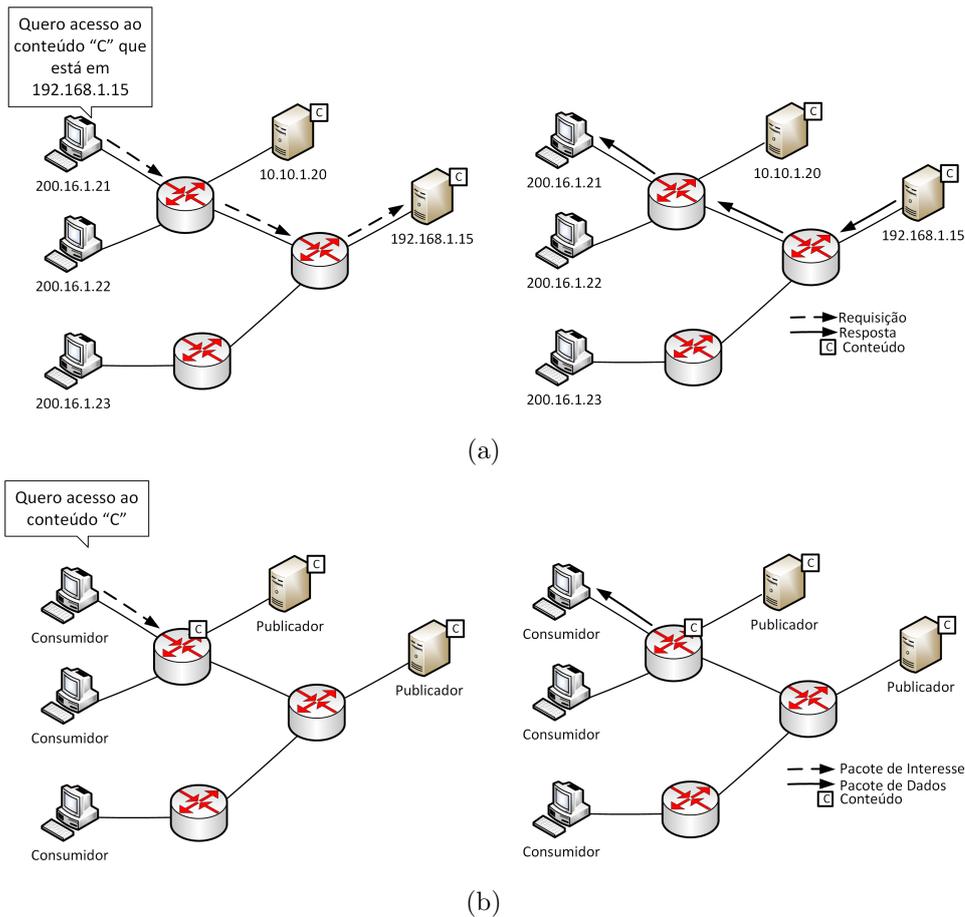


Figura 2.1 Comparativo entre requisição orientada à localização e requisição orientada ao conteúdo.

Assim, enquanto o modelo TCP/IP trafega pacotes endereçados a uma localização, a ICN visa trafegar pacotes nomeados de acordo com o conteúdo solicitado pelo usuário. Apesar dessa mudança de foco, essa arquitetura propõe-se a manter os mesmos princípios de simplicidade, robustez e escalabilidade da arquitetura atual. Com esses objetivos, (JACOBSON et al., 2009) faz uma relação entre os serviços e características de um modelo IP para um modelo ICN, de acordo com a Figura 2.2. Do lado esquerdo é possível observar as diferentes camadas do modelo IP e alguns protocolos característicos de cada uma delas, como protocolos de aplicação (SMTP, HTTP, RTP), de transporte (TCP, UDP), e assim por diante. Observa-se, portanto, que enquanto há a possibilidade de usar

diferentes protocolos em outras camadas, o protocolo IP está no centro da comunicação entre os *hosts* e todos os nós que se comunicam na rede precisam, necessariamente, utilizar esse mesmo protocolo. No caso das ICN, o protocolo IP é substituído pelos *chunks* de conteúdo, que são pedaços de conteúdo nomeado, que servem para fazer a comunicação entre um consumidor de conteúdo e um publicador, os outros protocolos, serviços e estratégias para que isso se concretize podem variar. Percebe-se, portanto, que ambas estruturas são responsáveis por realizar as principais atribuições propostas em cada arquitetura: no caso do IP, ele é o responsável por fazer a comunicação entre *hosts*, e no caso dos *chunks*, eles são responsáveis por permitir o compartilhamento de conteúdo entre os nós.

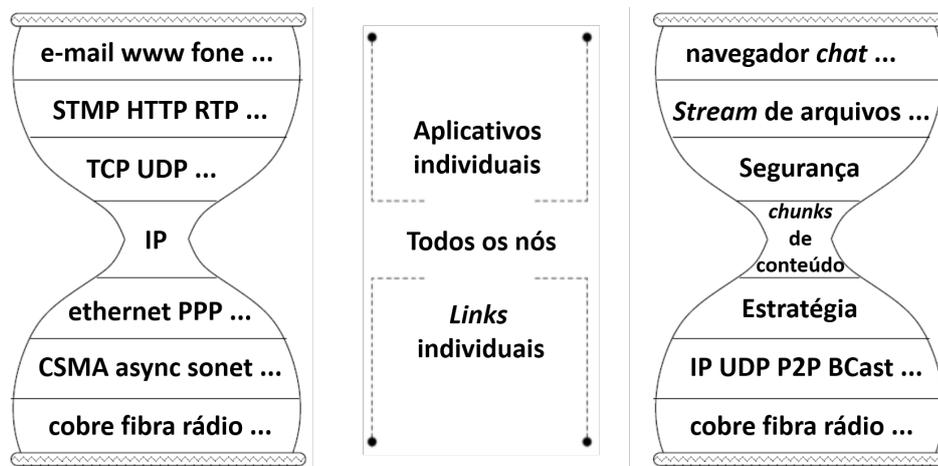


Figura 2.2 Relação entre o paradigma ICN e a arquitetura atual decorrente da mudança de pacotes IP para pacotes nomeados. Traduzida e adaptada de (JACOBSON et al., 2009)

Além disso, na representação da arquitetura ICN, as camadas estabelecidas pelo protocolo IP são substituídas por serviços. Na camada ICN são trafegados pacotes nomeados cujas aplicações individuais determinarão como será feita a segurança, o fluxo dos arquivos, quais serviços e protocolos utilizar. Enquanto os *links* individuais englobam diferentes estratégias que tentarão potencializar o uso de protocolos utilizados pela arquitetura atual, como TCP e o próprio IP, para realizar entrega de conteúdo nomeado.

Em vista desse paralelo entre IP e ICN, constata-se que para estabelecer uma arquitetura centrada na informação, é preciso fazer modificações nos objetivos, papéis e funcionalidades da rede. Na tabela 2.1 é possível observar um comparativo entre a arquitetura atual e a arquitetura ICN. A arquitetura atual é baseada em um modelo cliente/servidor, na qual o cliente é o solicitante de um serviço e um servidor é aquele que provê o serviço. Assim, o foco é permitir a comunicação entre esses dois *hosts* de acordo com seus endereços/localizações em um canal que seja seguro, visto que dados serão trafegados através dele.

Na arquitetura ICN esses conceitos são modificados e os papéis de cliente e servidor são substituídos por produtor e consumidor, no qual um produtor é aquele que publica um conteúdo e um consumidor é aquele que solicita conteúdo. Assim como no mo-

Arquitetura atual	Arquitetura centrada na informação
Servidor	Produtor
Cliente	Consumidor
Controle baseado no servidor	Controle baseado no consumidor
Comunicação Cliente/Servidor	Modelo Publica/Assina
Comunicação entre máquinas	Recuperação da informação
Topologia / Domínio	Escopo da informação
Unicast	Uni-multi-anycast unificados
Destino explícito	Destino implícito
Orientado a <i>hosts</i>	Orientado a dados
Nome do <i>host</i>	Nome do conteúdo
Canal seguro, autenticação de hosts	Segurança no próprio conteúdo

Tabela 2.1: Comparação entre os conceitos da arquitetura atual e da arquitetura centrada na informação. Adaptado de (OEHLMANN, 2013)

delo TCP/IP esses papéis não são fixos, então consumidor/produtor podem inverter seus papéis em determinados momentos. Além disso, a ICN se baseia em um modelo publica/assina, na qual um produtor só encaminha conteúdo como resposta ao interesse de um consumidor. Nessa arquitetura, não há necessidade de determinar a localização da informação, visto que existe a possibilidade da informação ser recuperada por diferentes nós da rede.

Em busca de utilizar o paradigma de ICN para superar os desafios atuais, alguns projetos foram criados com o intuito de desenvolver abordagens centradas na informação. Dentre os principais projetos estão o *Content Centric Network* (CCN) ou *Named Data Network* (NDN), o *Data Oriented Network Architecture* (DONA), o *Publish-Subscribe Internet Routing Paradigm* (PSIRP) e o *Network of Information* (NetInf). Esses projetos serão abordadas na seção 2.3 e buscam solucionar desafios na aplicação de ICN e diferem entre si sob aspectos como nomeação e roteamento.

Para realizar o roteamento de conteúdo nomeado, é necessário estabelecer toda uma nova arquitetura que se adapte aos conceitos e serviços propostos na abordagem ICN. Desta forma, a nova arquitetura conta com diferentes tipos de pacotes, estratégias para nomeação e roteamento de conteúdos que implicam diretamente em aspectos como segurança e escalabilidade, adição de novas estruturas aos equipamentos de rede e técnicas para fazer o encaminhamento baseado em conteúdo nomeado. Todos esses aspectos serão apresentados nas próximas seções.

2.2 ARQUITETURA ICN

Como já foi citado anteriormente, a comunicação na arquitetura ICN baseia-se no envio de conteúdo nomeado através da rede. Esses conteúdos são trafegados através de pacotes.

2.2.1 Tipos de Pacotes

Nas ICN, todos os dados que podem ser requisitados na rede são denominados Objetos de Dados Nomeados (do inglês, *Named Data Object* - NDO). Esses NDOs muitas vezes tem que ser transformados em unidades menores que serão transportadas na rede, como segmentos ou *chunks* (KUTSCHER et al., 2016). As estruturas que fazem o transporte dessas unidades podem ser de dois tipos: pacotes de interesse e pacotes de dados. Quando um nó deseja obter um conteúdo, ele envia um pacote de interesse com o nome do conteúdo para um prefixo conhecido ou via *broadcast* para os nós vizinhos. Ao receber esses pacotes, qualquer nó que tenha esse conteúdo armazenado em *cache* pode responder à requisição (OEHLMANN, 2013).

Um pacote de interesse é gerado por um nó consumidor que faz requisição para um determinado conteúdo. De acordo com (JACOBSON et al., 2009), um interesse pode ser composto pelo nome do conteúdo, por um campo que provê seletores ou especificadores que auxiliam na descoberta do dado e por um campo denominado *nonce*, que pode ser utilizado para detectar e prevenir *loops* de pacotes de interesse e remover duplicatas. Esse modelo de pacote de interesse pode ser observado na Figura 2.3.

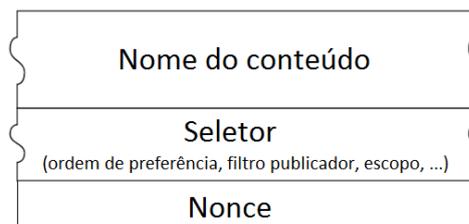


Figura 2.3 Modelo de pacote de interesse. Traduzido de (JACOBSON et al., 2009).

Na Figura 2.4 é representado o pacote de interesse utilizado no projeto CCN. O pacote é composto por campos como tipo de pacote, tamanho, nome, metadados e *payload*. Nesse projeto, os pacotes são caracterizados por um modelo TLV (do inglês, *Type, Length, Value*), no qual, para cada campo TLV, são especificados uma tupla composta por tipo, tamanho e valor. Desta forma, o campo Nome TLV, representa, na verdade, o tipo do nome, o tamanho e o nome propriamente dito. Além disso, apesar de o *payload* ser opcional, é comum que pacotes de interesse tenham *payload* vazio, visto que não há nenhum dado a ser transportado (AHMED; BOUK; KIM, 2016).

Como resposta a um pacote de interesse enviado por um consumidor, um publicador deve enviar um pacote de dados. Ele pode ser composto pelo nome do conteúdo, por uma assinatura que garanta a proveniência, pelas informações de segurança, como chave pública, e pelo próprio conteúdo (JACOBSON et al., 2009). Assim como o pacote de interesse, na proposta CCN o pacote de dados também é representado em um modelo TLV, como pode ser observado na Figura 2.5.

A comunicação entre os nós é sempre realizada através da troca de pacotes de interesse e dados através de um modelo publica/assina.

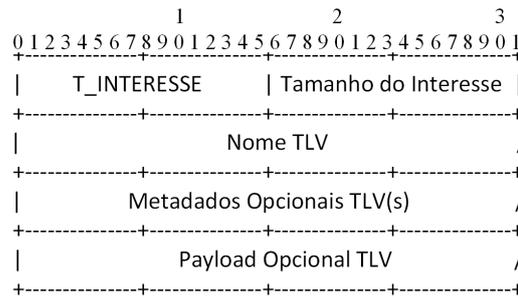


Figura 2.4 Pacote de Interesse CCN composto pelos campos tipo de pacote, tamanho do pacote, nome, metadados opcionais e *payload* opcional. Traduzida e adaptada de (IETF, 2014)

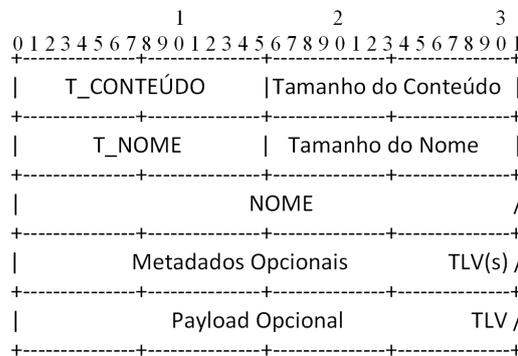


Figura 2.5 Pacote de Dados CCN composto pelos campos tipo, tamanho, nome, metadados opcionais e *payload* opcional. Traduzida e adaptada de (IETF, 2014)

2.2.2 Modelo Publica/Assina

Um modelo publica/assina define dois papéis principais que se relacionam ao envio de pacotes (de interesse e de dados): consumidores/assinantes e publicadores. Os consumidores são as entidades que solicitam conteúdo, através do envio de pacotes de interesse. Enquanto os publicadores são as entidades que provêem conteúdos, através do envio de pacotes de dados.

Na Figura 2.6 é demonstrada a troca de pacotes entre publicador e consumidor. Em a) um pacote de interesse é enviado de um consumidor a um publicador e em b) o pacote de dados referente ao interesse é encaminhado de volta ao consumidor. Observa-se que uma cópia do conteúdo é mantida nos equipamentos de rede que estavam no caminho percorrido pelo pacote de dados. E, embora o usuário não precise saber a localização de um publicador, os equipamentos de rede devem ter alguma informação que indique quem tem o conteúdo solicitado (KUTSCHER et al., 2016).

O modelo publica/assina é caracterizado pelo envio de um pacote de dados apenas em resposta a um pacote de interesse e é apropriado para sistemas com disseminação de conteúdos em grande escala, adequando-se ao perfil atual dos usuários e do tráfego. Nesse modelo os usuários são capazes de selecionar e filtrar as informações que deseja e

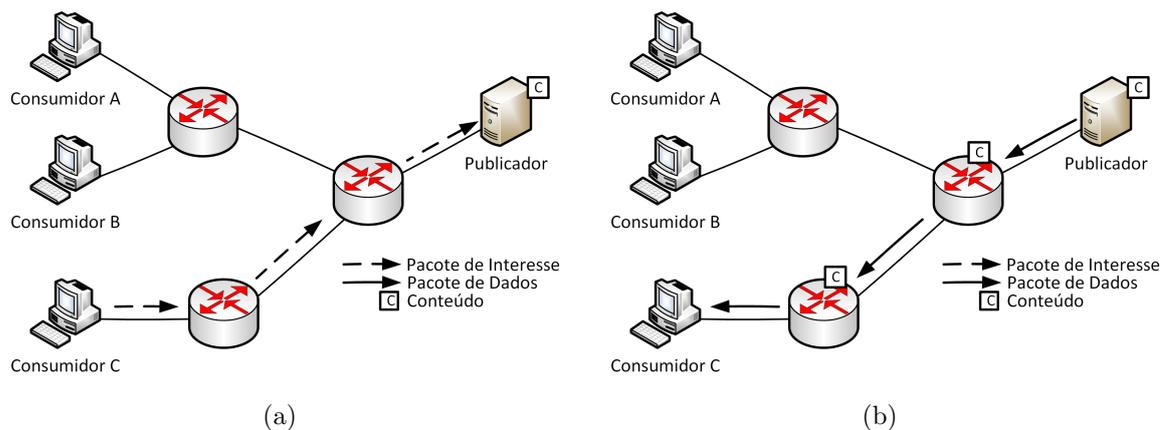


Figura 2.6 Encaminhamento de pacotes de interesse e de dados. Adaptado de (GUIMARAES, 2013)

solicitá-las diretamente à rede, evitando que os consumidores aceitem tudo que é enviado por um publicador (ARUMAITHURAI et al., 2014).

O principal propósito da ICN é conseguir conectar um publicador a um consumidor, em vez de rotear os pacotes a um endereço físico final. A forma como é feita essa comunicação depende, principalmente, do sistema de nomeação utilizado. Pois a partir dele serão definidas questões de segurança e roteamento.

2.2.3 Sistemas de Nomeação

Em uma ICN os equipamentos de rede e suas interfaces não são nomeados. Portanto, o sistema de nomeação é responsável por determinar apenas a formação do nome dos NDOs, que não dependerá de sua localização, uma vez que existem cópias de um mesmo NDO em diferentes lugares da rede (KUTSCHER et al., 2016). O sistema de nomeação influencia fatores como segurança e roteamento, e deve apresentar requisitos como unicidade, autenticação, persistência e escalabilidade. (BRITO; VELLOSO; MORAES, 2012) (AHLGREN et al., 2012).

Os dois principais sistemas de nomeação utilizados são a nomeação plana e a nomeação hierárquica. A nomeação hierárquica é composta por um agregado de prefixos legíveis que vão sendo adicionados à medida que o conteúdo desce de nível em um determinado escopo, com uma estrutura similar às URIs. Esse tipo de nomeação pode ser visto na Figura 2.7, na qual o prefixo representa um nome que é roteável, o sufixo representa o nome do NDO e a parte representa o pedaço da unidade transportada (segmento ou *chunk*)(JACOBSON et al., 2009).

A principal vantagem da nomeação hierárquica é a possibilidade de agregação de nomes, que melhora a escalabilidade do sistema de roteamento e pode ser tratada pelos mesmos mecanismos propostos para sanar a agregação de rotas do IP. Além disso, o fato de ser legível a seres humanos torna o processo de requisição de conteúdos mais simples, já que o usuário consegue identificar um conteúdo de acordo com sua nomenclatura.



Figura 2.7 Exemplo de nomeação hierárquica composta por prefixo, sufixo e parte do conteúdo.

No entanto, os mecanismos de segurança não são intrínsecos e devem ser estabelecidas outras estruturas capazes de verificar a integridade e autenticidade do conteúdo (BRITO; VELLOSO; MORAES, 2012) (KUTSCHER et al., 2016).

Já a nomeação plana é caracterizada por ser composta por uma cadeia de *bits* derivada por uma função *hash*, por exemplo. Esse tipo de nomeação se adéqua melhor aos requisitos de privacidade e segurança porque não é legível ao ser humano e porque na própria construção do nome do conteúdo é possível implantar mecanismos de integridade e autenticidade. Suas principais desvantagens envolvem a necessidade de algum sistema para resolução de nomes, a incompatibilidade com o *Domain Name System* (DNS), sendo necessária a implementação de algum mecanismo que faça o intermédio entre esses dois paradigmas (BRITO; VELLOSO; MORAES, 2012), e a necessidade de utilização de alguma estrutura que facilite a agregação de rotas (KUTSCHER et al., 2016).

Em (GHODSI et al., 2011) é feita uma avaliação dos dois tipos de nomeação levando em consideração a segurança, a escalabilidade e a flexibilidade. Do ponto de vista de segurança, a nomeação hierárquica consegue estabelecer uma relação entre o nome do conteúdo e a identidade do mundo real (do inglês, *Real-World Identity* - RWI) de forma intrínseca, mas há a necessidade de uma autoridade certificadora para garantir uma ligação entre o nome e a chave pertencente à entidade. Por exemplo, é possível observar a relação entre um conteúdo nomeado: `youtube/rollingstones/cd1/parte1` e a entidade responsável pelo conteúdo: `youtube`, em seguida, é necessário garantir que a chave de criptografia utilizada no NDO é, de fato, da entidade citada. Já no caso da nomeação plana, a construção do nome é realizada através da aplicação de uma função *hash* na própria chave pública. Desta forma, é possível garantir a relação entre nome e chave, mas a relação entre a RWI e a chave deve ser feita por uma autoridade externa.

Observa-se, portanto, que ambas abordagens lidam com a segurança de diferentes maneiras. Embora a nomeação hierárquica estabeleça relação entre nome e RWI, não há garantia de que o nome seja bem compreendido e que não seja ambíguo. Isso acaba prejudicando a flexibilidade dos mecanismos de segurança, pois além de uma entidade que consiga mapear uma chave pública a uma RWI, é necessário um mecanismo externo que garanta o mapeamento entre uma RWI e um nome. Desta forma, os nomes planos são considerados mais eficientes em termos de segurança e flexibilidade.

Do ponto de vista da escalabilidade, a nomeação hierárquica consegue reduzir tanto o tamanho quanto a frequência de atualização das tabelas de roteamento (GHODSI et al., 2011), enquanto a nomeação plana tem uma baixa capacidade de agregação, que acaba causando um aumento na quantidade de entradas de rotas e prejudica a escalabilidade. Além disso, os nomes planos não são legíveis, portanto, é necessário que haja um mapeamento entre nomes legíveis e ilegíveis para que o usuário consiga solicitar o conteúdo que deseja (ZHANG et al., 2017).

Além desses dois sistemas de nomeação principais, existem ainda sistemas que constroem o nome através de atributos do conteúdo, como categoria, formato, data, entre outros (CARZANIGA; RUTHERFORD; WOLF, 2004) e sistemas híbridos, que buscam maximizar as vantagens de cada sistema de nomeação citado em busca de superar os desafios associados a cada um (ZHANG et al., 2017).

2.2.4 Roteamento

O tipo de roteamento depende diretamente do tipo de nomeação utilizada. Apesar disso, existem requisitos que devem ser garantidos, como: orientação a conteúdo, robustez, eficiência e escalabilidade (JACOBSON et al., 2009) (BRITO; VELLOSO; MORAES, 2012) (OEHLMANN, 2013). Um dos maiores desafios, independente do tipo de nomeação, é que nas ICN as informações de localização são descartadas e os conteúdos devem ser encontrados através dos metadados e do próprio nome do NDO. Apesar de haver réplicas dos conteúdos espalhadas nos *caches* da rede, se não houver algum indicativo de onde esse conteúdo está, o encaminhamento pode se tornar ineficaz.

De acordo com (KUTSCHER et al., 2016), o processo de roteamento pode ser dividido em três etapas:

1. Resolução do nome: consiste em determinar a localização do NDO através do seu nome (`ufba/dcc/mata31/parte02` - prefixo é roteável)
2. Descoberta: realiza o envio do pacote de interesse do NDO para a localização identificada (publicador)
3. Entrega: envia o pacote de dados do NDO para o consumidor

(KUTSCHER et al., 2016) categorizam ainda três tipos de roteamento:

- Roteamento Rota-por-nome: na tabela do roteador existe um mapeamento entre os prefixos e a localização, desta forma, a etapa de resolução de nomes é eliminada. No entanto, se não houver um mecanismo de agregação, a quantidade de entradas na tabela se torna um problema, visto que cada conteúdo representará uma entrada. Uma vez estabelecida a rota para o publicador, é preciso que seja criado um identificador ou localizador para o consumidor ou que seja introduzido um mecanismo capaz de identificar a sua localização para que seja possível encaminhar o pacote de dados. Um dos mecanismos mais utilizados é o chamado “migalhas de pão” (ROSENSWEIG; KUROSE, 2009). Esse mecanismo é demonstrado na Figura 2.8: toda vez que um pacote de interesse é encaminhado, os nós intermediários armazenam a interface em que o pacote foi recebido (a). Desta forma, quando o pacote de dados é encontrado, ele é encaminhado pelo caminho inverso até chegar no consumidor, ainda que exista um caminho mais curto entre produtor e consumidor (b).
- Roteamento Pesquisa-por-nome: as etapas 1 e 2 (resolução do nome e descoberta) são utilizadas para fazer a tradução do nome para a localização e roteamento do pacote de interesse (não é definido especificamente como isso é feito, esse é um dos

desafios do roteamento em ICN). Já a etapa de entrega é implementada baseada no roteamento IP, desta forma, o localizador do consumidor é incluído no pacote de interesse. Uma vez que o publicador saiba a localização do consumidor, o pacote de dados é enviado. Um exemplo de resolução e descoberta pode ser observado no modelo PSIRP, que define um método baseado em um ponto de encontro ou *rendezvous*, conforme a Figura 2.9: em (1) o publicador anuncia o conteúdo C na rede e em (2) o consumidor faz a solicitação para esse mesmo conteúdo. No *rendezvous* é realizado o mapeamento (3) entre publicador e assinante e o endereço do publicador é enviado para o consumidor. Esse endereço pode ser o endereço IP ou algum outro mecanismo de localização proposto. Uma vez que o consumidor saiba o endereço do publicador, a troca de dados pode ocorrer via IP (4).

- Roteamento Híbrido: combinação dos dois tipos de roteamento listados.

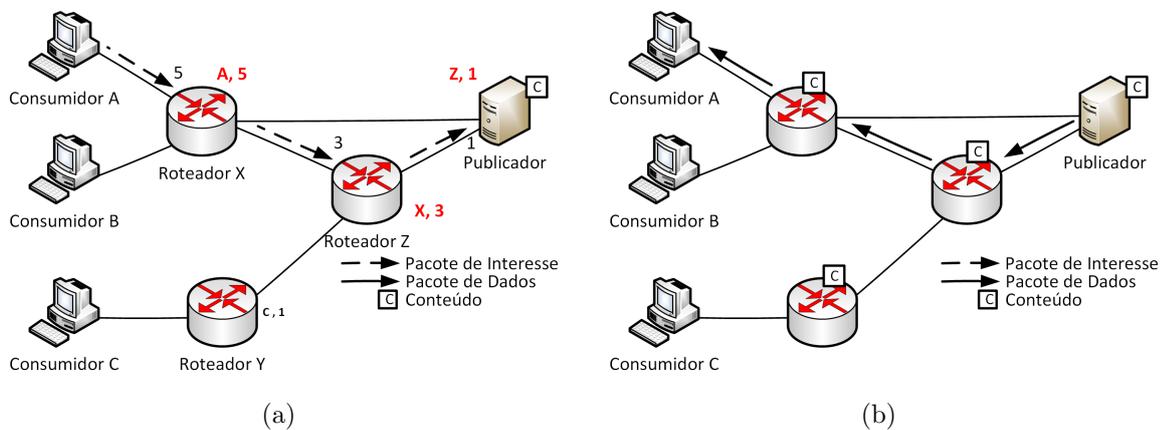


Figura 2.8 Representação do roteamento baseado em “migalhas de pão”.

Também é possível categorizar o tipo de roteamento de acordo com sua hierarquia. O roteamento não-hierárquico não é estruturado em níveis e todos os nós da rede tem capacidade de obter conteúdos válidos. As informações de roteamento precisam ser difundidas em escala global, e essa característica permite que os nós calculem melhores rotas para entrega mas pode se tornar pouco aplicável para redes mais robustas, visto que exige uma visão geral da rede (BRITO; VELLOSO; MORAES, 2012) (AHLGREN et al., 2012).

Já no roteamento hierárquico, os equipamentos de rede são distribuídos em diferentes níveis hierárquicos e a comunicação entre eles é determinada a partir da comunicação vertical entre nós que estejam dentro de um mesmo escopo e de níveis diferentes e a comunicação horizontal entre nós que estejam entre diferentes escopos mas no mesmo nível hierárquico. Essas relações hierárquicas permitem que sejam trocadas menos informações de controle, diminuindo o tráfego da rede (BRITO; VELLOSO; MORAES, 2012) (AHLGREN et al., 2012).

Na arquitetura ICN proposta em (JACOBSON et al., 2009), é possível observar como são relacionadas a estrutura hierárquica de nomeação e de roteamento, conforme Figura

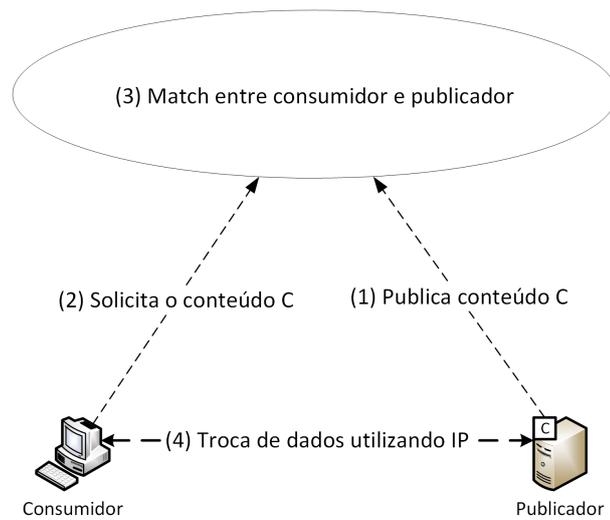


Figura 2.9 Roteamento baseado no modelo proposto no projeto PSIRP (AHLGREN et al., 2012)

2.10: é criada uma estrutura de árvore na qual cada parte do nome é encaminhada para uma folha da árvore, determinando a rota que deve ser percorrida para obtenção de um NDO já segmentado.

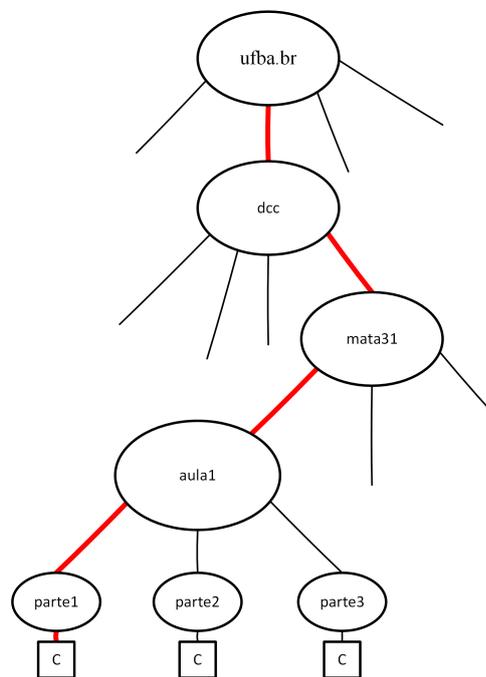


Figura 2.10 Modelo de roteamento em árvore. Fonte: Adaptado de (JACOBSON et al., 2009)

Independente do tipo de roteamento utilizado, (KUTSCHER et al., 2016; JACOBSON et al., 2009) definem algumas estruturas básicas que um roteador ICN deve ter para

garantir o funcionamento da rede com suporte à conteúdo nomeado.

2.2.5 Equipamentos de Rede

Os equipamentos de rede que compõem a arquitetura ICN são constituídos de *cache* ou *Content Store* (CS), Tabela de Interesses Pendentes (do inglês, *Pending Interest Table* - PIT) e Base de Informação de Encaminhamento (do inglês, *Forwarding Information Base* - FIB). Essas três estruturas são responsáveis por fazer armazenamento de conteúdo mais próximo do usuário, agregação de requisições e encaminhamento de pacotes nomeados através da rede. As principais características dessas estruturas e seu funcionamento serão explicados nas próximas seções.

2.2.5.1 Cache A primeira estrutura que compõe os roteadores e pode estar presente nos dispositivos dos usuários finais e outros equipamentos da rede são os *caches* ou *Content Store*. De acordo com (WANG, 1999), a utilização de *cache* deixa o conteúdo mais próximo do usuário, gerando melhoria na qualidade de serviço da rede e diminuindo o tráfego.

A utilização de *caches* locais para armazenamento de NDOs aumenta a tolerância a falhas, a resiliência e a disponibilidade, e possibilita o descarregamento de servidores e de partes da rede. Além disso, modifica o comportamento fim-a-fim das camadas de rede e de transporte da arquitetura atual, visto que uma solicitação pode ser respondida por diferentes nós (KUTSCHER et al., 2016). Esse comportamento pode ser visto na Figura 2.11: em a) o consumidor requisita um pacote de interesse, e em b) o pacote de dados referente ao interesse é encaminhado de volta ao consumidor por um nó intermediário que mantinha o conteúdo armazenado em *cache*. Assim, não houve a necessidade de alcançar o publicador, pois um dos roteadores percorridos já tinha o conteúdo. Esse tipo de armazenamento nos equipamentos de rede é denominado de *in-network caching*.

A ideia de utilizar *cache* para deixar o conteúdo mais próximo do usuário é uma característica das Redes de Distribuição de Conteúdo. No entanto, enquanto em uma CDN há um controle centralizado que gerencia as réplicas, em uma ICN as decisões de armazenamento são feitas pelo próprio equipamento e os equipamentos dos usuários também podem atuar como *cache* local. Desta forma, apesar do uso de *cache* ser algo comum desde o surgimento da *Web* e da adoção de CDN, o principal diferencial das ICN é a formação de uma rede de *cache* distribuída entre os equipamentos de rede e dos usuários. Assim, como os equipamentos de rede, como roteadores, tem capacidade de armazenamento bastante limitada, os principais desafios na área continuam envolvendo a definição de estratégias de armazenamento e de descarte de conteúdo que tornem a utilização desses *caches* mais eficiente (AHLGREN et al., 2012).

As estratégias de armazenamento também são chamadas de estratégia de descarte proativo, visto que ao decidir o que deve ou não ser armazenado, já se faz uma espécie de descarte. Enquanto as estratégias que removem conteúdo apenas quando a capacidade máxima do *cache* é atingida são chamadas de estratégias reativas. As principais estratégias de descarte em ICN são a LRU, que descarta o conteúdo que foi acessado menos recentemente, partindo do pressuposto de que se ele foi acessado a mais tempo, a

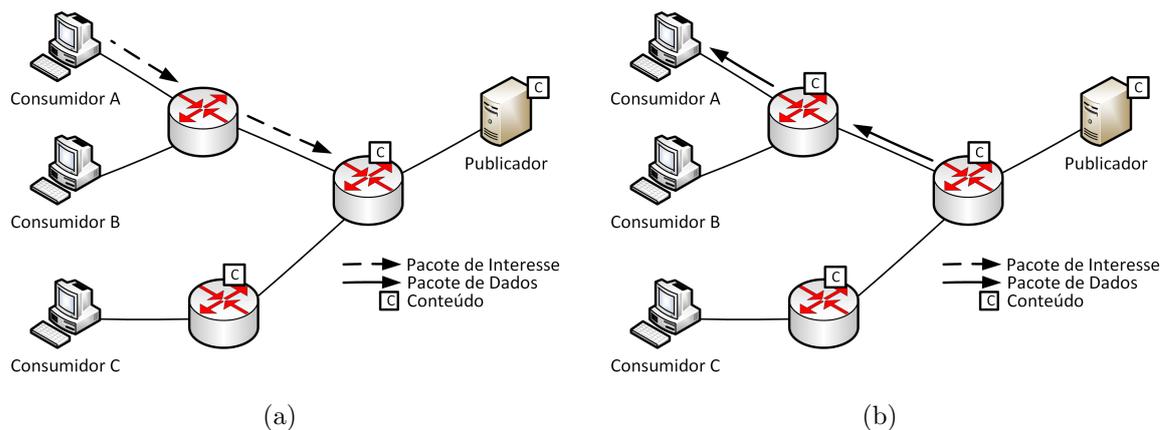


Figura 2.11 Uso do *cache* para deixar o conteúdo mais próximo do usuário. Adaptado de (GUIMARAES, 2013)

importância para ele na rede no momento atual é menor do que os outros conteúdos, e a LFU, que descarta o conteúdo que foi usado menos frequentemente. Utilizando a idéia de que se um conteúdo é acessado com mais frequência, ele deve ser mais importante na rede (CHE; TUNG; WANG, 2002).

Além da LFU e LRU, existem diversas outras estratégias propostas:

- FIFO: é baseada na política de atendimento de requisições em *buffer* e funciona da seguinte maneira: se o *buffer* ficar cheio e chegar uma nova mensagem, a primeira mensagem que chegou no *buffer* será descartada. Essa política parte do pressuposto de que mensagens novas não devem ser descartadas, visto que as mensagens mais antigas estão na rede a mais tempo e a probabilidade de terem chegado ao destino é maior. Contudo, essa política não leva em consideração aspectos como o conteúdo da mensagem e a quantidade de vezes que foi encaminhada, por exemplo (LINDGREN; PHANSE, 2006).
- MOFO (do inglês, *Most Forwarded*): descarta a mensagem que foi encaminhada mais vezes. Na implementação dessa política, cada mensagem utiliza uma variável contadora no cabeçalho, e cada vez que ela for encaminhada é adicionada uma unidade ao contador. Esse gerenciamento leva em consideração o fato de que as mensagens menos encaminhadas tem menos probabilidade de terem chegado ao destino, sendo assim, ao mantê-las no *cache*, a chance dela chegar aumenta. Então mesmo se a mensagem mais encaminhada for descartada desse *cache*, possivelmente não será perdida (LINDGREN; PHANSE, 2006).
- LRF (do inglês, *Least Recently Forwarded*): descarta a mensagem que foi encaminhada a mais tempo. Acredita-se que as mensagens que foram encaminhadas a mais tempo já chegaram ao destino e que não haverá perda em caso de remoção. No entanto, se nenhuma mensagem que está no *cache* tiver sido encaminhada, a

política LRF utiliza a FIFO para fazer o descarte de mensagens (GOMES et al., 2014).

A estratégia de descarte deve ser adequada ao cenário de utilização. Em casos de desenvolvimento de uma estratégia própria, é preciso considerar que as políticas de descarte devem ter heurísticas simples, visto que além das limitações de armazenamento esses dispositivos contam também com limitações de processamento. Portanto, é preciso haver um equilíbrio entre a complexidade da política criada e a capacidade de processamento desses dispositivos (MOREIRA et al., 2009).

2.2.5.2 Forwarding Information Base A segunda estrutura que compõe os roteadores é a FIB ou Base de Informações de Encaminhamento. Ela pode ser utilizada para realizar a primeira etapa do roteamento (resolução de nomes), visto que armazena um conjunto de prefixos pareados a um conjunto de interfaces de saída, e é responsável por determinar para onde os pacotes devem ser encaminhados (KUTSCHER et al., 2016).

As diferenças fundamentais entre a FIB proposta nas ICN e uma FIB IP podem ser observadas na Figura 2.12. Enquanto a FIB tradicional mantém o prefixo da rede destino, o próximo salto e a interface de saída, a FIB ICN mantém apenas o prefixo roteável (nome) do produtor e as interfaces de saída. Podendo ser observada mais uma adequação que visa substituir o roteamento baseado em localização por roteamento baseado em nome.

FIB IP			FIB ICN	
Endereço Destino	Próximo Salto	Interface	Prefixo	Interface
10.0.0.0/8	10.6.4.22	0	/ufba/	1,3
172.20.0.0/16	172.20.3.61	4	/youtube/brasil/	4
203.0.113.168/29	203.0.113.171	1	/netflix/filmes/	2
10.0.0.0/16	10.2.2.8	2	/facebook/	5

Figura 2.12 Comparação entre a Base de Informações de Encaminhamento no modelo IP e no modelo ICN.

As principais considerações que devem ser feitas em relação à FIB ICN é que, assim como na FIB IP, o tamanho dos prefixos podem impactar no desempenho do roteador e na entrega dos conteúdos. Na FIB tradicional essa relação é feita da seguinte forma: quanto maior o prefixo de rede, pior o desempenho do roteador, visto que serão feitas mais verificações para encontrar um *match* (TROTTER, 2001). Já na FIB ICN, as preocupações envolvem a diversidade de prefixos e o tamanho do campo da *string* que define o prefixo, já que diferentes prefixos tem tamanhos variados. De acordo com (AHLGREN et al., 2012), o número de prefixos que podem ser suportados para que seja feita apenas

uma busca na memória externa, ao considerar uma memória interna de 200Mb, é de 25% das estações ativas da Internet atual.

A FIB é utilizada apenas para roteamento de pacotes de interesse e por isso sempre mapeia um prefixo de um produtor a uma interface, sem manter informações sobre o solicitante. Quando um pacote de interesse deve ser encaminhado através da rede, é necessário verificar o prefixo do nome do conteúdo e identificar qual(s) a(s) interface(s) de saída para onde o interesse deve ser encaminhado. Caso não haja uma entrada representando o prefixo na FIB, o pacote de interesse pode ser descartado ou encaminhado via *broadcast* (JACOBSON et al., 2009; KUTSCHER et al., 2016). A entrega do pacote de dados é realizada através de outros mecanismos, como o algoritmo de “migalhas de pão” citado na seção 2.2.4, que faz uso de entradas na PIT para encaminhar pacotes de dados.

2.2.5.3 Pending Interest Table A PIT ou Tabela de Interesses Pendentes é a terceira estrutura que compõe os equipamentos de rede. A composição básica dessa tabela pode ser observada na Figura 2.13: um NDO é associado a uma ou mais interfaces que o solicitaram (AHLGREN et al., 2012; JACOBSON et al., 2009; KUTSCHER et al., 2016).

PIT	
Nome do conteúdo	Interface solicitante
/ufba/dcc/redes/icn/parte1	2,5
/ufba/dcc/redes/icn/parte2	2,5
/netflix/series/dexter/tmp1/ep1	1
/rfc/icn/pit/parte3	2,5

Figura 2.13 Representação da Tabela de Interesses Pendentes.

Assim, percebe-se que a PIT foi construída para evitar problemas de escalabilidade ao fazer agregação de solicitações para um mesmo conteúdo. Logo, se mais de um consumidor solicitar conteúdos iguais, é adicionada apenas uma interface correspondente na lista de interfaces solicitantes, não sendo necessária a criação de uma nova entrada na PIT nem o reenvio do pacote de interesse (JACOBSON et al., 2009). Essa característica acaba contribuindo para a redução do consumo da largura de banda no núcleo da rede, pois apesar de receber pacotes que representam um mesmo interesse, o roteador só o encaminha uma única vez.

Além da agregação de solicitações para um mesmo conteúdo, a PIT pode registrar o campo *nonce* (apresentado na Seção 2.2.1) presente no pacote de interesse para verificar se a requisição trata-se de um novo pacote de interesse ou de um interesse já encaminhado. Essa estratégia consegue impedir a formação de *loops* na rede. (GUIMARAES, 2013).

Como a PIT representa os interesses que ainda não foram satisfeitos pelos usuários, a forma mais comum de uma entrada na PIT ser descartada é através da chegada de um pacote do pacote de dados correspondente. No entanto, para evitar que solicitações para conteúdos inexistentes ou não encontrados ocupem o espaço da PIT por tempo indeterminado, é comum a adição de um campo TTL (do inglês, *Time-To-Live*) para expiração de requisições que estão na PIT (GUIMARAES, 2013). Assim, é de responsabilidade do usuário enviar novamente o seu interesse por um conteúdo solicitado que não chegou.

2.2.6 Encaminhamento e Aquisição de Conteúdos

O encaminhamento de pacotes e a aquisição de conteúdo são realizados utilizando as três estruturas citadas na seção anterior. Ao receber um pacote de interesse, o roteador faz verificações nas estruturas sempre de acordo com a seguinte ordem: *cache*, PIT, FIB. Na Figura 2.14 é possível observar um nó CCN composto por essas estruturas: são definidas as diferentes interfaces (que não são, necessariamente, físicas, podendo incluir a própria aplicação) e as três estruturas principais de um nó. Essas estruturas fazem parte do mecanismo de encaminhamento de pacotes e na chegada de um pacote de interesses a verificação é feita de acordo com a ordem de definição dos ponteiros: é verificado o *cache*, depois a PIT e depois a FIB (JACOBSON et al., 2009).

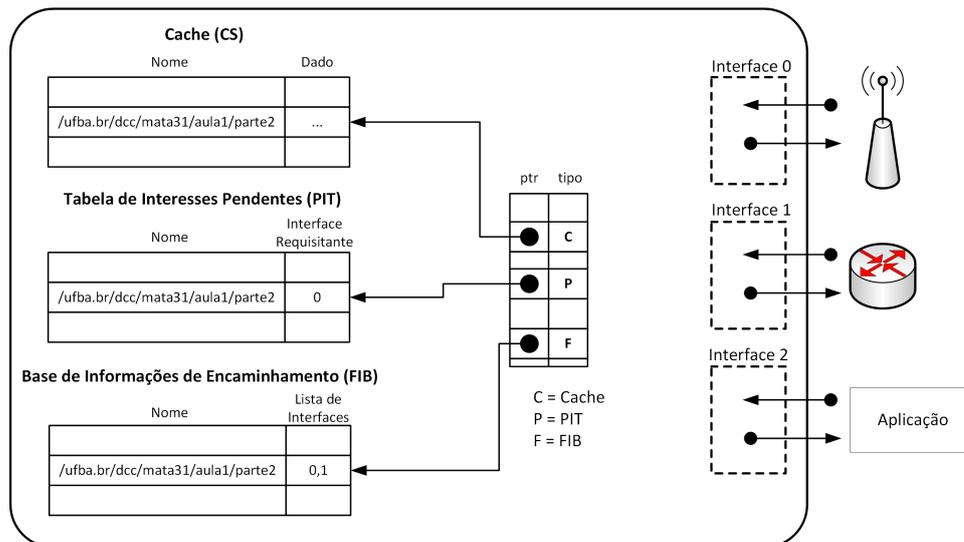


Figura 2.14 Funcionamento das estruturas básicas na arquitetura ICN. Adaptado de (JACOBSON et al., 2009)

Como o paradigma ICN se baseia no modelo publica/assina, o fluxo de verificações é iniciado, normalmente, com a chegada de um pacote de interesse. O Fluxograma 2.15 representa todo o processo realizado quando um pacote de interesse chega no roteador: a primeira estrutura que deve ser verificada é o *cache*, para determinar se existe um pacote de dados já armazenado que satisfaça àquela requisição. Se existir, o pacote de dados é enviado de volta à interface pela qual chegou o pacote de interesse. Se não existir, é preciso fazer a verificação para adição da entrada na PIT. Se já existir uma entrada

para aquele conteúdo, apenas a interface solicitante é adicionada e o pacote de interesse é descartado. No entanto, caso não exista uma entrada, é verificado se a FIB tem as informações de encaminhamento para aquele prefixo. Se a FIB não tiver as informações de encaminhamento, o pacote de interesse é descartado. Se ela tiver, é adicionada uma entrada na PIT com nome da NDO e interface solicitante e o interesse é encaminhado para o próximo nó.

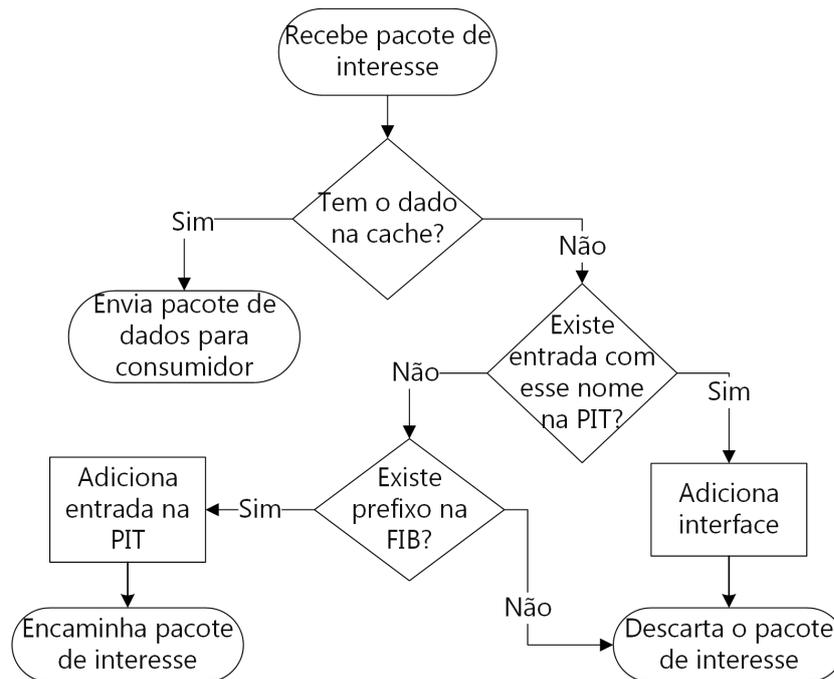


Figura 2.15 Fluxograma representando as ações realizadas por um roteador ao receber um pacote de interesse.

Ao receber um pacote de dados, o nó verifica duas coisas: se tem um interesse para aquele conteúdo que ainda não foi satisfeito e se o conteúdo está armazenado em *cache*. Normalmente, a ICN trabalha com uma estratégia que armazena todo pacote de dados que chega, então, se o *cache* não tem o conteúdo, ele adiciona. Além disso, é enviada uma réplica do pacote de dados para todas as interfaces que solicitaram aquele NDO.

Uma ICN pode ser implementada de diferentes formas e em diferentes níveis: em mais baixo nível, envolve a substituição a nível de pacote, cujo impacto causa mudanças nos principais fundamentos utilizados atualmente nas camadas de rede e transporte, sendo necessário redefinir os procedimentos de roteamento e encaminhamento. Já em mais alto nível, a rede é implementada sob a infraestrutura atual, com algumas adaptações que permitam estruturas nomeadas, utilização de *cache* nos equipamentos de rede, entre outros (KUTSCHER et al., 2016).

2.3 PROJETOS DE ICN

A ICN é uma proposta de arquitetura genérica com diferentes desafios e alternativas para questões como nomeação, roteamento, mobilidade, entre outros. Diferentes instituições começaram a definir projetos para a evolução e aplicação da arquitetura ICN. Dentre os principais projetos estão o CCN, o DONA, o PSIRP e o NetInf. Alguns trabalhos (MARTINS; ALBERTI, 2011; AHLGREN et al., 2012; XYLOMENOS et al., 2014; LIU et al., 2017) tem abordado as diferentes características dessas propostas, bem como seus principais desafios e contribuições.

CCN/NDN¹ é uma proposta ICN que faz parte do projeto NDN² criado em 2010 pelo *Palo Alto Research Center*. Esse projeto visa estabelecer uma alternativa que seja capaz de ampliar as vantagens da arquitetura atual e identificar e resolver os principais desafios, como escalabilidade de roteamento, velocidade de encaminhamento, privacidade e proteção de conteúdo, entre outros (NDN, 2010). Suas principais características são nomeação hierárquica, roteamento baseado em nomes e a utilização do algoritmo “migas de pão” para rotear conteúdos (ZHANG et al., 2010). Essas características básicas fundamentam e influenciam o desenvolvimento de pesquisas em NDN, visto que a determinação da nomeação, por exemplo, impacta em diversos fatores. Desta forma, tem sido desenvolvidas documentações e ambientes de experimentação para incentivar e validar pesquisas utilizando as premissas adotadas no projeto .

O DONA (KOPONEN et al., 2007) apresenta uma abordagem *clean slate* orientada a dados que traz novas propostas para nomeação e sistema de resolução de nomes com o objetivo de alcançar persistência, disponibilidade e autenticação sem necessidade de estruturas externas. O DONA faz uso de nomeação plana e, conseqüentemente, a persistência e a autenticação dos dados é intrínseca ao conteúdo. Além disso, para garantir a disponibilidade e lidar com o roteamento, o DONA propõe um mecanismo de resolução de nomes. Esse mecanismo tem o objetivo de fazer o mapeamento entre um conteúdo e sua localização e utiliza *Resolution Handlers* (RHs) para encontrar a cópia mais próxima do cliente.

O PSIRP (DIMITROV; KOPTCHEV, 2010) foi criado em 2009 com o objetivo de substituir o modelo IP por um modelo do tipo publica/assina. Esse modelo consiste nas atividades de solicitação e encaminhamento de conteúdos, na qual um cliente faz a assinatura e qualquer nó que tenha o conteúdo disponível pode servi-lo. Para isso, é utilizado um sistema de mapeamento (*rendezvous*) que identifica qual nó tem o conteúdo e envia o endereço do nó para o assinante. Com o endereço do publicador, o assinante pode enviar a solicitação diretamente. As evoluções do PSIRP consistem no projeto PURSUIT³.

O NetInf⁴ (DANNEWITZ, 2009) foi desenvolvido no projeto *Design for the Future Internet* (4WARD) e também fez parte do SAIL⁵ (*Scalable Adaptive Internet Solution*).

¹Nesta dissertação, CCN e NDN referem-se ao mesmo projeto

²<https://named-data.net/>

³<http://www.fp7-pursuit.eu/PursuitWeb/>

⁴<http://www.netinf.org/>

⁵<http://www.sail-project.eu/>

O objetivo do projeto é utilizar dados nomeados e *cache* para aplicar as características do modelo *peer-to-peer* na própria arquitetura da rede. Para isso, foi criado o protocolo NetInf (KUTSCHER; FARRELL; DAVIES, 2013) que define o princípios e os detalhes do projeto. O propósito do protocolo é encaminhar requisições de NDOs e transferir dados para um consumidor. Isso é feito utilizando nomeação plana e um sistema de resolução de nomes, que pode ser uma funcionalidade dos próprios equipamentos de rede.

Na Tabela 2.2 é possível observar um comparativo entre as diferentes propostas descritas e suas principais características. Observa-se portanto, que existem abordagens com características evolucionárias e que buscam se basear na arquitetura atual e aproveitar as soluções já existentes utilizando IP, enquanto outros projetos tem um caracter mais *clean slate* e propõem novas alternativas que não são compatíveis com a arquitetura atual. A importância de ter essas diferentes abordagens é que as pesquisas crescem em diferentes direções e acabam cobrindo mais possibilidades.

	DONA	CCN	PSIRP	NetInf
Nomeação	Plana	Hierárquica	Plana	Plana
Legibilidade dos nomes	Não	Sim	Não	Não
Segurança	Assinatura, PKI independente	Assinatura, autoridade certificadora externa	Assinatura, PKI independente	Assinatura ou <i>hash</i> , PKI independente
Agregação de rotas	Publicador/ explícito	Publicador	Escopo/ explícito	Publicador
Roteamento dos pacotes de interesse	Baseado em nomes (via RHs)	Baseado em nomes	Sistema de Resolução de nomes	Híbrida
Roteamento dos pacotes de dados	Caminho re-verso ou via IP	Caminho reverso	Roteamento baseado em Filtro de Bloom	Caminho re-verso ou via IP
Transporte	IP	Diferentes protocolos, incluindo IP	IP/PSIRP	Diferentes protocolos, incluindo IP

Tabela 2.2: Principais características dos projetos de ICN. Traduzido e adaptado de (AHLGREN et al., 2012)

Além desses projetos mais genéricos, outros projetos também foram desenvolvidos com o objetivo de encontrar soluções para desafios mais específicos, como mobilidade, apoio a centro de dados, entre outros. Dentre esses projetos, existe o MOBILITYFIRST (VENKATARAMANI et al., 2014), o GREENICN⁶, o RIFE⁷ e o BONVOYAGE⁸. Além

⁶<http://www.greenicn.org/>

⁷<https://rife-project.eu/>

⁸<http://bonvoyage2020>

disso, em 2012 foi criado um grupo de trabalho no IRTF (Internet Research Task Force)⁹ para determinar padrões e protocolos para a arquitetura ICN através do desenvolvimento de RFCs. Todas essas iniciativas evidenciam a proporção do crescimento de pesquisas em ICN e de alternativas que possam auxiliar o desenvolvimento da arquitetura atual.

2.4 CONSIDERAÇÕES FINAIS

Neste capítulo foram abordados os principais fundamentos teóricos em relação às Redes Centradas na Informação. Foram expostos o objetivo da arquitetura, seus principais componentes, os tipos de pacote envolvidos no compartilhamento de conteúdo, modelos de roteamento e projetos existentes que se baseiam nessa estrutura. No capítulo seguinte serão expostos os trabalhos relacionados que consideram diferentes alternativas, inclusive o uso de ICN, para prover descarregamento de dados.

⁹<https://trac.ietf.org/trac/irtf/wiki/icnrg>

TRABALHOS RELACIONADOS

As estratégias de descarregamento de dados em redes móveis visam utilizar novas tecnologias e técnicas para realizar a entrega de pacotes, com o intuito de diminuir a sobrecarga nos servidores e no núcleo da rede e otimizar o uso dos recursos existentes (AIJAZ; AGH-VAMI; AMANI, 2013). Diversos trabalhos propõem a adoção de soluções para tratar descarregamento, como a utilização de redes Wi-Fi (LEE et al., 2013; POULARAKIS et al., 2016; DIMATTEO et al., 2011), de redes tolerantes a atrasos e interrupções (do inglês, *Delay-Tolerant Networks*) (LI et al., 2014), redes MESH (APOSTOLARAS et al., 2016), entre outras.

Neste capítulo serão abordadas algumas soluções que utilizam redes Wi-Fi, estratégias cientes de contexto e densificação de redes para prover descarregamento de dados.

3.1 DESCARREGAMENTO DE DADOS UTILIZANDO REDES WI-FI

A utilização de redes Wi-Fi tem sido uma alternativa de descarregamento adotada pelas operadoras por ser algo de baixo custo de implementação. Diversos trabalhos tem sido desenvolvidos nessa linha e trazem propostas que envolvem o uso de estratégias *on-the-spot* e *delayed*, aluguel de largura de banda e criação de *frameworks* que consideram diferentes aspectos da rede e do usuário.

Em (LEE et al., 2013) foi feito um experimento com 75 usuários, durante duas semanas e meia, para avaliar a taxa de descarregamento obtida através da utilização de redes Wi-Fi. Para isso, os autores desenvolveram uma aplicação para *iPhone* que rastreia conexões Wi-Fi e envia para o servidor informações como localização das estações de acesso, duração da conexão e taxa de transferência de dados. Essas informações foram utilizadas como *trace* para realização de simulações que compararam a utilização de dois tipos de estratégias de descarregamento: a *on-the-spot*, que consiste na configuração padrão dos dispositivos, na qual a transferência dos dados é feita a partir da rede Wi-Fi e, em casos de desconexão, continuam a transferência através da rede de dados móveis; e a *delayed*, que determina um prazo máximo para término da transferência dos dados, enquanto o tempo estiver dentro do limite, os dados são trafegados apenas pela rede

Wi-Fi, ainda que haja desconexões a rede móvel só é acionada se o prazo limite para a transferência for ultrapassado. Os resultados obtidos mostraram que mesmo sem utilizar nenhuma estratégia de atraso, a utilização do Wi-Fi alcança uma taxa de descarregamento de 65% e uma economia de energia de 55%. Nas estratégias com atraso, o descarregamento só é vantajoso se o tempo limite de transferência for superior a 10 minutos. Em casos em que o tempo limite é de uma hora, as taxas de descarregamento e economia de energia alcançam 29% e 20%, respectivamente.

Em (POULARAKIS et al., 2016) é proposto um esquema de aluguel de largura de banda utilizando pontos de acesso residenciais com *cache* com o intuito de reduzir o congestionamento nas redes móveis e melhorar a qualidade de experiência do usuário sem sobrecarregar o núcleo da rede. Desta forma, foi feita a avaliação de um *trace*, composto por 167 usuários residenciais em Portugal para avaliar a frequência e os momentos em que os pontos de acesso residenciais eram utilizados. Observou-se que o maior uso da banda ocorria durante fins de semana e que a maioria dos usuários utilizava o Wi-Fi menos de cinco horas por dia, evidenciando ociosidade dos pontos de acesso em momentos que não havia conexão. Além disso, foi avaliado os momentos do dia em que os usuários mais utilizavam a banda. Todas essas avaliações foram levadas em consideração na criação de um *framework* que relacionasse incentivos, *cache* e políticas de roteamento. Os resultados obtidos com a aplicação do *framework* demonstraram que em áreas com grande densidade demográfica e largura de banda ociosa, é possível reduzir pela metade o custo das operadoras. Algumas aplicações práticas nesse sentido já estão em desenvolvimento. No Brasil, operadoras habilitam uma rede auxiliar à rede residencial que pode ser acessada por seus clientes sem causar impactos na velocidade contratada pelo usuário e mantendo sistemas de recompensa em caso de novas assinaturas.

Em (LI et al., 2014) é criado um *framework* que leva em consideração a heterogeneidade dos dados e dos usuários e a capacidade de armazenamento limitado dos dispositivos para avaliar o uso de redes tolerantes a atrasos e interrupções como uma alternativa para prover descarregamento de dados em redes móveis. Os autores definem que a rede tem diferentes usuários, do ponto de vista de interesses por dados e necessidade de privacidade, que os dados tem diferentes tamanhos e sensibilidade a atrasos, que os dispositivos tem capacidade de armazenamento limitado e que nem todos os dispositivos provêm descarregamento de dados. Para lidar com o descarregamento são estabelecidos três algoritmos adequados a cenários diferentes: o *Greedy Algorithm* (GA), recomendado para cenário genérico, o *Approximation Algorithm* (AA), usado para cenários com dados cujo tempo de vida é pequeno, e o *Homogeneous Algorithm* (HA), para cenário com tamanho de conteúdo e taxa de contato homogêneas. Os resultados demonstraram que embora o AA alcance resultados similares ao GA quando o cenário é heterogêneo e tem dados com tempo de vida curto, o GA alcança os melhores resultados para cenários heterogêneos genéricos e o HA com cenários homogêneos. Apesar disso, ao avaliar os algoritmos com um *trace* real, o AA conseguiu resultados similares ao GA, sendo que a complexidade do AA é menor. Possibilitando uma alternativa para descarregamento de dados baseado em DTNs em um cenário real.

3.2 DESCARREGAMENTO DE DADOS CIENTE DE CONTEXTO

Além da adição de estruturas como pontos de acesso Wi-Fi ou pequenas células, algumas propostas (POULARAKIS et al., 2016; LI et al., 2014; ELBAMBY et al., 2014; PANTISANO et al., 2015) levam em consideração o uso de *cache* nos equipamentos de rede. No entanto, como já foi citado anteriormente, os benefícios trazidos com o uso de *cache* podem ser ampliados quando adicionada alguma inteligência ao sistema.

Por tais motivos, o projeto MACACO¹ (do inglês, *Mobile context-Adaptive Caching for Content-centric networking*) foi criado com o objetivo de explorar a adição de *cache* em diferentes pontos da rede como um mecanismo para sanar os problemas decorrentes do alto volume de tráfego nas redes móveis. O projeto apresenta propostas que envolvem soluções cientes de contexto social e de recursos oportunistas pensando em estratégias que englobem tanto o núcleo da rede quando os dispositivos móveis e que direcionem à arquitetura ICN.

Em (WU et al., 2017) é proposto um *framework* para utilização de descarregamento de dados colaborativo através da cooperação dispositivo-a-dispositivo. O trabalho traz como cenário um conjunto de dispositivos móveis que estão fisicamente próximos e que tem interesse por um mesmo conteúdo. A intenção é que a estação base envie o conteúdo uma única vez e que o dispositivo receptor encaminhe o conteúdo para os seus vizinhos. Embora a exploração da proximidade dos usuários para descarregamento de dados seja levada em consideração, não é a posição geográfica do dispositivo móvel que caracteriza sua escolha para recepção e transmissão dos dados, o contexto, nesse caso, é a capacidade energética e o poder de transmissão.

(KO; LEE; PACK, 2017) apresentam um esquema de gerenciamento de localização ciente de mobilidade, em redes celulares de pequeno porte, cujo objetivo é minimizar a frequência de atualizações de localização desnecessárias e maximizar o descarregamento de dados. Comumente, o gerenciamento da localização dos dispositivos móveis é realizado através do estabelecimento de áreas de localização e cada área de localização mantém um identificador único. Caso o dispositivo se mova de uma área para outra, ele deve enviar uma mensagem para a entidade de gerenciamento de mobilidade. O problema ocorre quando existem dispositivos com alta mobilidade e há a necessidade de fazer atualizações de localização com frequência, visto que as femto-células tem um raio de alcance pequeno. No entanto, caso as operações de localização não sejam realizadas, há uma diminuição do descarregamento de dados. Para resolver esse problema, o esquema criado pelos autores usa um *Dual Home Agent*, que mantém duas áreas de localização: a da femto-célula atual e a da macro-célula. Assim, quando há mudança de localização entre femto-célula e macro-célula, não há necessidade de fazer atualização. Além disso, os autores levam em consideração a probabilidade do nó enviar pacotes e permanecer dentro do alcance da femto-célula para determinar se deve ser feita atualização de uma femto-célula para outra. Os resultados obtidos demonstraram que o esquema criado consegue melhores resultados em diferentes ambientes quando comparados a outros esquemas.

¹<http://macaco.inria.fr/>

3.3 DESCARREGAMENTO DE DADOS CIENTE DE CONTEXTO EM SCN

Em (ELBAMBY et al., 2014) é descrita uma solução que utiliza agrupamento ciente de contexto para aumentar a eficiência em uma rede de *cache* caracterizada como uma rede celular de pequeno porte. Essa estratégia leva em consideração o fato de que nem todos os usuários tem interesses em comum, portanto, devem ser caracterizados e servidos de maneira distinta. Desta forma, é proposta uma solução dividida em duas etapas: a primeira é a etapa de associação, que busca agrupar os usuários de forma a aumentar a taxa de acerto nos *caches*; e a segunda é uma proposta de gerenciamento de *cache* baseada em *regret-minimization learning*.

O cenário consiste em uma rede heterogênea composta por equipamentos de usuários, macro estação base e sub estações base. Nesse cenário, os usuários são agrupados de acordo com sua similaridade em relação à frequência dos conteúdos que solicitam. Para isso, eles utilizam um algoritmo de agrupamento espectral e calculam a similaridade levando em consideração a frequência das requisições através da lei de similaridade do cosseno. Depois de realizar os agrupamentos, é feita uma associação entre os grupos de usuários e as SCBS, independente da relação Sinal/Ruído.

A estratégia de gerenciamento de *cache* proposta é proativa e reativa. Ela é proativa porque nem todo conteúdo que chega é armazenado, a SCBS baseia-se em um vetor contendo um conjunto de ações para cada conteúdo, para determinar se o conteúdo deve ser armazenado ou não. A probabilidade do conteúdo ser armazenado é calculada através do histórico da própria SCBS de acordo com àquele conteúdo. O objetivo da estratégia é manter os conteúdos mais populares armazenados, mas sem tirar completamente a chance de armazenamento dos conteúdos menos populares. Quando é estabelecido que um conteúdo deve ser armazenado mas o limite de capacidade de armazenamento já foi atingido, é necessário determinar qual conteúdo deve ser substituído. A proposta de remoção também é probabilística e busca remover os conteúdos menos populares de acordo com a distribuição de probabilidade Gibbs-Sampling. Os resultados obtidos demonstraram uma diminuição de 27% e 42% no atraso quando utilizado o algoritmo de agrupamento sem e com aprendizado, respectivamente. Além disso, foi obtida uma taxa de descarregamento de dados de 280% e 90% quando comparado à quantidade obtida em um cenário com agrupamento randômico de usuários.

De acordo com (PANTISANO et al., 2015), a eficiência nas estratégias de *cache* utilizadas em redes celulares de pequeno porte está diretamente proporcional com a capacidade das sub estações base de conseguirem prever os interesses dos usuários. Desta forma, o autor propõe um algoritmo de associação entre usuários e *caches* com a intenção de fazer uma melhor alocação da largura de banda, aumentar QoS e reduzir retransmissões. O cálculo de similaridade se baseia em técnicas de Filtro Colaborativo e é realizado levando em consideração a frequência da média de requisições recebidas por cada SCBS, da média de requisições para cada conteúdo e da média de requisições de cada usuário para todos os conteúdos. A intenção é explorar as relações entre as requisições de conteúdo e os usuários, levando em consideração a probabilidade de um conteúdo ser requisitado por outro usuário. Os resultados demonstraram ganhos em relação à alocação de largura de banda e taxa de acerto com melhoras de até 25%.

Ambos trabalhos relacionados atuam em cenários de SCN com o intuito de prover descarregamento de dados e melhorias em relação à utilização da rede através de abordagens ciente de contexto. A relação entre esses trabalhos e a iSim pode ser observada na Tabela 3.1, na qual é feito um comparativo entre o cenário, objetivo, contexto utilizado, função de similaridade, entre outras características. De acordo com esse sumário, é possível observar que o trabalho de (PANTISANO et al., 2015) diferencia-se da iSim em relação ao objetivo principal, que envolve melhoria na alocação de banda, no QoS e na redução de retransmissões. No entanto, o descarregamento de dados também pode ter influência em alguns desses aspectos, principalmente na melhoria do QoS, visto que diminui a sobrecarga das estações principais.

	iSim	(ELBAMBY et al., 2014)	(PANTISANO et al., 2015)
Cenário	Redes heterogêneas	Redes heterogêneas	Redes heterogêneas
Objetivo	Descarregamento de dados	Descarregamento de dados	Melhorar alocação de banda, melhorar QoS e Reduzir retransmissões
Contexto	Interesses dos usuários (frequência e diversidade)	Interesses dos usuários (frequência)	Cache
Função de Similaridade	Similaridade do Cosseno + Índice de Jaccard	Similaridade do Cosseno	Função própria
Algoritmo de Agrupamento	k vizinhos próximos	Agrupamento espectral	Baseado em matching game
Estratégia de Gerenciamento de cache	LFU	Estratégia própria	LFU

Tabela 3.1: Comparação entre a iSim e os trabalhos relacionados ((ELBAMBY et al., 2014) e (PANTISANO et al., 2015))

Além disso, o trabalho de (PANTISANO et al., 2015) é ciente de *cache* e determina uma função própria para calcular a similaridade entre os usuários, essa função baseia-se nas frequências de requisições dos vizinhos em relação ao conteúdo, como foi citado anteriormente. Além disso, o agrupamento é realizado através de um *matching game*. O trabalho de (ELBAMBY et al., 2014) é mais parecido com a iSim porque ambos baseiam-se no mesmo cenário de aplicação e utilizam os interesses dos usuários por conteúdo como o contexto para calcular a afinidade entre os usuários e agrupá-los. No entanto, a principal diferença entre esses trabalhos encontra-se na função de similaridade utilizada, visto que a iSim propõe uma soma ponderada entre dois aspectos – diversidade e frequência – enquanto (ELBAMBY et al., 2014) considera apenas a frequência das requisições. Além

disso, os algoritmos de agrupamento utilizados são diferentes e em (ELBAMBY et al., 2014) é criada uma estratégia própria para realizar o gerenciamento de cache.

3.4 CONSIDERAÇÕES FINAIS

Este capítulo apresentou algumas estratégias de descarregamento de dados utilizando diferentes tecnologias e técnicas que visam minimizar os efeitos do aumento do tráfego nas redes móveis e aumentar a qualidade de serviço e de experiência. As alternativas expostas envolveram utilização de redes Wi-Fi, densificação de redes e utilização de métodos cientes de contexto. Na seção seguinte será apresentada a iSim, estratégia proposta nesta dissertação, que visa potencializar os efeitos do descarregamento considerando a afinidade entre usuários.

ISIM: ESTRATÉGIA DE AGRUPAMENTO POR INTERESSES SIMILARES

Neste capítulo é apresentada a iSim (interests' Similarity), uma estratégia ciente de contexto que visa utilizar a afinidade dos usuários em relação aos interesses por conteúdo para agrupá-los e associá-los a diferentes *caches*. Desta forma, será exposta a arquitetura adequada ao uso da estratégia, a função de similaridade e o algoritmo de similaridade e agrupamento, além das formas de comunicação que podem ser associadas à estratégia para maximizar os ganhos em relação ao descarregamento de dados.

4.1 DESCRIÇÃO GERAL

A inclusão de *cache* tem sido uma das alternativas para lidar com descarregamento de dados em diferentes níveis. Grandes empresas, como o Netflix, utilizam Redes de Distribuição de Conteúdo para colocar servidores que atendam os usuários locais em diferentes pontos geográficos, eliminando a necessidade de trafegar grandes volumes de dados através da Internet.

Com a migração crescente do tráfego das redes fixas para as redes móveis, diferentes estratégias de descarregamento tem sido propostas para lidar com esse problema, como foi visto no Capítulo 3. Uma das estratégias utilizadas é a adição de *cache* nos equipamentos de rede, seguindo um modelo com características centradas na informação, associado à medidas que tornem o armazenamento e descarte de conteúdo mais inteligente para potencializar o uso do *cache*, alcançando melhores taxa de acerto e descarregamento de dados.

A iSim é uma estratégia ciente de contexto que visa utilizar a afinidade dos usuários em relação aos interesses por conteúdo para agrupá-los e associá-los, de forma lógica, a diferentes *caches*. A afinidade corresponde à aproximação entre os usuários do ponto de vista das suas preferências em relação aos conteúdos disponíveis na rede. Desta forma, dois usuários que solicitam conteúdos em comum em uma alta frequência tem maior afinidade do que usuários que requisitam muitos conteúdos distintos.

Se pensarmos, por exemplo, em estudantes dentro de uma universidade que tem interesses por conjuntos de conteúdos relacionados às suas disciplinas, é possível observar que há uma maior convergência entre pessoas que estão em áreas que compartilham conteúdos em comum. Um estudante de Engenharia da Computação e um de Ciência da Computação provavelmente terão mais afinidade do ponto de vista de interesses por conteúdo do que um estudante de Engenharia da Computação e um estudante de Farmácia. Essa lógica pode ser aplicada a diferentes cenários, como estádios de futebol, onde torcedores de times diferentes geralmente tem interesse por conteúdos distintos (cada um tem interesse em conteúdos do seu próprio time), shows, convenções, entre outros.

A iSim propõe-se a calcular a afinidade entre diferentes usuários de acordo com seus interesses por conteúdo para fazer agrupamentos a partir da similaridade entre eles. Esses agrupamentos são associados aos *caches* disponíveis com a intenção de aumentar a taxa de acerto em cada *cache*, visto que aumenta-se a probabilidade de usuários agrupados requisitarem os mesmos conteúdos. Com o aumento da taxa de acerto nos *caches* locais, é possível diminuir a quantidade de requisições à estação principal, culminando no descarregamento de dados. As próximas seções irão apresentar as características da iSim, como cenários de aplicação, estrutura proposta, função de similaridade e algoritmo de agrupamento.

4.2 CENÁRIO DE APLICAÇÃO

Um exemplo de caso real em que a iSim poderia ser utilizada é um cenário universitário. Normalmente, os estudantes que fazem cursos em áreas afins frequentam os mesmos locais, visto que a universidade tende a associar geograficamente os cursos da mesma área. Na Universidade Federal da Bahia, por exemplo, os cursos de Estatística, Matemática e Computação estão associados, principalmente, ao Instituto de Matemática. Assim como na Escola Politécnica concentram-se os cursos de Engenharia. No entanto, há casos em que a associação entre os cursos não é tão evidente, como no Instituto de Humanidades, Artes e Ciências, que mantém os cursos de Bacharelado Interdisciplinar em Artes, Ciências e Tecnologias, Humanidades e Saúde. Em todos esses casos, têm-se estudantes que utilizam os mesmos espaços e que tem diversos interesses acadêmicos, sendo que os interesses dos estudantes por conteúdo podem estar diretamente relacionado com o curso frequentado. Logo, dois estudantes de Engenharia Civil tendem a ter mais interesses em comum do que um estudante de Engenharia Civil e um estudante de Engenharia Elétrica. Esse processo também é evidenciado no caso do Instituto de Humanidades, Artes e Ciências.

Como o conteúdo acadêmico está concentrado na Superintendência de Tecnologia da Informação, pode-se considerar que há uma sobrecarga na estação principal que, nesse caso, recebe e responde todas as requisições. Desta forma, ao colocar *caches* mais próximos dos usuários, é possível melhorar o tempo de resposta e o descarregamento de dados. No entanto, como dentro de um mesmo Instituto existem estudantes com interesses por conteúdos distintos, é importante que haja uma associação entre os usuários (realizada através da iSim) de forma que usuários com interesses em comum se associem ao mesmo *cache*, como pode ser observado na Figura 4.1, que traz a Superintendência de Tecnologia

da Informação como estação principal e os institutos com diferentes *caches* associados aos grupos de usuários afins.

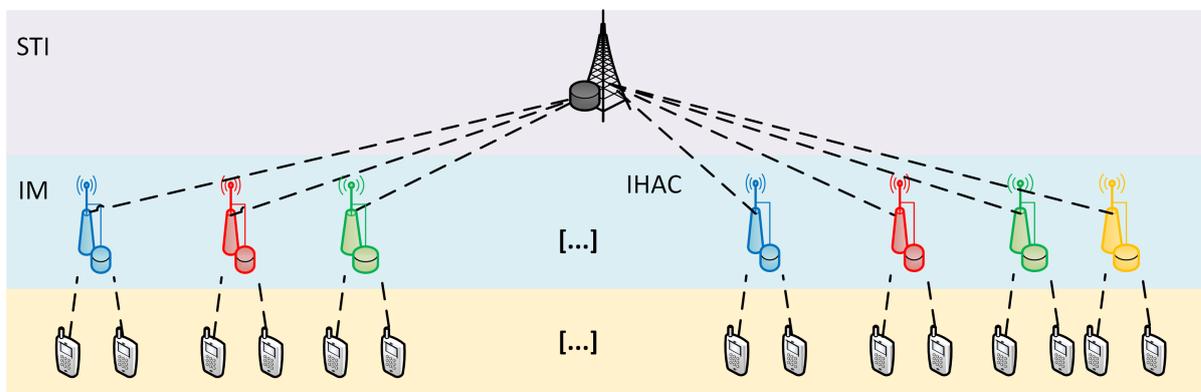


Figura 4.1 Representação de um cenário real em que a iSim pode ser utilizada.

4.2.1 Cenário Técnico

Alguns estudos já demonstram a associação entre Redes Centradas na Informação e Redes Celulares de Pequeno Porte com o objetivo de otimizar recursos de armazenamento e melhorar a utilização da largura de banda. A arquitetura ICN tem como princípio fundamental a utilização de *caches* em equipamentos de rede para prover conteúdos aos usuários de forma mais eficiente. Esse princípio se relaciona diretamente com a necessidade de fazer descarregamento de dados, refletindo na melhor utilização dos recursos de rede.

A iSim adequa-se a cenários onde há uma estação principal e estações secundárias, com capacidade de cobertura e armazenamento inferiores. Esse tipo de cenário pode ser observado como uma representação das Redes Celulares de Pequeno Porte. As SCN são comumente chamadas de redes híbridas, visto que são compostas por diferentes tipos de infraestrutura e tecnologias, e geralmente são resultantes de técnicas de densificação de redes. A densificação consiste na inserção de aparato tecnológico de baixa capacidade, comparado às infraestruturas já implantadas, que servem para aumentar a cobertura do sinal e prover descarregamento de dados (ANPALAGAN, 2015; TELECOMINSIGHTS, 2011).

A densificação de redes pode ocorrer através da densificação do espaço ou da frequência. A densificação do espaço ocorre quando são utilizadas diferentes tecnologias para complementar ou substituir a infraestrutura existente, enquanto a densificação por frequência está relacionada ao aumento no uso de espectros de frequência e diferentes faixas (TES-SCOTECHNOLOGIES, 2014).

A densificação de redes através do espaço pode ser desenvolvida utilizando diferentes tecnologias, a depender do objetivo da implantação. É possível observar na Tabela 4.1 que existem diversos tipos de nós que se diferenciam de acordo com sua capacidade de cobertura e poder de transmissão (QUEK et al., 2013). Nesta dissertação, chamamos

de redes celulares de pequeno porte as redes formadas por macro estação (do inglês, *macrocell*) e pequenas células.

Tipos de nós	Poder de transmissão	Cobertura
Macro estação	46 dBm	Alguns km
Pico-célula	23-30 dBm	<300 m
Femto-célula	<23 dBm	<50 m
Repetidor	30 dBm	300 m
Unidade de Rádio Remota	46 dBm	Alguns km

Tabela 4.1: Comparação entre macro-células e diferentes tipos de células pequenas

As tecnologias citadas geralmente são implantadas em pontos estratégicos, onde há mais volume de tráfego e mais conexões, como estações de metrô, universidades, estádios, entre outros. As macro-células são caracterizadas por prover uma taxa de dados mínima sob restrições de atraso e indisponibilidade, o alcance dela é de alguns quilômetros, com poder de transmissão de 46 dBm e capacidade de conexão com milhares de clientes. Já as pico-células e femto-células, que são comumente utilizadas para fazer densificação, tem poder e alcance inferiores e podem servir algumas dezenas de usuários. Elas geralmente são dispostas de maneira centralizada e são utilizadas para aumentar a cobertura tanto de ambientes internos quanto externos.

A densificação pode ocorrer, basicamente, de duas maneiras: distribuída ou como um complemento à estrutura existente. Na Figura 4.2, é possível observar que a macro estação pode ser substituída por pequenas estações que são capazes de prover a mesma cobertura. Enquanto, a outra alternativa é a adição de células de capacidade inferior ao perímetro de cobertura da estação principal com o objetivo de aumentar a capacidade e atingir maior taxa de dados.

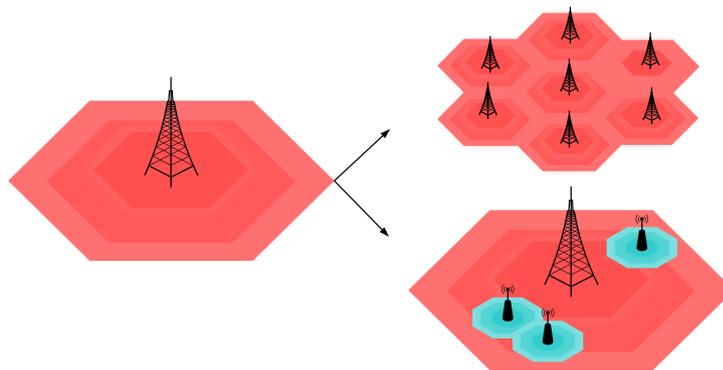


Figura 4.2 Representação dos tipos de densificação: distribuída ou como complemento à estrutura principal.

A iSim foi elaborada para lidar com ambientes onde é realizada a densificação como complemento à uma estrutura principal. Para isso, foram definidos alguns papéis que representam as estruturas de densificação associadas à arquitetura centrada no conteúdo.

4.3 ESTRUTURA DA ICN

4.3.1 Equipamentos de Rede

A iSim se baseia no mecanismo de adição de células como complemento à estrutura principal e categoriza os elementos que compõem a rede em três níveis de acordo com suas características:

- a MBS é a estação principal, tem maior cobertura e é capaz de prover todos os conteúdos que estão disponíveis na rede;
- a SCBS tem capacidade de armazenamento limitada e é utilizada para armazenar réplicas de conteúdo mais próximas dos usuários;
- os dispositivos móveis (DM) são os clientes/assinantes de conteúdo que fazem requisições através do envio de pacotes de interesse.

Um cenário genérico que representa essas três estruturas pode ser observado na Figura 4.3, onde existe uma MBS, algumas SCBS distribuídas com o intuito de aumentar a cobertura e prover descarregamento de dados através da utilização de *cache*, e os dispositivos móveis que representam os usuários.

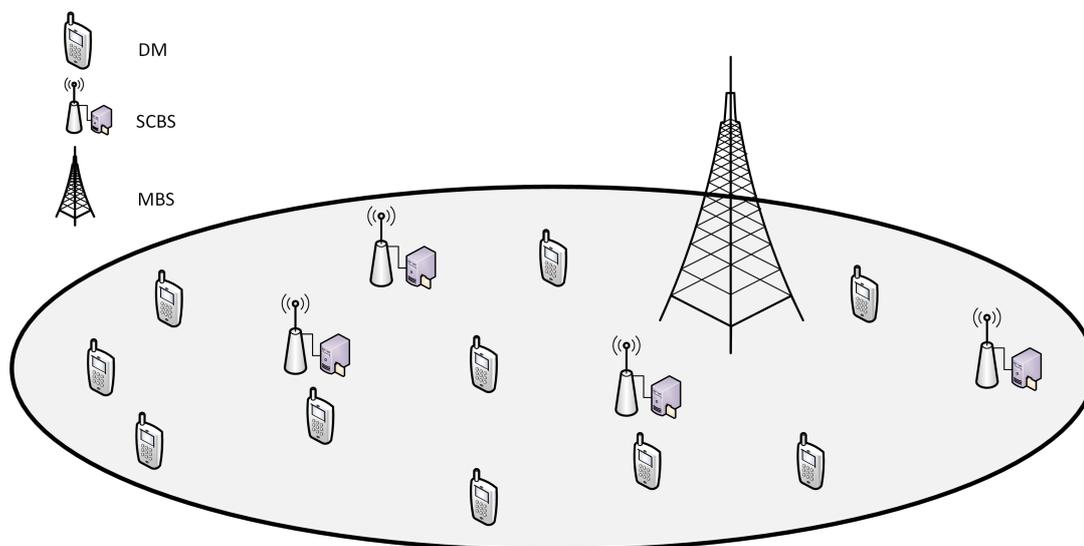


Figura 4.3 Estrutura genérica de uma SCN na qual a iSim foi baseada.

Como a MBS e a SCBS são equipamentos de rede e a iSim baseia-se em uma arquitetura centrada na informação, ambas foram adaptadas para lidar com conteúdo nomeado. Desta forma, para utilizar a iSim, é preciso definir o *cache*, a PIT e a FIB que compõem nós ICN.

O *cache* definido nos equipamentos de rede é uma estrutura semelhante ao *cache* descrito na Seção 2.2.5.1 e pode ser observado na Figura 4.4. Ele deve ser composto pelos dois atributos obrigatórios, que são o nome do conteúdo e o conteúdo propriamente dito.

Além disso, para lidar com políticas de descarte diferentes da FIFO, é necessário adicionar campos que mantenham as informações que serão utilizadas para realizar a substituição de conteúdos. Na Figura 4.4¹, por exemplo, o atributo adicionado é a frequência de envio do conteúdo, para que seja possível utilizar a política de descarte LFU. Caso seja escolhida a política LRU, é necessário adicionar o campo de último envio. Logo, os atributos que irão compor o *cache* dependem do tipo de descarte que será utilizado.

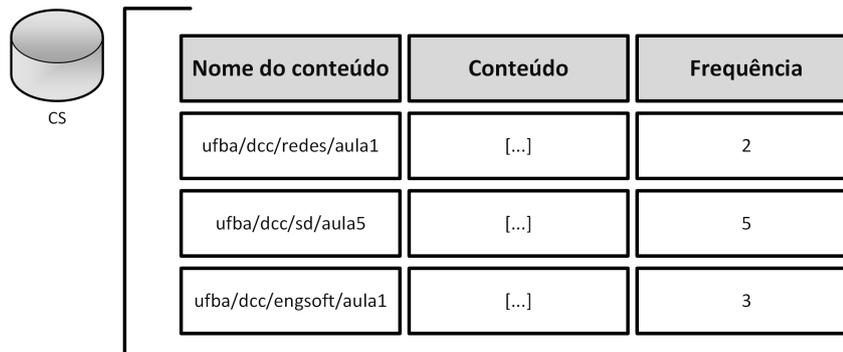


Figura 4.4 Estrutura do *cache* em uma rede na qual a iSim é adotada.

Além das políticas de descarte LFU e LRU que são comumente utilizadas, o padrão ICN, do ponto de vista de armazenamento, é utilizar uma política do tipo LCE (do inglês, *Leave Copy Everywhere*), que é caracterizada por manter uma cópia dos conteúdos em todos os nós que estão no caminho entre um assinante e um publicador. Desta forma, a iSim parte do pressuposto de que todo conteúdo que chega deve ser armazenado em *cache*. Assim, a estratégia não faz uso de nenhuma política de descarte proativa, visto que armazena tudo que chega ao nó. Logo, um conteúdo só é descartado quando outro conteúdo precisa ser armazenado e não há mais espaço disponível.

Tanto a MBS quanto a SCBS tem *cache* em sua composição, a diferença, no entanto, é a capacidade de armazenamento de ambas. Parte-se do pressuposto de que uma MBS tem *cache* de tamanho ilimitado, isso significa que a MBS é capaz de armazenar todos os conteúdos disponíveis na rede e que não é necessário adicionar nenhum atributo no *cache* para utilização de políticas de descarte. Já a SCBS tem *cache* de tamanho limitado e mantém cópia de apenas uma porcentagem da quantidade de conteúdo total disponível na rede.

Como a MBS tem todos os conteúdos que serão solicitados, não há necessidade de criar uma tabela para armazenar interesses pendentes, visto que todos os interesses podem ser resolvidos por ela. Desta forma, a PIT é uma estrutura utilizada apenas na SCBS.

A PIT foi definida de acordo com as premissas de ICN e é composta pelo nome do conteúdo e as interfaces que enviaram pacotes de requisição para aqueles conteúdos, conforme a Figura 4.5. Como foi citado anteriormente na Seção 2.2.5, a PIT deve ser capaz de permitir a agregação de requisições. Desta forma, quando um dispositivo faz requisição para um conteúdo e já existe uma entrada na PIT, a requisição é descartada

¹A imagem é apenas uma representação da implementação, desta forma, não são demonstrados detalhes em relação ao tamanho do pacote ou quantidade de *bits*

e o dispositivo é adicionado na lista de interfaces solicitantes, evitando desperdício de recurso e diminuindo a sobrecarga da rede.



Nome do conteúdo	Interfaces Solicitantes
ufba/dcc/redes/aula1	DM01, DM04,DM07
ufba/dcc/sd/aula5	DM09
ufba/dcc/engsoft/aula1	DM03,DM05

Figura 4.5 Estrutura da PIT em uma rede na qual a iSim é adotada.

A FIB é a estrutura responsável por manter as informações para realização do encaminhamento de pacotes de interesse. Essa estrutura faz o mapeamento entre um prefixo e uma interface de saída. Como a MBS não irá encaminhar pacotes de interesse, também não há a necessidade da adição de uma FIB. Desta forma, a estrutura que precisaria manter uma FIB é a SCBS. No entanto, a iSim atua em níveis superiores (aplicação) e como a estratégia não influencia ou relaciona-se com questões de roteamento, os pacotes foram trafegados sob uma rede IP. Desta forma, na Figura 4.5, a lista de interfaces solicitantes é convertida em uma lista de endereços IP utilizados no roteamento dos pacotes de interesse.

Além da SCBS e da MBS, o terceiro equipamento utilizado é o dispositivo móvel. Cada dispositivo móvel representa um usuário que faz solicitações para conteúdos. Desta forma, nenhuma das estruturas definidas acima é adotada. No entanto, para utilização da iSim, foi adicionada uma estrutura chamada tabela de requisições (do inglês, *Requisition Table* - RT), que pode ser observada na Figura 4.6. Ela é composta pelo nome do conteúdo e a frequência de solicitações, que determina a quantidade de vezes que um DM requisitou aquele conteúdo. Essa tabela é utilizada como insumo para determinar a afinidade entre os usuários e seu uso será explicado na seção 4.5.



Nome do conteúdo	Frequência de solicitação
ufba/dcc/redes/aula1	5
ufba/dcc/sd/aula5	3
ufba/dcc/engsoft/aula1	1

Figura 4.6 Estrutura da RT em uma rede na qual a iSim é adotada.

A comunicação entre DM, SCBS e MBS é realizada através do envio de pacotes. Os tipos de pacotes definidos para utilização da iSim são expostos na próxima seção.

4.3.2 Tipos de Pacote

Toda comunicação entre SCBS, MBS e DM é realizada através do envio de pacotes. Como foi dito no Capítulo 2, uma ICN é caracterizada por determinar dois tipos de pacotes: pacotes de interesse e pacotes de dados. Para utilização da iSim, os pacotes de interesse e de dados foram definidos conforme a Figura 4.7². O pacote de interesse mantém o nome do conteúdo, o tipo de pacote, a quantidade de saltos, o endereço do nó assinante, o tipo de nó que encaminhou o interesse (um DM ou uma SCBS) e um *payload* vazio. Já o pacote de dados tem apenas o nome do conteúdo, o tipo de pacote e o *payload* contendo o conteúdo.

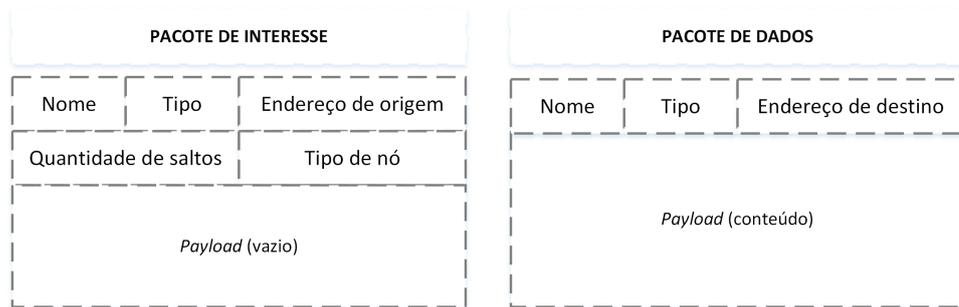


Figura 4.7 Estrutura dos pacotes de interesse e de dados definidos para dar suporte à iSim.

Além desses dois tipos de pacotes, foi preciso definir os pacotes de controle. Um pacote de controle pode ter sua estrutura associada à de um pacote de interesse (com *payload* vazio) ou de dados, a depender da sua finalidade. Esse tipo de pacote é enviado apenas pelos DMs e pela MBS, com propósitos diferentes. De acordo com a Figura 4.8, é possível observar, do lado esquerdo, o pacote de controle que é enviado pelo DM. Ele é composto pelo nome do conteúdo, que nesse caso representa o nome do próprio DM (por exemplo, *wirelessHost1*), pelo tipo de pacote e pelo *payload* contendo a tabela de requisições do DM. Quando o pacote de controle é enviado pela MBS (lado direito da figura), os campos de cabeçalho são apenas o nome do conteúdo, representando o nome da SCBS que o DM deve se associar (nesse caso, *wiredHost1*), o tipo de pacote e um *payload* vazio.

4.3.3 Encaminhamento de Pacotes

Cada dispositivo descrito na Seção 4.3.1 é responsável por manipular os tipos de pacotes de acordo com a sua funcionalidade. Como trata-se de uma ICN, a comunicação entre os dispositivos é realizada através do envio de pacotes de interesse, dados ou controle. A relação entre os dispositivos, seus papéis na rede e os tipos de pacote que eles recebem e enviam pode ser observada na Tabela 4.2.

Os DMs representam usuários finais interessados em conteúdos disponibilizados pela rede. Desta forma, a função dos DMs é a de fazer requisições de conteúdo através do

²A imagem é apenas uma representação da implementação e por isso não tem detalhes em relação ao tamanho do pacote ou quantidade de *bits*.

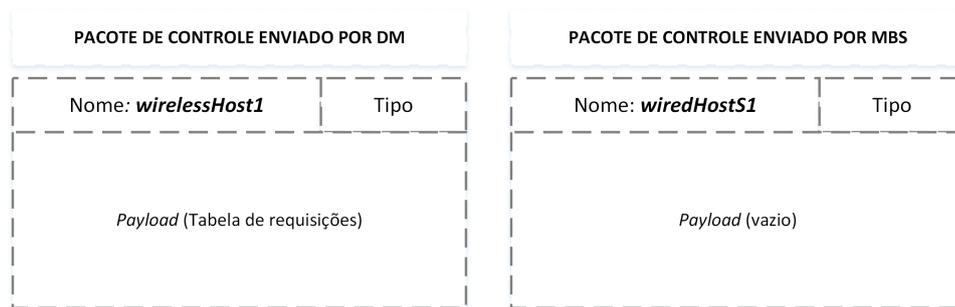


Figura 4.8 Estrutura dos pacotes de controle definidos para dar suporte à iSim.

	DM	MBS	SCBS
Função	Requisitar Conteúdos	Prover Conteúdos Calcular similaridade Definir agrupamentos	Prover Conteúdos
Pacotes recebidos	Pacotes de dados Pacotes de controle	Pacotes de interesse Pacotes de controle	Pacotes de interesse Pacote de dados
Pacotes enviados	Pacotes de interesse Pacotes de controle	Pacotes de dados Pacotes de controle	Pacotes de interesse Pacote de dados

Tabela 4.2: Relação entre os dispositivos e suas funcionalidades e pacotes trafegados.

envio de pacotes de interesse. Como foi observado anteriormente, o DM mantém uma RT que deve ser atualizada toda vez que há o envio de um pacote de interesse. A RT é enviada como *payload* de um pacote de controle e serve de insumo para o cálculo de similaridade entre os nós. Além disso, os DMs recebem pacotes de dados, como resposta aos pacotes de interesse, e pacotes de controle, que indicam o *cache* que eles devem se associar.

Já a MBS e a SCBS tem como principal função prover conteúdos aos usuários. Desta forma, a funcionalidade básica de ambas é a de encaminhar pacotes de dados. No entanto, além dessa função, a MBS é responsável por calcular a similaridade e definir os agrupamentos e enviar pacotes de controle que indiquem a qual *cache* o DM deve se conectar. Determinada as funcionalidades e os pacotes enviados, observa-se que a MBS pode receber pacotes de interesse por conteúdo e pacotes de controle dos DMs. Enquanto a SCBS recebe pacotes de interesse do usuário e pacotes de dados da MBS e caso a SCBS não tenha o conteúdo armazenado em *cache*, ela deverá enviar pacote de interesse e receberá como resposta um pacote de dados.

Cada dispositivo irá se comportar de maneira distinta de acordo com a funcionalidade que executa e com o tipo de pacote que recebe. Os DMs podem enviar apenas dois tipos de pacotes: pacotes de interesse e pacotes de controle. O envio dos pacotes de interesse pelo DM é o único envio que ocorre sem ser decorrente do recebimento de outro pacote. Já o envio de pacotes de controle deve ocorrer em períodos predefinidos em que deverá ser realizado o cálculo de similaridade e agrupamento. Do ponto de vista de recebimento do

pacote, o DM pode receber pacotes de dados como resposta a seus interesses ou pacotes de controle como resposta a seus pacotes de controle. Ao receber um pacote de dados, a única ação que o DM faz é descartar o pacote, visto que ele não precisa armazená-lo ou encaminhá-lo. Ao receber um pacote de controle com o nome da SCBS a qual deve se associar, o DM deve conectar-se a ela e deixar de fazer requisições à MBS.

O processo de recebimento de pacotes de interesse na MBS está descrito na Figura 4.9: ao receber pacotes de interesse (1), verifica-se no *cache* qual o conteúdo correspondente de acordo com o nome do NDO recebido (2 e 3), uma vez encontrado o conteúdo, é criado um pacote de dados (4) que deve ser encaminhado ao usuário (5). Como uma das premissas utilizadas é de que a MBS mantém em *cache* todos os conteúdos disponíveis na rede, não foi considerada a possibilidade de chegar um pacote de interesse para um conteúdo inexistente, logo, não há necessidade de criação de uma PIT. O outro tipo de pacote recebido pela MBS é o pacote de controle. Quando a MBS recebe um pacote de controle ela adiciona a RT do usuário (que chega no *payload* do pacote de controle) à uma matriz que representa conteúdos x frequência de todos os usuários, e apenas quando todos os pacotes de controle chegam, o processo de cálculo de similaridade e agrupamento começam. O processamento dos pacotes de controle na MBS serão melhor descritos na Seção 4.5, onde será explicado o algoritmo de similaridade e agrupamento.

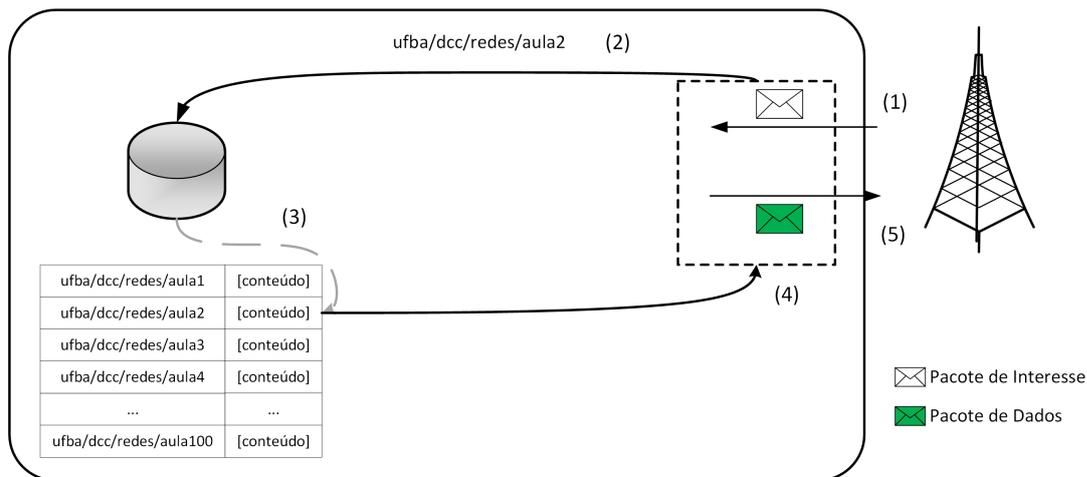


Figura 4.9 Processamento de pacotes de interesse na MBS.

A SCBS pode receber e solicitar pacotes de interesse e dados. Ao receber um pacote de interesse, verifica-se no *cache* se existe um pacote de dados relacionado àquele conteúdo, se existir, encaminha-se o pacote de dados para o nó solicitante (semelhante ao esquema descrito para a MBS). Caso contrário, os procedimentos são realizados conforme a Figura 4.10(a): ao receber o pacote de interesse (1), verifica-se no *cache* se existe um NDO armazenado (2 e 3), caso não exista, é adicionada uma entrada na PIT com o nome do conteúdo e a interface solicitante (4). Em seguida, a SCBS encaminha o pacote de interesse (5 e 6) para a MBS, solicitando o conteúdo. Na Figura 4.10(b), a SCBS recebe o pacote de dados enviado como resposta pela MBS (1) e ele é adicionado no *cache* (2 e 3) no lugar de um conteúdo que estava armazenado, visto que não tinha mais espaço disponível.

Depois de adicionar o conteúdo ao *cache*, verifica-se se existe alguma entrada para aquele conteúdo na PIT (4), se houver, é preciso encaminhar uma réplica do conteúdo para as interfaces solicitantes (5 e 6) e apagar a entrada da PIT (7).

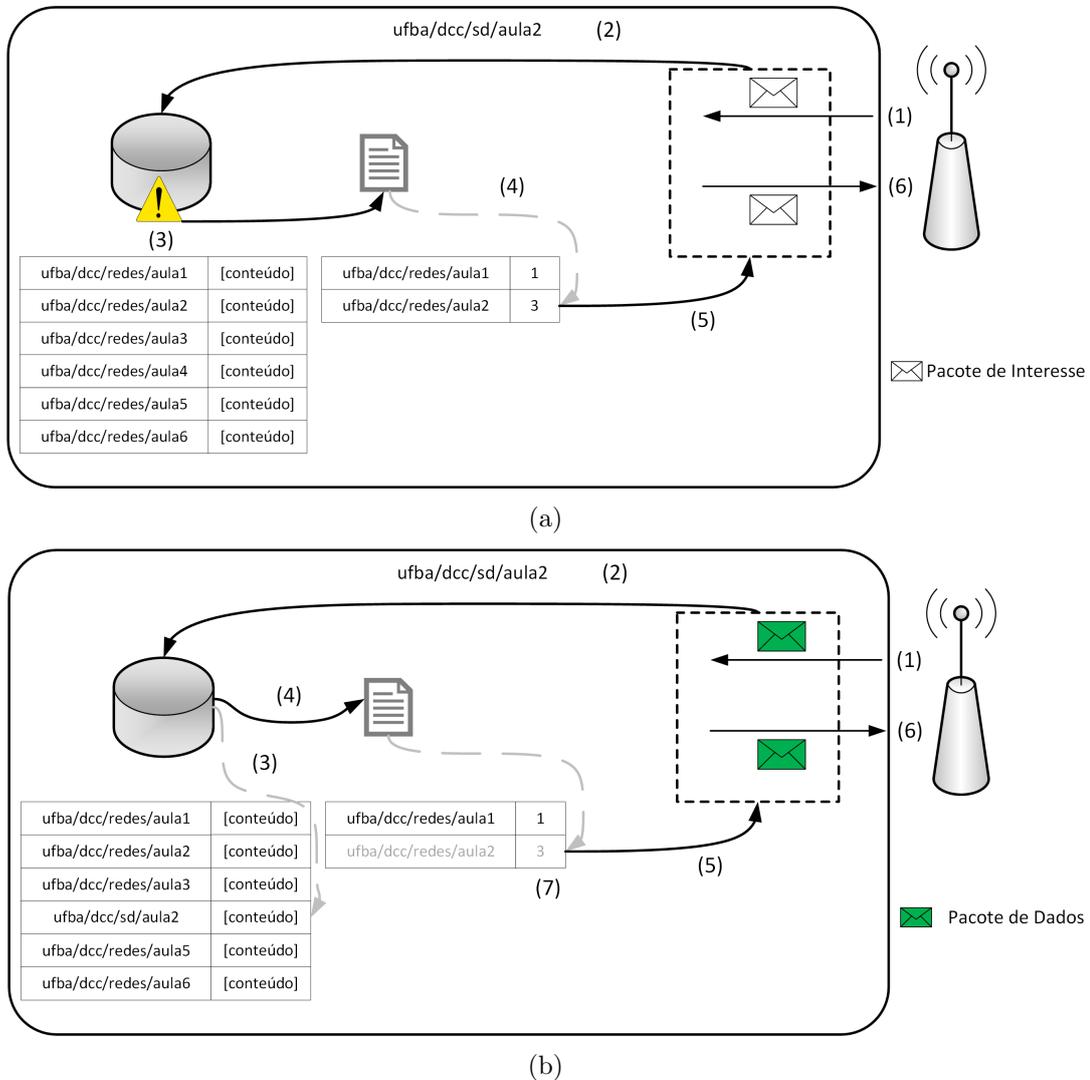


Figura 4.10 Processamento de pacotes de interesse e de dados na SCBS

De acordo com as diferentes funcionalidades de cada estrutura, é possível observar que o envio de pacotes de dados pode ocorrer de três maneiras distintas:

- a MBS envia o conteúdo diretamente para o DM;
- a SCBS tem o conteúdo armazenado em *cache* e envia para o DM;
- a SCBS não tem o conteúdo armazenado e a MBS envia o conteúdo para ela.

Cada uma dessas situações ocorre de acordo com a configuração da rede em diferentes momentos. O processo de envio de pacotes ao decorrer do tempo pode ser observado no Diagrama de Sequência exposto na Figura 4.11. Inicialmente, os DMs se comunicam diretamente com a MBS enviando pacotes de interesse e recebendo pacotes de dados. No entanto, em determinado momento t' , os DMs devem enviar suas RTs através de pacotes de controle para a MBS, que aplica as funções de similaridade e agrupamento e envia um pacote de controle com a SCBS que o DM deve se conectar. Após a formação dos agrupamentos, os usuários são servidos pelas SCBS. No primeiro momento, a SCBS fará muitas requisições à MBS porque sua *cache* é inicializada vazia, mas, após um período de tempo, a SCBS será capaz de servir os DMs com os conteúdos armazenados em *cache* e fará requisições à MBS quando não tiver o conteúdo.

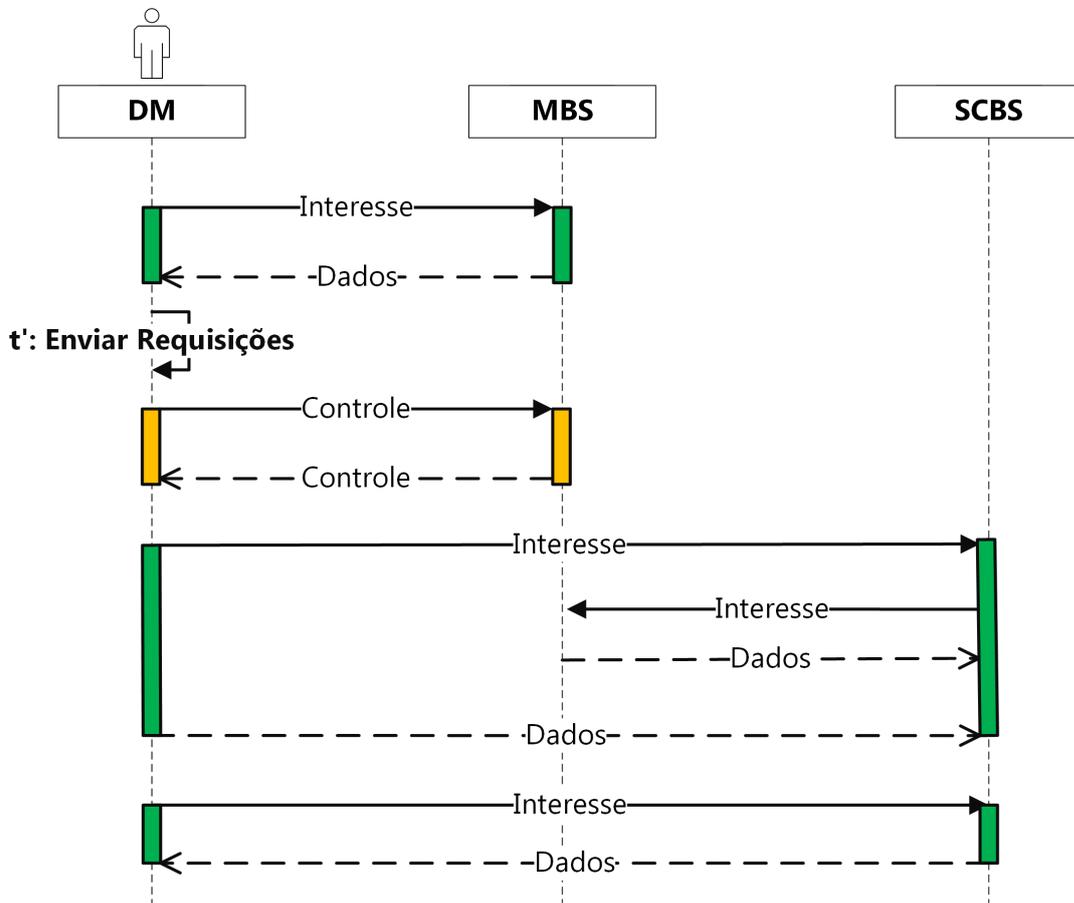


Figura 4.11 Diagrama de Sequência que ilustra o encaminhamento de pacotes em diferentes momentos.

Uma vez descrita a arquitetura da ICN e como é processada a troca de pacotes entre os diferentes equipamentos de rede, é possível definir como é realizado o cálculo de similaridade entre os usuários e a formação dos agrupamentos.

4.4 FUNÇÃO DE SIMILARIDADE

As medidas de similaridade ou dissimilaridade são funções definidas para calcular quão semelhantes ou distintos são dois indivíduos, elas variam de acordo com o tipo de atributo que consideram e na maneira de calcular. Existem funções que consideram tanto a presença quanto a ausência de atributos conjuntos e funções que são mais apropriadas para determinação de frequência. Dentre as funções de similaridade mais utilizadas estão a distância de Manhattan, a Similaridade do Cosseno, o Índice de Sorensen ou Dice e o Índice de Jaccard.

A distância de *Manhattan* ou *City Block* (KRAUSS, 1987) é medida através da Equação 4.1. Essa equação quantifica a distância entre dois pontos, sendo que quanto mais próximo de zero, melhor. Se tivermos, portanto, dois indivíduos que fizeram requisições em comum para os conteúdos x, y, z de acordo com a mesma frequência $a = b = (1, 1, 1)$, o resultado do somatório será 0, logo, os dois pontos são exatamente iguais. Se a frequência for $a = (1, 1, 1)$ e $c = (2, 2, 1)$ temos um somatório equivalente à 2. Logo, no segundo caso, os indivíduos são menos similares. Desta forma, percebe-se que a distância de *Manhattan* é uma medida de dissimilaridade e que o resultado não está dentro de um intervalo definido. De acordo com (CLIFFORD; STEPHENSON, 1975), os índices cujo valor resultante estão dentro de um intervalo $[0, 1]$ são mais adequados pois são menos sensíveis a mudança do que os coeficientes que tendem ao infinito.

$$Manhattan = \sum_{j=1}^k |a_j - b_j| \quad (4.1)$$

A Similaridade do Cosseno representada pela Equação 4.2, mede o ângulo entre dois vetores levando em consideração apenas o conjunto intersecção dos atributos. Essa medida é adequada em cenários em que a quantidade de atributos comuns e distintos não são tão discrepantes, visto que a medida leva em consideração apenas atributos comuns. Assim, se tivermos dois indivíduos a, b que solicitam um único conteúdo em comum uma única vez, a similaridade entre esses dois indivíduos será igual a 1, ainda que eles solicitem n outros conteúdos distintos.

$$Cosseno = \frac{F_i \cdot F_j}{\|F_i\| \|F_j\|} \quad (4.2)$$

O Índice de Sorensen ou Dice (DICE, 1945) foi criado para avaliar a relação entre dois indivíduos dentro de um ecossistema e pode ser observado na Equação 4.3. O h representa os atributos em comum entre duas espécies e a e b representam o conjunto total de atributos em a e em b , respectivamente.

$$Dice = \frac{2h}{a + b} \quad (4.3)$$

O Índice de Jaccard (JACCARD, 1912) também considera a relação entre atributos comuns e atributos distintos e é composto pela relação entre uma intersecção com uma união, como pode ser observado na Equação 4.4.

$$Jaccard = \frac{|C_i \cap C_j|}{|C_i \cup C_j|} \quad (4.4)$$

Essas medidas de similaridade são utilizadas em diferentes trabalhos com o intuito de calcular a afinidade entre textos, *strings*, usuários, entre outros. O método escolhido depende das características do ambiente a ser avaliado e do objetivo da similaridade. No cenário de redes, é comum calcular a similaridade de usuários para fazer recomendações e para construção de agrupamentos em busca de melhorias do ponto de vista da rede e do usuário (ELBAMBY et al., 2014; PANTISANO et al., 2014).

A iSim é uma estratégia que visa calcular a similaridade entre usuários de acordo com os grupos de conteúdo que os usuários tem interesse. Para determinar o grau de similaridade, foi criada uma função que tem o objetivo de quantificar a semelhança entre diferentes usuários de acordo com seus interesses.

Para calcular a similaridade entre os usuários, foram utilizados dois métodos de cálculo de similaridade: o Índice de Jaccard e a similaridade do cosseno. Portanto, na iSim, a similaridade entre os usuários é medida através da Equação 4.5. A primeira parte representa o Índice de Jaccard (JACCARD, 1912), ela foi utilizada para medir a similaridade de acordo com os conteúdos em comum que dois nós (i, j) tem interesse. Já a segunda parte é calculada utilizando a lei de similaridade do cosseno em relação à frequência de requisições de conteúdos em comum.

Desta forma, C_i e C_j representam os conjuntos de conteúdos que foram solicitados por i e por j , respectivamente. Sendo que, durante o cálculo da similaridade é utilizada a quantidade de elementos nos conjuntos interseção e união, não os conjuntos propriamente dito; F_i e F_j representam a frequência, em i e em j , em que os conteúdos comuns foram requisitados; e β define um peso entre as partes da equação e permite que seja avaliado o impacto delas no cálculo de similaridade. A utilização do parâmetro β permite flexibilidade e adaptabilidade no cálculo da similaridade, visto que permite o aumento ou diminuição do impacto das partes da equação de acordo com as características do cenário.

$$S_{i,j} = \beta \left(\frac{|C_i \cap C_j|}{|C_i \cup C_j|} \right) + (1 - \beta) \left(\frac{F_i \cdot F_j}{\|F_i\| \|F_j\|} \right) \quad (4.5)$$

Vale ressaltar, que não estamos avaliando a similaridade entre usuário e *cache*, nem entre dois conteúdos, mas a similaridade entre usuários de acordo com seus interesses por conteúdo. O algoritmo utilizado para realizar esse cálculo de similaridade e determinar o agrupamento é descrito na próxima seção.

4.5 ALGORITMO DE SIMILARIDADE E AGRUPAMENTO

O algoritmo de similaridade e agrupamento tem a função de estabelecer as associações entre nós que tenham interesses similares. Desta forma, a primeira coisa a ser feita é calcular a similaridade conforme a Equação 4.5. No entanto, para que a equação seja aplicada, é preciso que a MBS tenha recebido os pacotes de controle contendo as RTs de cada DM. Para isso, é estabelecido um instante de tempo t a partir do qual todos os DMs devem enviar um pacote de controle contendo a sua tabela de requisições. Cada tabela

de requisição recebida pela MBS é adicionada a uma matriz $N \times C$, exposta na Figura 4.12, onde N é a quantidade de nós da rede e C a quantidade de conteúdos disponíveis. Ao receber uma tabela de requisição, a MBS verifica o ID do nó que enviou a requisição, adiciona na linha da matriz referente àquele ID e preenche a frequência da solicitação dos conteúdos de acordo com a posição do conteúdo na coluna da matriz. Por exemplo, o nó com $ID = 0$ requisitou o *cont1* duas vezes, não requisitou o *cont2*, requisitou o *cont3* cinco vezes, e assim por diante.

		Frequência de requisições para os conteúdos (C)				
		cont1	cont2	cont3	cont4	cont5
ID nó (N)	0	2	0	5	0	1
	1	1	5	0	0	2
	2	3	2	0	5	2
	3	0	1	1	3	0

Figura 4.12 Demonstração da formação da matriz $N \times C$.

Depois de receber as RTs dos DMs, inicia-se a execução do Algoritmo 1. As linhas 2-9 representam o cálculo da similaridade entre os usuários a partir das tabelas de requisições enviadas: para cada DM faz-se uma comparação entre as suas requisições e as dos outros nós da rede de acordo com os conteúdos que foram requisitados e a frequência das requisições. A similaridade (linha 5) é calculada de acordo com a Equação 4.5 e os resultados dos cálculos de similaridade geram uma matriz $N \times N$, conforme Figura 4.13, na qual o índice de cada linha/coluna da matriz corresponde ao ID de um dos nós que compõe a rede e a linha representa a similaridade entre um nó e os demais. A linha 0, por exemplo, equivale à similaridade entre o nó 0 e todos os outros nós da rede $\{0, 1, 2, \dots, N\}$. Desta forma, a matriz é caracterizada como uma matriz simétrica, na qual $n_{ij} = n_{ji}$ e cuja diagonal principal é nula, pois representa a similaridade entre um nó e ele mesmo.

Após a formação da matriz contendo a similaridade entre todos os nós da rede, inicia-se a formação dos agrupamentos. A quantidade de grupos existentes será igual à quantidade de SCBS disponíveis. Logo, para cada SCBS (linha 11) é feita uma ordenação decrescente entre os elementos de cada linha da matriz (linha 12), para que os nós sejam ordenados de acordo com a similaridade. Cada elemento da matriz é um tipo abstrato de dados formado pelo ID do nó e pela similaridade, isso garante que, mesmo após a ordenação, sejam mantidos os IDs dos nós que mudaram de posição, como observado na Figura 4.14.

Na realização dos agrupamentos é utilizado o algoritmo k Nearest Neighbor (PETERSON, 2009). Uma vez que a similaridade entre os nós já foi estabelecida e ordenada, é calculada a média entre um nó seus vizinhos (linha 13), obtendo assim a média de similaridade de cada nó da rede. O nó com maior média (linha 14) é escolhido como líder do grupo e inicia-se a formação do agrupamento com os seus vizinhos mais similares (linha 15). Sendo que a determinação dos vizinhos pode ser feita de duas maneiras:

		ID nó (N)			
		0	1	2	3
ID nó (N)	0	0	0,5	0,4	0,2
	1	0,5	0	0,75	0,2
	2	0,4	0,75	0	0,4
	3	0,2	0,2	0,4	0

Figura 4.13 Demonstração da formação da matriz $N \times N$.

Algoritmo 1: Algoritmo de Agrupamento

Entrada: Matriz $N \times C$

```

1 início
2   para cada  $i \in N$  faça
3     para cada  $(j > i) \in N$  faça
4       para cada  $w \in C$  faça
5         | Calcula os valores de  $C_i \cap C_j$ ,  $C_i \cup C_j$ ,  $F_i \cdot F_j$  e  $\|F_i\| \|F_j\|$ ;
6       fim
7       Calcula a similaridade entre  $i$  e  $j$ ;
8       Constrói uma matriz  $N \times N$  com a similaridade dos nós;
9     fim
10  fim
11  para cada  $z \in s$  faça
12    Aplica um método de ordenação em cada linha da matriz;
13    Calcula a média aritmética entre os  $k$  vizinhos mais similares de cada nó;
14    Identifica o nó com maior média de similaridade;
15    Forma um agrupamento com o nó com maior média e seus  $k$  vizinhos mais similares;
16    Envia mensagem de controle para os nós do agrupamento formado identificando o servidor de distribuição que eles devem se associar;
17    Zerar os valores de similaridade dos nós que já foram agrupados;
18  fim
19 fim

```

Saída: s agrupamentos, sendo s a quantidade de SCBS

- Agrupamentos de tamanho fixo: a quantidade de nós em cada grupo deve ser a mesma (ou muito próxima), e essa quantidade é medida através da relação entre a quantidade de nós alcançados e a quantidade de SCBS implantadas. Desta forma,

ID nó (N)	ID do nó comparado - similaridade				
	0	1 - 0,5	2 - 0,4	3 - 0,2	0 - 0
1	2 - 0,75	0 - 0,5	3 - 0,2	1 - 0	
2	1 - 0,75	0 - 0,4	3 - 0,4	2 - 0	
3	2 - 0,4	0 - 0,2	1 - 0,2	3 - 0	

Figura 4.14 Demonstração da ordenação da matriz NxN e da utilização de um tipo abstrato de dados para manutenção dos identificadores.

em um cenário com 20 nós e quatro SCBS, cada grupo deveria ser formado por cinco DMs ($20/4 = 5$).

- Agrupamentos por nível de similaridade: a quantidade de nós em cada grupo pode ser diferente e essa formação depende de um limiar de similaridade. Para cada nó líder, são associados todos os nós cuja similaridade com o líder é maior ou igual ao limiar predeterminado.

Independente da forma de agrupamento, a formação se dá pelo nó que tem a maior média e os seus k primeiros vizinhos, que representam os vizinhos com maior similaridade. Ao identificar os membros do agrupamento, a MBS envia um pacote de controle para cada usuário identificando a SCBS que ele deve se associar (linha 16). A partir desse momento, os DM fazem requisições apenas para as SCBS, deixando de requisitar à MBS.

No Algoritmo 1 a complexidade do cálculo de similaridade é de $O(CN^2)$ enquanto a realização do agrupamento tem complexidade $O(N^2 \log(N))$ (quando utilizado o algoritmo de QuickSort para ordenação). Sendo C a quantidade de conteúdos e N a quantidade de usuários.

4.6 FORMAS DE COMUNICAÇÃO

Além da arquitetura de ICN e do algoritmo de similaridade e agrupamento, a iSim pode utilizar-se de duas formas distintas de comunicação: comunicação direta e comunicação indireta. O funcionamento desses mecanismos de comunicação podem ser observados nas Figuras 4.15 e 4.16. Partindo do pressuposto de que um DM envia um conteúdo nomeado para a SCBS1 e o único nó da rede que possui o pacote de dados associado a esse conteúdo é a MBS, a comunicação poderá funcionar da seguinte forma:

- Comunicação Direta, conforme Figura 4.15: ao receber um pacote de interesse de um dispositivo móvel (1), SCBS1 verifica a sua *cache* (2) e percebe que não existe

um conteúdo relacionado ao interesse. Desta forma, é adicionada uma entrada na PIT (3) e o pacote de interesse é encaminhado diretamente a MBS (4). A MBS acessa o *cache* para obter o conteúdo solicitado (5,6) e envia o pacote de dados para SCBS1 (7), SCBS1 armazena o pacote de dados no seu *cache* (8), apaga a entrada da PIT (9) e encaminha o pacote de dados para o dispositivo móvel solicitante (10);

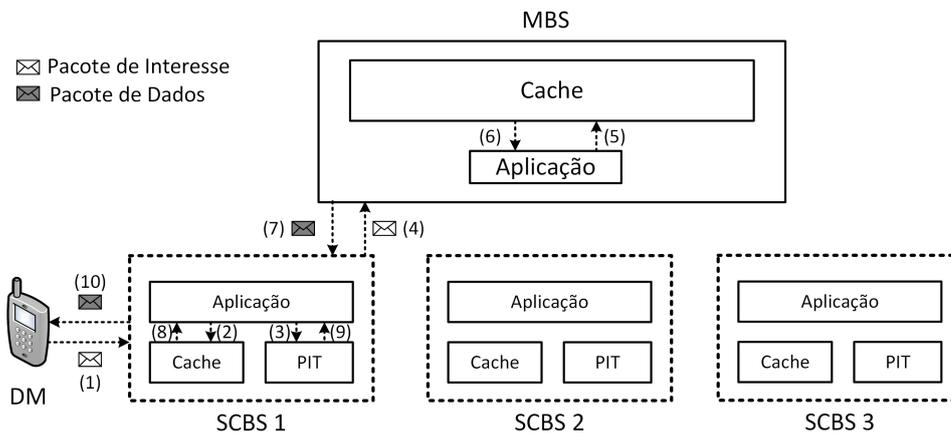


Figura 4.15 Encaminhamento de mensagens utilizando comunicação direta.

- Comunicação Indireta, conforme Figura 4.16: o DM envia um pacote de interesse para SCBS1 (1), como SCBS1 não tem o conteúdo referente ao interesse, ela encaminha o interesse à SCBS2 (2) e a SCBS2 realiza o mesmo processo de verificação e encaminhamento (3). Ao chegar na SCBS3, o pacote de interesse já percorreu todos os nós da rede daquele nível, portanto, a única entidade que pode resolver o interesse é a MBS. Logo, o servidor SCBS3 envia o pacote de interesse para a MBS (4) e a MBS envia o pacote de dados diretamente para SCBS1 (5). Isso é possível porque mesmo o pacote sendo encaminhado entre as SCBS, o endereço da primeira SCBS solicitante é mantido nas mensagens. Por fim, a SCBS1 encaminha o pacote de dados para o DM (6).

4.7 CONSIDERAÇÕES FINAIS

Nesta seção foi apresentada a iSim, uma estratégia de agrupamento de usuários similares de acordo com suas requisições de conteúdo. Essa estratégia conta com uma arquitetura ICN e pode ser implementada utilizando diferentes formas de comunicação. No próximo capítulo será apresentado o cenário de validação, as métricas, fatores e parâmetros utilizados na avaliação da estratégia e os resultados obtidos.

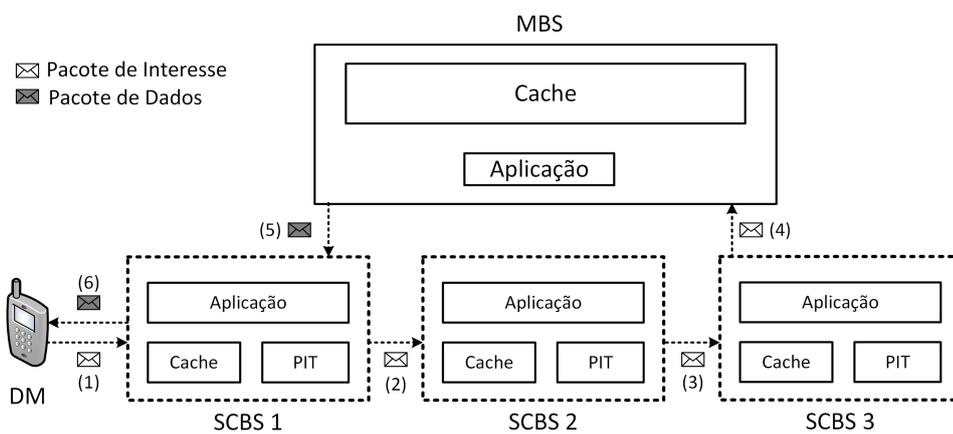


Figura 4.16 Encaminhamento de mensagens utilizando comunicação indireta.

AVALIAÇÃO EXPERIMENTAL

A avaliação experimental da iSim visa analisar o desempenho da estratégia de acordo com as métricas de taxa de acerto na SCBS e descarregamento de dados na MBS. Desta forma, neste capítulo é apresentado o método de valiação utilizado para realização das implementações da arquitetura ICN e da estratégia, são definidos os cenários, fatores, métricas e parâmetros para avaliação e é feita uma análise dos resultados obtidos.

5.1 SIMULAÇÃO

Como foi citado na seção 1, a técnica de validação utilizada foi a simulação de eventos discretos, que é caracterizada por garantir que o estado do sistema mudará apenas quando ocorrer um evento em um instante de tempo predeterminado. Ao utilizar essa técnica foi possível estabelecer rotinas de evento, variáveis de estado e contadores estatísticos para avaliação da proposta.

O simulador escolhido para implementação foi o OMNeT++, um simulador baseado em C++ que conta com diferentes *frameworks* que complementam suas funcionalidades principais. Dentre os *frameworks* desenvolvidos, existe o INET, que se trata de um *framework* para desenvolvimento de redes móveis e é composto por diferentes classes que suportam cenários sem fio e de mobilidade. Esse *framework* foi utilizado para desenvolver o cenário de aplicação, a arquitetura ICN e todas as características da iSim. Os detalhes sobre essa implementação podem ser observados no Apêndice A.

5.2 CENÁRIO DE APLICAÇÃO

Para validação da iSim, foi escolhido um cenário de Redes Centradas na Informação associado às Redes Celulares de Pequeno Porte. Enquanto a utilização de Redes Celulares de Pequeno Porte visa fazer densificação da rede para diminuir a sobrecarga nas estações principais e aumentar a cobertura, capacidade e tempos de resposta da rede, as Redes Centradas na Informação tem o intuito de determinar uma arquitetura cujo foco principal é o compartilhamento de conteúdo e, para isso, faz uso de *caches* nos próprios equipamentos de rede.

Como foi citado na Seção 4, a iSim foi desenvolvida para cenários onde há uma estação principal e algumas estações secundárias com *caches* associados. Desta forma, o cenário escolhido para validação conta com uma *Macro Base Station*, que representa a estação principal e três *Small-Cell Base Stations* que representam estações secundárias, com capacidade e cobertura inferiores. Além disso, o cenário conta com uma quantidade variável de dispositivos móveis.

O encaminhamento de pacotes de dados entre essas estações e os dispositivos móveis pode ser realizado de três formas distintas, a depender do momento da simulação. No período de treinamento, todos os usuários solicitam conteúdo diretamente à MBS, que envia o conteúdo de volta ao usuário. Esse período é essencial para a criação de um histórico de requisição dos usuários, que será utilizado como insumo no cálculo da similaridade e na formação dos agrupamentos. Passado o período de treinamento, é possível estabelecer a afinidade dos usuários de acordo com seus históricos de requisições e associar os agrupamentos aos *caches* existentes. No primeiro momento, os *caches* estarão vazios e farão solicitação de conteúdos à MBS, no entanto, à medida que a rede se estabiliza e os *caches* são povoados, a quantidade de requisições para a MBS diminui e os *caches* locais serão capazes de enviar conteúdo diretamente aos usuários. O cenário e as formas de envio de conteúdo podem ser observados na Figura 5.1.

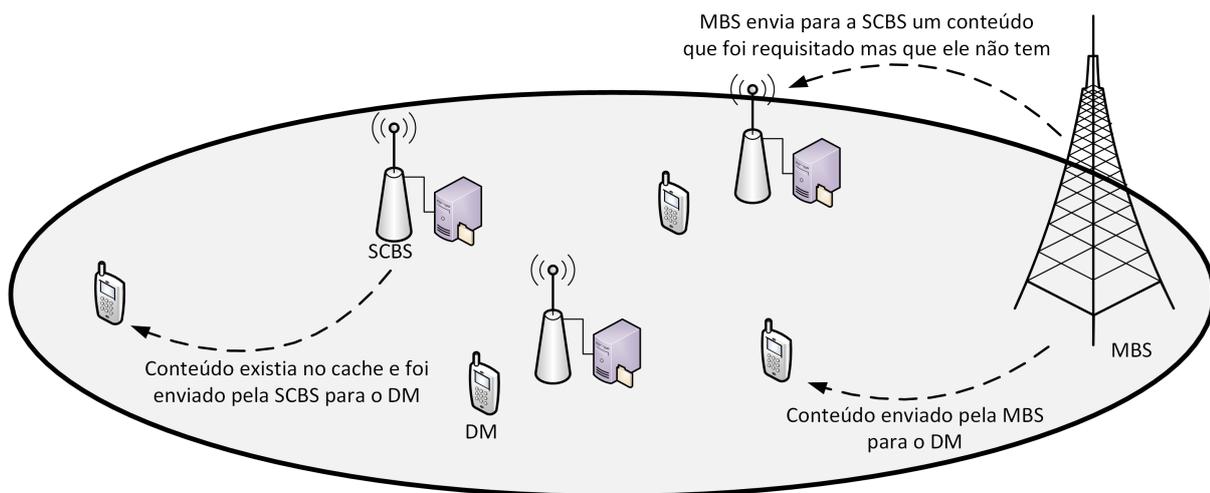


Figura 5.1 Cenário genérico utilizado para validação da iSim.

Como trata-se de uma Rede Celular de Pequeno Porte Centrada na Informação, além das características intrínsecas a um ambiente sem fio, os dispositivos de rede devem se basear em uma arquitetura ICN para se comunicar. Desta forma, foi necessário implementar algumas características de ICN nesses equipamentos.

5.3 ARQUITETURA ICN

Para definir a arquitetura ICN na MBS e na SCBS, foi necessário implementar algumas das estruturas principais que compõem os nós ICN. Apesar de existirem três estruturas padrões: *cache*, PIT e FIB, para fins de avaliação, não houve a necessidade da imple-

mentação de uma FIB que mapeasse os prefixos do conteúdo para um próximo salto. Como o foco do trabalho não envolve questões de roteamento, a ICN implementada trafega dados encapsulados por pacotes IP.

A estrutura de *cache* foi definida na MBS e na SCBS e foi desenvolvida como uma lista de uma classe cujos atributos são nome do conteúdo, conteúdo, frequência e último envio. Desta forma, é permitido utilizar as seguintes políticas de descarte: FIFO, LFU, LRU e LFRU (do inglês, *Least Frequently Recently Used*).

O *cache* da MBS tem tamanho máximo equivalente à quantidade de conteúdo disponível na rede, enquanto o *cache* das SCBS tem, cada um, tamanhos que variam de 10% a 30% da quantidade de conteúdo disponível. Como o padrão ICN é armazenar todos os conteúdos, depois do período de inicialização dos *caches*, todo conteúdo que chega deverá substituir um conteúdo já existente.

Como a MBS tem todos os conteúdos que serão solicitados, não há necessidade de criar uma tabela para armazenar interesses pendentes, visto que todos os interesses podem ser resolvidos por ela. Desta forma, a PIT é uma estrutura utilizada apenas na SCBS. A PIT, assim como o *cache*, é definida através da lista de uma classe que contém o nome do conteúdo e uma lista de endereços IPs que solicitaram aquele conteúdo. Ela permite a agregação de requisições, logo, quando um dispositivo faz requisição para um conteúdo e já existe uma entrada na PIT, a requisição é descartada e o endereço IP do dispositivo é adicionado na lista de interfaces solicitantes.

Além da SCBS e da MBS, o terceiro equipamento utilizado é o dispositivo móvel. Cada dispositivo móvel representa um usuário que faz solicitações para conteúdos e os usuários precisam manter a tabela de requisições atualizada, para enviá-la para a MBS para calcular a similaridade e a formação dos agrupamentos. A RT foi implementada como uma lista na qual cada posição representa o ID de um conteúdo (por exemplo, a posição 30 representa o conteúdo www.ufba.br/redes/aula30) e cada valor representa a quantidade de vezes que aquele conteúdo foi solicitado.

Além dessas estruturas, foi preciso definir os tipos de pacotes que trafegam na rede. Assim, foi definida uma classe genérica que representa um pacote CCN e é composta pelos atributos nome do conteúdo, tipo de pacote, quantidade de saltos, endereço IP do nó assinante, tipo de nó que encaminhou o interesse e um *payload* que, a depender do tipo de pacote, irá conter ou não conteúdo.

Cada DM requisita um conjunto específico de conteúdos seguindo a distribuição de popularidade de Zipf e cujo intervalo entre as requisições é determinado por um modelo de *Poisson*. De acordo com esse cenário genérico que baseia-se em uma arquitetura ICN para Redes Móveis, foram determinados diferentes fatores, métricas e parâmetros para avaliação.

5.4 CENÁRIOS, MÉTRICAS, FATORES E PARÂMETROS

Os fatores utilizados podem ser vistos na primeira coluna da Tabela 5.1. A similaridade representa a utilização da equação de similaridade e o β representa o peso usado na equação. A comunicação retrata a direção no encaminhamento dos pacotes, o tamanho do *cache* indica espaço disponível para armazenamento, e o α representa o parâmetro da

Fatores	Grupo 1	Grupo 2	Grupo 3
Similaridade	Sim - Não	Sim - Não	Sim - Não
Comunicação	Direta	Direta	Indireta - Direta
Tamanho do <i>Cache</i>	10% - 20% - 30%	20%	20%
β (Funç. Similaridade)	0 - 0.5 - 1	0 - 0.5 - 1	0.5
α (Distr. Zipf)	1	0.6 - 0.8 - 1	1

Tabela 5.1: Fatores e Cenários de Avaliação.

Distribuição de *Zipf*, que é responsável por definir a popularidade dos conteúdos.

De acordo com esses fatores foram definidos três cenários de avaliação: o Grupo 1 compara o impacto do parâmetro β e do Tamanho da *Cache* em relação à taxa de acerto; o Grupo 2 avalia o impacto do parâmetro Zipf e do parâmetro β na taxa de acerto; e, por fim, o Grupo 3 analisa a utilização da similaridade e da hierarquia em relação à taxa de acerto e ao descarregamento de dados. Além disso, são avaliados os efeitos na variação da quantidade de nós e de conteúdos e a necessidade de recalculer a similaridade entre os usuários.

Os experimentos foram avaliados de acordo com duas métricas: taxa de acerto na SCBS e descarregamento de dados na MBS. O descarregamento de dados visa medir a quantidade de requisições que deixaram de ser feitas na MBS e pode ser representado pela Equação 5.1, na qual $rMax$ representa os interesses que foram requisitados quando todas as requisições eram feitas diretamente na MBS e $rMin$ são as requisições recebidas após a formação dos agrupamentos.

$$off = 100 \times \frac{rMax - rMin}{rMax} \quad (5.1)$$

Já a taxa de acerto mede a quantidade de vezes que uma SCBS recebeu um pacote de interesse e tinha o pacote de dados no *cache*. Essa taxa é medida de acordo com a Equação 5.2, na qual $match$ representa os interesses que foram resolvidos ao chegar na SCBS e $iRec$ o total de todos os interesses que a SCBS recebeu. A taxa de acerto total é calculada através da média aritmética da taxa de acerto das três SCBS.

$$hr = 100 \times \frac{match}{iRec} \quad (5.2)$$

A taxa de acerto e a taxa de descarregamento de dados são, geralmente, diretamente proporcionais. Assim, o objetivo é aumentar a quantidade de requisições resolvidas nas SCBS para que a MBS receba menos requisições.

5.5 AMBIENTE DE SIMULAÇÃO

A solução proposta foi implementada no simulador OMNeT++ utilizando o *framework* INET, conforme parâmetros da Tabela 5.2.

A área simulada é de 600x800m onde estão distribuídos dispositivos móveis que se movimentam de acordo com o modelo de mobilidade *Mass Mobility* e requisitam conteúdo

Parâmetro	Valor
Quantidade de dispositivos móveis na rede	30
Quantidade de MBS	1
Quantidade de SCBS	3
Quantidade de conteúdos disponíveis	100
Política de descarte de <i>cache</i>	LFU
Modelo de Mobilidade	<i>MassMobility</i>
Tipo de Propagação	<i>ConstantSpeedPropagation</i>
Frequência	2.4GHz
Perda	<i>FreeSpacePathLoss</i>
Ruído	-110dBm

Tabela 5.2: Parâmetros da Simulação.

seguindo o modelo de Poisson. O meio físico sem fio tem propagação do tipo *Constant Speed Propagation*, frequência de 2.4GHz, perda do tipo *Free Space Path Loss* e ruído de -110dBm, todos esses parâmetros já vieram especificados no INET. Nesse cenário foram adicionadas uma MBS e três SCBS.

Na simulação, os dispositivos são associados às SCBS de forma imperativa, de acordo com a estratégia utilizada. De acordo com a quantidade de conteúdos disponíveis na rede, foi selecionado um subconjunto de conteúdos para cada dispositivo para determinar perfis diferentes para os usuários. Para cada subconjunto foi aplicado o modelo de popularidade de *Zipf*.

O tamanho do *cache* da SCBS varia entre 10% e 30% da quantidade de conteúdos e a estratégia de descarte utilizada é a LFU. O parâmetro β , que determina o peso na equação de similaridade, é variado entre 0 e 1 e são comparados cenários com comunicação direta e indireta. Os resultados foram gerados com um Intervalo de Confiança de 95%.

5.6 ANÁLISE DOS RESULTADOS

Nesta seção são apresentados os resultados e as análises dos diferentes cenários em relação à taxa de acerto e de descarregamento de dados de acordo com as variações de fatores apresentadas na Tabela 5.1.

5.6.1 Grupo 1: Relação entre Tamanho do Cache, β e Taxa de Acerto

A Figura 5.2 expõe o comportamento da taxa de acerto em relação ao uso da função de similaridade e da variação do tamanho do *cache*. Este experimento foi executado com 30 dispositivos móveis, 100 conteúdos disponíveis, comunicação direta e $\alpha = 1$. Foram comparadas as variações da equação de similaridade, bem como um cenário de agrupamento randômico, também chamado de sem similaridade.

Em relação ao uso da equação de similaridade, para todos os tamanhos de *cache* (10, 20, 30), é possível observar que quando apenas a quantidade de conteúdos em comum é levada em consideração ($\beta = 1$), a taxa de acerto é superior a quando apenas a frequência

é utilizada ($\beta = 0$). Isso pode ser justificado devido ao fato de que ao contabilizar apenas a frequência, é possível que dois usuários sejam considerados bastante similares mesmo tendo um único conteúdo em comum. Por exemplo, quando dois nós tem um conteúdo em comum, ainda que requisitado uma única vez, de acordo com a função da similaridade do cosseno, esses dois nós terão similaridade igual a 1. Dessa forma, como o Índice de Jaccard leva em consideração a relação entre os conteúdos requisitados em comum e todos os conteúdos requisitados pelos dois nós, os agrupamentos baseados nessa relação de similaridade conseguem alcançar resultados superiores. Apesar disso, os melhores resultados foram obtidos quando ambos ($\beta = 0.5$) foram utilizados, evidenciando a importância de uma equação de similaridade que considere esses dois fatores.

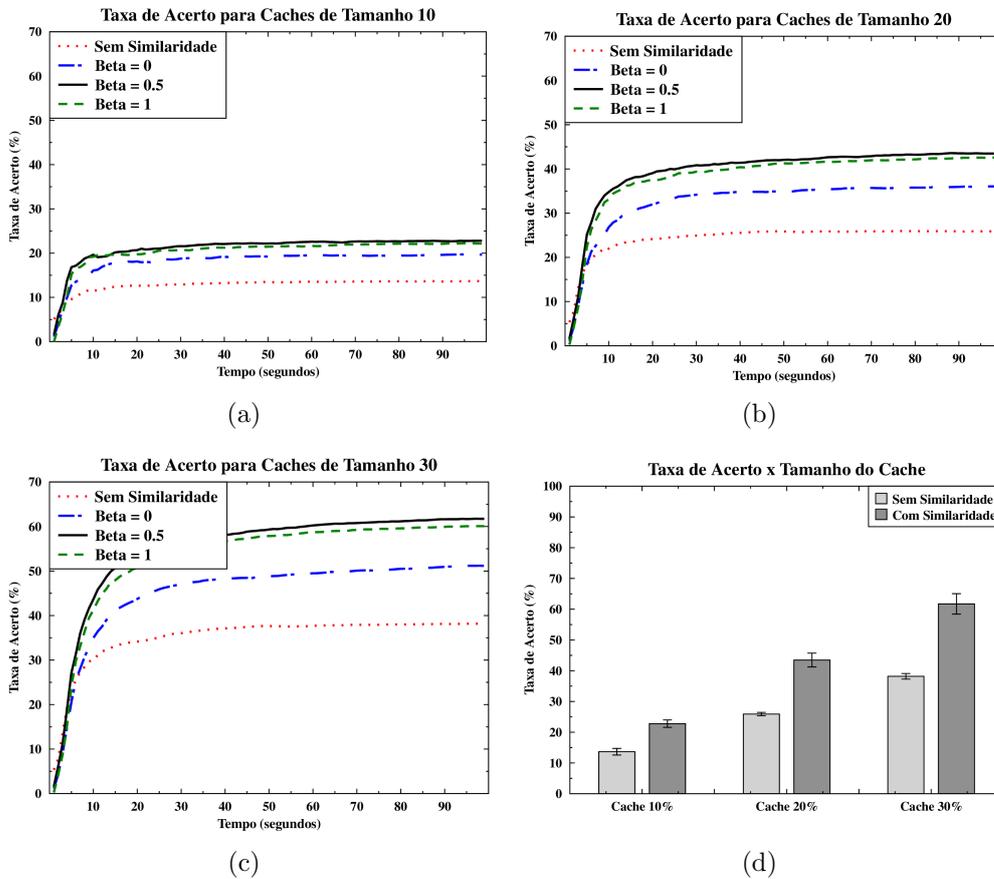


Figura 5.2 Relação entre o Tamanho do *Cache* e a Taxa de Acerto.

A distinção entre as taxas de acerto dos gráficos das Figuras 5.2(a) 5.2(b) 5.2(c) se dá devido ao aumento do tamanho do *cache*. Desta forma, é possível observar que há um crescimento constante proporcional à capacidade de armazenamento, sendo que, nos casos sem similaridade a taxa de acerto corresponde ao tamanho do *cache*. Logo, para *cache* de tamanho 10, é observada uma taxa de acerto de 10%, enquanto para *caches* de tamanho 20 e 30 a taxa de acerto é de 20% e 30%, respectivamente. Quanto utilizada a equação de similaridade, as taxas de acerto conseguem atingir, aproximadamente, o

dobro da capacidade de armazenamento.

Ao relacionar os tamanhos do *cache* 10 e 20, conforme a Figura 5.2(d), observa-se que a taxa de acerto obtida sem similaridade com um *cache* = 20 alcança valores próximos ao melhor valor obtido com uso de similaridade em um *cache* = 10. Ao se comparar os *caches* de tamanho 20 e 30, percebe-se que com o uso da similaridade em um *cache* = 20 é possível obter valores superiores ao cenário sem similaridade em um *cache* = 30. Isso evidencia que a utilização de agrupamentos levando em consideração a similaridade dos usuários torna, de fato, o gerenciamento de *cache* mais eficiente e possibilita economia de recursos, visto que os mesmos valores de taxa de acerto podem ser obtidos com *caches* de tamanho inferior. Além disso, na Figura 5.3 é possível observar que a taxa de descarregamento de dados na MBS é diretamente proporcional à taxa de acerto obtida pelas SCBS, alcançando, aproximadamente, os mesmos valores.

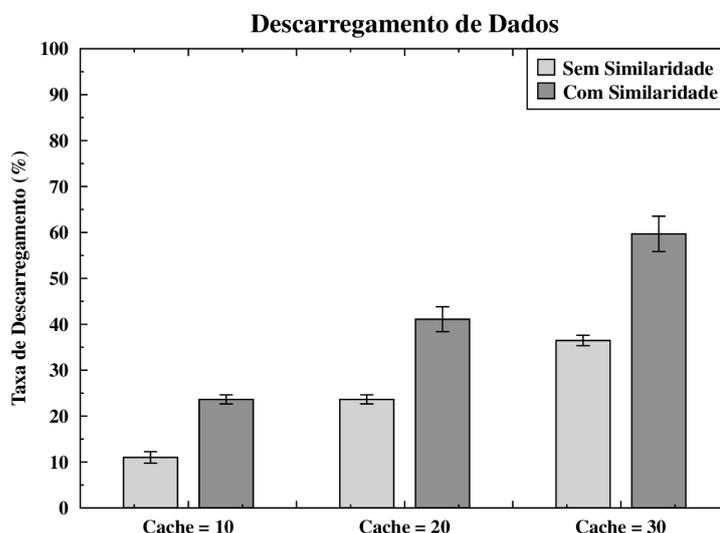


Figura 5.3 Efeito do tamanho do cache na Taxa de Descarregamento de Dados.

5.6.2 Grupo 2: Relação entre o parâmetro α da Lei de Zipf e a Taxa de Acerto

A Figura 5.4 demonstra os impactos da variação na popularidade dos conteúdos na taxa de acerto. Este experimento foi executado com 30 dispositivos móveis, 100 conteúdos e tamanho de *cache* = 20. O parâmetro α foi variado em 0.6, 0.8 e 1.

A Figura 5.4 evidencia que mesmo com diferentes valores para o parâmetro Zipf, o comportamento da função de similaridade permanece o mesmo: os resultados com valor de $\beta = 0.5$ ou $\beta = 1$ são superiores a quando considera-se apenas a frequência. No entanto, os efeitos do cálculo de similaridade são baixos, visto que o cenário sem similaridade consegue atingir resultados próximos.

Apesar do comportamento da função de similaridade permanecer o mesmo, é possível observar na Figura 5.5 que os valores de taxa de acerto alcançados são distintos: quanto menor o valor do parâmetro Zipf, maior a taxa de acerto obtida. Isso acontece porque

quando o parâmetro Zipf é menor, há uma maior quantidade de conteúdos populares que, conseqüentemente, são requisitados mais vezes. Como a política de descarte utilizada é a LFU, os conteúdos mais populares ficam armazenados por mais tempo, satisfazendo as requisições e culminando no aumento da taxa de acerto.

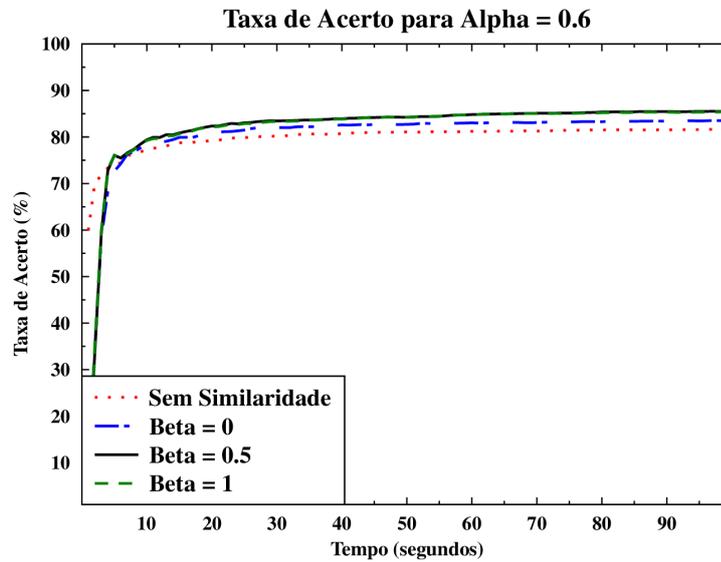


Figura 5.4 Efeito do parâmetro Zipf na Taxa de Acerto.

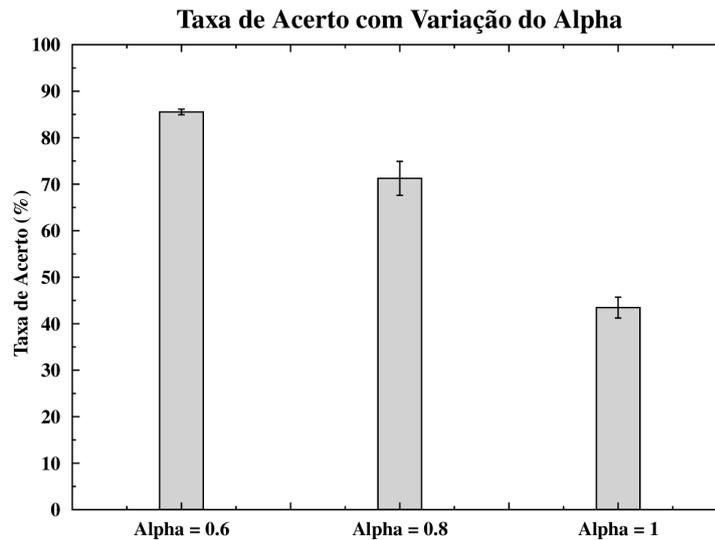


Figura 5.5 Efeito do parâmetro Zipf na Taxa de Acerto.

5.6.3 Grupo 3: Relação entre Similaridade e Comunicação na Taxa de Acerto e no Descarregamento de Dados

Na Figura 5.6 é possível observar o comportamento entre similaridade e comunicação em relação à taxa de descarregamento. Apesar de a taxa de acerto e do descarregamento de dados serem, comumente, proporcional um ao outro, em cenários com comunicação indireta é possível observar comportamento oposto.

De acordo com a Figura 5.6, a taxa de descarregamento em um cenário sem similaridade e com comunicação direta é de 22% obtida no cenário sem similaridade e com comunicação direta. Esta melhora é decorrente da adição das três SCBS, cada uma com *cache* de tamanho = 20. Como vimos na Seção 5.6.1, quando a similaridade não é levada em consideração, a taxa de acerto é proporcional ao tamanho do *cache*. Logo, se o mesmo experimento fosse executado com *cache* de tamanho 30, a taxa de descarregamento obtida pela adição das SCBS seria um valor próximo de 30.

Ainda na Figura 5.6, o cenário com comunicação direta e com similaridade consegue obter resultados superiores ao cenário com comunicação indireta e sem similaridade, evidenciando que a utilização de agrupamentos influencia mais no descarregamento do que a forma de comunicação.

Os melhores resultados para descarregamento de dados são aqueles obtidos quando os grupos são formados de acordo com a similaridade dos nós e quando todas as SCBS são percorridas antes de buscar na estação central (comunicação indireta).

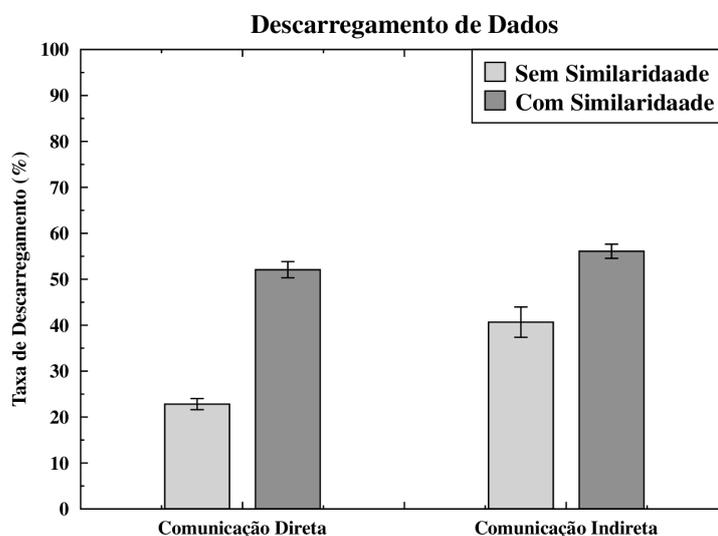


Figura 5.6 Influência da Similaridade e da Hierarquia no Descarregamento de Dados.

Em relação a avaliação da taxa de acerto nos cenários com e sem similaridade e comunicação direta e indireta, é possível observar, de acordo com a Figura 5.7, que os cenários de comunicação direta obtém taxa de acerto superior aos cenários de comunicação indireta. Isso acontece porque quando a comunicação é indireta, as SBCS recebem pacotes de interesse que não pertencem aos usuários do grupo que está associado a elas, portanto,

a probabilidade delas terem o conteúdo armazenado em *cache* é menor. Apesar disso, o grupo de usuários associado a cada SCBS não sofre nenhum impacto decorrente dessa diminuição da taxa de acerto, visto que não há substituição em *cache* dos conteúdos que foram solicitados por outra SCBS. No entanto, há um aumento no tempo de resposta ao usuário decorrente da comunicação indireta, visto que pode aumentar a quantidade de saltos para alcançar o conteúdo.

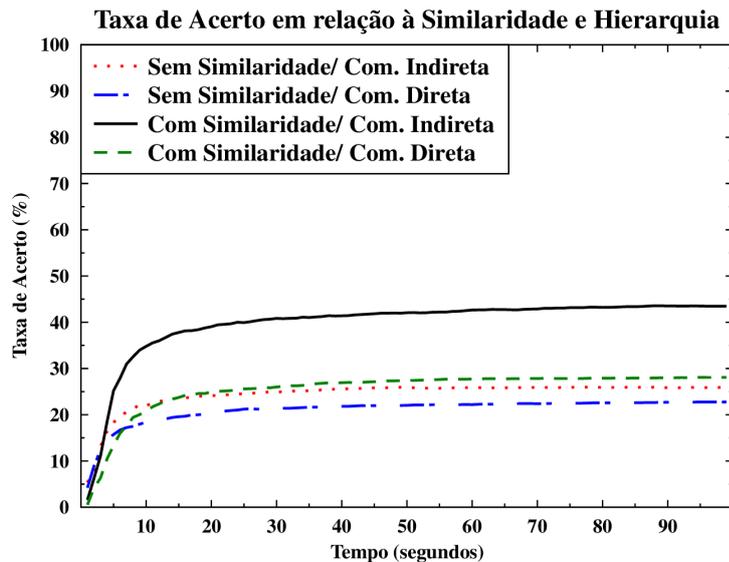


Figura 5.7 Influência da Similaridade e da Hierarquia na Taxa de Acerto.

5.6.4 Avaliação da variação na quantidade de dispositivos móveis e conteúdo

Alguns outros fatores que também podem influenciar no cálculo da similaridade e na formação dos agrupamentos são a quantidade de dispositivos móveis na rede e a quantidade de conteúdos disponíveis. Na Figura 5.8 é possível observar a comparação entre cenários em que a similaridade é levada em consideração mas há variação na quantidade de dispositivos móveis na rede. Assim, observa-se que com 30 DMs, para *cache* de tamanho 20, é alcançada uma taxa de 40% de descarregamento de dados (conforme as análises realizadas na Seção 5.6.1). Com um aumento de 50% na quantidade de nós, a degradação na taxa de descarregamento é de aproximadamente 5%, resultando em valores estatisticamente semelhantes e indicando que a quantidade de dispositivos não tem alta interferência na taxa de descarregamento quando os interesses dos dispositivos continuam similares. Apesar disso, a adição de DMs pode causar aumento no uso da largura de banda, na perda de pacotes e, conseqüentemente, no tempo de resposta ao usuário.

Em relação à variação na quantidade de conteúdos, observa-se na Figura 5.9 a comparação entre dois cenários: um cenário com quantidade de conteúdo igual a 100 e outro cenário com quantidade de conteúdo igual a 200. No cenário com 100 conteúdos, ao utilizar um tamanho de *cache* capaz de armazenar 20% do conteúdo existente ($cache = 20$), é

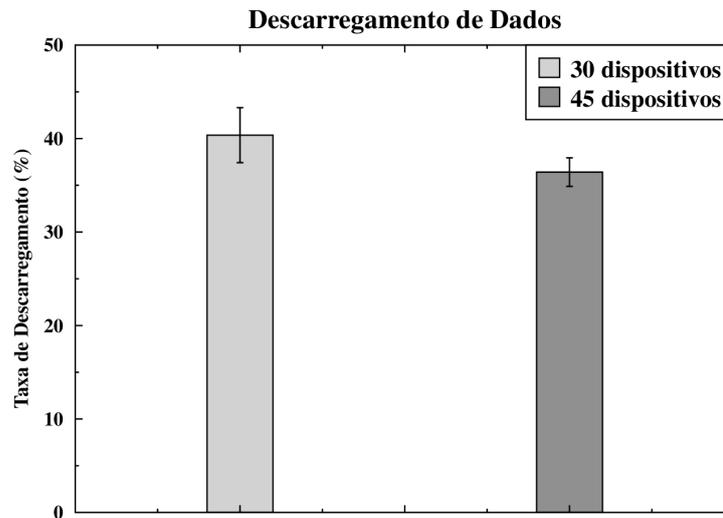


Figura 5.8 Efeito da variação da quantidade de dispositivos na Taxa de Descarregamento de Dados.

possível obter uma taxa de descarregamento de, aproximadamente, 40%. Com o mesmo tamanho de *cache* ($cache = 20$), mas em um cenário com o dobro de conteúdos, há uma diminuição na taxa, que decai para a marca de 25%. No entanto, para a nova quantidade de conteúdos, esse tamanho representa apenas 10% da quantidade total. Assim, mesmo com o dobro de conteúdos, foi possível obter uma taxa semelhante ao cenário sem similaridade com apenas 100 conteúdos. Ao avaliar o cenário com 200 conteúdos e tamanho de $cache = 40$ é possível obter uma taxa de descarregamento de, aproximadamente, 45%, valor ligeiramente superior ao cenário com tamanho de *cache* equivalente à metade dos conteúdos. Desta forma, observa-se que mesmo com o aumento na quantidade de conteúdos, ainda é possível manter uma boa taxa de descarregamento, quando mantida a proporção no tamanho do *cache*.

5.6.5 Avaliação do recálculo dos grupos

Nos experimentos anteriores, todos os nós eram distribuídos de maneira uniforme e não foi considerada a entrada/saída de nós. Neste experimento foi avaliada a necessidade de recálculo de agrupamentos em um cenário dinâmico. Assim, esse experimento foi executado com 45 nós, sendo que 30 nós foram introduzidos no início da simulação, enquanto os outros 15 nós aparecem a partir do instante 100. Para esse experimento, o algoritmo de agrupamento foi executado três vezes: no instante 1, no instante 150 e no instante 300. Sendo que no instante 1 foi executado apenas para os 30 nós e nos outros para os 45 nós. Para aumentar a sensibilidade em relação às mudanças de taxa de acerto, considerou-se uma janela de observação em relação ao histórico das últimas requisições (*matches* e requisições não atendidas pelo *cache* local). Essa janela foi definida de forma empírica considerando os últimos 40 segundos, tempo suficiente para aumentar a granularidade e visualizar o efeito do recálculo.

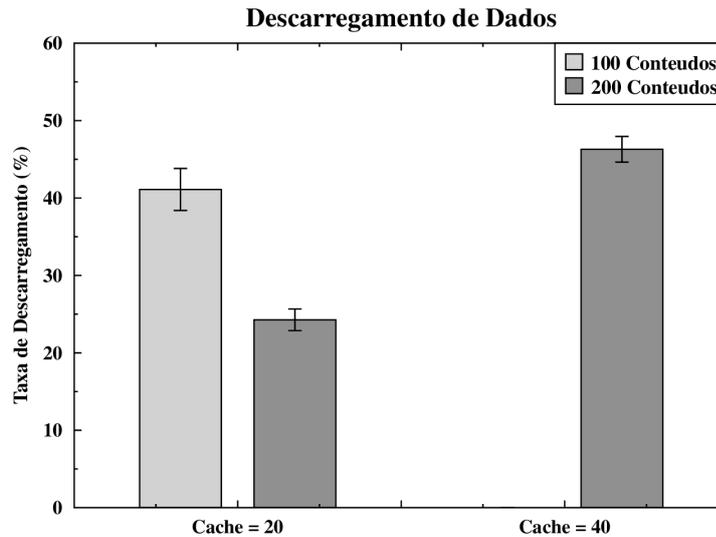


Figura 5.9 Efeito da quantidade de conteúdos na Taxa de Descarregamento de Dados.

Na Figura 5.10 são demonstrados os possíveis efeitos do reagrupamento dos usuários. Nesta figura foram avaliados os casos sem similaridade e com similaridade. No cenário com similaridade foram consideradas amostras distintas em relação às requisições de conteúdo. A avaliação das amostras de forma separada é importante para demonstrar as mudanças decorrentes da formação dos novos agrupamentos e associações entre SCBS. No instante 100 há uma queda, em todas as linhas do gráfico, referente à adição de novos usuários na rede. Essa queda permanece até o instante 150. Ao observar o cenário “Com Similaridade II”, é possível verificar que a partir de $t = 150$ a taxa de acerto começa a estabilizar, com sentido ascendente, enquanto o cenário “Com Similaridade I” tem uma queda ainda mais acentuada. É importante ressaltar, que ambos cenários tem os mesmos parâmetros, o motivo dessa divergência entre os resultados deve-se à mudança de SCBS.

Supondo que existem três grupos, A, B e C formados pelos 30 usuários que inicializaram junto com o sistema, e que os grupos estão associados às SCBS 1, 2 e 3, respectivamente, ao recalculer os agrupamentos com os novos usuários, há a possibilidade na mudança de associação entre os grupos e as SCBS. Como os grupos são distintos, os interesses também são, assim, há um período de readequação às solicitações dos novos usuários, que acabam buscando uma grande quantidade de conteúdos que não está armazenada em *cache*. Portanto, o grupo A, que inicialmente estava associado à SCBS 1, pode ser recalculado e associado à SCBS 2, por exemplo. Logo, essa queda da linha 2, representada no instante 150, corresponde às mudanças de grupos nas SCBS. Esse mesmo comportamento pode ser observado no instante 300.

Desta forma, uma vez que os agrupamentos iniciais são estabelecidos, há também a possibilidade de fazer a associação entre os usuários que entram na rede e o conjunto de conteúdos armazenado nas SCBS naquele instante. Isso evita que haja mudanças de grupos de usuários entre as SCBS mas pode causar desgaste na taxa de acerto ao longo do tempo, já que os grupos deixam de ser estabelecidos entre os próprios usuários.

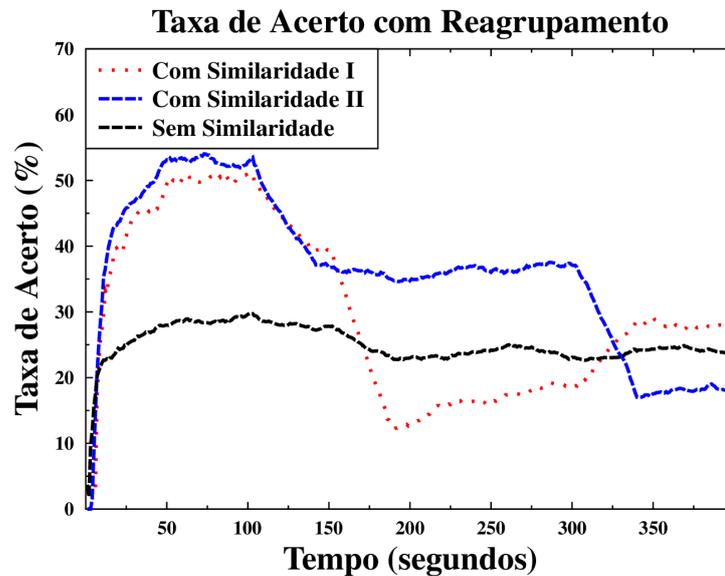


Figura 5.10 Efeito do reagrupamento de Usuários.

O cenário “Sem Similaridade” apesar de ter desempenho inferior ao uso da similaridade, apresenta-se mais estável durante a entrada e saída dos usuários, visto que os ganhos não serão acentuados, uma vez que os grupos são formados aleatoriamente.

Embora o cenário avaliado tenha como característica a mobilidade dos nós, pressupõe-se sua conectividade à qualquer SCBS. No entanto, em um cenário real, é comum que um nó esteja fisicamente distante de um *cache* que seria mais apropriado. Logo, a construção dos agrupamentos entre os nós só pode ser garantida de maneira lógica, já que não há meios de garantir que os nós estarão ao alcance da SCBS recomendada. Assim, a mensagem de controle contendo o endereço da SCBS que o dispositivo deve se associar, é encaminhada como uma recomendação. Caso esteja ao alcance daquela antena, o dispositivo deve priorizá-la, independente da relação Sinal/Ruído; no entanto, caso não esteja, o dispositivo deve se conectar a SCBS com sinal mais forte.

5.7 CONSIDERAÇÕES FINAIS

Este capítulo abordou o processo de avaliação experimental, indicando o método de validação utilizado, os mecanismos implementados e descrição dos cenários, fatores, métricas e parâmetros para avaliação experimental. Além disso, foi realizada uma análise dos experimentos de acordo com os resultados obtidos para os diferentes fatores. No próximo capítulo serão expostas as conclusões do trabalho e as principais contribuições.

CONCLUSÕES E TRABALHOS FUTUROS

A densificação de redes associada à utilização de Redes Centradas na Informação é uma alternativa que vem sendo considerada como uma das soluções para o problema de sobrecarga nas estações centrais. As células adicionadas à infraestrutura devem ser capazes de servir os usuários associados de maneira mais eficiente possível, sendo que sua eficiência é medida, principalmente, através da taxa de acerto. Em busca de aumentar essa taxa de acerto e prover descarregamento de dados, nesta dissertação foi proposta uma estratégia de cálculo de similaridade e agrupamento de usuários de acordo com seus interesses por conteúdo. O objetivo é associar usuários que têm interesses similares, desta forma, os conteúdos mais populares para cada grupo de usuário seria mais requisitado e, provavelmente, ficaria mais tempo em *cache*, aumentando a taxa de acerto nas SCBS e diminuindo a sobrecarga na estação principal.

O cálculo da similaridade levou em consideração tanto a semelhança e a quantidade total dos conteúdos solicitados entre os nós quanto a frequência das solicitações dos conteúdos em comum. A partir da definição da função de similaridade, foram determinados os agrupamentos e feita a associação entre os grupos e cada SCBS. A função de similaridade foi avaliada levando em consideração diferentes fatores e seus impactos na rede. Assim, foi avaliada a variação do tamanho do *cache*, a popularidade dos conteúdos, o uso da similaridade na formação dos agrupamentos, entre outros. Com essas avaliações, foi possível observar que utilizar uma equação de similaridade com o Índice de Jaccard para medir a relação entre conteúdos em comum e conteúdo total e o uso da similaridade do cosseno para medir a frequência dos conteúdos em comum, traz melhores resultados quando comparado a cenários que utilizam apenas um desses fatores.

Do ponto de vista do descarregamento de dados, a adição de *cache* proposta na arquitetura ICN consegue prover descarregamento de dados proporcional à capacidade de armazenamento de cada *cache*. Portanto, é recomendado associar alguma inteligência ao *cache* para otimização dos recursos de armazenamento disponível. Assim, observou-se que a formação de agrupamentos baseados na similaridade dos usuários permitiu um melhor aproveitamento do *cache*, gerando aumento na taxa de acerto e melhor descarregamento de dados.

Outros aspectos observados foram a influência do aumento na quantidade de dispositivos e de conteúdos. O aumento na quantidade de dispositivos não gerou impacto do ponto de vista de descarregamento, indicando que a adição de usuários com interesses semelhantes aos usuários já existentes não prejudica o comportamento do *cache*, no entanto, a adição de dispositivos pode causar impactos em relação ao tempo de resposta e consumo de largura de banda. Em relação à quantidade de conteúdos, mesmo com o aumento de conteúdo, a iSim conseguiu manter um bom resultado quando o *cache* aumenta de forma proporcional.

Além disso, o cenário de SCN é adequado para diferentes propósitos, e a iSim pode considerar cenários diversificados, visto que pode atender cenários com diferentes características em relação à requisição de conteúdos. Assim, é possível utilizar a iSim em cenários como *shoppings*, estádios de futebol, eventos de cultura pop e universidades, para fazer recomendação de *cache* aos usuários e tornar a rede mais eficiente, melhorando o descarregamento de dados e diminuindo o tempo de resposta ao usuário.

6.1 CONTRIBUIÇÕES

As principais contribuições desse trabalho envolvem a formulação de uma equação de similaridade para desenvolvimento de um cenário de Redes Centradas na Informação aplicado às Redes Móveis com o intuito de fazer descarregamento de dados. De acordo com os objetivos estabelecidos, foi possível definir as seguintes contribuições:

- Avaliação do impacto da utilização de Redes Centradas na Informação para auxiliar no problema de descarregamento de dados em Redes Celulares.
- Desenvolvimento de ambiente de simulação composto por uma rede de infraestrutura do tipo *Small Cell Network* com uma arquitetura baseada em Redes Centradas na Informação.
- Desenvolvimento de uma função de similaridade baseada no Índice de Jaccard e na Similaridade do Cosseno e de um algoritmo de agrupamento baseado no kNN.
- Avaliação experimental da iSIM levando em consideração diferentes fatores e parâmetros.

A estrutura inicial do simulador foi apresentada no DCC DEMO DAY de 2016 sob o título “Gerenciamento de Cache Ciente de Similaridade em uma Rede MESH Centrada no Conteúdo”¹. Depois dessa apresentação, muitas alterações foram realizadas de acordo com o cenário definido para validação.

A iSim foi publicada na trilha principal do XXXV Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC) sob o título: **Explorando a afinidade de usuários para descarregamento de dados mais eficiente em redes celulares de pequeno porte**, com autoria de Adriana Viriato Ribeiro, Leobino N. Sampaio e Artur Ziviani.

¹<https://youtu.be/IPR5qpOLL8M>

Além disso, houve uma cooperação para o desenvolvimento de um artigo, também publicado na trilha principal do SBRC, na área de Redes Centradas na Informação com foco em segurança, intitulado: Um Modelo de Rede Centrada na Informação Resiliente a Ataques de Negação de Serviços por Inundação de Interesses, com autoria de Nilton Flávio S. Seixas, Adriana Viriato Ribeiro e Leobino N. Sampaio.

6.2 TRABALHOS FUTUROS

De acordo com a avaliação realizada, foi possível constatar que a iSim consegue atingir melhores resultados quando considera tanto a diversidade quanto a frequência dos conteúdos nos cenários observados. No entanto, para avaliar a adequação da estratégia e seu comportamento de maneira mais ampla, é necessário realizar novos experimentos envolvendo maiores variações na quantidade de conteúdo, de dispositivos, de modelos de popularidade de conteúdo, de políticas de descarte de *cache*, entre outros fatores.

Outro aspecto que deve ser abordado nas experimentações é o efeito da mobilidade dos nós na estratégia proposta. Para isso, é preciso também considerar a distância de um usuário para seus semelhantes e ajustar o mecanismo de recomendação de SCBS para o usuário. Em particular, quando o usuário não estiver ao alcance da SCBS recomendada, ele pode conectar-se à SCBS com melhor sinal. Assim, outras estratégias de associação entre um móvel móvel e a SCBS recomendada podem ser alvo de trabalhos futuros.

Em relação à evolução da iSim, é possível utilizar-se da inteligência considerada no agrupamento dos usuários para prover mecanismos que favoreçam a eficiência do *cache*. Desta forma, pode-se investigar mecanismos de recomendação de conteúdo aos usuários baseados no seu histórico de requisições e nas características de outros usuários similares. Além disso, é possível criar uma estratégia de descarte que considere as características dos agrupamentos, a frequência de solicitação dos conteúdos e o próprio mecanismo de recomendação para identificar o conteúdo que deve ser substituído, considerando não apenas o seu histórico mas também a possibilidade de requisições futuras.

REFERÊNCIAS BIBLIOGRÁFICAS

- AHLGREN, B. et al. A survey of information-centric networking. *IEEE Communications Magazine*, IEEE, v. 50, n. 7, 2012.
- AHMED, S. H.; BOUK, S. H.; KIM, D. *Content-centric networks: an overview, applications and research challenges*. Springer, 2016.
- AIJAZ, A.; AGHVAMI, H.; AMANI, M. A survey on mobile data offloading: technical and business perspectives. *IEEE Wireless Communications*, IEEE, v. 20, n. 2, p. 104–112, 2013.
- ANPALAGAN, A. *Design and Deployment of Small Cell Networks*. Cambridge University Press, 2015.
- APOSTOLARAS, A. et al. A mechanism for mobile data offloading to wireless mesh networks. *IEEE Transactions on Wireless Communications*, IEEE, v. 15, n. 9, p. 5984–5997, 2016.
- ARUMAITHURAI, M. et al. *Enabling Publish/Subscribe in ICN*. 2014.
- BHUSHAN, N. et al. Network Densification: The Dominant Theme for Wireless Evolution into 5G. *IEEE Communications Magazine*, v. 52, n. 2, p. 82–89, February 2014. ISSN 0163-6804.
- BRITO, G. M. de; VELLOSO, P. B.; MORAES, I. M. Redes orientadas a conteúdo: Um novo paradigma para a internet. *Minicursos do Simpósio Brasileiro de Redes de Computadores-SBRC*, v. 2012, p. 211–264, 2012.
- CAMPISTA, M. E. M. et al. Challenges and research directions for the future inter-networking. *IEEE Communications Surveys & Tutorials*, v. 16, n. 2, 2014.
- CARZANIGA, A.; RUTHERFORD, M. J.; WOLF, A. L. A routing scheme for content-based networking. In: IEEE. *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*. 2004. v. 2, p. 918–928.
- CHANG, Z. et al. Context-Aware Data Caching for 5G Heterogeneous Small Cells Networks. In: IEEE. *Communications (ICC), 2016 IEEE International Conference on*. 2016. p. 1–6.
- CHE, H.; TUNG, Y.; WANG, Z. Hierarchical web caching systems: Modeling, design and experimental results. *IEEE Journal on Selected Areas in Communications*, IEEE, v. 20, n. 7, p. 1305–1314, 2002.

CHEN, M. et al. Mobility-aware caching and computation offloading in 5G ultra-dense cellular networks. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 16, n. 7, p. 974, 2016.

CHEN, Z.; KOUNTOURIS, M. Cache-enabled Small Cell Networks with Local User Interest Correlation. In: *IEEE 16th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. 2015. p. 680–684.

CISCO. *Cisco VNI Forecast and Methodology, 2015-2020*. 2016. Disponível em: <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html>.

CLARK, D. The design philosophy of the darpa internet protocols. *ACM SIGCOMM Computer Communication Review*, ACM, v. 18, n. 4, p. 106–114, 1988.

CLIFFORD, H.; STEPHENSON, W. An introduction to numerical taxonomy. *London: Acad. Press. Google Scholar*, 1975.

DANNEWITZ, C. Netinf: An information-centric design for the future internet. In: *Proc. 3rd GI/ITG KuVS Workshop on The Future Internet*. 2009. p. 1–3.

DICE, L. R. Measures of the amount of ecologic association between species. *Ecology*, Wiley Online Library, v. 26, n. 3, p. 297–302, 1945.

DIMATTEO, S. et al. Cellular traffic offloading through wifi networks. In: *IEEE. Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on*. 2011. p. 192–201.

DIMITROV, V.; KOPTCHEV, V. Psirp project—publish-subscribe internet routing paradigm: new ideas for future internet. In: *ACM. Proceedings of the 11th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing on International Conference on Computer Systems and Technologies*. 2010. p. 167–171.

ELBAMBY, M. S. et al. Content-aware User Clustering and Caching in Wireless Small Cell Networks. In: *11th International Symposium on Wireless Communications Systems (ISWCS)*. 2014. p. 945–949.

FELDMANN, A. Internet clean-slate design: what and why? *ACM SIGCOMM Computer Communication Review*, ACM, v. 37, n. 3, p. 59–64, 2007.

FOUNDATION, O. N. Software-defined networking: The new norm for networks. *ONF White Paper*, v. 2, p. 2–6, 2012.

GHODSI, A. et al. Naming in content-oriented architectures. In: *ACM. Proceedings of the ACM SIGCOMM workshop on Information-centric networking*. 2011. p. 1–6.

GOMES, E. da N. et al. Uso de mecanismo de remoção de mensagens obsoletas em conjunto com políticas de gerenciamento de *Buffer* para as redes tolerantes a atrasos e interrupções. In: . Congresso em Sistemas Sociais e Eventos de Grandes Massas: Ampliando Desafios da Computação (CSBC) joint with the WPerformance conference, 2014.

GUIMARAES, F. de Q. *Modelagem de Ataque de Negação de Serviço nas Redes Centradas em Conteúdo*. 2013.

HAJRI, S. E.; ASSAAD, M. Caching Improvement Using Adaptive User Clustering. *arXiv preprint arXiv:1605.09602*, 2016.

HAN, B. et al. Cellular traffic offloading through opportunistic communications: a case study. In: ACM. *Proceedings of the 5th ACM workshop on Challenged networks*. 2010. p. 31–38.

IETF. *CCN Packet Formats*. 2014. Disponível em: <https://www.ietf.org/proceedings/interim/2014/09/27/icnrg/slides/slides-interim-2014-icnrg-2-3.pdf>.

IOANNOU, A.; WEBER, S. A Survey of Caching Policies and Forwarding Mechanisms in Information-Centric Networking. *IEEE Communications Surveys & Tutorials*, p. 1553–877, 2016.

JACCARD, P. The distribution of the flora in the alpine zone. *New Phytologist*, v. 11, n. 2, p. 37–50, fev. 1912.

JACOBSON, V. et al. Networking named content. In: ACM. *Proceedings of the 5th international conference on Emerging networking experiments and technologies*. 2009. p. 1–12.

JAIN, R. *Art of Computer Systems Performance Analysis Techniques For Experimental Design Measurements Simulation And Modeling*. Wiley Computer Publishing, 1991.

KIM, J.; SHIN, D.; KO, Y.-B. TOP-CCN: topology aware content centric networking for mobile ad hoc networks. In: IEEE. *Networks (ICON), 2013 19th IEEE International Conference on*. 2013. p. 1–6.

KO, H.; LEE, J.; PACK, S. Malm: Mobility-aware location management scheme in femto/macrocell networks. *IEEE Transactions on Mobile Computing*, IEEE, 2017.

KOPONEN, T. et al. A data-oriented (and beyond) network architecture. In: ACM. *ACM SIGCOMM Computer Communication Review*. 2007. v. 37, n. 4, p. 181–192.

KRAUSS, E. *Taxicab geometry: an adventure in non-Euclidean geometry*. Dover Publications, 1987.

KUTSCHER, D. et al. *Information-Centric Networking (ICN) Research Challenges*. 2016.

KUTSCHER, D.; FARRELL, S.; DAVIES, E. *The NetInf Protocol*. 2013.

- LEE, K. et al. Mobile data offloading: How much can wifi deliver? *IEEE/ACM Transactions on Networking (TON)*, IEEE Press, v. 21, n. 2, p. 536–550, 2013.
- LI, Y. et al. Multiple mobile data offloading through disruption tolerant networks. *IEEE Transactions on Mobile Computing*, IEEE, v. 13, n. 7, p. 1579–1596, 2014.
- LINDGREN, A.; PHANSE, K. S. Evaluation of queueing policies and forwarding strategies for routing in intermittently connected networks. In: IEEE. *Communication System Software and Middleware, 2006. Comsware 2006. First International Conference on*. 2006. p. 1–10.
- LIU, X. et al. Information-centric mobile ad hoc networks and content routing: a survey. *Ad Hoc Networks*, Elsevier, v. 58, p. 255–268, 2017.
- MARTINS, B. M.; ALBERTI, A. M. *Redes Centradas na Informação: Uma Comparação de Abordagens*. SBC, 2011.
- MOREIRA, M. D. D. et al. Internet do futuro: Um novo horizonte. In: . *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, 2009.
- NDN. *NDN Project Overview*. 2010. Disponível em: (<https://named-data.net/project/>).
- NETFLIX. *How Netflix Works With ISPs Around the Globe to Deliver a Great Viewing Experience*. 2016.
- OEHLMANN, F. Content-centric networking. *Innovative Internet Technologies and Mobile Communications (IITM)*, v. 43, 2013.
- PANTISANO, F. et al. Cache-aware user association in backhaul-constrained small cell networks. In: IEEE. *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2014 12th International Symposium on*. 2014. p. 37–42.
- PANTISANO, F. et al. Match to Cache: Joint User Association and Backhaul Allocation in Cache-aware Small Cell Networks. In: *IEEE International Conference on Communications (ICC)*. 2015. p. 3082–3087.
- PETERSON, L. E. K-nearest neighbor. *Scholarpedia*, v. 4, n. 2, p. 1883, 2009.
- POULARAKIS, K. et al. Mobile data offloading through caching in residential 802.11 wireless networks. *IEEE Transactions on Network and Service Management*, IEEE, v. 13, n. 1, p. 71–84, 2016.
- QUEK, T. Q. et al. *Small cell networks: Deployment, PHY techniques, and resource management*. Cambridge University Press, 2013.
- REXFORD, J.; DOVROLIS, C. Future internet architecture: Clean slate versus evolutionary research. *Communications of the ACM*, v. 53, n. 9, 2010.

- ROH, J. et al. Femtocell traffic offload scheme for core networks. In: IEEE. *New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on*. 2011. p. 1–5.
- ROSENSWEIG, E. J.; KUROSE, J. Breadcrumbs: Efficient, best-effort content location in cache networks. In: IEEE. *INFOCOM 2009, IEEE*. 2009. p. 2631–2635.
- SOFIA, R. C. User-centric networks. COPELABS/ULHT, 2011.
- STUCKMANN, P.; ZIMMERMANN, R. European research on future internet design. *IEEE Wireless Communications*, October 2009.
- TELECOMINSIGHTS. *LTE: Network densification with Heterogeneous Networks*. 2011. Disponível em: <http://trends-in-telecoms.blogspot.com.br/2011/06/lte-network-densification-with.html>.
- TESSCOTECHNOLOGIES. *Cellular Coverage/Capacity... the Small Cell Revolution*. 2014. Disponível em: <https://www.tessco.com/knowledge-center/2014/articles/cellular-coverage>.
- TROTTER, G. *Terminology for Forwarding Information Base (FIB) based Router Performance*. 2001.
- VENKATARAMANI, A. et al. Mobilityfirst: a mobility-centric and trustworthy internet architecture. *ACM SIGCOMM Computer Communication Review*, ACM, v. 44, n. 3, p. 74–80, 2014.
- WANG, J. A survey of web caching schemes for the internet. *ACM SIGCOMM Computer Communication Review*, ACM, v. 29, n. 5, p. 36–46, 1999.
- WU, Y. et al. Energy-aware cooperative traffic offloading via device-to-device cooperations: an analytical approach. *IEEE Transactions on Mobile Computing*, IEEE, v. 16, n. 1, p. 97–114, 2017.
- XYLOMENOS, G. et al. A survey of information-centric networking research. *IEEE Communications Surveys & Tutorials*, IEEE, v. 16, n. 2, p. 1024–1049, 2014.
- ZHANG, G.; LI, Y.; LIN, T. Caching in information centric networking: A survey. *Computer Networks*, Elsevier, v. 57, n. 16, p. 3128–3141, 2013.
- ZHANG, H. et al. *Uniform information with a hybrid naming (hn) scheme*. 2017.
- ZHANG, L. et al. *Named Data Networking (NDN) Project*. 2010. Disponível em: <http://named-data.net/techreport/TR001ndn-proj.pdf>.

AMBIENTE DE SIMULAÇÃO

A iSim foi implementada e avaliada no simulador OMNeT++, versão 4.6. Trata-se de um simulador de eventos discretos baseado em C++ cujas principais características são robustez, modularização e capacidade de extensão. O OMNeT++ conta com uma documentação bastante extensa e diversos projetos foram desenvolvidos para aumentar as funcionalidades do simulador e facilitar a criação de diferentes cenários, como criação de VANETs, criação de redes móveis *ad hoc*, entre outros.

O *framework* INET¹ é o projeto do OMNeT++ recomendado para criação de redes móveis. O INET foi utilizado, em sua versão 3.2.1, como projeto referência para o desenvolvimento desse trabalho e algumas de suas classes foram utilizadas na implementação da arquitetura e dos algoritmos descritos na Seção 4.

Para realizar uma simulação no OMNeT++ é necessário, basicamente, construir três tipos de arquivos: um arquivo na linguagem NED (do inglês, *Network Description*), que permite ao usuário definir módulos simples, compostos e conexões entre módulos; um arquivo C++, que determina o comportamento de cada módulo simples; e um arquivo de configuração com extensão .ini, na qual são definidos alguns parâmetros como tempo de simulação, modelo de rede utilizado, quantidade de amostras, entre outros.

Na execução desse trabalho foi utilizado o arcabouço do INET, cujos módulos que compõem uma rede sem fio básica já vem implementados. Na Figura A.1 pode ser observado o arquivo .NED que caracteriza um dispositivo móvel, trata-se da composição de diversos módulos simples que representam as estruturas do dispositivo para formação do módulo composto *wirelessHost*. Como já foi citado, os módulos simples são implementados através de uma classe C++ utilizando as bibliotecas básicas do C++ e do próprio simulador. Desta forma, o comportamento dos equipamentos foi definido através da criação de módulos de aplicação.

A Figura A.2 representa as três classes principais que foram implementadas para determinar o comportamento da aplicação dos módulos e suas principais funcionalidades. As mudanças do ponto de vista de configuração da rede, mobilidade, entre outras coisas, foram modificadas nas classes já existentes e no arquivo de inicialização.

¹<https://inet.omnetpp.org/>

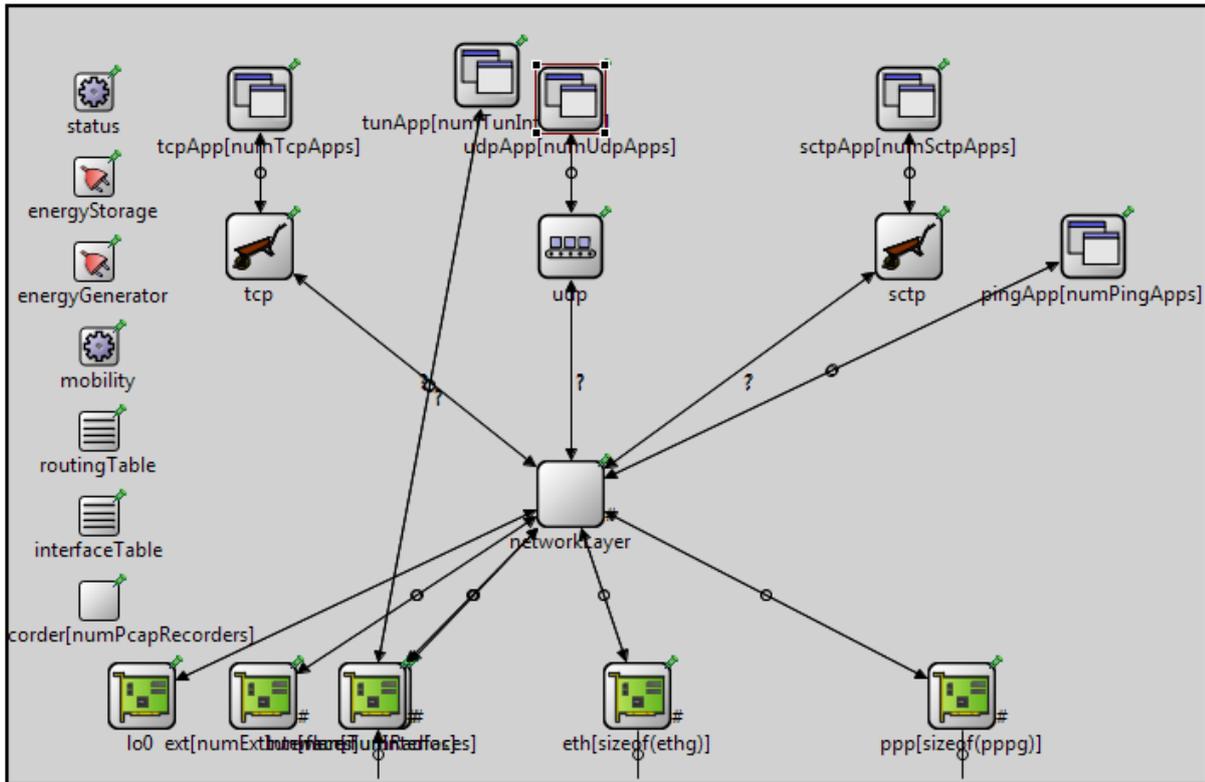


Figura A.1 Composição do módulo *wirelessHost* que representa os dispositivos móveis.

A aplicação do DM mantém cinco métodos principais:

- **zipfDistribution()**: Nesse método é definido um perfil para cada usuário e aplicada a distribuição de *Zipf*. A definição do perfil de cada usuário foi necessária porque ao utilizar a distribuição de *Zipf* com todos os conteúdos disponíveis na rede, não houve diferenciação entre os interesses dos usuários. Desta forma, para cada dispositivo móvel foi estabelecido um intervalo de conteúdos que ele tem interesse. Por exemplo, o dispositivo 1 tem interesse pelos conteúdos de 1 a 50, o dispositivo 2 tem interesse pelos conteúdos de 40 a 90, o dispositivo 3 tem interesse pelos conteúdos de 30 a 80, e assim por diante. Para cada conjunto de conteúdos definido para um usuário foi aplicada a distribuição de *Zipf* para calcular a probabilidade de cada conteúdo ser escolhido.
- **getZipf()**: Toda vez que um usuário deseja fazer uma requisição, é gerado um número aleatório entre 0 e 1 que irá determinar qual conteúdo será requisitado de acordo com as probabilidades calculadas na função *zipfDistribution()*. Por exemplo, se o valor sorteado for 0.5, o vetor de probabilidades será percorrido até encontrar uma probabilidade que seja maior do que 0.5. Em seguida, o índice da probabilidade é escolhido e será gerada uma requisição para o conteúdo de nome: *www.ufba.br/im/dcc/id+INDICE*.

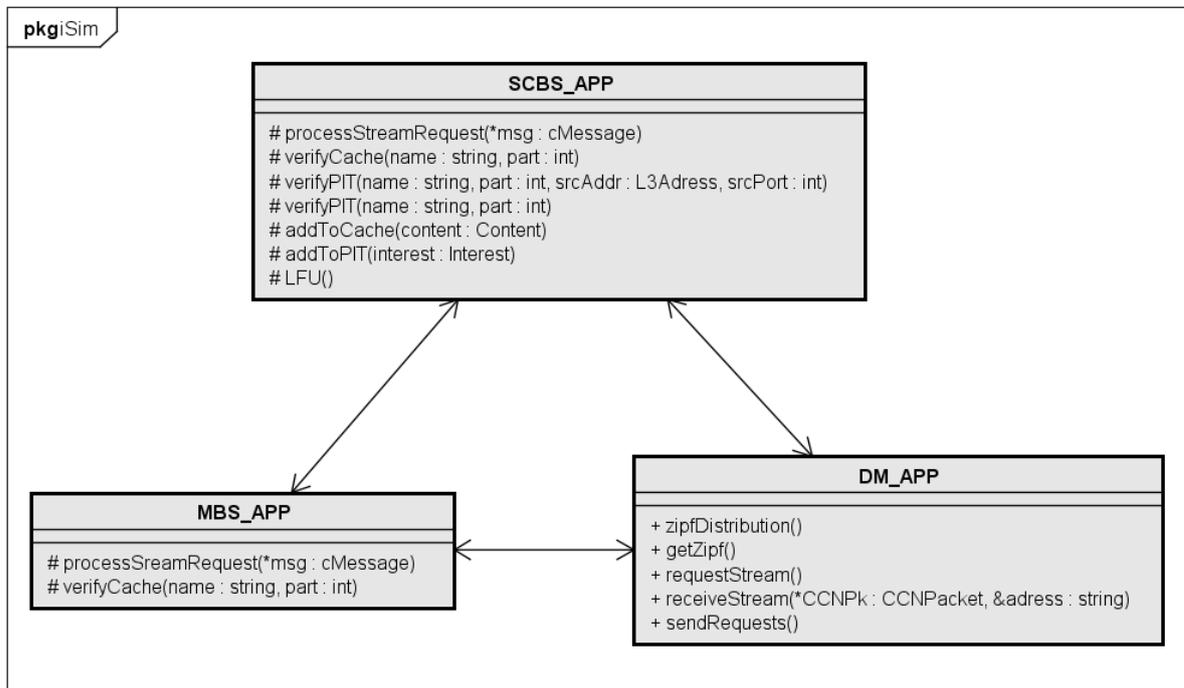


Figura A.2 Diagrama de classes das aplicações criadas.

- **requestStream()**: Esse método representa a criação e envio de um pacote de interesses. Inicialmente é acionada a função *getZipf()* para determinar qual conteúdo será requisitado. Em seguida, é criado um pacote do tipo **CCNPacket**, composto pelos seguintes atributos: nome, quantidade de saltos, tipo, endereço de origem, tipo de nó que está encaminhando (dispositivo móvel ou SCBS) e *payload*. Depois de criado, o pacote é encaminhado e é adicionada uma entrada na tabela de requisições. Essa tabela é um vetor composto pela frequência de vezes que cada conteúdo foi solicitado, sendo que o índice da tabela representa o índice do conteúdo. Desta forma, se $requisitions[30] = 5$, isso quer dizer que o nó solicitou o conteúdo `www.ufba.br/im/dcc/id30` cinco vezes.
- **receiveStream()**: Um DM só pode receber dois tipos de pacotes: pacotes de dados e pacotes de controle. Ao receber um pacote de dados em resposta à uma solicitação, o pacote é descartado. No entanto, toda vez que um DM recebe um pacote de controle é porque ele deve associar-se a uma nova SCBS.
- **sendRequests()**: Como já foi citado, todo usuário mantém uma tabela de requisições composta por uma associação entre os conteúdos requisitados e pela frequência dessas solicitações. Esse método é responsável por criar um pacote de controle contendo a lista de requisições e enviá-lo à MBS. Nos casos em que avaliamos o cenário que não leva em consideração a similaridade entre os usuários, não há necessidade do envio da tabela de requisições. Nessa situação, os usuários são associados às SCBS de acordo com seu identificador. Por exemplo, os usuários de 1 a 10

ficam na SCBS 1, os usuários de 11 a 20 ficam na SCBS 2, e assim sucessivamente.

Além desses métodos implementados, existe o método de inicialização, que define as configurações iniciais dos atributos, o método de finalização, que grava todas as funções estatísticas criadas, e o método *handleMessageWhenUp()*, que define o intervalo entre criação das mensagens através da distribuição de *Poisson*.

Como uma SCBS deve ter capacidade de armazenamento limitada e funções diferentes de um DM, os métodos criados foram:

- **processStreamRequest()**: Uma SCBS recebe apenas dois tipos de pacotes: pacotes de interesse e pacotes de dados. Ao receber um pacote de interesses é feita uma verificação para saber se o conteúdo solicitado está armazenado em *cache*. Se o conteúdo existir, um pacote de dados é enviado ao nó solicitante. Caso contrário, é feita uma verificação na PIT para examinar se aquele conteúdo já foi requisitado por algum nó. Se o conteúdo não foi requisitado, é adicionada uma entrada na PIT com o nome do conteúdo e a interface do nó solicitante é adicionada à lista de interfaces. Caso o conteúdo já tenha sido requisitado por outro nó, não será criada uma nova entrada na PIT, mas a interface do novo solicitante será adicionada à entrada já existente. Assim, é possível observar que a PIT é definida em uma classe C++ com duas variáveis: nome do conteúdo e uma lista de endereços. Para garantir as duas formas de comunicação, quando a comunicação é do tipo indireta, ao receber um pacote de interesses também é verificado se aquele interesse partiu de um DM ou de uma SCBS. Caso o interesse tenha sido solicitado por uma SCBS, é verificada a quantidade de saltos que já foram executados, caso já tenha passado por todas as SCBS, o interesse é encaminhado para a MBS, caso contrário, o interesse é encaminhado para a próxima SCBS. Foi determinado um anel no qual SCBS1 encaminha para SCBS2, que encaminha para SCBS3, que encaminha para a SCBS1, caso a quantidade de saltos esteja dentro do limite. Além disso, ao encaminhar um interesse que chegou de outra SCBS, o endereço da SCBS original é mantido, para que o retorno do pacote de dados seja realizado de forma mais eficiente.
- **verifyCache()**: Esse método compara o nome do conteúdo requisitado com todos os conteúdos que existem no *cache*. Caso encontre um conteúdo, é retornado a posição (índice) do conteúdo no *cache*, caso contrário, retorna -1.
- **verifyPIT()**: Existem dois métodos criados para fazer verificação na PIT. O primeiro método é utilizado quando a SCBS recebe um pacote de interesse, desta forma, a função recebe parâmetros como nome do conteúdo, endereço de origem e porta. Assim, é possível fazer a verificação se há uma entrada na PIT para aquele conteúdo. Se não houver, uma entrada é criada com o nome do conteúdo, o endereço e a porta do nó solicitante. Caso já exista uma entrada, é verificado o nó que solicitou, se for o mesmo nó, o pacote de interesse é descartado. Se forem nós distintos, é criada mais uma entrada com endereço de origem e porta do novo nó solicitante e o pacote de interesse é descartado. O segundo método de verificação da PIT é utilizado ao receber um pacote de dados. Desta forma, verifica-se apenas

se há entradas na PIT para o nome do conteúdo que chegou. Se houver entrada, uma cópia do conteúdo é encaminhada para a lista de nós solicitantes.

- **addToCache()**: Ao receber um pacote de dados como resposta ao envio de um pacote de interesses, o pacote de dados é adicionado ao *cache* e é feita uma verificação para determinar qual a entrada da PIT relacionada àquele pacote. Ao identificar a entrada, é encaminhada uma réplica do conteúdo para todos os nós que solicitaram. Toda vez que uma réplica de um conteúdo mantido em *cache* é encaminhada, a frequência daquele conteúdo é incrementada no *cache*.
- **addToPIT()**: Essa função representa um *push back* para a PIT e é utilizada toda vez que uma entrada deve ser adicionada.
- **LFU()**: Como o *cache* tem tamanho limitado, toda vez que um novo conteúdo deve ser adicionado, é feita uma verificação para determinar se ainda há espaço disponível para armazenamento. Se não houver espaço, é preciso remover um conteúdo existente. O modelo de *cache* implementado é reativo, logo, não é feito nenhum tratamento ou estrutura de prevenção no armazenamento de conteúdos. Assim, todo conteúdo que chega a uma SCBS tem que ser adicionado ao seu *cache*. Para determinar qual conteúdo deve ser removido, foi implementada a política de descarte *Least Frequently Used* (LFU), que descarta os conteúdos requisitados com menos frequência.

Como a métrica de taxa de acerto é calculada em cada SCBS, também são mantidos os valores referentes à quantidade de interesses recebidos, quantidade de *matches* e a taxa de acerto ao longo do tempo.

A MBS reproduz os mesmos processos da SCBS em relação ao recebimento de pacotes de interesse, verificação do *cache* e encaminhamento de pacote de dados. No entanto, o método **processStreamRequest()** é mais complexo pois também faz o processamento dos pacotes de controle que são utilizados para realização do cálculo de similaridade e agrupamento.

Ao receber os pacotes de controle encaminhados pelos dispositivos móveis, a MBS mantém um contador verificando a quantidade de tabelas recebidas e a quantidade de nós que estão ao seu alcance. Desta forma, quando a quantidade de nós que enviaram as tabelas de requisições é igual a quantidade total de nós, é iniciado o cálculo da similaridade.

A similaridade é calculada par a par entre cada nó da rede de acordo com suas tabelas de requisições. Uma vez calculada as similaridades, o par: id de nó + similaridade é armazenado em uma matriz de estrutura de dados do tipo *struct*. Com a matriz de similaridade entre cada nó da rede é calculada a média de cada nó com os demais e verificado o nó que contém maior média de similaridade. O agrupamento é feito entre o nó com maior média de similaridade e os seus *k* vizinhos.

O valor de *k* pode variar de acordo com o cenário utilizado: o valor de *k* pode ser calculado de maneira uniforme para garantir que em cada SCBS tenha aproximadamente a mesma quantidade de dispositivos móveis associados. Ou o valor de *k* pode ser determinado de forma dinâmica a partir de um limiar de similaridade. Por exemplo, se o nó

com ID 15 foi escolhido como o nó com maior média de similaridade, são escolhidos os seus k vizinhos cujo valor de similaridade com ele é igual ou superior a 0.7. Desta forma, os agrupamentos são formados dependendo apenas do nível de similaridade entre os nós.

Se houverem nós que não conseguiram alcançar o limiar de similaridade entre os nós escolhidos como líderes, é feita uma verificação para determinar com qual líder ele é mais similar, independente de conseguir alcançar o limiar.

Ao determinar os membros de um agrupamento, são enviados pacotes de controle cujo nome do conteúdo é o nome da SCBS a qual os nós devem ser associados. Não foi realizada nenhuma determinação para calcular distância física entre os nós e uma SCBS, assim, os agrupamentos foi associados de acordo com a ordem de formação: o grupo 1 foi associado à SCBS 1, o grupo 2 foi associado à SCBS 2 e assim por diante.

Além dos módulos de aplicação criados para determinar as funcionalidades de cada dispositivo, foi criado um arquivo de configuração. O arquivo de configuração determina o tempo de simulação (que variou entre 300 e 600 segundos de acordo com os cenários, cujos primeiros 100 segundos foram considerados tempo transiente), quantidade de amostras, área de simulação e alguns parâmetros como modelo de mobilidade.

Ao final de toda simulação, são gerados vetores e/ou escalares que representam as taxas calculadas durante a simulação. Esses valores podem ser exportados em diferentes formatos para geração e análise dos gráficos.

Para cada resultado foram gerados média e desvio padrão. Nos gráficos em linha o desvio padrão não foi exibido porque o intervalo entre as médias é de apenas um segundo, e a utilização do desvio padrão causava prejuízo na visualização. No entanto, o desvio padrão foi utilizado em todos os gráficos em barra.

Esses resultados foram avaliados de acordo com diferentes cenários, métricas, fatores e parâmetros.

Explorando a afinidade de usuários para descarregamento de dados mais eficiente em redes celulares de pequeno porte

Adriana Viriato Ribeiro¹, Leobino N. Sampaio¹, Artur Ziviani²

¹Programa de Pós-Graduação em Ciência da Computação (PGCOMP)
Instituto de Matemática – Universidade Federal da Bahia (UFBA)
Salvador – BA – Brasil

²Laboratório Nacional de Computação Científica (LNCC)
Petrópolis, RJ - Brasil

adrianaavr@dcc.ufba.br, leobino@ufba.br, ziviani@lncc.br

Abstract. *Network-densification by Small Cell Networks has become the main alternative for mobile network operators when dealing with exponential traffic-growth. A Content-Centric Network can be an alternative to improve data offloading in this scenario. In this paper, we propose a user clustering strategy to take advantage of the similarity function that considers frequency and common-content requisitions to correlate users. Simulation results show that the proposed strategy can be adaptive to different scenarios and is able to increase hit ratio and minimize data offloading by 20% and 30%, respectively.*

Resumo. *A densificação de redes celulares através do uso de estações-base de pequeno porte é uma recente alternativa, adotada por operadoras de telefonia, para lidar com o crescente volume de tráfego de dados. No intuito de alcançar melhor taxa de descarregamento de dados, esse tipo de técnica tem sido associada às Redes Orientadas a Conteúdo. Neste artigo é proposta uma estratégia de agrupamento de usuários que explora a combinação da similaridade entre eles a partir dos conteúdos solicitados em comum e da frequência de solicitação a esses conteúdos. Os resultados de simulação demonstram a capacidade de adaptação da estratégia proposta a diferentes cenários e evidenciam melhoras de 20% em relação à taxa de acerto e de 30% ao descarregamento de dados.*

1. Introdução

A densificação de redes tem sido vislumbrada como uma das alternativas para as operadoras de telefonia móvel lidar com o crescente volume de tráfego de dados decorrente, principalmente, de serviços *streaming* [Aijaz et al. 2013, Bhushan et al. 2014]. Trata-se de uma maior oferta do serviço de telefonia a partir do aumento do número de antenas de transmissão para expandir a cobertura e prover descarregamento de dados (do inglês, *Data offloading*). Essas antenas geralmente são instaladas em pequenas estações-base (do inglês, *Small-Cell Base Station* – SCBS). As SCBS são constituídas por meio da implantação de equipamentos e tecnologias de transmissão de curto alcance, oferecendo, desta forma, redes celulares de pequeno porte (tradução adaptada do termo em inglês, *Small Cell Networks* – SCNs).¹

¹Na literatura, a expressão em inglês *Small cell* refere-se às chamadas “células pequenas” que englobam femto-células, pico-células e microcélulas. Essas redes aumentam a cobertura do serviço prestado pelas operadoras e são utilizadas pelos usuários na redução do congestionamento das macro-células.

As SCNs oferecem serviços complementares aos disponibilizados pelas macro estações-base (do inglês, *Macro Base Station* – MBS) com o propósito de aumentar a cobertura do serviço e atender às demandas de tráfego locais, melhorando os tempos de resposta da rede. Tais objetivos tem sido potencializados através das Redes Orientadas à Conteúdo (ROC) [Jacobson et al. 2009]. Propostas descritas na literatura sugerem o uso de ROCs em cenários de SCN para manter os conteúdos mais populares na SCBS e, conseqüentemente, mais próximos dos consumidores finais [ElBamby et al. 2014, Pantisano et al. 2015, Chen and Kountouris 2015, Hajri and Assaad 2016]. Dado que as políticas clássicas de gerenciamento de *cache*, tais como *Least Frequently Used* (LFU) e *Least Recently Used* (LRU), não levam em consideração as preferências individuais dos usuários, trabalhos mais recentes têm sugerido o uso de técnicas cientes de contexto [Chang et al. 2016] e de agrupamento com o objetivo de reunir usuários com base em características similares e associá-los às SCBS mais apropriadas [Ioannou and Weber 2016]. Dessa forma, espera-se que, através de agrupamentos, os *caches* consigam alcançar melhores taxas de acerto e descarregar a MBS [ElBamby et al. 2014, Pantisano et al. 2015].

Apesar do agrupamento apresentar ganhos em relação ao descarregamento de dados, a maioria dos trabalhos relatados na literatura [ElBamby et al. 2014, Pantisano et al. 2015] que recorrem a esta estratégia leva em consideração apenas a frequência dos conteúdos na definição dos grupos que serão associados às SCBS. O problema, contudo, é que a frequência dos conteúdos não relata de forma precisa o nível de afinidade entre os usuários, visto que, dois consumidores podem ter interesse apenas por um único conteúdo em comum e, ainda assim, serem considerados similares em decorrência da frequência de solicitações e da função de similaridade utilizada.

Este trabalho, portanto, apresenta uma estratégia que explora a formação de grupos de usuários similares em cenários de uso de *caches* em SCN. A técnica de agrupamento proposta se baseia nos interesses dos consumidores de conteúdos e visa obter melhores taxas de descarregamento de dados na MBS através do aumento da taxa de acerto nas SCBS. A estratégia proposta busca identificar grupos que retratem melhor o nível de afinidade entre os usuários considerando, não somente a frequência de solicitações, mas também a quantidade de conteúdos solicitados em comum. Os resultados obtidos demonstraram que quando a similaridade leva em consideração tanto a quantidade de conteúdos em comum quanto a frequência, as taxas de acerto e descarregamento são superiores a quando apenas um fator é utilizado. Além disso, a função de similaridade proposta pode ser adaptada a diferentes cenários, uma vez que a influência da frequência e do conteúdo pode ser ajustada a partir da atribuição de um peso às diferentes partes da equação.

O restante deste artigo está organizado da seguinte forma. A Seção 2 discute brevemente as principais propostas que fazem uso de técnicas de agrupamento em cenários de SCN. A Seção 3 apresenta a estratégia adotada. A Seção 4 descreve o ambiente de avaliação adotado e a Seção 5 analisa os principais resultados experimentais obtidos. Por fim, a Seção 6 conclui o artigo e discute possíveis trabalhos futuros.

2. Gerenciamento de *Cache* em Redes Orientadas à Conteúdo

As Redes Orientadas à Conteúdo [Jacobson et al. 2009] tem como característica a desvinculação entre um conteúdo e seu localizador, com o objetivo de focar em *qual* conteúdo o usuário quer, em vez de *onde* o conteúdo está localizado. Essa abordagem

apresenta novas estruturas que dão suporte ao tráfego de conteúdo nomeado: a *Pending Interest Table* (PIT) armazena as requisições que ainda não foram atendidas, a *Forwarding Information Base* (FIB) mantém as informações de encaminhamento e a *Content Store* é uma estrutura de armazenamento similar a um *cache*. A utilização do *cache* nos próprios equipamentos de rede é um dos diferenciais dessa abordagem e essa característica tem sido utilizada em diferentes contextos.

Existem diversas políticas de gerenciamento de *cache* [Ioannou and Weber 2016] com diversas adaptações para se adequar aos diferentes cenários, no entanto, essas estratégias geralmente não levam em consideração as preferências individuais dos usuários. Desta forma, começaram a ser criadas estratégias que fizessem agrupamentos entre usuários com características similares para que os *caches* conseguissem alcançar melhores taxas de acerto e descarregamento.

Em [ElBamby et al. 2014] é descrita uma solução que faz uma associação entre usuários e *caches* de SCBS. A solução visa melhorar o descarregamento de dados na MBS e diminuir o tempo de resposta ao usuário. A associação entre usuário e SCBS é feita de acordo com a similaridade entre eles, que é calculada utilizando a lei do cosseno em relação à frequência dos conteúdos requisitados. Após a associação, os *caches* das SCBS são atualizados de acordo com um método de aprendizado por regressão também proposto no artigo. A estratégia conseguiu minimizar o atraso em 42% e 27% e obteve um ganho de 280% e 90% quando comparado com o agrupamento feito de forma aleatória e com esquemas sem agrupamentos.

[Pantisano et al. 2015] propõe um algoritmo de associação entre usuários e *caches* com a intenção de fazer uma melhor alocação da largura de banda, aumentar QoS do usuário e reduzir retransmissões. O cálculo de similaridade se baseia em técnicas de Filtro Colaborativo e é realizado levando em consideração a frequência da média de requisições recebidas por cada SCBS, da média de requisições para cada conteúdo e da média de requisições de cada usuário para todos os conteúdos. A intenção é explorar as relações entre as requisições de conteúdo e os usuários, levando em consideração a probabilidade de um conteúdo ser requisitado por outro usuário. Os resultados demonstraram ganhos em relação à alocação de largura de banda e taxa de acerto com melhoras de até 25%.

3. Estratégia de Agrupamento por Interesses Similares

A solução desenvolvida visa aumentar a taxa de acerto nas SCBS para diminuir a sobrecarga na MBS. Para tanto, a solução conta com o uso de agrupamentos baseados na similaridade de usuários em relação aos seus interesses. Nesta seção serão descritos a estrutura da rede e os tipos de pacotes trafegados, a função de similaridade, o algoritmo de agrupamento e as formas de comunicação existentes.

3.1. Estrutura da Rede e Tipos de Pacotes

O cenário é composto por três entidades distintas:

- a MBS é a estação principal, capaz de prover todos os conteúdos que estão disponíveis na rede;
- a SCBS tem capacidade de armazenamento limitada e é utilizada para armazenar réplicas de conteúdo mais próximas dos usuários;

- os dispositivos móveis (DM) são os clientes/assinantes de conteúdo que fazem requisições através do envio de pacotes de interesse.

A comunicação entre os equipamentos é feita através do envio de pacotes. Os pacotes de interesse tem a finalidade de requisitar conteúdos nomeados, os pacotes de dados também são nomeados e comportam o conteúdo propriamente dito, e os pacotes de controle são pacotes de dados cujo conteúdo é utilizado na manutenção da rede.

Cada entidade é responsável por manipular esses pacotes de acordo com a sua funcionalidade. Os dispositivos móveis são responsáveis pelas requisições de dados através do envio de pacotes de interesse. Cada dispositivo contém uma tabela de requisições que é composta pelo nome do conteúdo e a frequência em que foi solicitado. Desta forma, toda vez que um pacote de interesse é enviado, a tabela de requisições é atualizada. Essa tabela é enviada à MBS através de um pacote de controle e é utilizada para realização do cálculo de similaridade e agrupamento entre os dispositivos.

As principais funções da MBS são as de processar pacotes de interesse e controle. Ao receber um pacote de interesse, a MBS faz uma busca no *cache* para encontrar o pacote de dados referente àquele interesse e enviá-lo ao nó solicitante. A MBS é capaz de prover todos os conteúdos disponíveis na rede. Além disso, é responsável por realizar o cálculo de similaridade e a formação dos agrupamentos, conforme a Seção 3.2.

A SCBS pode receber e solicitar pacotes de interesse e dados. Ao receber um pacote de interesse, verifica no *cache* se existe um pacote de dados relacionado àquele conteúdo, se existir, encaminha o pacote de dados para o nó solicitante. Caso contrário, o nome do conteúdo e o nó solicitante são adicionados à PIT. Em seguida, a SCBS solicita o conteúdo à MBS e, ao recebê-lo, adiciona no seu *cache* e envia para o nó requisitante. Vale ressaltar que a ROC faz agregação de requisições, portanto, se mais de um nó solicitar o mesmo conteúdo, apenas uma entrada na PIT é adicionada com o nome do conteúdo e a lista de endereços dos nós solicitantes. Desta forma, o pacote de interesse é enviado uma única vez através da rede e o pacote de dados é replicado para todos os nós assinantes.

De acordo com as diferentes funcionalidades de cada entidade, é possível observar que o envio de pacotes de dados pode ocorrer de três maneiras distintas, como pode ser observado na Figura 1:

- a MBS envia o conteúdo diretamente para o DM;
- a SCBS tem o conteúdo armazenado em *cache* e envia para o DM;
- a SCBS não tem o conteúdo armazenado e a MBS envia o conteúdo para ela.

Cada uma dessas situações ocorre de acordo com a configuração da rede em determinado instante. Antes da formação dos grupos, todos os pacotes de dados são enviados pela MBS. Após a formação dos agrupamentos, os usuários são servidos pelas SCBS. A MBS só enviará conteúdos para a SCBS quando a SCBS não tiver o conteúdo solicitado armazenado em *cache*. Em uma ROC, um pacote de dados só é enviado em resposta a um pacote de interesse, logo, toda vez que a SCBS necessitar de um conteúdo que não existe no seu *cache*, deverá enviar uma requisição através do envio de um pacote de interesse.

Como foi citado anteriormente, além dos pacotes de interesse e de dados, existem os pacotes de controle. Os pacotes de controle são utilizados para realizar a formação dos agrupamentos. Os primeiros pacotes de controle enviados contém as tabelas de

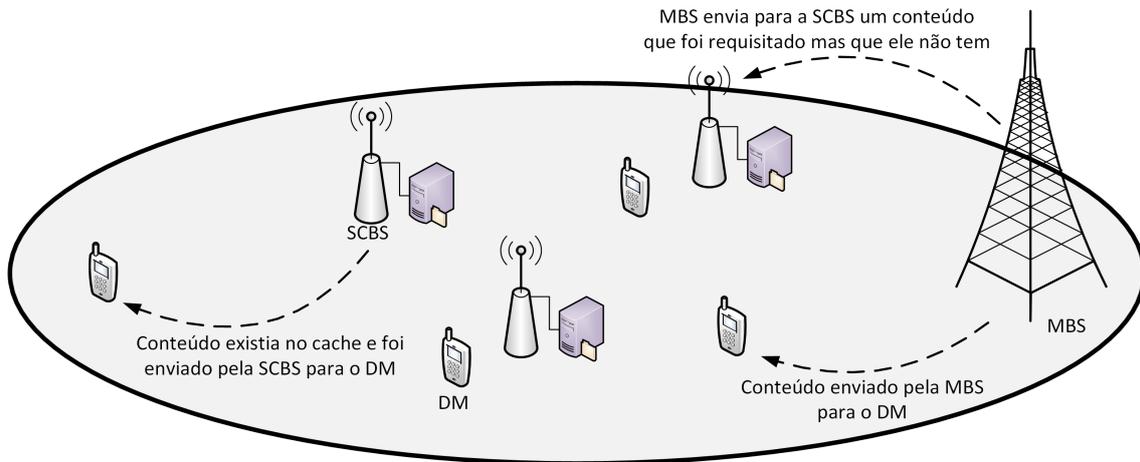


Figura 1. Estrutura da SCN utilizada.

requisições dos usuários e são utilizados como insumo para o cálculo de similaridade e formação dos agrupamentos.

3.2. Função de Similaridade

A função de similaridade tem o objetivo de quantificar a semelhança entre diferentes usuários de acordo com seus interesses.

A similaridade entre os usuários é medida através da Equação 1. A primeira parte representa o Índice de Jaccard [Jaccard 1912], comumente utilizado para medir a similaridade através das características semelhantes e distintas entre dois indivíduos. Ela foi utilizada para medir a similaridade de acordo com os conteúdos em comum que dois nós (i, j) tem interesse. Já a segunda parte é calculada utilizando a lei de similaridade do cosseno em relação à frequência de requisições de conteúdos em comum.

Desta forma, C_i e C_j representam os conjuntos de conteúdos que foram solicitados por i e por j , respectivamente, e F_i e F_j representam a frequência, em i e em j , em que os conteúdos comuns foram requisitados. O β define um peso entre as partes da equação e permite que seja avaliado o impacto delas no cálculo de similaridade.

$$S_{i,j} = \beta \left(\frac{|C_i \cap C_j|}{|C_i \cup C_j|} \right) + (1 - \beta) \left(\frac{F_i \cdot F_j}{\|F_i\| \|F_j\|} \right) \quad (1)$$

Vale ressaltar, que não estamos avaliando a similaridade entre usuário e *cache*, nem entre dois conteúdos, mas a similaridade entre usuários de acordo com seus interesses por conteúdo. Além disso, neste trabalho, o cenário “com similaridade” representa o uso da função de similaridade na formação dos agrupamentos. Enquanto no cenário “sem similaridade” os membros são adicionados de acordo com sua identificação. Por exemplo, os nós de 1 a 10 formam o primeiro agrupamento, os nós de 10 a 20 formam o segundo, e assim por diante. Isso não quer dizer que não exista similaridade entre os usuários, mas que a similaridade não está sendo levada em consideração na formação dos grupos.

3.3. Algoritmo de Agrupamento

O algoritmo de agrupamento (Algoritmo 1) é executado na MBS e tem a função de estabelecer as associações entre nós que tenham interesses similares. Esse trabalho avalia o efeito de apenas uma execução do algoritmo.

Os agrupamentos são determinados utilizando as tabelas de requisições dos nós como insumo. A partir de um determinado instante de tempo t , cada usuário envia uma única vez um pacote de controle contendo a sua tabela de requisições. Cada tabela de requisição recebida pela MBS é adicionada a uma matriz $N \times C$, sendo N a quantidade de nós da rede e C a quantidade de conteúdos disponíveis. Ao receber a tabela de requisição, a MBS verifica o ID do nó que enviou a requisição e adiciona na linha da matriz referente àquele ID. Depois de receber as requisições de todos os nós da rede, é inicializado o cálculo da similaridade. Para saber quando todas as requisições foram recebidas, foi utilizado um contador inicializado em zero e incrementado até a quantidade de nós.

Algoritmo 1: Algoritmo de Agrupamento

Entrada: Matriz $N \times C$

```
1 início
2   para cada  $i \in N$  faça
3     para cada  $j \in N$  faça
4       para cada  $w \in C$  faça
5         Calcule os valores de  $C_i \cap C_j$ ,  $C_i \cup C_j$ ,  $F_i \cdot F_j$  e  $\|F_i\| \|F_j\|$ ;
6       fim
7     Calcule a similaridade entre  $i$  e  $j$ ;
8     Constrói uma matriz  $N \times N$  com a similaridade dos nós;
9   fim
10  fim
11  para cada  $z \in s$  faça
12    Aplique o método de ordenação por seleção em cada linha da matriz;
13    Calcule a média aritmética entre os  $k$  vizinhos mais similares de cada
        nó;
14    Identifique o nó com maior média de similaridade;
15    Forme um agrupamento com o nó com maior média e seus  $k$  vizinhos
        mais similares;
16    Envie mensagem de controle para os nós do agrupamento formado
        identificando o servidor de distribuição que eles devem se associar;
17    Zere os valores de similaridade dos nós que já foram agrupados;
18  fim
19 fim
```

Saída: s agrupamentos, sendo s a quantidade de SCBS

O cálculo da similaridade é realizado par a par entre cada nó da rede, gerando uma matriz $N \times N$. O índice de cada linha da matriz corresponde ao ID de um dos nós que compõe a rede e a linha representa a similaridade entre um nó e os demais. A linha 1, por exemplo, equivale à similaridade entre o nó 1 e todos os outros nós da rede $\{1, 2, 3, \dots, N\}$. Além disso, a diagonal principal dessa matriz é uma diagonal nula, pois representa a similaridade entre um nó e ele mesmo.

Após a formação da matriz contendo a similaridade entre todos os nós da rede, é feita uma ordenação decrescente entre os elementos de cada linha da matriz, para que os nós sejam ordenados de acordo com a similaridade. Cada elemento da matriz é uma *struct* formada pelo ID do nó e pela similaridade, isso garante que, mesmo após a ordenação, sejam mantidos os IDs dos nós que mudaram de posição.

Com a matriz de similaridade ordenada, o próximo passo é calcular a média de similaridade entre um nó e os seus k vizinhos mais próximos (do inglês, k *Nearest Neighbor*), obtendo assim a média de similaridade de cada nó da rede. Em seguida, o nó com maior média é identificado e inicia-se a formação do agrupamento.

O agrupamento é formado pelo nó que tem a maior média e os seus k primeiros vizinhos, que representam os vizinhos com maior similaridade. O valor de k é calculado por $(N/S)-1$. Sendo que N é a quantidade de nós, S é a quantidade de SCBS e a subtração por 1 é justificada pelo fato de que o primeiro nó do agrupamento já foi determinado (nó com maior média de similaridade). Esse cálculo garante que os nós sejam distribuídos de maneira uniforme entre as SCBS. A distribuição uniforme foi feita com o intuito de equilibrar a quantidade de usuários por SCBS, para que a quantidade de usuários por antenna não interferisse nos resultados do efeito da equação de similaridade. Por exemplo, supõe-se que uma antenna com dois usuários conectados teria uma taxa de acerto maior do que uma antenna com 20 nós conectados, ainda que fossem bastante similares.

Ao identificar os membros do agrupamento, a MBS envia um pacote de controle para cada usuário identificando a SCBS que ele deve se associar. A partir desse momento, os DM fazem requisições apenas para as SCBS, deixando de requisitar à MBS.

No Algoritmo 1 a complexidade do cálculo de similaridade é de $O(CN^2)$ enquanto a realização do agrupamento tem complexidade $O(N^3)$. Sendo C a quantidade de conteúdos e N a quantidade de usuários.

3.4. Formas de Comunicação

Após a formação dos agrupamentos, existem duas possibilidades de comunicação. Partindo do pressuposto de que um DM envia um conteúdo nomeado para a SCBS1 e o único nó da rede que possui o pacote de dados associado a esse conteúdo é a MBS, é possível que a comunicação ocorra de duas maneiras:

- Comunicação Direta, conforme Figura 2: ao receber um pacote de interesse de um dispositivo móvel (1), SCBS1 verifica a sua *cache* (2) e percebe que não existe um conteúdo relacionado ao interesse. Desta forma, é adicionada uma entrada na PIT (3) e o pacote de interesse é encaminhado diretamente a MBS (4). A MBS acessa o *cache* para obter o conteúdo solicitado (5,6) e envia o pacote de dados para SCBS1 (7), SCBS1 armazena o pacote de dados no seu *cache* (8), apaga a entrada da PIT (9) e encaminha o pacote de dados para o dispositivo móvel solicitante (10);
- Comunicação Indireta, conforme Figura 3: o DM envia um pacote de interesse para SCBS1 (1), como SCBS1 não tem o conteúdo referente ao interesse, ela encaminha o interesse à SCBS2 (2) e a SCBS2 realiza o mesmo processo de verificação e encaminhamento (3). Ao chegar na SCBS3, o pacote de interesse já percorreu todos os nós da rede daquele nível, portanto, a única entidade que pode resolver

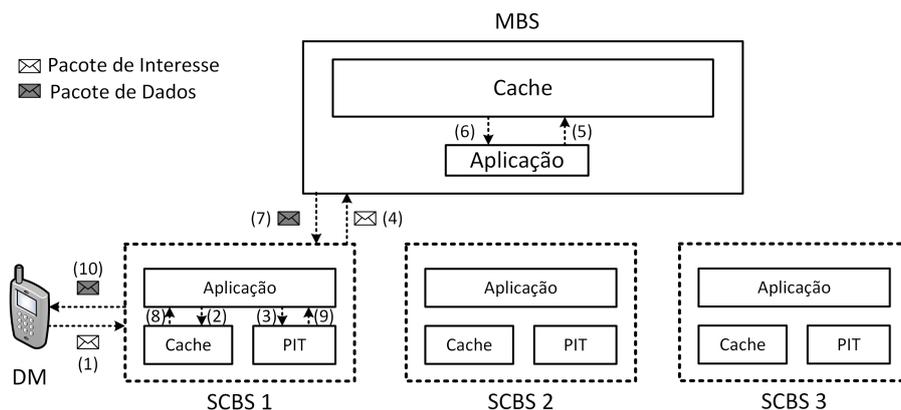


Figura 2. Comunicação Indireta.

o interesse é a MBS. Logo, o servidor SCBS3 envia o pacote de interesse para a MBS (4) e a MBS envia o pacote de dados diretamente para SCBS1 (5). Isso é possível porque mesmo o pacote sendo encaminhado entre as SCBS, o endereço da primeira SCBS solicitante é mantido nas mensagens. Por fim, a SCBS1 encaminha o pacote de dados para o DM (6).

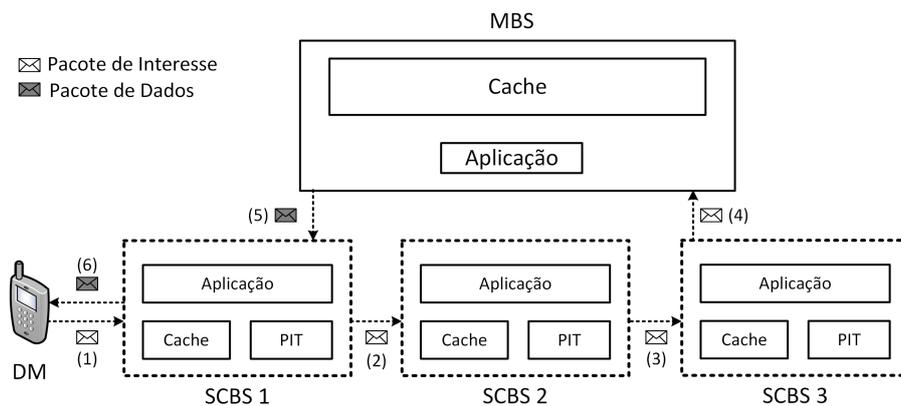


Figura 3. Comunicação Direta.

4. Experimentos

Para realizar os experimentos, foi necessário determinar cenários de avaliação, métricas, fatores e parâmetros. Todos esses aspectos serão abordados nessa seção.

4.1. Cenários, Métricas, Fatores e Parâmetros

Os fatores utilizados podem ser vistos na primeira coluna da Tabela 1. A similaridade representa a utilização da equação de similaridade e o β representa o peso usado na equação. A comunicação retrata a direção no encaminhamento dos pacotes, o tamanho do *cache* indica espaço disponível para armazenamento, e o α representa o parâmetro da Distribuição de Zipf, que é responsável por definir a popularidade dos conteúdos.

De acordo com esses fatores foram definidos três cenários de avaliação: o Grupo 1 compara o impacto do parâmetro β e do Tamanho da *Cache* em relação à taxa de acerto;

Tabela 1. Fatores e Cenários de Avaliação.

Fatores	Grupo 1	Grupo 2	Grupo 3
Similaridade	Sim - Não	Sim - Não	Sim - Não
Comunicação	Direta	Direta	Indireta - Direta
Tamanho do <i>Cache</i>	10 - 20 - 30	20	20
β (Funç. Similaridade)	0 - 0.5 - 1	0 - 0.5 - 1	0.5
α (Distr. Zipf)	1	0.6 - 0.8 - 1	1

o Grupo 2 avalia o impacto do parâmetro Zipf e do parâmetro β na taxa de acerto; e, por fim, o Grupo 3 analisa a utilização da similaridade e da hierarquia em relação à taxa de acerto e ao descarregamento de dados.

Os experimentos foram avaliados de acordo com duas métricas: taxa de acerto na SCBS e descarregamento de dados na MBS. O descarregamento de dados visa medir a quantidade de requisições que deixaram de ser feitas na MBS e pode ser representado pela Equação 2, na qual $rMax$ representa os interesses que foram requisitados quando todas as requisições eram feitas diretamente na MBS e $rMin$ são as requisições recebidas após a formação dos agrupamentos.

$$off = 100 \times \frac{rMax - rMin}{rMax} \quad (2)$$

Já a taxa de acerto mede a quantidade de vezes que uma SCBS recebeu um pacote de interesse e tinha o pacote de dados no *cache*. Essa taxa é medida de acordo com a Equação 3, na qual $match$ representa os interesses que foram resolvidos ao chegar na SCBS e $iRec$ o total de todos os interesses que a SCBS recebeu. A taxa de acerto total é calculada através da média aritmética da taxa de acerto das três SCBS.

$$hr = 100 \times \frac{match}{iRec} \quad (3)$$

4.2. Ambiente de Simulação

A solução proposta foi implementada no simulador OMNeT++ utilizando o *framework* INET, conforme parâmetros da Tabela 2.

A área simulada é de 600x800m onde estão distribuídos 30 dispositivos móveis que se movimentam de acordo com o modelo de mobilidade *Mass Mobility* e requisitam conteúdo seguindo o modelo de Poisson. O meio físico sem fio tem propagação do tipo *Constant Speed Propagation*, frequência de 2.4GHz, perda do tipo *Free Space Path Loss* e ruído de -110dBm, todos esses parâmetros já vieram especificados no INET. Foi criada uma MBS que alcança toda área de simulação e foram criadas três SCBS cujo alcance é de 300m. Na simulação, cada SCBS é associada a 10 dispositivos móveis, de forma imperativa, de acordo com a estratégia utilizada. Existem 100 conteúdos disponíveis na rede e, para determinar perfis diferentes para os usuários, foi selecionado um conjunto de conteúdo para cada dispositivo. Para cada subconjunto de conteúdo é aplicado o modelo de popularidade de *Zipf*. O tamanho do *cache* da SCBS varia entre 10% e 30% da

Tabela 2. Parâmetros da Simulação.

Parâmetro	Valor
Quantidade de dispositivos móveis na rede	30
Quantidade de MBS	1
Quantidade de SCBS	3
Quantidade de conteúdos disponíveis	100
Política de descarte de <i>cache</i>	LFU
Modelo de Mobilidade	<i>MassMobility</i>
Tipo de Propagação	<i>ConstantSpeedPropagation</i>
Frequência	2.4GHz
Perda	<i>FreeSpacePathLoss</i>
Ruído	-110dBm

quantidade de conteúdos e a estratégia de descarte utilizada é a LFU. O parâmetro β , que determina o peso na equação de similaridade, é variado entre 0 e 1 e são comparados cenários com comunicação direta e indireta. Cada experimento foi executado 10 vezes com um Intervalo de Confiança de 95%.

5. Análise dos Resultados

Nesta seção são apresentados os resultados e as análises dos diferentes cenários em relação à taxa de acerto e de descarregamento de dados de acordo com as variações de fatores apresentadas na Tabela 1.

5.1. Grupo 1: Relação entre Tamanho do *Cache*, β e Taxa de Acerto

A Figura 4 expõe o comportamento da taxa de acerto em relação ao uso da função de similaridade e da variação do tamanho do *cache*. Em relação ao uso da equação de similaridade, para todos os tamanhos de *cache* (10, 20, 30), é possível observar que quando apenas a quantidade de conteúdos em comum é levada em consideração ($\beta = 1$), a taxa de acerto é superior a quando apenas a frequência é utilizada ($\beta = 0$). Isso pode ser justificado devido ao fato de que ao contabilizar apenas a frequência, é possível que dois usuários sejam considerados bastante similares mesmo tendo um único conteúdo em comum. Por exemplo, quando dois nós tem um conteúdo em comum, ainda que requisitado uma única vez, de acordo com a função da similaridade do cosseno, esses dois nós terão similaridade igual a 1. Dessa forma, como o Índice de Jaccard leva em consideração a relação entre os conteúdos requisitados em comum e todos os conteúdos requisitados pelos dois nós, os agrupamentos baseados nessa relação de similaridade conseguem alcançar resultados superiores. Apesar disso, os melhores resultados foram obtidos quando ambos ($\beta = 0.5$) foram utilizados, evidenciando a importância de uma equação de similaridade que considere esses dois fatores.

Vale ressaltar que ao levar em consideração apenas a frequência de conteúdos ($\beta = 0$) calculada pela similaridade do cosseno, não estamos replicando o cenário descrito em [ElBamby et al. 2014], visto que o autor calcula similaridade usuário-*cache* e neste trabalho é calculada a similaridade usuário-usuário.

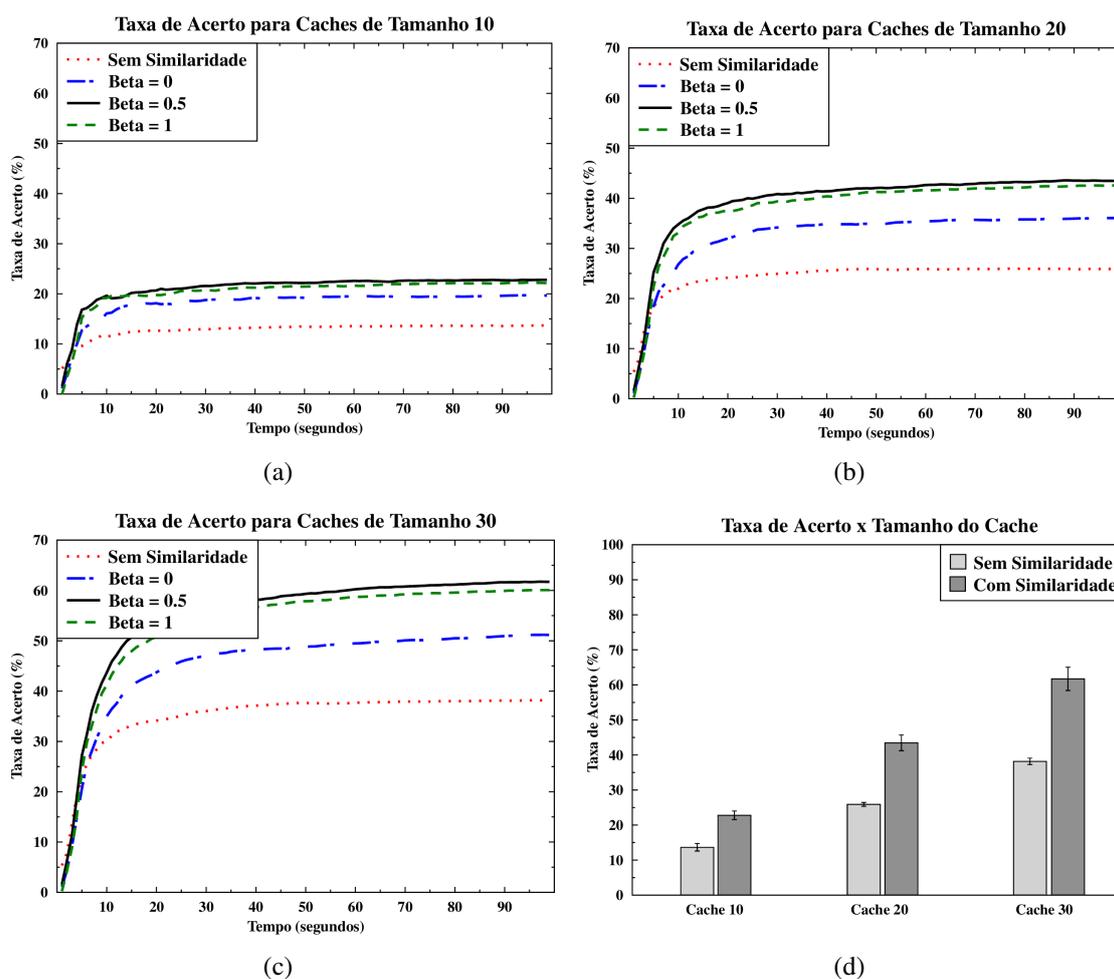


Figura 4. Relação entre o Tamanho do *Cache* e a Taxa de Acerto.

A distinção entre as taxas de acerto dos gráficos das Figuras 4(a) 4(b) 4(c) se dá devido ao aumento do tamanho do *cache*. No entanto, ao relacionar os tamanhos do *cache* 10 e 20, conforme a Figura 4(d), observa-se que a taxa de acerto obtida sem similaridade com um *cache* = 20 alcança valores próximos ao melhor valor obtido com uso de similaridade em um *cache* = 10. Ao se comparar os *caches* de tamanho 20 e 30, percebe-se que com o uso da similaridade em um *cache* = 20 é possível obter valores superiores ao cenário sem similaridade em um *cache* = 30. Isso evidencia que a utilização de agrupamentos levando em consideração a similaridade dos usuários torna, de fato, o gerenciamento de *cache* mais eficiente e possibilita economia de recursos, visto que os mesmos valores de taxa de acerto podem ser obtidos com *caches* de tamanho inferior.

5.2. Grupo 2: Relação entre o parâmetro α da Lei de Zipf e a Taxa de Acerto

A Figura 5(a) evidencia que mesmo com diferentes valores para o parâmetro Zipf, o comportamento da função de similaridade permanece o mesmo: os resultados com valor de $\beta = 0.5$ ou $\beta = 1$ são superiores a quando considera-se apenas a frequência. No entanto, é possível observar na Figura 5(b) que, apesar do comportamento da função de similaridade permanecer o mesmo, os valores de taxa de acerto alcançados são distintos. Quanto menor o valor do parâmetro Zipf, maior a taxa de acerto obtida. Isso acontece porque

quando o parâmetro Zipf é menor, há uma maior quantidade de conteúdos populares que, conseqüentemente, são requisitados mais vezes. Como a política de descarte utilizada é a LFU, os conteúdos mais populares ficam armazenados por mais tempo, satisfazendo as requisições e culminando no aumento da taxa de acerto.

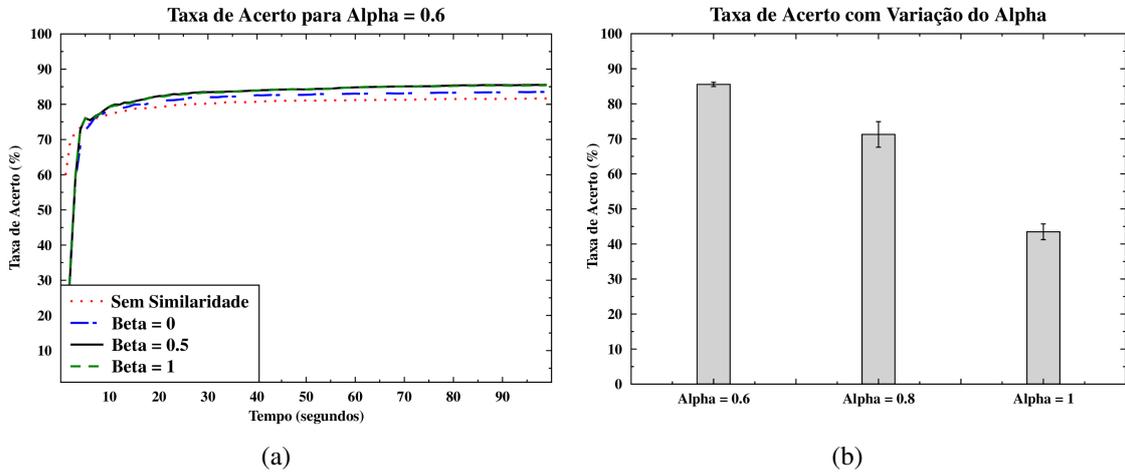


Figura 5. Efeito do parâmetro Zipf na Taxa de Acerto.

5.3. Grupo 3: Relação entre Similaridade e Comunicação na Taxa de Acerto e no Descarregamento de Dados

Na Figura 6(a) é possível observar o comportamento entre similaridade e comunicação em relação à taxa de descarregamento. A taxa de descarregamento de aproximadamente 22% obtida no cenário sem similaridade e com comunicação direta, decorre da adição das três SCBS. O cenário com comunicação indireta e com similaridade consegue obter resultados superiores ao cenário com comunicação direta e sem similaridade, evidenciando que a utilização de agrupamentos influencia mais no descarregamento do que a forma de comunicação. Os melhores resultados foram obtidos com comunicação indireta e similaridade.

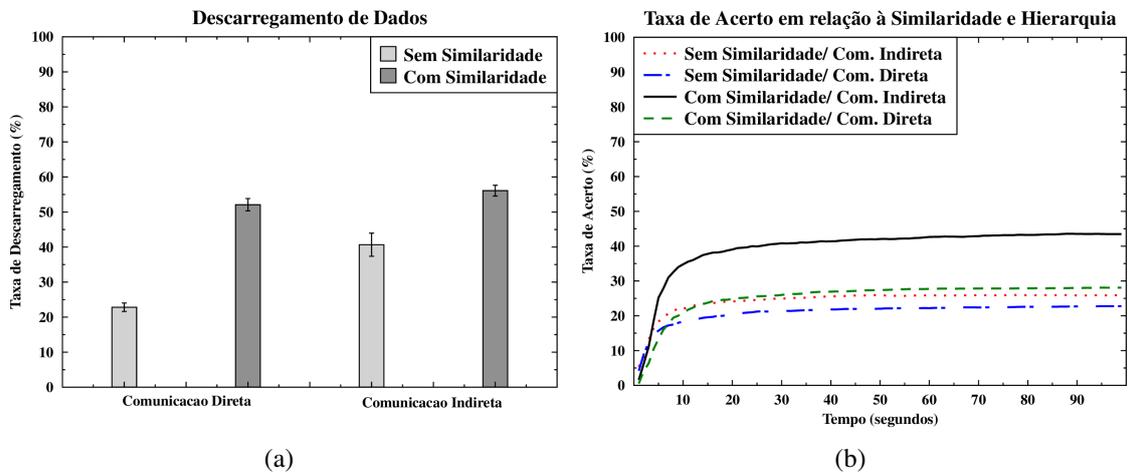


Figura 6. Influência da Similaridade e da Hierarquia na Taxa de Acerto e de Descarregamento de Dados.

Em relação a avaliação da taxa de acerto nos cenários com e sem similaridade e comunicação direta e indireta, é possível observar, de acordo com a Figura 6(b), que os cenários de comunicação direta obtém taxa de acerto superior aos cenários de comunicação indireta. Isso acontece porque quando a comunicação é indireta, as SBCS recebem pacotes de interesse que não pertencem aos usuários do grupo que está associado a elas, portanto, a probabilidade delas terem o conteúdo armazenado em *cache* é menor. Apesar disso, o grupo de usuários associado a cada SCBS não sofre nenhum impacto decorrente dessa diminuição da taxa de acerto, visto que não há substituição em *cache* dos conteúdos que foram solicitados por outra SCBS. No entanto, há um aumento no tempo de resposta ao usuário decorrente da comunicação indireta, visto que pode aumentar a quantidade de saltos para alcançar o conteúdo.

Na Figura 7 foi feito um experimento com apenas 30 conteúdos, no qual todos os usuários requisitam os conteúdos disponíveis com diferentes frequências, logo, apenas a parte da equação que representa a similaridade do cosseno irá refletir a similaridade dos usuários. Assim, é possível observar que, contabilizando apenas os conteúdos em comum ($\beta = 1$), a taxa de acerto é inferior a quando a frequência é considerada ($\beta = 0$). Isso acontece porque como todos os usuários requisitam todos os conteúdos, o Índice de Jaccard não consegue detectar a similaridade entre os usuários, pois os conjuntos de conteúdo serão sempre iguais. Desta forma, ao utilizar $\beta = 1$, os resultados são semelhantes a um cenário sem similaridade.

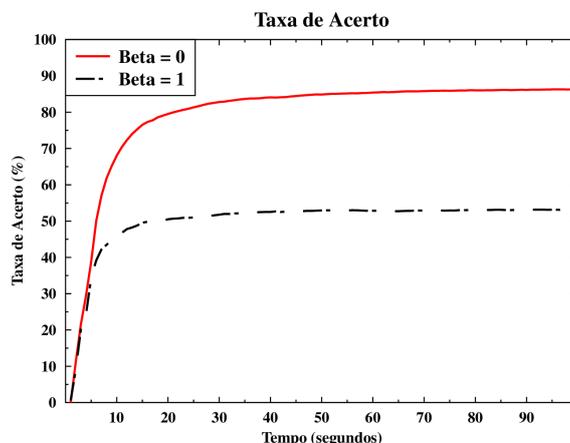


Figura 7. Representação da adaptatividade da função de similaridade.

Também já foi exposto que quando dois nós tem um único conteúdo em comum solicitado uma única vez, de acordo com a similaridade do cosseno, esses nós são caracterizados com grau de similaridade máximo. Dessa forma, fica evidente a importância da utilização de ambas as partes da função de similaridade (Equação 1) durante a formação dos agrupamentos, tornando o cálculo de similaridade adaptativo às características do ambiente e capaz de superar lacunas de cada indicador de similaridade.

6. Conclusão e Trabalhos Futuros

Os resultados obtidos demonstraram que, ao utilizar a função de similaridade proposta, foi possível obter valores de taxa de acerto superiores ao cenário aleatório mesmo com tamanho de *cache* inferior, evidenciando a economia de recursos e aumento da taxa de acerto.

Ao utilizar a comunicação indireta há um aumento no tempo de resposta ao usuário e degradação da taxa de acerto nas SCBS. Apesar disso, os melhores resultados para descarregamento de dados na MBS envolvem a comunicação indireta e o uso de similaridade.

Como trabalhos futuros, pretende-se aumentar a quantidade de conteúdos e de nós, também levando em consideração os nós que não se conectam a todas as SCBS. Além disso, será avaliada a entrada e saída de nós no ambiente para verificar a necessidade de recalcular a similaridade entre os usuários e a distribuição dos nós nas SCBS não será realizada uniformemente.

Agradecimentos

Os autores agradecem o apoio da CAPES e do CNPq.

Referências

- Aijaz, A., Aghvami, H., and Amani, M. (2013). A Survey on Mobile Data Offloading: Technical and Business Perspectives. *IEEE Wireless Communications*, 20(2):104–112.
- Bhushan, N., Li, J., Malladi, D., Gilmore, R., Brenner, D., Damnjanovic, A., Sukhavasi, R. T., Patel, C., and Geirhofer, S. (2014). Network Densification: The Dominant Theme for Wireless Evolution into 5G. *IEEE Communications Magazine*, 52(2):82–89.
- Chang, Z., Gu, Y., Han, Z., Chen, X., and Ristaniemi, T. (2016). Context-Aware Data Caching for 5G Heterogeneous Small Cells Networks. In *Communications (ICC), 2016 IEEE International Conference on*, pages 1–6. IEEE.
- Chen, Z. and Kountouris, M. (2015). Cache-enabled Small Cell Networks with Local User Interest Correlation. In *IEEE 16th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 680–684.
- ElBamby, M. S., Bennis, M., Saad, W., and Latva-Aho, M. (2014). Content-aware User Clustering and Caching in Wireless Small Cell Networks. In *11th International Symposium on Wireless Communications Systems (ISWCS)*, pages 945–949.
- Hajri, S. E. and Assaad, M. (2016). Caching Improvement Using Adaptive User Clustering. *arXiv preprint arXiv:1605.09602*.
- Ioannou, A. and Weber, S. (2016). A Survey of Caching Policies and Forwarding Mechanisms in Information-Centric Networking. *IEEE Communications Surveys & Tutorials*, pages 1553–877.
- Jaccard, P. (1912). The distribution of the flora in the alpine zone. *New Phytologist*, 11(2):37–50.
- Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H., and Braynard, R. L. (2009). Networking Named Content. In *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, pages 1–12.
- Pantisano, F., Bennis, M., Saad, W., and Debbah, M. (2015). Match to Cache: Joint User Association and Backhaul Allocation in Cache-aware Small Cell Networks. In *IEEE International Conference on Communications (ICC)*, pages 3082–3087.

Affinity-Based User Clustering for Efficient Edge Caching in Content-Centric Cellular Networks

Adriana Viriato Ribeiro¹, Leobino N. Sampaio,¹ and Artur Ziviani²

¹Universidade Federal da Bahia (UFBA), Salvador, BA, Brazil

²National Laboratory for Scientific Computing (LNCC), Petrópolis, RJ, Brazil

Abstract—Network densification by small cell networks has become the main alternative for mobile network operators to deal with an ever-increasing traffic growth. In this context, Content-Centric Networking (CCN) emerges as an alternative to improve data offloading by the promotion of in-network caching. In this paper, we propose a user clustering scheme that takes advantage of the affinity among users with respect to the frequency of content requisition and the common interest for content for a more efficient edge caching. The proposed strategy is evaluated in a varied set of scenarios, including different cache sizes, communication models, or concentration levels of content popularity. Simulation results show that the proposed strategy increases both cache hit ratio and data offloading in Content-Centric Cellular Networks.

Index Terms—Data Offloading, Clustering, Content-Centric Networking, Mobile Networks, 5G

I. INTRODUCTION

Cellular network densification is the main strategy to save backhaul bandwidth in recent years [1]. Essentially, high density cellular networks involve the deployment of low-cost antennas in wireless scenarios to complement the transmission capacity of current 5G cellular networks [2]. To this aim, small cell networks coupled with macrocell base stations favor traffic distribution and alleviate backhaul bottlenecks. Despite recent advances, the research in high density 5G cellular networks is still in its initial stage of development [3]. There is a number of open issues that must be addressed before the telecommunication industry fully adopts cellular network densification. One challenging aspect is the coordination of the interference caused by adjacent base stations, which limits the overall transmission capacity of the antennas [1].

In this context, edge caching in cellular networks emerges as a consequence of the performance constraints that the process of cellular network densification still faces. Cache-enabled base stations reduce the distance between users and the data consumed by their applications, which obviously improves the network throughput and the general quality of experience (QoE) perceived by users [4]. Yet, content-oriented traffic distribution is a notable advantage for scenarios of dense cellular networks, since applications can fetch contents locally, instead of retrieving them from the macrocells. Such benefits thus render the Information-Centric Networking (ICN) concept [5] highly suitable for the low-power and short-range setups of dense cellular networks, in particular to support data offloading at the network edges [6]. Moreover, ICN is

a collaborative approach in its nature, where local caches store contents from prior requests to serve them further to other similar requests. Consequently, ICN adequately fulfills requirements of cellular networks deployed in scenarios featured by numerous users that likely share popular contents, e.g., universities, music festivals, and sport arenas.

In particular, Content-Centric Networking (CCN) [7] is a specialization of ICN that specifically promotes in-network content caching. In a CCN, cache-equipped network nodes store and forward contents through Content Store (CS) data structures that seek to make available to users the most popular contents, whereas limited to the CS capacity. The contents on these CS data structures are maintained by replacement policies, e.g., First In First Out (FIFO), Least Recently Used (LRU), or Least-Frequently Used (LFU) [5]. Given that such policies disregard the individual user preferences, these policies either drop or hold contents without considering potential common interests of the users. To address this issue, recent studies have proposed context-aware and cluster-based management approaches for in-network edge caching to group mobile users sharing similar content interests [8]–[10]. These works then associate clusters with base stations according to the interests of users to improve the cache hit ratio at small cells, while offloading data traffic from the macrocells.

Cluster-based approaches effectively offload data traffic in cellular network scenarios [11]–[13]. However, the works presented so far in the literature strongly rely on frequencies of content requests to form clusters [8], [9]. In other words, despite two users having interests for just a single content, they might be misclassified as similar and associated with the same cluster, if the frequency of requests for this particular content is too high. To mitigate such bias, this paper extends existing clustering approaches to take into account not only the frequency of content requests, but also the variety of contents shared by users with common interests. As a result, we introduce a new similarity measure through which cache-equipped base stations can efficiently serve content to users by high-affinity clusters.

We thus propose in this article the interest Similarity (iSim) strategy to provide affinity-based user clustering for efficient edge caching in Content-Centric Cellular Networks. This strategy implements a flexible method to group users according to the number of common requests and the total quantity of distinct contents they share as a common interest (i.e., diversity level). Together, these two parameters enable iSim to achieve

a higher hit ratio in edge caches at small cells and, as a consequence, a better level of data offloading at macrocells through a more accurate evaluation of user similarities.

The main contributions of the iSim strategy are thus as follows:

- iSim provides a flexible method to group users based on their affinity concerning the frequency of requests to contents of common interest and the diversity level of the shared content between these users;
- The proposed strategy not only increases the cache hit ratio at network edges through cache-enabled small cell base stations, but also it reduces the traffic load at macrocells contributing to data offloading and the saving of network bandwidth.

The reminder of this article is organized as follows. Section II briefly discusses the general approach adopted for user clustering in mobile scenarios. Section III presents the iSim strategy by describing the method used to accomplish high-density clusters for information-centric cellular networks. Section IV describes the experimental study conducted through the OMNeT++ simulator. Section V presents our analysis based on the obtained simulation results. Finally, Section VI concludes the paper and discusses future work.

II. CONTEXT-AWARE AND CLUSTER-BASED CONTENT DELIVERING IN CELLULAR NETWORKS

The growing interest for content caching in mobile network scenarios has motivated a number of investigations on data offloading through clustering algorithms. In general, these investigations propose clustering methods to group users with similar interests and associate them with cache-enabled base stations located at network edges.

In [8], authors use a cosine similarity measure to group users within clusters based on content popularities. For cache maintenance, they propose a reinforcement minimization learning method to optimize SCBS's caches according to interests of a group. As a result, each base station implements an individual cache replacement policy. A complementary work, presented in [14], investigates the performance of content clustering by addressing the content replication problem at the cluster level. Accordingly, they adopt a hierarchical network of caches to avoid the complexity of dealing with content-level granularity. To this end, their approach replicates clusters of contents instead of individual contents.

Pantisano et al. [9] propose a collaborative-filter scheme to estimate the backhaul usage for each SCBS. The main goals consist in updating the network bandwidth according to the backhaul usage and to identify which user equipments to serve. The collaborative-filter helps on the estimation of file popularities calculated through the requests made for all SCBS's. Consequently, it is possible to correlate different files and users, and then calculate the probability of a file being requested by another user's equipment. To accomplish this goal, the authors carried out the user association as a matching game, which considers the maximum storage capacity of each SCBS, the maximum backhaul bandwidth allocation, the transmission capacity bottleneck, and the minimum data rate requirement.

In short, related work basically relies on content popularity among users calculated through the frequency of interests issued by them. Despite [9] using different contents and network features to define a user association, they do not take into account distinct aspects of the user content interests, as our proposal does. Therefore, in contrast to previous related work, we introduce a similarity measure that combines different aspects of user content interests to form affinity-based user clusters for a more efficient edge caching in Content-Centric Cellular Networks.

III. ISIM: AFFINITY-BASED USER CLUSTERING

The iSim strategy uses a high-affinity similarity measure to group users in Content-Centric Cellular Networks. We designed the proposed strategy based on the CCN architecture and present the main iSim characteristics in this section. Figure 1 presents a general cellular network scenario in which iSim takes place.

A. iSim communication architecture

In Figure 1, there are essentially three roles played by network nodes, namely: (i) User Equipment – UE; (ii) Small-Cell Base Station – SCBS; and (iii) Macrocell Base Station – MBS. UE sends interest packets to request contents from the base stations. SCBS consists of a cache-enabled base station that delivers contents fetched from its local cache. When content requests are not locally matched, the SCBS forwards interest packets to other base stations. Finally, a MBS is a high-capacity cache-enabled base station that serves all other base stations and UEs. A MBS is also responsible for forming clusters and associate them with the most appropriated SCBS.

When a UE requests for a content, the SCBS and the MBS transfer interest and data packets through collaborative or non-collaborative content-centric communication models, as depicted in Figure 1. In a collaborative communication, the SCBS searches a content in every node that is in the same level, and only if no SCBS has the content the request is forwarded to MBS. A non-collaborative scenario is featured by a direct communication between SCBS and MBS, if the SCBS does not have the content.

B. User clustering in Content-Centric Networks

CCN is one of the current ICN architectures that provides resources to deal with named data. For this purpose, CCN defines three basic structures: (i) The Pending Interest Table (PIT) maintains the interests issued by users that have not been resolved yet, as a bread crumb; (ii) The Content Store (CS) contains a copy of some contents (like a cache); and (iii) The Forwarding Information Base (FIB) is a mapping between a producer and an output interface. The proposed affinity-based algorithm intends to provide a better use of CS, storing the most requested contents by a group of similar users and, consequently, help decreasing the number of PIT entries.

Figure 1 also shows the PIT and CS data structures. PIT stores content identifiers in the first field and users identifiers in the second one. In respect to CS, apart from content

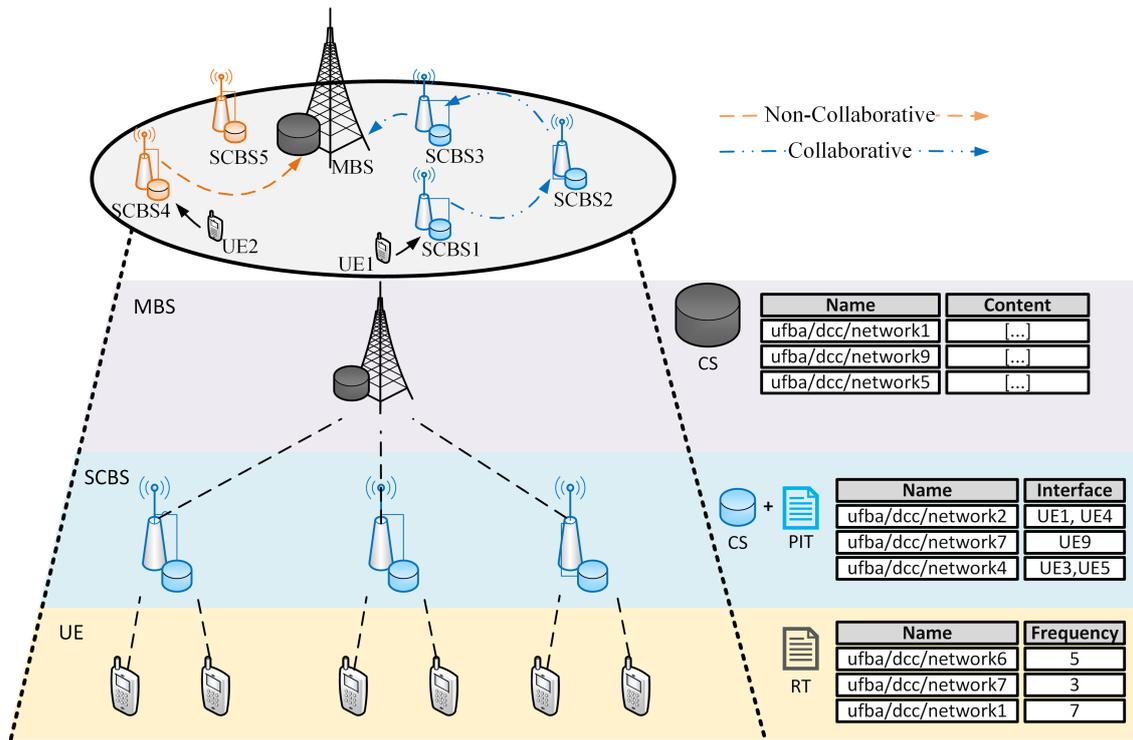


Fig. 1: Content-Centric Cellular Networks scenario in which iSim takes place.

identifiers, it stores the content itself and the frequency of users' requests, to use the LFU cache maintenance policy. In addition to such regular CCN data structures, iSim also adopts the Requisition Table (RT) that user equipments use to send up-to-date information of their interests (e.g., request frequency for each content).

C. Similarity measure for high-affinity users

We define affinity-related users as those that not only frequently request the same contents, but also share a common interest for a large variety of diverse contents. User similarity calculated through such a definition enables cache replacement policies to maintain more useful contents in caches, thereby improving the hit ratio within affinity-based clusters. Therefore, to accomplish this goal, iSim adopts the similarity measure S , defined in Equation 1. $S_{i,j}$ consists of the weighted sum of the Jaccard index and the Cosine similarity, used to calculate the affinity relation between two users i, j based on content diversity and request frequency, respectively. Accordingly, C_i and C_j represent the two sets of contents requested by i and j , whereas F_i and F_j represent the frequency of common contents requested by i and j . The result is a similarity measure $S_{i,j}$ that measures the affinity level between users i and j . Through the weight $0 \leq \beta \leq 1$, the similarity measure $S_{i,j}$ can be adapted to diverse cellular network scenarios by establishing the relative importance of the content diversity and request frequency in each case.

$$S_{i,j} = \beta \left(\frac{|C_i \cap C_j|}{|C_i \cup C_j|} \right) + (1 - \beta) \left(\frac{F_i \cdot F_j}{\|F_i\| \|F_j\|} \right) \quad (1)$$

D. Affinity-based clustering algorithm

iSim centrally calculates users' similarities at MBS to form G clusters, whereas G is the number of SCBS. To this aim, the MBS receives RT information from all user devices to build a frequency of content requests matrix $M = [m_{u,c}]$, for $u = \{1, 2, \dots, nU\}$ and $c = \{1, 2, \dots, nC\}$, and assuming that nU and nC express the number of users and contents within a network scenario, respectively. Thus, each $m_{u,c}$ element gives the number of requests for a content per each user. The MBS then uses M as an input parameter of Algorithm 1, that procedures as follows. Firstly, Equation 1 is used to calculate the similarities among users according to each $m_{u,c}$ element (lines 1-8). The results are then used to build a symmetric matrix $U = [u_{i,j}]$, for $i = \{1, 2, \dots, nU\}$ and $j = \{1, 2, \dots, nU\}$, i.e., $u_{i,j} = u_{j,i}$, for all indexes i and j . Each $u_{i,j}$ element thus contains the similarity among users i and j . Afterwards, the algorithm initiates the clustering process that classifies users into groups according to the similarities from U . iSim uses the k-Nearest Neighbor (kNN) algorithm to accomplish such a goal.

iSim runs the kNN algorithm based on an ordered similarity list. According to this list, the algorithm calculates the mean between the node and its k nearest neighbors. The node with the highest mean is then defined as a cluster leader. The cluster formation is then started through the cluster leader and its most similar neighbors. When a cluster is formed, nodes belonging to the cluster are removed from U and the process (lines 11-14) is repeated until all groups are formed. Finally, upon forming clusters, the MBS sends a message notifying which SCBS the node should be associated with.

Algorithm 1 Similarity and Grouping Algorithm

```

1: procedure SIMILARITY FUNCTION(Matrix  $M = [m_{uc}]$ )
2:   for  $i \in nU$  do
3:     for  $j > i \in nU$  do
4:       for  $c \in nC$  do
5:         Calculate  $C_i \cap C_j, C_i \cup C_j$ 
6:         Calculate  $F_i \cdot F_j \in \|F_i\| \|F_j\|$ 
7:       Calculate similarity using Equation 1
8:        $u_{i,j} = u_{j,i}$ 
9: procedure CLUSTERING FUNCTION(Matrix  $U = [u_{ij}]$ )
10:  for  $w \in G$  do
11:    Apply a sorting algorithm for each  $[u_{ij}]$  line
12:    Calculate average between similarity's nodes
13:    Choose the node with best average
14:    Group the node chosen and its k nearest neighbors
15: Result:  $G$  groups with  $k + 1$  members

```

IV. SIMULATION-BASED EXPERIMENTAL STUDY

We evaluated iSim's affinity-based clustering method through a simulation-based experimental study. The next subsections describe the goals of this experimental study, the adopted methodology, and the simulation environment.

A. Experimental study goals and evaluation methodology

As mentioned before, iSim has been designed to accomplish a better level of data offloading in cellular networks through high-affinity user clusters. Therefore, we formulated our research question to find out to which extent iSim's similarity measure contributes to data traffic reduction at macrocells through content delivery at small cells. Thus, in order to answer this question, the experimental study evaluates the following performance metrics:

- MBS Data offloading (*off*). It consists of the relative number of content requests satisfied at small cells by adopting iSim that would be downloaded from the macrocell if the strategy were not adopted.
- SCBS Cache-hit ratio (*hr*). It describes the relative number of contents fetched at small cells.

In order to measure the Data offloading metric (*off*), we adopt Equation 2. The *rMax* variable represents the interest that would be requested directly from MBS, whereas *rMin* consists of the requests received after clusters are formed.

$$off = 100 \times \frac{rMax - rMin}{rMax} \quad (2)$$

We use Equation 3 to measure the cache hit ratio (*hr_i*) at each SCBS. The *match* variable represents the satisfied interests at the SCBS, whereas *iRec* denotes the total number of interests received. Thus, *hr* is calculated through the average between all existing SCBS's hit ratio, as in Equation 4, where *G* is the number of SCBS.

$$hr_i = 100 \times \frac{match_i}{iRec_i} \quad (3)$$

$$hr = \frac{\sum_{i=1}^G hr_i}{G} \quad (4)$$

To analyze these metrics, we followed a methodology that consists of repeating the experiments 10 times to obtain results corresponding to the mean with 95% confidence intervals for each simulation setup.

Additionally, we determined a training period of 200 seconds to map users' preferences according to the configuration parameter α of a Zipf distribution [15], which defines the relevance and frequency of a content in an ordered list. In a Zipf distribution, the first content is the most common, the second is the next most common, and so on. Therefore, the popularity of a content *i* is given by Equation 5. Hence, the request rate for the *ith* most popular content is proportional to $1/i^\alpha$.

$$z_i = \frac{1/i^\alpha}{\sum_{n=1}^N 1/n^\alpha} \quad (5)$$

B. Simulation environment

To carry out the simulations, we used the OMNeT++ simulator, supported by the INET framework. We use this simulation environment to reproduce a cellular network scenario composed of SCBS, MBS, and user equipments. In this environment, the users' movement pattern relies on a Mass Mobility model and content requests follow a Poisson distribution. Further, we adopt the Zipf distribution for content popularity, which is varied according to the users' profiles. Table I summarizes the parameters adopted for the simulations.

TABLE I: Simulation parameters.

Parameter	Value
Quantity of UEs	30
Quantity of MBS	1
Quantity of SCBS	3
Quantity of contents	100
Caching Policy	LFU
Mobility Model	Mass Mobility
Propagation	Constant Speed Propagation
Frequency	2.4GHz
Loss	Free Space Path Loss
Noise	-110dBm

V. PERFORMANCE EVALUATION

The previously described experimental simulation study enables us to evaluate the iSim performance and its suitability to cellular network scenarios. The following subsections present our analysis regarding the cache size, communication models, content popularity, and the iSim's adaptability to different CCN scenarios.

A. Cache size and iSim's similarity measure effect

Figure 2 presents the results of varying the cache size and the effects of iSim's β parameter on the hit ratio. We evaluate three sizes for the SCBS's caches: 10%, 20%, and 30% of the number of available contents. Therefore, the caches at the

SCBS's contain a copy of some contents downloaded from a higher-capacity cache of the MBS, which stores the whole content used in experiments (i.e., 100 contents).

The results demonstrate that no matter the β value using in iSim's similarity measure, iSim outperforms random clustering for all configured cache sizes. Besides that, the larger the cache size, the higher the hit ratio for any scenario, because more contents are stored in local caches, as it would be expected.

Figure 2 also shows that the use of either the Cosine similarity (i.e., $\beta = 0$) or the Jaccard index (i.e., $\beta = 1$) presents a worse performance when compared with their joint use (i.e., $\beta = 0.5$) for all cache sizes. Thus, the weighted sum of these parts, as the iSim's similarity measure suggests, leads to the best performance. Furthermore, between Jaccard index and Cosine similarity, the Jaccard index seems to represent better the user-affinity in the evaluated scenario as consistently the results for using just the Jaccard index are close to the results for joint use of the two parts of the similarity definition. Moreover, notice that when $\beta = 0$, iSim only takes into account the Cosine similarity measure, which is an approach equivalent to the one adopted by recent related works, such as [8]. That is, iSim's similarity measure outperforms related work since it also relies on the Jaccard index to form high-affinity user clusters (e.g., $\beta = 0.5$).

Finally, Figure 2(d) summarizes the relation between cache size and hit ratio. Results attest that the iSim experienced higher hit ratio proportional gains as the cache size increases. When we compare the use of both scenarios, we can see that for a cache size of 10% the iSim can achieve results next to a random clustering for cache size of 20%. Further, when we compare the iSim for a cache size of 20%, the results are better than the random scenario for a cache size of 30%, showing that iSim provides resource savings. Since cache hit ratio and data offloading are directly proportional, every interest satisfied in a SCBS means an equivalent data offload from the MBS. Hence, the data offloading on MBS is directly related to the cache hit ratio on SCBS.

B. Communication models

Figure 3 analyzes the iSim's similarity measure with respect to the communication models adopted in the network under evaluation. Figure 3(a) shows that in both communication models iSim provides better levels of data offloading. Hence, employing the iSim strategy has more influence on data offloading than the communication model. However, by just evaluating the communication model, we can notice that iSim slightly favors higher data offloading in collaborative SCBS's communications.

Besides data offloading, Figure 3(b) illustrates that iSim improves the cache hit ratio in either collaborative or non-collaborative models while collaborative communication decreases the hit ratio due to the increase in number of interests received for each SCBS. The non-collaborative model achieves better hit ratios for one rather obvious reason. Since the SCBS does not request missing contents from adjacent base stations, contents are always fetched by the MBS. Conversely, the non-collaborative model incurs in less data offloading at the MBS.

The result shows that even in such non-collaborative model iSim achieves better hit ratio since it clusters affine users at edge base stations that are likely to request the same contents.

C. Content popularity

The Zipf distribution defines the content popularity concentration and directly influences on the frequency and diversity of requests. The larger the α parameter, the smaller the number of popular contents and higher the frequency of requests for such data. Therefore, when the α parameter increases, there is a little group of highly popular contents that will be frequently requested, but the other requests will have more variety and will provide more caching changes and miss ratio. Therefore, with a smaller α we have better cache hit ratios, as shown in Figure 4(a).

In a scenario with a smaller α , we have a larger group of more frequently requests for contents and a better distribution between them. Since we are using LFU as a replacement cache strategy, a larger group of popular contents improves the results of cache hit ratio, as shown in Figure 4(a). However, the difference between applying iSim or not becomes smaller in such scenarios. As the number of popular content is larger and the users usually request such contents, even with the random clustering, the local caches are able to storage the most requested contents, leaving less room for improvement, as shown in Figure 4(b).

D. iSim's adaptability

One important advantage of iSim's similarity measure resides in its adaptability to different ICN scenarios. In networks featured by low-diversity user requests, the Cosine similarity achieves higher cache hit ratios, whereas the Jaccard index favors those networks featured by high-diversity requests among users. For unknown scenarios, the coupled use of these terms (i.e., $\beta = 0.5$) is suggested. This flexibility in adapting the similarity measure can be used to adjust iSim according to the features of each scenario. Since the MBS centrally calculates user similarity and forms clusters according to the user's RTs, it's also possible to improve MBS intelligence to provide an automatic distribution of weight between Cosine similarity and Jaccard index depending on the requests characteristics, such content diversity, and frequency. This also can be used to dynamically adapt to network changes.

VI. CONCLUSION

Content-Centric Networking (CCN) emerges as an alternative to improve data offloading in cellular networks through in-network caching. In this context, we introduce iSim, an affinity-based user clustering strategy for efficient edge caching in Content-Centric Cellular Networks. iSim relies on a similarity measure that captures a combination of different aspects of content interest by mobile users to form high-affinity user clusters. These affinity-based user clusters drive the contents kept at edge caching. This improves the data offloading at the MBS by increasing the cache hit ratio at small cells.

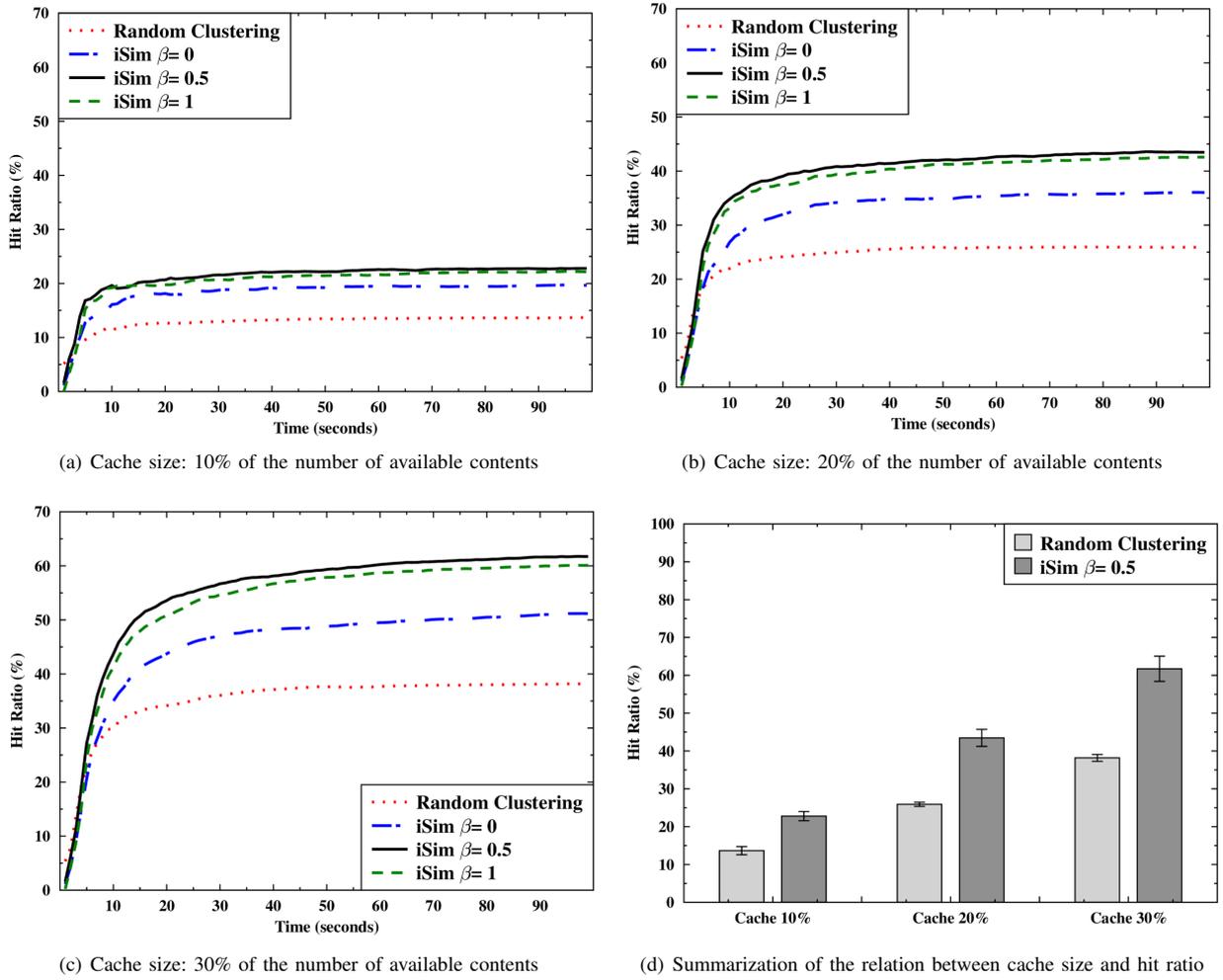


Fig. 2: Influence of the cache size in iSim's performance.

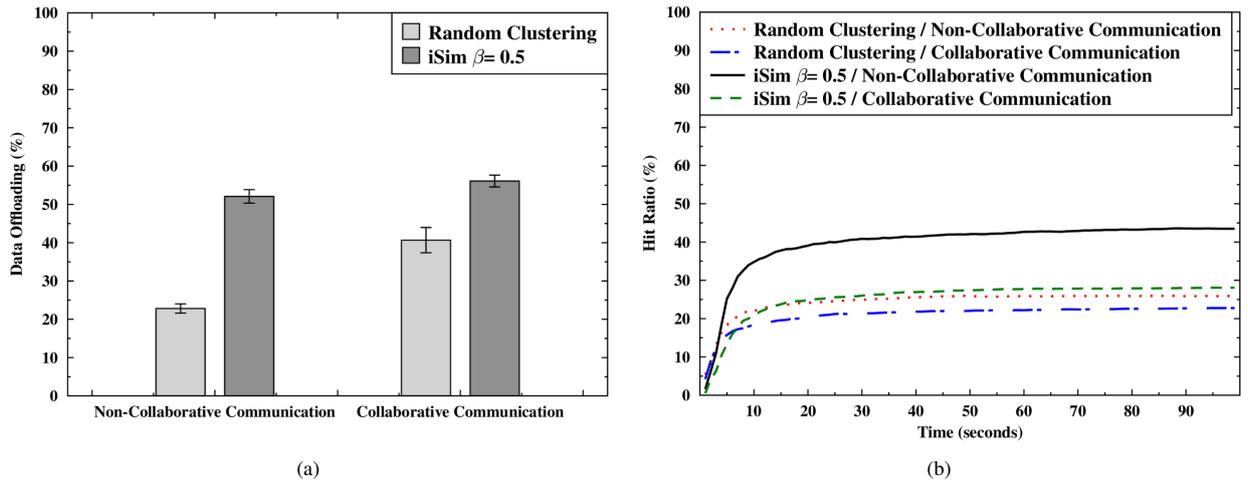


Fig. 3: iSim's data offloading and hit ratio in respect to the collaborative and non-collaborative communication models.

iSim provides a flexible method to group users based on their affinity concerning the frequency of requests to

contents of common interest and the diversity level of the shared content between these users. The proposed strategy

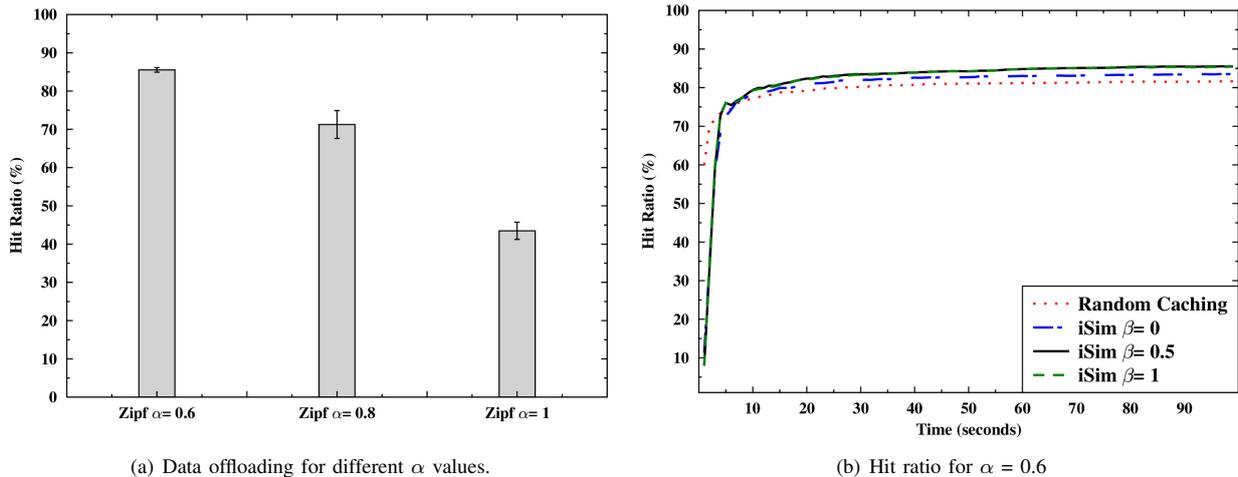


Fig. 4: iSim's data offloading and hit ratio in respect to Zipf Distribution.

is evaluated in a varied set of scenarios, including different cache sizes, communication models, or concentration levels of content popularity. Simulation results show that the proposed strategy increases both cache hit ratio and data offloading in Content-Centric Cellular scenarios. The proposed strategy not only increases the cache hit ratio at network edges through cache-enabled small cell base stations, but also it reduces the traffic load at macrocells contributing to data offloading and the saving of network bandwidth. Although the performance gains by iSim depend on the concentration level of content popularity and cache size, the iSim's combined similarity measure can be adapted to different scenarios.

As future work, we intend to design adaptive cache policies that build upon characteristics of the affinity-based user clusters. Another promising direction for future work is the application of the affinity-based user clusters to also design content recommendation systems. Such content recommendation systems might then be coupled with edge caching so that the most recommended contents may be prefetched for an enhanced cache performance and user experience.

ACKNOWLEDGEMENT

A. V. Ribeiro and L. N. Sampaio acknowledge the financial support from CAPES through grant 1460330. A. Ziviani acknowledges the partial support from CNPq through grants 465.560/2014-8 (INCT in Data Science – INCT-CiD) and 308.729/2015-3.

REFERENCES

- [1] B. Soret, K. I. Pedersen, N. T. K. Jørgensen, and V. Fernández-López, "Interference coordination for dense wireless networks," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 102–109, January 2015.
- [2] X. Ge, S. Tu, G. Mao, C. X. Wang, and T. Han, "5G Ultra-Dense Cellular Networks," *IEEE Wireless Communications*, vol. 23, no. 1, pp. 72–79, February 2016.
- [3] N. Bhushan, J. Li, D. Malladi, R. Gilmore, D. Brenner, A. Damnjanovic, R. T. Sukhvasi, C. Patel, and S. Geirhofer, "Network Densification: The Dominant Theme for Wireless Evolution into 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 82–89, February 2014.
- [4] M. Chen, Y. Hao, M. Qiu, J. Song, D. Wu, and I. Humar, "Mobility-aware caching and computation offloading in 5g ultra-dense cellular networks," *Sensors*, vol. 16, no. 7, p. 974, 2016.
- [5] A. Ioannou and S. Weber, "A survey of caching policies and forwarding mechanisms in information-centric networking," *IEEE Communications Surveys Tutorials*, vol. 18, no. 4, pp. 2847–2886, Fourthquarter 2016.
- [6] F. Rebecchi, M. D. De Amorim, V. Conan, A. Passarella, R. Bruno, and M. Conti, "Data offloading techniques in cellular networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 580–603, 2015.
- [7] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking Named Content," in *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, 2009, pp. 1–12.
- [8] M. S. ElBamby, M. Bennis, W. Saad, and M. Latva-Aho, "Content-aware User Clustering and Caching in Wireless Small Cell Networks," in *11th International Symposium on Wireless Communications Systems (ISWCS)*, 2014, pp. 945–949.
- [9] F. Pantisano, M. Bennis, W. Saad, and M. Debbah, "Match to cache: Joint user association and backhaul allocation in cache-aware small cell networks," in *2015 IEEE International Conference on Communications (ICC)*, June 2015, pp. 3082–3087.
- [10] Z. Chang, Y. Gu, Z. Han, X. Chen, and T. Ristaniemi, "Context-Aware Data Caching for 5G Heterogeneous Small Cells Networks," in *Communications (ICC), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1–6.
- [11] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Communications Surveys Tutorials*, vol. PP, no. 99, pp. 1–1, 2017.
- [12] S. Bassooy, H. Farooq, M. A. Imran, and A. Imran, "Coordinated multi-point clustering schemes: A survey," *IEEE Communications Surveys Tutorials*, vol. 19, no. 2, pp. 743–764, Secondquarter 2017.
- [13] P. V. Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A survey of machine learning techniques applied to self organizing cellular networks," *IEEE Communications Surveys Tutorials*, vol. PP, no. 99, pp. 1–1, 2017.
- [14] L. Gkatzikis, V. Sourlas, C. Fischione, I. Koutsopoulos, and G. Dán, "Clustered content replication for hierarchical content delivery networks," in *2015 IEEE International Conference on Communications (ICC)*, June 2015, pp. 5872–5877.
- [15] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: Evidence and implications," in *INFOCOM'99, Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 1. IEEE, 1999, pp. 126–134.