

A Competitive Analysis of BitTorrent-like Algorithms for Interactive Video-on-Demand Streaming over MANETs

C. Rodrigues, and V. Rocha

Abstract—This article carries out a competitive analysis between BitTorrent-like algorithms for interactive video-on-demand streaming over mobile ad hoc networks. To this end, we herein mainly delve into five recent proposals of this kind, and assess the operational download (DO) and the discontinuity-time (DT) performance metrics in different scenarios by means of simulations, considering two client interactivity profiles, two mobility models, and various network sizes. Although final results indicate that all five proposals may provide suitable system quality of service and client quality of experience in certain scenarios, the one named as BitTorrent Interactive Broadcast - Network is overall the most competitive one. For instance, compared to other literature proposals, it may provide optimizations of up to 106% and 86% at the DO and DT metrics, respectively, thus confirming its attractive performance.

Index Terms—Streaming, BitTorrent, MANET, Interactivity.

I. INTRODUÇÃO

Diante da popularização dos dispositivos móveis e do crescimento ininterrupto do tráfego de dados das aplicações multimídia nas redes de comunicação [1], [2], pesquisas recentes têm se dedicado ao desenvolvimento de soluções para oferecer *streaming* interativo de vídeo sob demanda em redes móveis ad hoc (do inglês, *Mobile Ad Hoc Networks* - MANETs) [3], [4], [5].

Em uma MANET, os nós móveis operam autonomamente como servidores, clientes e roteadores, sem dependência de infraestrutura fixa. Esses nós são responsáveis pela própria gerência da comunicação existente na rede. A MANET tem, assim, as características de autoconfiguração, autoformação e automanutenção. Essa versatilidade operacional suscita especial interesse em inúmeras áreas de aplicação, e.g., entretenimento, ensino, negócios, operações militares e segurança [6].

Dentre as pesquisas realizadas, cinco propostas recentes examinam a adaptação do protocolo de replicação de arquivos BitTorrent [7], a saber: Algoritmo de Distância Mínima com Reciprocidade Indireta (AD-RI) [8], BitTorrent Interativo para MANET (BTM-I) [3], BitTorrent Interativo de Aplicação (BTI-A) [9], BitTorrent Interativo de Rede (BTI-R) [9], e BitTorrent Interativo com *Broadcast* - Rede (BIB-R) [10].

Essas cinco propostas datam a partir do ano de 2018 e, salvo melhor juízo, são as únicas baseadas no BitTorrent que tratam sobre a interatividade do cliente [11], [12], [13]. Todavia, em

que pesem os satisfatórios resultados apresentados nos trabalhos individuais, não há uma obra única que reúna as cinco propostas para uma análise competitiva. Sendo assim, algumas questões ainda se constituem em lacunas de investigação, tais como: Qual dessas propostas é a mais eficiente? Como o padrão de mobilidade e o perfil de interatividade do cliente impactam essas propostas? Como o crescimento da rede afeta a escalabilidade das propostas? Ainda é possível promover algum tipo de otimização nessas propostas?

Essa conjuntura evidencia a atualidade deste tema de pesquisa, bem como impele para a realização de trabalhos subsequentes para ratificar os resultados alcançados e ampliar os escopos então considerados. Neste contexto, este artigo tem o objetivo de realizar uma análise competitiva entre as propostas AD-RI, BTM-I, BTI-A, BTI-R e BIB-R. Para este fim, por meio de simulações, são avaliadas as métricas de desempenho *download* operacional (*DO*) e tempo de descontinuidade (*TD*) em distintos cenários de *streaming* de vídeo sob demanda, considerando dois perfis de interatividade, dois modelos de mobilidade e diferentes tamanhos de rede.

Para maior abrangência de escopo, os experimentos de avaliação incluem também duas outras soluções para *streaming* interativo em MANETs. A primeira é o algoritmo BitTorrent Interativo com *Broadcast* - Aplicação (BIB-A), que é uma variante aqui apresentada do algoritmo BIB-R. Seu diferencial é uma maior simplicidade de projeto. O objetivo de sua inclusão é verificar se é possível ter o mesmo (ou similar) nível de desempenho que BIB-R, usando uma solução de menor complexidade operacional.

A segunda solução é o *Delay-Sensitive Segment Scheduling Algorithm* (DSSSA) [4], que se constitui na mais recente proposta da literatura que não é baseada no BitTorrent. Sua inclusão permite avaliar a eficiência da abordagem geral de replicação do BitTorrent (presente nas cinco propostas) comparativamente a uma distinta forma de replicação.

Ante o exposto, a principal contribuição deste artigo é prover novos subsídios para o desenvolvimento de protocolos reais que possam oferecer adequados níveis de QoS (*Quality of Service*) do sistema, e de QoE (*Quality of Experience*) do cliente, considerando o estado da arte de pesquisas sobre *streaming* interativo de vídeo sob demanda em MANETs.

O restante deste artigo é organizado como segue. A Seção II revisa a operação do conhecido BitTorrent e discute suas limitações para redes sem fio e realização de *streaming*. Na Seção III, tem-se a apresentação de cada uma das cinco propostas a serem analisadas competitivamente. A Seção IV

apresenta experimentos, resultados e análises. Conclusões finais e trabalhos futuros constituem a Seção V.

II. FUNDAMENTOS

A. BitTorrent: Políticas de Seleção

O protocolo BitTorrent opera segundo uma arquitetura de rede *peer-to-peer* (P2P). Seu núcleo é constituído por uma política de seleção de pedaços e uma de seleção de *peers*. Essas duas políticas governam o processo de replicação de arquivos, considerando a transmissão dos pedaços dos arquivos via conexões TCP estabelecidas entre os *peers* da rede.

Swarm é a denominação do conjunto de *peers* que trabalha cooperativamente para replicar os pedaços dos arquivos, e *tracker* é a entidade coordenadora da comunicação entre os *peers* do *swarm* [7]. As operações das duas políticas de seleção supracitadas são sucintamente revisadas a seguir, sob a visão de um *peer p* pertencente a um *swarm* da rede. Para maior detalhamento, sugerem-se os trabalhos de [7], [14].

A política de seleção de *peers* serve para que um *peer p* decida que outros *peers* podem requisitar seus pedaços. A capacidade de transmissão do *peer p* é dividida em cinco *slots* lógicos de dados: quatro *slots* para *upload*, e um *slot* para *download*. Ademais, o *peer p* pode ser um *leecher* (se não possui todos os pedaços do arquivo), ou um *seed* (se possui todos os pedaços do arquivo). Os dois processos de *unchoking* (desbloqueio) a seguir são então periodicamente executados.

1) *Regular unchoking*: a cada intervalo $\delta t = 10$ s, se o *peer p* é um *leecher* (*seed*), então os três *peers* que transmitem (recebem) pedaços para o mesmo (a partir do mesmo) com as mais altas taxas de *upload* (*download*) são selecionados. Um *slot* é então alocado para cada *peer* selecionado.

2) *Optimistic unchoking*: a cada intervalo $\delta t = 30$ s, o *peer p* (*leecher* ou *seed*) seleciona aleatoriamente um quarto *peer*, e um *slot* é alocado para o mesmo.

Por sua vez, a política de seleção de pedaços serve para que um *peer p* decida que pedaços requisitar ao ser selecionado por um outro *peer* em um processo de *unchoking*. Para tanto, o *peer p* mantém uma lista do número de cópias de cada pedaço. Essa informação é usada para definir um conjunto de pedaços mais raros (i.e., menos replicados), como explicado a seguir.

Seja m o número de cópias do pedaço mais raro. O índice de cada pedaço com m cópias é então adicionado ao conjunto de pedaços mais raros. Após ser selecionado, o *peer p* requisita o próximo pedaço com base no seu conjunto de pedaços mais raros, bem como nos pedaços disponíveis localmente no *peer* que o selecionou. Após receber o pedaço solicitado, o *peer p* avisa aos demais *peers* do *swarm* sobre o fato. A replicação de pedaços não é feita, portanto, de forma estritamente sequencial (i.e., gulosa), pois a transmissão de pedaços prioriza os pedaços mais raros (i.e., menos replicados).

B. Paradigmas: Layered Architecture e Cross Layer

A operação do BitTorrent não tem consciência da topologia física da rede de comunicação subjacente. As políticas de seleção anteriores atuam sobre uma topologia lógica, considerando apenas a camada de aplicação da arquitetura TCP/IP. A razão é que a concepção do BitTorrent é pensada para redes

com fio, caracterizadas por canais de comunicação estáveis e baixa perda de pacotes de dados [15].

Porém, a seleção de *peers* considerando apenas a topologia lógica pode comprometer o desempenho. Isso porque, sob a topologia lógica, *peers* vizinhos (i.e., *peers* que trocam pedaços) aparecem como diretamente conectados (i.e., enlace direto de um único salto), mas eles podem na verdade estar a múltiplos saltos entre si na real topologia física, usando *peers* intermediários para encaminhamento de pedaços [15].

Neste contexto, há dois paradigmas de desenvolvimento possíveis de adotar para implementar adaptações do BitTorrent: *layered architecture* e *cross layer*. Sob o primeiro, a rede é dividida em camadas distintas e define-se uma hierarquia de serviços a serem providos pelas camadas individualmente. Inexiste a comunicação direta entre camadas não adjacentes, e a comunicação entre camadas adjacentes se restringe a chamadas de procedimentos e respostas. Simplicidade é a principal vantagem, porém não se adquire a consciência da topologia física da rede subjacente.

Sob o segundo, i.e., *cross layer*, a hierarquia de serviços é flexibilizada, permitindo que camadas não adjacentes se comuniquem. Por um lado, as implementações se tornam mais complexas, podendo estabelecer dependência de informações entre várias camadas. Em contrapartida, as aplicações podem então ter consciência da topologia física da rede subjacente para obter melhores desempenhos [11].

III. ADAPTAÇÕES DO BITTORRENT

Esta seção revisa as cinco propostas que são o foco deste trabalho: AD-RI, BTM-I, BTI-A, BTI-R e BIB-R. Para tanto, a organização desta seção é estruturada como segue. A Subseção III-A apresenta os critérios usualmente empregados para a seleção de *peers*. A Subseção III-B descreve as políticas de seleção de *peers*. Por fim, a Subseção III-C explica as políticas de seleção de pedaços.

A. Critérios para Seleção de Peers

As cinco propostas em discussão utilizam (parcial ou totalmente) os critérios a seguir para que um *peer p* selecione um *peer g*, onde se assume que o *peer g* é vizinho do *peer p*, ambos pertencentes a um *swarm* de uma MANET.

1) *Taxa de upload* (tx_{up}): dentre os vizinhos do *peer p*, o *peer g* é aquele que mais rapidamente transmite pedaços para o *peer p* em média. Em caso de empate, a seleção é aleatória. Este critério reforça a *reciprocidade direta*, i.e., apenas aqueles *peers* que contribuem diretamente para o *peer p* podem receber pedaços a partir dele. Daí, *peers* egoístas, i.e., *peers* que apenas querem receber pedaços, são eliminados do sistema. O efeito colateral é que *peers* lentos, i.e., *peers* de baixa capacidade de transmissão, têm dificuldades de participar do processo de replicação, passando a depender da seleção proveniente de um processo como *optimistic unchoking*. O valor de tx_{up} é computado pelo próprio *peer p*.

2) *Taxa de download* (tx_{dn}): dentre os vizinhos do *peer p*, o *peer g* é aquele que mais rapidamente baixa (i.e., recebe) pedaços a partir do *peer p* em média. Em caso de empate, a seleção é aleatória. Este critério é utilizado no caso de o *peer*

p já possuir todos os pedaços do arquivo, ou seja, ser um *seed*. O valor de tx_{dw} é informado ao *peer* p pelo *peer* g .

3) *Reciprocidade indireta* (r_{ind}): dentre os vizinhos do *peer* p , o *peer* g é aquele que mais transmite pedaços em média a outros *peers* distintos do *peer* p , i.e., não se consideram os pedaços transmitidos para o *peer* p . Em caso de empate, a seleção é aleatória. Este critério tenta uniformizar a distribuição de dados ao estimular um compartilhamento de pedaços mais altruísta. O valor de r_{ind} é informado ao *peer* p pelo *peer* g .

3) *Contagem de saltos* (c_{sal}): dentre os vizinhos do *peer* p , o *peer* g é o mais próximo do *peer* p , em termos do número de saltos (i.e., roteadores) constituintes da rota física de ligação. Em caso de empate, a seleção é aleatória. Este critério objetiva a eliminação do *overhead* da computação de rotas longas. Estas rotas usualmente acabam sendo inúteis, pois tendem a ser mais lentas que rotas mais curtas. Além disso, este critério otimiza o uso da capacidade de transmissão de *peers* intermediários, evitando o seu uso desnecessariamente. O valor de c_{sal} é informado ao *peer* p pelo protocolo de roteamento.

4) *Distância geográfica* (d_{geo}): dentre os vizinhos do *peer* p , o *peer* g é o mais próximo do *peer* p , em unidades de distância física. Em caso de empate, a escolha é aleatória. Este critério estimula compartilhamentos de dados entre *peers* que estão fisicamente próximos entre si. Espera-se uma otimização do uso da capacidade do sistema, pois os dados são majoritariamente trafegados dentro dos mesmos grupos de *peers* e, ainda, diminui-se o *overhead* de mensagens para descobrimento de novas rotas. Além disso, este critério pode mitigar restrições das tecnologias móveis, e.g., sinais de baixa intensidade e limitação de baterias. O valor de d_{geo} é obtido pelo *peer* p a partir de um *Global Positioning System* (GPS).

B. Políticas de Seleção de Peers

As políticas de seleção de *peers* das propostas em questão são constituídas individualmente por dois processos de *unchoking*, como ocorre no BitTorrent original. Por simplicidade, esses dois processos são aqui nomeados também como *regular unchoking* e *optimistic unchoking*, respectivamente.

Para explicação dos processos de *unchoking*, assumamos, como antes, que o *peer* p do *swarm* tem sua capacidade de transmissão dividida em cinco *slots* lógicos de dados: quatro *slots* para *upload*, e um *slot* para *download*. As Tabelas I e II trazem em suas linhas os passos sequenciais dos processos de *regular unchoking* e *optimistic unchoking* de cada proposta.

Note que as políticas de AD-RI, BTM-I e BTI-A seguem o paradigma *layered architecture*, enquanto que aquelas de BTI-R e BIB-R seguem a abordagem *cross layer*. Essa condição é constatada pelo emprego do critério c_{sal} , existente apenas nestas duas últimas. Outra diferença diz respeito ao número e lógica de uso dos critérios, bem como ao intervalo entre ocorrências sucessivas dos *unchokings*. Por exemplo, AD-RI é a que utiliza o maior número de critérios (i.e., $tx_{up}, tx_{dw}, d_{geo}, r_{ind}$), enquanto BIB-R é a que tem o maior intervalo (i.e., $\delta t = 200$ s). Os experimentos apresentados mais adiante (vide Seção IV) evidenciam os benefícios advindos do emprego dos critérios (em conjunto ou isoladamente) durante a operação dos algoritmos em análise.

C. Seleção de Pedaços

De forma geral, um mesmo critério é utilizado nas cinco propostas. Este critério resulta da combinação de uma janela deslizante, W , com um *buffer* interior, V . Para a sua explicação, assumamos que: os *peers* p e g pertencem ao mesmo *swarm*, o *peer* g é um vizinho do *peer* p , e um arquivo F está sendo reproduzido pelo *peer* g .

O arquivo F está dividido em u pedaços. Seja d o pedaço correspondente ao ponto atual de reprodução, e seja w o tamanho de W , em número de pedaços. Daí, W abrange os seguintes pedaços: $[d; d+w]$. Além disso, W é dinamicamente atualizada conforme ocorre a reprodução. Por exemplo, seja Δ_{play} o número de pedaços consecutivos que acabam de ser reproduzidos. O primeiro pedaço de W é então atualizado para $d + \Delta_{play}$, e o último pedaço é atualizado para $d + w + \Delta_{play}$.

Seja v o tamanho de V , em número de pedaços. O primeiro pedaço de V é sempre coincidente com o primeiro pedaço de W . Além disso, antes de o *peer* g solicitar o próximo pedaço ao *peer* p , é sempre verificado se todos os pedaços de V foram baixados (recebidos). Em caso afirmativo, o *peer* g solicita o pedaço mais raro pertencente a W que ainda não foi baixado; caso contrário, o *peer* g solicita o próximo pedaço consecutivo dentro de V ainda não baixado.

Os pedaços transmitidos por meio de *peers* intermediários são localmente armazenados nesses *peers*. Isso é feito para compensá-los pelo uso de suas capacidades de transmissão para enviar pedaços ao longo de rotas que conectam *peers* extremos (i.e., origem e destino). Para tanto, o protocolo de roteamento deve analisar os pacotes de dados para extrair os pedaços e, então, passá-los para a camada de aplicação local.

A única diferença que existe entre as cinco propostas é a seguinte. Em BTM-I e BIB-R, sempre que um *peer* do *swarm* transmite um pedaço pela primeira vez, este pedaço é necessariamente transmitido via *broadcast*. Ou seja, BTM-I e BIB-R enviam pedaços via *unicast* (usando TCP) e *broadcast* (usando UDP), enquanto que as demais admitem apenas transmissão *unicast* (usando TCP).

Sobre os valores de configuração para os parâmetros w e v , tem-se o seguinte. Os experimentos apresentados nos trabalhos individuais das cinco propostas não determinam valores específicos, havendo na verdade delimitações de regiões de valores em que se observa a otimização do desempenho das soluções. Daí, para uma análise imparcial e inspirando-se em trabalhos anteriores (e.g., [16], [3]), admite-se a seguinte formulação para configuração desses parâmetros: $w \approx 0,3 \times f_{size}$ e $v \approx 0,75 \times w$, onde f_{size} é o tamanho do arquivo F , medido em número de pedaços.

IV. AVALIAÇÃO DE DESEMPENHO

Os experimentos de avaliação de desempenho são baseados em simulações. Admite-se uma MANET constituída por n *peers*, com cada *peer* atuando como um cliente móvel que realiza ações de interatividade durante a reprodução de um arquivo de vídeo F . Dentre os n *peers*, há um *tracker* que opera de forma contínua e é diretamente alcançável pelos demais *peers* da rede.

Cada *peer* tem sua capacidade de transferência de dados dividida em cinco *slots* lógicos: quatro para *upload* e um para

TABELA I
POLÍTICA DE SELEÇÃO DE PEERS: REGULAR UNCHOKING.

Passos	AD-RI ($\delta t = 10$ s) [8]	BTM-I ($\delta t = 10$ s) [3]	BTI-A ($\delta t = 20$ s) [9]	BTI-R ($\delta t = 20$ s) [9]	BIB-R ($\delta t = 200$ s) [10]
1°	Os vizinhos do <i>peer p</i> são ordenados pela tx_{up} (ou pela tx_{dw} , se o <i>peer p</i> é um <i>seed</i>).	Idêntico à AD-RI.	Idêntico à AD-RI.	Os vizinhos do <i>peer p</i> , que estão dentro do limite de $c_{sal} = 2$, são ordenados pela tx_{up} (ou pela tx_{dw} , se o <i>peer p</i> é um <i>seed</i>).	Idêntico à BTI-R.
2°	Os dois vizinhos de mais altas tx_{up} (ou tx_{dw}) são selecionados; e um <i>slot</i> é alocado para cada um dos mesmos.	Os dez vizinhos de mais altas tx_{up} (ou tx_{dw}) são ordenados pela d_{geo} .	Idêntico à AD-RI.	Idêntico à AD-RI.	Idêntico à AD-RI.
3°	Os vizinhos remanescentes são ordenados pela d_{geo} .	Os três vizinhos de menores d_{geo} são selecionados; e um <i>slot</i> é alocado para cada um dos mesmos.	—	—	Os dois primeiros vizinhos remanescentes são ordenados pela r_{ind} .
4°	Os três vizinhos de menores d_{geo} são ordenados pela r_{ind} .	—	—	—	O vizinho de maior r_{ind} é selecionado; e um <i>slot</i> é alocado para o mesmo.
5°	O vizinho de maior r_{ind} é selecionado; e um <i>slot</i> é alocado para o mesmo.	—	—	—	—

TABELA II
POLÍTICA DE SELEÇÃO DE PEERS: OPTIMISTIC UNCHOKING.

Passos	AD-RI ($\delta t = 30$ s) [8]	BTM-I ($\delta t = 30$ s) [3]	BTI-A ($\delta t = 20$ s) [9]	BTI-R ($\delta t = 20$ s) [9]	BIB-R ($\delta t = 200$ s) [10]
1°	o <i>peer p</i> seleciona um vizinho aleatoriamente; e um <i>slot</i> é alocado para o mesmo.	Idêntico à AD-RI.	Os vizinhos do <i>peer p</i> são ordenados pela tx_{dw} .	Os vizinhos do <i>peer p</i> , que estão fora do limite de $c_{sal} = 2$, são ordenados pela tx_{dw} .	Idêntico à BTI-R.
2°	—	—	Os dois vizinhos de mais altas tx_{dw} são selecionados; e um <i>slot</i> é alocado para cada um dos mesmos.	Idêntico à BTI-A.	O vizinho de mais alta tx_{dw} é selecionado; e um <i>slot</i> é alocado para o mesmo.

download. A taxa de transferência de cada *slot* pode variar, pois depende da qualidade da conexão sem fio. Mas, para simulação, assume-se essa taxa em 1 Mbps em média. Para exemplificar, o padrão 802.11g suporta até 54 Mbps no total.

Como mencionado e justificado no início deste artigo, para competição com as cinco propostas em foco, os experimentos incluem ainda duas outras soluções: BIB-A e DSSSA, que são brevemente explicadas a seguir. A proposta BIB-A é inédita e consiste em uma variante simples da proposta BIB-R, explicada anteriormente. A diferença é que, na BIB-A, não é utilizado o critério c_{sal} (i.e., *contagem de saltos*), tornando-a então de abordagem *layered architecture* e, portanto, de menor complexidade operacional.

Por sua vez, DSSSA segue o paradigma *cross layer* e sua operação se baseia no escalonamento dos pedaços do arquivo na seguinte sequência. Primeiro, são transmitidos os pedaços

com instantes de reprodução bem próximos do corrente (i.e., pedaços urgentes). Segundo, são transmitidos os pedaços que não são urgentes, considerando os correspondentes instantes de reprodução (i.e., pedaços normais). Terceiro, são transmitidos os pedaços que estão replicados apenas em alguns poucos *peers* (i.e., pedaços mais raros). Além disso, como um mesmo pedaço pode estar armazenado em distintos *peers*, DSSSA sempre seleciona a rota mais rápida [17].

TABELA III
PERFIS DE INTERATIVIDADE.

Parâmetro	Baixa (IB)	Alta (IA)
λ	0,005/s	0,025/s
L	14,5% de f_{size}	1,5% de f_{size}
$p_{play}; p_{pause}$	0,89; 0,01	0,55; 0,15
$p_{j f}; p_{j b}$	0,05; 0,05	0,15; 0,15

O restante desta seção é organizado como segue. A Subseção IV-A trata sobre a mobilidade e os perfis de interatividade dos clientes. A Subseção IV-B explica os modelos de simulação e as métricas de avaliação. Por fim, A Subseção IV-C traz os experimentos, resultados e análises.

A. Mobilidade e Interatividade

Os *peers* da MANET se movimentam sobre uma área plana de 100 m², livre de obstáculos, com velocidade entre 0 e 2 m/s, e com direção entre 0 e 360°. Essa mobilidade é definida a partir dos seguintes simuladores: *Small World in Motion* (SWIM) [18] e *Bonnmotion* [19].

O modelo do SWIM baseia-se em três premissas: (i) o indivíduo tende a visitar lugares populares ou próximos do lugar onde passa a maior parte do tempo; (ii) o indivíduo permanece a maior parte do tempo em poucos lugares (e.g., escritórios e escolas), enquanto permanece por um curto período de tempo em vários lugares (e.g., bancos e restaurantes); (iii) a velocidade do indivíduo depende de quão próximo se está do destino final. Os parâmetros e valores de configuração do simulador são os mesmos considerados em [18].

No caso do *Bonnmotion*, tem-se o modelo de mobilidade *Pursue* [20]. Neste caso os indivíduos perseguem um certo alvo. Embora os movimentos de cada indivíduo sejam independentes entre si, a perseguição por um alvo leva a um comportamento coletivo. Este cenário pode corresponder a um indivíduo que transmite um vídeo enquanto caminha dentro de uma área limitada, e.g., saguão de aeroporto, campus universitário, dentre outros. Os parâmetros e valores de configuração do simulador são os mesmos usados em [21].

A principal diferença entre os modelos SWIM e *Pursue* é que, neste último, os *peers* estão mais concentrados (i.e., próximos), levando a uma comunicação mais estável. Além disso, no modelo SWIM, é possível haver pequenos silos, ou seja, grupos de *peers* que não podem se comunicar porque estão fora do alcance (de rádio) um do outro.

Sobre a interatividade dos clientes, tem-se que as ações podem ser: *Play*, *Pause*, *Jump Forwards* (JF), e *Jump Backwards* (JB). A ação *Play* é a reprodução contínua. A ação *Pause* indica que a reprodução é pausada. As ações JB e JF representam um salto para um ponto de reprodução anterior e posterior ao corrente, respectivamente.

As ações *Play*, *Pause*, JF e JB são geradas de acordo com uma distribuição de Poisson com taxa λ e probabilidades p_{play} , p_{pause} , p_{jf} e p_{jb} , respectivamente. As ações *Play*, *Pause*, JF e JB possuem uma mesma extensão L , computada como um percentual do tamanho total do arquivo, f_{size} . Neste contexto, são considerados dois perfis de interatividade do cliente: *interatividade baixa* (IB) e *interatividade alta* (IA). Por fim, a Tabela III lista os valores numéricos dos parâmetros de configuração, que são os mesmos utilizados em [8], [3].

B. Simulação e Métricas

Os experimentos são realizados no simulador PeerSim [22]. A plataforma computacional é um Intel Core i7 (2,6 GHz), 24 GB de RAM, executando o sistema operacional GNU/Linux. Os resultados da simulação têm intervalos de confiança de

95% que estão dentro do limite de 5% dos valores médios reportados, tendo sido realizadas 10 execuções (rodadas).

Na camada de rede, o roteamento de pacotes é baseado no protocolo *Ad hoc On-Demand Distance Vector* (AODV) do simulador WiFi Direct [23]. As informações das camadas MAC/PHY são consolidadas pelo atraso na transferência e pelo percentual de perda de pacotes, ambos baseados na força do sinal entre dois *peers*. O arquivo a ser distribuído por *streaming* é um vídeo codificado a 300 kbps. Este arquivo é dividido em 390 pedaços de 256 kB, sendo que cada pedaço é dividido em blocos de 16 kB. Assim, o vídeo possui um tamanho aproximado total de 100 MB. Neste caso, tem-se que: $f_{size} = 390$; $v = 90$; e $w = 120$ (vide Subseção III-C). Apesar de o bloco ser a unidade fundamental de envio pela rede física, a análise considera somente os pedaços transmitidos.

Para avaliação das propostas, são consideradas as seguintes duas métricas de desempenho: *download* operacional (*DO*), que estima a taxa média do *peer* para receber o vídeo; e o tempo de descontinuidade (*TD*), que estima o tempo médio total de interrupção do *peer* durante a reprodução do vídeo. A análise conjunta dessas duas métricas permite inferir tanto sobre a QoS entregue pelo sistema, quanto sobre a QoE percebida pelo cliente.

Para aceitável QoS, assume-se $DO \geq 300$ kbps. Este valor é uma das opções para taxa de codificação de vídeos na Internet [24], [25]. Para aceitável QoE, assume-se $TD \leq 600$ s. Este valor representa 25% do tempo total de transmissão nos experimentos apresentados mais adiante, o que está dentro dos limites considerados na literatura ou pode ser mitigado por técnicas específicas (e.g., *buffering*, *prefetching*, etc.), quando desejável [26], [27], [28].

C. Experimentos, Resultados e Análises

As propostas AD-RI, BTM-I, BTI-A, BTI-R, BIB-R, BIB-A e DSSSA são analisadas conjuntamente nesta subseção. Os cenários consistem de MANETs categorizadas pelo tamanho, medido em número de *peers*: redes pequenas ($n = 25$); redes médias ($50 \leq n \leq 75$); e redes grandes ($n \geq 100$). Como a área de cobertura se mantém fixa (i.e., 100 m²), quanto menor é a rede, maior tende a ser a dispersão dos *peers*, e vice-versa.

Para o modelo SWIM, as Figuras 1a-1b e 2a-2b comparam as propostas para as métricas *DO* e *TD*, respectivamente. Dos resultados observados, tem-se as conclusões parciais a seguir.

- Em todos tamanhos de rede, BIB-R tem desempenho superior. Isso ocorre pelo uso de *broadcast* e consciência da topologia física. Ademais, BIB-A, que é uma variante mais simples de BIB-R (por considerar apenas topologia lógica), aparece no melhor caso apenas como a terceira melhor proposta. Isso evidencia a importância da abordagem *cross layer* em comparação à *layered architecture*.
- Para redes grandes, BIB-A e BTM-I pouco diferem entre si. Essa similitude advém do uso de *broadcast* e do critério de seleção de *peers tx_{up}*, presente em ambas propostas. Por sua vez, a pequena diferença observada se deve ao uso do critério d_{geo} , presente apenas em BTM-I.
- Para redes médias e grandes, BTI-A, AD-RI e DSSSA não apresentam níveis adequados de QoS e QoE. O

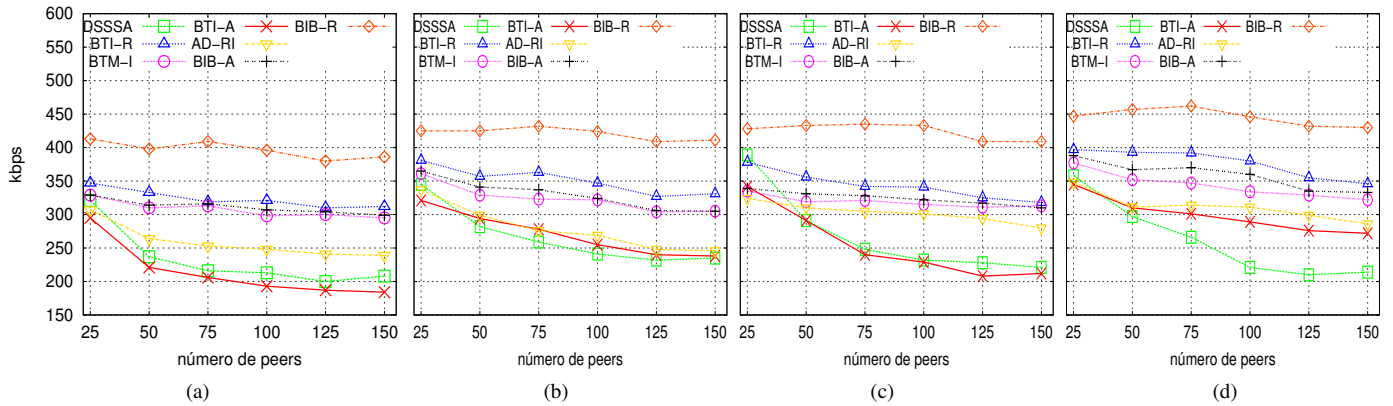


Fig. 1. Comparação no *download* operacional: (a) SWIM – IB; (b) SWIM – IA; (c) *Pursue* – IB (d) *Pursue* – IA.

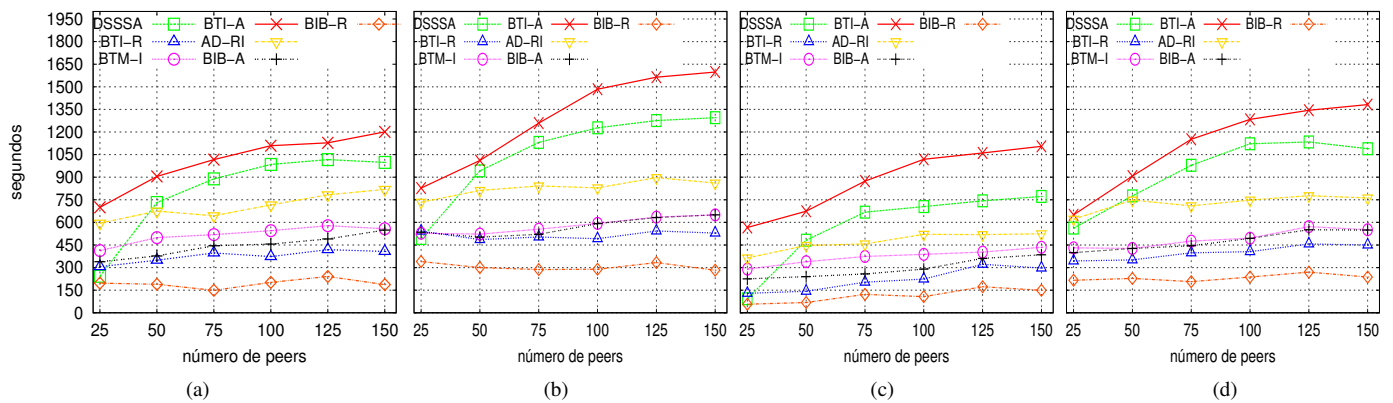


Fig. 2. Comparação no tempo de descontinuidade: (a) SWIM – IB; (b) SWIM – IA; (c) *Pursue* – IB (d) *Pursue* – IA.

desempenho insatisfatório ocorre por não ser considerado a topologia física na seleção de *peers*. Apesar de BIB-A e BTM-I também não levarem em conta a topologia física da rede, existe uma compensação de perda de desempenho devido ao uso de *broadcast*.

- Para redes pequenas, as propostas apresentam os melhores valores absolutos para as métricas *DO* e *TD*. Isso resulta do fato de a dispersão física dos *peers* propiciar um menor nível de interferência (concorrência pelo meio comum) entre as comunicações estabelecidas entre os *peers*, resultando na maior efetividade dos critérios de seleção propostos, seja quando utilizados conjuntamente ou isoladamente.
- Quanto mais interativo é o cliente, maior otimização ocorre nas métricas *DO* e *TD*. Contrariamente, quanto menos interativo, mais negativamente impactadas são essas métricas. Isso ocorre porque, quanto maior a interatividade, menos novas transmissões devem existir, pois mais pedaços anteriormente solicitados (portanto, já disponíveis localmente) tendem a ser necessários.

Neste contexto, a Tabela IV classifica as propostas de acordo com o desempenho, considerando os resultados das métricas *DO* e *TD* indicados nas figuras anteriores. Em geral, pode-se então observar que as propostas que utilizam o critério c_{sal} (i.e., BIB-R e BTI-R) são as mais competitivas, sendo usual-

mente a primeira e segunda melhor escolha, respectivamente. Na sequência, as propostas mais competitivas são as que fazem uso de *broadcast* (i.e., BTM-I e BIB-A).

A única exceção para a constatação geral anterior ocorre em redes pequenas. Neste caso, DSSSA aparece como a segunda melhor escolha na métrica *TD*. Isso porque em redes pequenas, como mencionado, existe usualmente menor interferência (i.e., menos concorrência pelo meio comum) entre as comunicações dos *peers*. Daí, o procedimento de escolher a rota física mais rápida (executado apenas em DSSSA) termina sendo decisivo para otimização. Por outro lado, quando a rede aumenta de tamanho (ocasionando mais interferência nas comunicações), esse procedimento perde relevância, dado que as rotas físicas passam a apresentar semelhantes retardos. Nesta situação, o uso de c_{sal} e de *broadcast* se tornam mais efetivos.

Para uma avaliação quantitativa, a Tabela V mostra as otimizações máximas e mínimas (i.e., ganhos em *DO* e reduções em *TD*) alcançadas pelas propostas. Os valores positivos (negativos) significam ganhos (reduções) percentuais em relação aos valores alcançados por DSSSA. Por exemplo, os resultados sob SWIM mostram que, em comparação à DSSSA, a proposta BIB-R produz um aumento em *DO* de até 90%, e uma redução em *TD* de até 83%.

Para o modelo *Pursue*, tem-se as as Figuras 1c-1d e 2c-2d. Os resultados mostram que as propostas têm um compor-

TABELA IV
ANÁLISE COMPETITIVA SOB OS MODELOS SWIM E PURSUE.

Perfil	Métrica	IB				IA			
		DO		TD		DO		TD	
Tamanho		$n = 25$	$n > 25$	$n = 25$	$n > 25$	$n = 25$	$n > 25$	$n = 25$	$n > 25$
SWIM	1°	BIB-R	BIB-R	BIB-R	BIB-R	BIB-R	BIB-R	BIB-R	BIB-R
	2°	BTI-R	BTI-R	DSSSA	BTI-R	BTI-R	BTI-R	DSSSA	BTI-R
	3°	BIB-A	BIB-A	BTI-R	BIB-A	BIB-A	BIB-A	BTI-R	BIB-A
	4°	BTM-I	BTM-I	BIB-A	BTM-I	BTM-I	BTM-I	BIB-A	BTM-I
	5°	DSSSA	AD-RI	BTM-I	AD-RI	DSSSA	AD-RI	BTM-I	AD-RI
	6°	AD-RI	DSSSA	AD-RI	DSSSA	AD-RI	BTI-A	AD-RI	DSSSA
	7°	BTI-A	BTI-A	BTI-A	BTI-A	BTI-A	DSSSA	BTI-A	BTI-A
Pursue	1°	BIB-R	BIB-R	BIB-R	BIB-R	BIB-R	BIB-R	BIB-R	BIB-R
	2°	DSSSA	BTI-R	DSSSA	BTI-R	BTI-R	BTI-R	BTI-R	BTI-R
	3°	BTI-R	BIB-A	BTI-R	BIB-A	BIB-A	BIB-A	BIB-A	BIB-A
	4°	BTI-A	BTM-I	BIB-A	BTM-I	BTM-I	BTM-I	BTM-I	BTM-I
	5°	BIB-A	AD-RI	BTM-I	AD-RI	DSSSA	AD-RI	DSSSA	AD-RI
	6°	BTM-I	DSSSA	AD-RI	DSSSA	AD-RI	BTI-A	AD-RI	DSSSA
	7°	AD-RI	BTI-A	BTI-A	BTI-A	BTI-A	DSSSA	BTI-A	BTI-A

TABELA V
OTIMIZAÇÕES ALCANÇADAS PELAS PROPOSTAS BASEADAS EM BITTORRENT COM RELAÇÃO À DSSSA.

BTI-R		BTM-I		BTI-A		AD-RI		BIB-A		BIB-R		Métrica / Modelo
Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	
8%	55%	2%	50%	4%	-12%	-2%	21%	2%	52%	23%	90%	DO / SWIM
10%	-62%	7%	71%	7%	189%	-8%	146%	9%	-54%	-19%	-83%	TD / SWIM
-3%	69%	5%	57%	0%	32%	-3%	41%	8%	60%	10%	106%	DO / Pursue
39%	-70%	-23%	212%	16%	510%	-4%	292%	-29%	142%	-39%	-86%	TD / Pursue

tamento similar àquele observado sob SWIM. A diferença está apenas no fato de que os valores absolutos são maiores. Isso porque os *peers* no modelo *Pursue*, como mencionado, encontram-se fisicamente próximos entre si (i.e., mais concentrados). Daí, os canais de comunicação entre *peers* tornam-se mais estáveis e, conseqüentemente, há uma maior efetividade para obtenção dos pedaços. Isso posto, tem-se que as conclusões parciais atingidas para SWIM se aplicam de forma análoga ao caso do modelo *Pursue*.

Por fim, ainda para o modelo *Pursue*, as Tabelas IV e V também mostram a classificação geral das propostas e as otimizações atingidas pelas propostas baseadas no BitTorrent em comparação à DSSSA, respectivamente. Por exemplo, os resultados mostram que, em comparação à DSSSA, a proposta BIB-R produz um aumento em *DO* de até 106%, e uma redução em *TD* de até 86%.

V. CONCLUSÕES FINAIS E TRABALHOS FUTUROS

Este artigo realizou uma análise competitiva entre cinco algoritmos recentes da literatura para o serviço de *streaming* interativo de vídeo sob demanda em MANETs. Os algoritmos se constituem em adaptações do conhecido protocolo de replicação de arquivos BitTorrent.

Os experimentos de avaliação se basearam em simulações, considerando a análise de cenários de *streaming* caracterizados por diferentes tamanhos de rede, além de dois perfis de interatividade e dois modelos de mobilidade do cliente. Dentre os resultados alcançados, podem-se destacar os seguintes:

- Os cinco algoritmos apresentaram níveis adequados de QoS (i.e., taxa de *download* superior a 300 kbps) e de

QoE (i.e., tempo de descontinuidade total abaixo de 25% do tempo de reprodução) em cenários específicos.

- O algoritmo nomeado de BIB-R [10] se mostrou o mais eficiente dentre os analisados, provendo otimizações de até 106%, na taxa de *download*, e de até 86% no tempo de descontinuidade, em comparação à proposta da literatura DSSSA [4]. Esse resultado evidenciou a importância da consciência da topologia física da rede e, conseqüentemente, do emprego da abordagem *cross layer* para o projeto de novos algoritmos.
- O padrão de mobilidade dos clientes influenciou o desempenho das propostas. Para uma mesma área física e mesmo tamanho de rede (em número de *peers*), um padrão que leva à dispersão dos *peers* prejudica a taxa de *download* e o tempo de descontinuidade. De forma contrária, um padrão que estimula a concentração dos *peers* otimiza essas métricas.
- O perfil de interatividade dos clientes também se mostrou influente no desempenho das propostas. Quanto mais interativo é o cliente, melhores são a taxa de *download* e o tempo de descontinuidade. De forma contrária, quanto menos interativo é o cliente, piores são a taxa de *download* e o tempo de descontinuidade.
- Para uma mesma área de cobertura, o aumento do tamanho da rede (em número *peers*) influenciou a escalabilidade das propostas. Quanto maior é a rede, mais negativamente afetadas são a taxa de *download* e o tempo de descontinuidade. De forma contrária, quanto menor é a rede, mais eficiente se torna a proposta.

Por fim, como trabalhos futuros e contemplando limitações

desta pesquisa, apontam-se: complementar esta análise com modelos analíticos e medições reais, incluindo peculiaridades das camadas inferiores; e realizar comparações com outras propostas da literatura ante, e.g., diferentes modelos de mobilidade e interatividade, áreas de cobertura, protocolos de roteamento, e codificações de arquivo (e.g., [2], [29], [30]).

REFERÊNCIAS

- [1] CISCO, "Cisco annual internet report (2018–2023)," *Cisco White Paper*, 2020, document ID:1551296909190103. [Online]. Available: <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>
- [2] Y. Liu and J. Y. B. Lee, "Post-Streaming Rate Analysis—A New Approach to Mobile Video Streaming with Predictable Performance," *IEEE Transactions on Mobile Computing*, vol. 16, no. 12, pp. 3488–3501, 2017, doi: 10.1109/TMC.2017.2694418.
- [3] C. K. S. Rodrigues and V. E. M. Rocha, "Uma Adaptação do BitTorrent para Streaming de Vídeo sob Demanda Interativo em Redes Móveis Ad Hoc," in *Anais do XVIII Workshop em Desempenho de Sistemas Computacionais e de Comunicação*, Belém, PA, Brasil, 2019, doi:10.5753/wperformance.2019.6459.
- [4] C.-C. Hu, C.-F. Lai, J.-G. Hou, and Y.-M. Huang, "Timely scheduling algorithm for P2P streaming over MANETs," *Computer Networks*, vol. 127, pp. 56 – 67, 2017, doi:10.1016/j.comnet.2017.08.001.
- [5] S. Ghali, R. Mishra, A. S. Baghel, and S. Sharma, "Routing protocol development for quality of service optimization of video-on-demand system over mobile ad hoc networks," *International Journal of Communication Systems*, vol. 31, no. 2, p. e3452, 2018, doi:10.1002/dac.3452.
- [6] J. Wang, B. Xie, and D. P. Agrawal, *Journey from Mobile Ad Hoc Networks to Wireless Mesh Networks*. London: Springer London, 2009, pp. 1–30.
- [7] B. Cohen, "Incentives build robustness in BitTorrent," in *First Workshop on Economics of Peer-to-Peer System*, Berkeley, USA, 2003.
- [8] C. K. S. Rodrigues, "Efficient BitTorrent-Like Algorithms for Interactive On-Demand Multimedia Streaming over MANETs," in *Proceedings of the 24th Brazilian Symposium on Multimedia and the Web*, Salvador, Bahia, Brazil, 2018, doi:10.1145/3243082.3243083.
- [9] C. K. S. Rodrigues and V. E. M. Rocha, "Towards adapting BitTorrent for interactive on-demand multimedia streaming over MANETs," in *Proceedings of the 25th Brazilian Symposium on Multimedia and the Web*, Rio de Janeiro, RJ, Brazil, 2019, doi:10.1145/3323503.3360618.
- [10] C. Rodrigues and V. Rocha, "BIB-R: Uma Nova Adaptação do BitTorrent para Streaming de Vídeo sob Demanda ante Clientes Interativos em MANETs," in *Anais do XIX Workshop em Desempenho de Sistemas Computacionais e de Comunicação*, Cuiabá, MT, Brasil, 2020, doi: 10.5753/wperformance.2020.11101.
- [11] S. Mantzouratos, G. Gardikis, H. Koumaras, and A. Kourtis, "Survey of cross-layer proposals for video streaming over Mobile Ad hoc Networks (MANETs)," in *2012 International Conference on Telecommunications and Multimedia (TEMU)*, Chania, Greece, 2012, pp. 101–106.
- [12] K. D. Fleury, M. and N. N. Qadri, "Video streaming over MANETs: An overview of techniques," *Multimedia Tools Applications*, vol. 6, pp. 23 749–23 782, 2019, doi:10.1007/s11042-019-7679-0.
- [13] N. Shah, S. Abid, D. Qian, and W. Mehmood, "A survey of P2P content sharing in MANETs," *Computers & Electrical Engineering*, vol. 57, pp. 55 – 68, 2017, doi: 10.1016/j.compeleceng.2016.12.013.
- [14] A. Legout, G. Urvoy-Keller, and P. Michiardi, "Rarest First and Choke Algorithms Are Enough," in *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement*, Rio de Janeiro, RJ, Brazil, 2006. [Online]. Available: <https://doi.org/10.1145/1177080.1177106>
- [15] M. Rethfeldt, B. Beichler, P. Danielis, F. Uster, C. Haubelt, and D. Timmermann, "MeNTor: A wireless-mesh-network-aware data dissemination overlay based on BitTorrent," *Ad Hoc Networks*, vol. 79, pp. 146 – 159, 2018, doi:10.1016/j.adhoc.2018.06.013.
- [16] C. K. S. Rodrigues and V. E. M. Rocha, "BT-MANET: A Novel BitTorrent-Like Algorithm for Video On-Demand Streaming over MANETs," *IEEE Latin America Transactions*, vol. 17, no. 01, pp. 78–84, 2019, doi: 10.1109/TLA.2019.8826698.
- [17] C.-C. Hu, "Delay-sensitive routing in multi-rate MANETs," *Journal of Parallel and Distributed Computing*, vol. 71, no. 1, pp. 53 – 61, 2011, doi: 10.1016/j.jpdc.2010.08.018.
- [18] A. Mei and J. Stefa, "SWIM: A Simple Model to Generate Small Mobile Worlds," in *IEEE INFOCOM*, Rio de Janeiro, RJ, Brazil, 2009, doi:10.1109/INFCOM.2009.5062134.
- [19] N. Aschenbruck, R. Ernst, E. Gerhards-Padilla, and M. Schwamborn, "BonnMotion: A Mobility Scenario Generation and Analysis Tool," in *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques*, Brussels, Belgium, 2010, pp. 51:1–51:10, doi:10.4108/ICST.SIMUTOOLS2010.8684.
- [20] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, 2002, doi:10.1002/wcm.72.
- [21] S. Alla, "A study of mobility models based on spatial node distribution and area coverage," Master's thesis, University of North Texas, 2017.
- [22] A. Montresor and M. Jelasity, "PeerSim: A scalable P2P simulator," in *IEEE Ninth International Conference on Peer-to-Peer Computing*, Seattle, WA, USA, 2009, doi: 10.1109/P2P.2009.5284506.
- [23] L. Baresi, N. Derakhshan, and S. Guinea, "WiDiSi: A Wi-Fi direct simulator," in *IEEE Wireless Communications and Networking Conference*, Doha, Qatar, 2016, doi:10.1109/WCNC.2016.7565169.
- [24] M. Schmitt, J. Redi, P. Cesar, and D. Bulterman, "1Mbps is enough: Video quality and individual idiosyncrasies in multiparty HD video-conferencing," in *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, 2016, pp. 1–6, doi: 10.1109/QoMEX.2016.7498961.
- [25] A. Bentalb, B. Taani, A. C. Begen, C. Timmerer, and R. Zimmermann, "A survey on bitrate adaptation schemes for streaming media over http," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 562–585, 2019, doi: 10.1109/COMST.2018.2862938.
- [26] C. K. S. Rodrigues, "On the Optimization of BitTorrent-Like Protocols for Interactive On-Demand Streaming Systems," *International Journal of Computer Networks and Communications*, vol. 6, pp. 39–58, 2014, doi:10.5121/ijcnc.2014.6503.
- [27] F. Xhafa, L. Barolli, and F. Amato, *Advances on P2P, Parallel, Grid, Cloud and Internet Computing: Proceedings of the 11th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, 1st ed. Springer Publishing Company, Incorporated, 2016.
- [28] L. Barolli, F. Amato, F. Moscato, T. Enokido, and M. Takizawa, *Advanced Information Networking and Applications: Proceedings of the 34th International Conference on Advanced Information Networking and Applications (AINA-2020)*, 1st ed. Springer International Publishing, 2020, doi:10.1007/978-3-030-44041-1.
- [29] P. Smiti, S. Srivastava, and N. Rakesh, "Video and Audio Streaming Issues in Multimedia Application," in *2018 8th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, 2018, pp. 360–365, doi:10.1109/CONFLUENCE.2018.8442823.
- [30] W. Castellanos, J. C. Guerri, and M. Chacon, "Improving video transmission over heterogeneous networks," *IEEE Latin America Transactions*, vol. 16, no. 12, pp. 2987–2995, 2018, doi:10.1109/TLA.2018.8804266.



Carlo K. da S. Rodrigues é Doutor em Engenharia de Sistemas e Computação pela Universidade Federal do Rio de Janeiro (UFRJ, 2006) e Mestre em Sistemas e Computação pelo Instituto Militar de Engenharia (IME, 2000). É professor do Centro de Matemática, Computação e Cognição da Universidade Federal do ABC (UFABC), no curso de Ciência da Computação. Atua na subárea de Redes de Computadores. <http://lattes.cnpq.br/4860474255962383>.



Vladimir E. M. Rocha é Doutor em Engenharia de Computação pela Escola Politécnica da Universidade de São Paulo (POLI-USP, 2017), e Mestre em Ciência da Computação pelo Instituto de Matemática e Estatística da mesma universidade (IME-USP, 2005). É professor do Centro de Matemática, Computação e Cognição (CMCC) da UFABC, no curso de Ciência da Computação. Atua em Sistemas Distribuídos. <http://lattes.cnpq.br/1225632464417175>.