

Análise do Paradigma BitTorrent para Streaming de Vídeo sob Demanda na Internet ante Acesso Sequencial Particionado Sincronizado

Carlo Kleber da Silva Rodrigues
Centro de Matemática, Computação e Cognição
Universidade Federal do ABC (UFABC)
Santo André, São Paulo, Brasil
carlo.kleber@ufabc.edu.br

Vladimir Rocha
Centro de Matemática, Computação e Cognição
Universidade Federal do ABC (UFABC)
Santo André, São Paulo, Brasil
vladimir.rocha@ufabc.edu.br

ABSTRACT

Considering the video-on-demand (VoD) streaming service and a mostly sequential-data access pattern by users, this article has the two following goals. Firstly, to analyze the BitTorrent paradigm's efficiency under this access-pattern type. Secondly, to propose and analyze a new policy for data selection by users, denoted as Smart Policy, focused on this access-pattern type. For that, simulations are carried out in different VoD streaming scenarios, evaluating a variety of performance metrics. Compared to previous proposals in the literature, the final results highlight optimizations of up to 24,3% and 100% at the download rate and the data waiting time, respectively. Conclusions and directions for future work close this article.

CCS Concepts

•Networks → Network performance analysis; Network algorithms;

Keywords

BitTorrent; Streaming; VoD; Acesso Sequencial; Internet.

1. INTRODUÇÃO

Aplicações de *streaming* de vídeo sob demanda (do inglês, *Video on Demand* - VoD) na Internet são cada vez mais populares (e.g., [8, 36]). Todavia, há em paralelo um crescente exaurimento da capacidade de transmissão de dados das redes de comunicação subjacentes, que resulta no comprometimento da qualidade de serviço (do inglês, *Quality of Service* - QoS) do sistema e da qualidade de experiência (do inglês, *Quality of Experience* - QoE) percebida pelos usuários. Em resposta, novos protocolos de *streaming* têm então sido propostos na literatura recente (e.g., [11, 18, 37]).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Os protocolos de *streaming* usualmente são desenvolvidos sob as arquiteturas cliente/servidor (C/S) e *peer-to-peer* (P2P). Sob a primeira, há um servidor lógico centralizado que disponibiliza o conteúdo acessado pelos usuários do sistema. Sob o segunda, os próprios usuários (i.e., *peers*) cooperam na disseminação do conteúdo. A escolha de uma arquitetura é função principal do tipo de aplicação, levando-se em conta, por exemplo, a infraestrutura de rede preexistente juntamente com a precificação do serviço final (e.g., [1, 39]).

Atendo-se especificamente à arquitetura P2P, a literatura recente apresenta várias adaptações do conhecido protocolo BitTorrent [9, 16], as quais têm se revelado como promissoras soluções para mitigar o exaurimento de capacidade antes mencionado (e.g., [30, 14, 26]). A filosofia base do BitTorrent é dividir o conteúdo a ser transmitido em pedaços de tamanho fixo e, então, estimular todos os usuários, que desejam receber este conteúdo, a trabalhar cooperativamente na distribuição dos pedaços. Sob este estímulo, os usuários que mais compartilham seus pedaços são também aqueles que mais rapidamente recebem outros pedaços.

As adaptações do BitTorrent notadamente devem considerar a análise do comportamento interativo dos usuários (i.e., ações de *play*, *pause*, *skip*, etc.) enquanto assistem aos conteúdos. Isso para que seja possível melhor definir a política de seleção de pedaços do protocolo, i.e., a ordem com que os pedaços são requisitados ao sistema durante a reprodução. O comportamento interativo dos usuários tem sido tema de inúmeros trabalhos, considerando diferentes tipos de mídia contínua e serviços (e.g., [35, 6, 20]).

É fato que as mais recentes adaptações assumem um comportamento interativo bem flexível, em que as ações realizadas durante a reprodução são variadas e probabilisticamente determinadas (e.g., [29, 32]). Supondo então um padrão de acesso bem restritivo, aqui denominado de *Sequencial Particionado Sincronizado* (SPS), no qual usuários chegam no mesmo instante e desejam assistir sequencialmente a um mesmo conjunto de segmentos do vídeo, as seguintes indagações devem ser respondidas: Essas adaptações ainda se mostram eficientes? É possível otimizar essas adaptações modificando apenas a sua política de seleção de pedaços? Como avaliar especificamente uma política de seleção de pedaços?

Este artigo possui então os dois objetivos mencionados a seguir. Primeiro, analisar o paradigma BitTorrent, considerando o algoritmo Interactive BitTorrent - Application

(IB-A) [30], ante o acesso SPS em redes cabeadas e o serviço de *streaming* de VoD. Este algoritmo é uma das mais recentes adaptações do BitTorrent. Segundo, propor e analisar uma nova política de seleção de pedaços, denominada de *Política Esperta*. Para tanto, são realizadas simulações em diferentes cenários, em que são avaliadas sete métricas de desempenho. Neste contexto, a principal contribuição deste trabalho é, portanto, prover novos subsídios teóricos para o desenvolvimento de protocolos de *streaming* de VoD.

O restante deste artigo tem a seguinte organização. A Seção 2 discorre sobre trabalhos relacionados. A Seção 3 revisa o algoritmo IB-A. Na Seção 4, tem-se a proposta da *Política Esperta*. A Seção 5 detalha o padrão de acesso SPS. Na Seção 6, tem-se experimentos e resultados. Por fim, conclusões e trabalhos futuros constituem a Seção 7.

2. TRABALHOS RELACIONADOS

Os trabalhos relacionados se subdividem em dois grupos. O primeiro abrange as propostas de adaptações do BitTorrent para *streaming* de VoD. O segundo inclui estudos referentes ao comportamento interativo dos usuários, os quais impactam a definição das políticas de seleção de pedaços dos protocolos. A seguir são então citados alguns dos mais recentes e importantes trabalhos desses dois grupos.

Pertencentes ao primeiro grupo, podem ser citados, por exemplo, os trabalhos de [29, 30, 12, 34, 17, 10]. As propostas de adaptações se baseiam essencialmente na modificação da política de seleção de *peers* e/ou da política de seleção de pedaços do BitTorrent original.

A política de seleção de *peers* trata sobre como um *peer* autoriza (escolhe) outros *peers* a receber os pedaços que ele já possui. A diferenciação entre as propostas reside principalmente no número e tipos de critérios (e.g., taxa de *upload*, distância geográfica, reciprocidade indireta, etc.) então utilizados. A política de seleção de pedaços contempla a forma como um *peer* do sistema decide (escolhe) qual próximo pedaço do conteúdo deve ser solicitado aos outros *peers*. Usualmente utiliza-se uma janela deslizante para delimitar os pedaços que podem ser solicitados em cada instante de tempo. A diferenciação entre as propostas se dá pelo tamanho e número de janelas que são utilizadas, bem como a forma de acesso (e.g., sequencial ou não sequencial) dos dados dentro da janela.

De acordo com a literatura, as propostas de adaptações se mostram no geral eficientes, com alguma supremacia de uma em relação a outra a depender do cenário (e.g., redes cabeadas ou móveis? usuários muito ou pouco interativos? redes pequenas ou grandes?) (e.g., [29, 26]). Todavia, como realçado ao início deste trabalho, os experimentos são usualmente de escopo generalista por assumirem um comportamento bem flexível dos usuários, em que as ações interativas, quando existem, são diversas e probabilisticamente determinadas. Diferentemente, no caso da pesquisa aqui realizada, assume-se o exame específico do padrão de acesso SPS.

Neste contexto, para efeito de avaliação do paradigma BitTorrent, esta pesquisa examina o algoritmo IB-A, cuja operação é revisada na Seção 3. Sua escolha se dá por ser um dos mais recentes trabalhos da literatura, além de seguir a abordagem de projeto *layered architecture* [19]. Esta abordagem se revela mais adequada para redes do tipo cabeada, que é o alvo dos experimentos realizados mais adiante.

Concernente ao segundo grupo de trabalhos, podem ser citados, por exemplo, as seguintes obras: [6, 35, 20, 5, 3,

38, 2, 28, 7, 25, 27]. Estas obras incluem análises das ações de interatividade dos usuários enquanto os conteúdos são reproduzidos. Os cenários investigados contemplam diferentes plataformas e aplicações de *streaming*, incluindo, por exemplo, YouTube, PPLive, Joost, e PPStream, bem como variados tipos de mídia contínua, e.g., filmes e músicas.

Os resultados advindos desse segundo grupo potencialmente geram subsídios para distintas áreas técnicas, as quais podem atuar em conjunto para o provimento de adequados níveis de QoS e QoE. Por exemplo, os produtores de conteúdo podem melhor direcionar os projetos visuais de telas de navegação das aplicações (e.g., cores, tamanhos, número de botões, etc.), conforme o número médio de ações de interatividade dos usuários. Também, os projetistas das redes de comunicação subjacentes (e.g., provedores de conteúdo) podem melhor dimensionar e escolher os equipamentos comutadores de tráfego e tipos de cabeamento (e.g., roteadores, *switches*, fibra óptica, etc.), conforme os requisitos de tempo de resposta das aplicações.

Em complemento, esses trabalhos do segundo grupo também possibilitam o desenvolvimento de geradores de cargas sintéticas, que se mostram valiosos para o processo de validação de propostas de novos protocolos quando da impossibilidade de obtenção ou utilização de *datasets* reais. Finalmente, em relação à pesquisa aqui desenvolvida, esses trabalhos comprovam a existência de padrões de acesso não tão generalistas e flexíveis, caracterizados por baixa dispersão temporal e espacial [28], o que acaba por ressaltar a importância desta pesquisa.

Ante o exposto, esta pesquisa tem seu ineditismo revelado pelos seguintes dois aspectos: realiza pela primeira vez uma análise específica de uma recente proposta de adaptação do BitTorrent ante o acesso SPS e, ainda, propõe e avalia uma nova política de seleção de pedaços. Neste sentido, esta pesquisa traz importantes subsídios teóricos para o projeto de protocolos reais para o serviço de *streaming* de VoD.

3. ALGORITMO: INTERACTIVE BITTORRENT - APPLICATION (IB-A)

Esta seção revisa o algoritmo IB-A por meio da explicação de suas duas políticas constituintes: seleção de *peers* e seleção de pedaços. Para esta explicação, assuma que cada *peer* da rede representa um usuário distinto do sistema, com sua capacidade de transferência de dados dividida em cinco *slots* lógicos de dados, sendo: $(y_1 + y_2)$ *slots* para *upload*, e um *slot* para *download*. Ademais, assuma ainda que os *peers* desejam reproduzir localmente um vídeo, o qual está dividido em pedaços de mesmo tamanho.

3.1 Política de Seleção de Peers

Considere o ponto de vista de um *peer* *A* qualquer. São realizados dois processos de autorização (escolha) de *peers* periodicamente: *Regular Unchoking* e *Altruistic Unchoking*, como descrito a seguir.

Regular Unchoking, a cada δt segundos:

1. *Peers* que desejam receber pedaços do *peer* *A* são ordenados em função da taxa de *upload* de dados para o próprio *peer* *A*.
2. Os $(y_1 - 1)$ *peers* de maiores taxas são selecionados pelo *peer* *A*, e um *slot* de dados é alocado para cada um deles.

- Os u primeiros *peers* remanescentes são ordenados em função da *reciprocidade indireta*, onde $u < (y_1 - 1)$. Este critério se refere à taxa de *upload* de dados para *peers* diferentes de A . Quanto maior é esta taxa, maior é a *reciprocidade indireta*.
- O *peer* de maior *reciprocidade indireta* é selecionado pelo *peer* A , e um *slot* de dados é alocado para ele.

Altruistic Unchoking, a cada δt segundos:

- Peers* que desejam receber pedaços do *peer* A são ordenados em função da taxa de *download* de dados a partir do próprio *peer* A .
- Os y_2 *peers* de maiores taxas são selecionados pelo *peer* A , e um *slot* de dados é alocado para cada um deles.

3.2 Política de Seleção de Pedaços

A operação desta política é baseada no emprego de uma janela deslizante, W , com um *buffer* interior, V , em cada *peer* da rede. Para a sua explicação, assumamos que: os *peers* A e B pertencem ao sistema, e um vídeo L está sendo reproduzido pelo *peer* B que, se necessário, solicitará o próximo pedaço do vídeo ao *peer* A , a ser recebido por meio de um *slot* lógico de dados.

O vídeo L tem tamanho l , em número de pedaços, i.e., os pedaços de L são: $1, 2, 3, \dots, l$. 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* B solicitar o próximo pedaço ao *peer* A , é sempre verificado se todos os pedaços de V foram recuperados (baixados). Em caso afirmativo, o *peer* B solicita o pedaço mais raro (i.e., menos replicado) pertencente a W que ainda não foi baixado; caso contrário, o *peer* B 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., *peers* 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. Por fim, sobre os valores de configuração para os parâmetros w e v , tem-se as formulações $w \approx 0,37 \times l$ e $v \approx 0,4 \times w$, em conformidade com a proposta original.

4. NOVA PROPOSTA: POLÍTICA ESPERTA

Esta seção explica a operação da nova política de seleção de pedaços, denotada aqui por *Política Esperta*. Para facilidade de apresentação, assumem-se a definição de parâmetros e as considerações já feitas na Seção 3.

Enquanto o *buffer* V do *peer* não tiver sido preenchido, o próximo pedaço a solicitar é o pedaço consecutivo ainda faltante dentro do mesmo. Por outro lado, caso o *buffer*

V esteja preenchido e assumindo que o primeiro pedaço da janela é o pedaço x , então o tamanho correspondente ao restante do vídeo (i.e., $l - (v + x)$) é dividido em k partes idênticas de tamanho $D = (l - (v + x))/k$, em número de pedaços. O próximo pedaço é então solicitado probabilisticamente a partir de uma das k partes definidas, como explicado a seguir.

Cada i -ésima parte, $1 \leq i \leq k$, é escolhida com probabilidade $p_i = p/(i + 1)$, com p obtido a partir do somatório $\sum_{i=1}^k p_i = 1$. A solicitação dos pedaços em cada uma das k partes pode ser feita de forma sequencial ou segundo o pedaço mais raro (i.e., menos replicado), gerando respectivamente duas variantes: *EspertaSequencial* e *EspertaMaisRaro*. A Figura 1 ilustra este entendimento para $k = 3$ e $x = 1$.

5. ACESSO: SEQUENCIAL PARTICIONADO SINCRONIZADO (SPS)

A seguir é explicado o padrão de acesso *Sequencial Particionado Sincronizado* (SPS). Para facilidade de exposição, assumem-se a definição de parâmetros e as considerações feitas nas seções anteriores.

Os *peers* se conectam ao sistema no mesmo instante de tempo e, em seguida, solicitam os mesmos pedaços do vídeo L obedecendo ao seguinte. Os pedaços solicitados são consecutivos, constituindo z segmentos de tamanho individual Q . Por sua vez, os z segmentos são separados entre si por intervalos (i.e., saltos na reprodução) de tamanho individual J . Os tamanhos Q e J são medidos em número de pedaços, cujos valores variam conforme o domínio da aplicação de *streaming* em análise. Por fim, a Figura 2 ilustra este padrão de acesso.

6. AVALIAÇÃO DE DESEMPENHO

6.1 Configuração Sistêmica

Os experimentos são realizados via simulação, considerando uma rede de sobreposição composta por n *peers*. Nos experimentos, varia-se n de 20 a 100, e se assume a rede em três categorias: pequena (de 20 a 30 *peers*); média (de 40 a 60 *peers*); e grande (de 70 a 100 *peers*). A capacidade de transferência de dados de cada *peer* é de 4 Mbps. Entre os n *peers*, há um *tracker* (i.e., um *peer* que coordena a comunicação entre os *peers*) e, inicialmente, um *seed* (i.e., um *peer* que já possui todos os pedaços do vídeo L).

Para implementar a conectividade física entre os *peers*, é utilizada a topologia de rede baseada em King [13]. Esta topologia representa uma situação realista de *hosts* de Internet com conexão cabeada. Nesta topologia, a média da latência fim a fim é de 200 ms. Em se tratando da recepção dos dados, por ser uma rede cabeada, assume-se uma baixa perda de pacotes, com um valor aleatório entre 0% e 5%.

Por fim, cada *peer* é um usuário que acessa um sistema de *streaming* sob demanda para assistir a um vídeo L de tamanho l , medido em número de pedaços de tamanho 256 kB cada, codificado a 300 kbps. Todos os *peers* seguem o padrão de acesso SPS.

6.2 Modelo de Simulação e Métricas

O modelo de simulação empregado nos experimentos é desenvolvido no simulador PeerSim [21]. O *hardware* utilizado é um Intel Core i7 (2,6 GHz), 24 GB de RAM, executando o sistema operacional GNU/Linux. Os resultados obtidos têm

Política Esperta

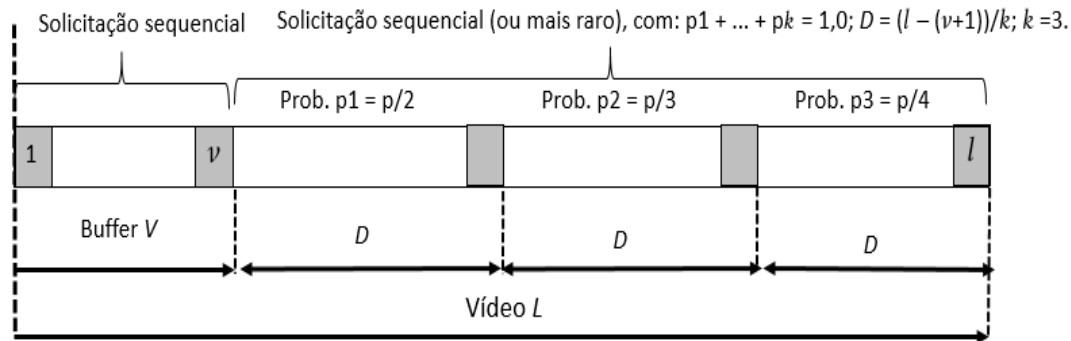


Figura 1: Política de solicitação de pedaços.

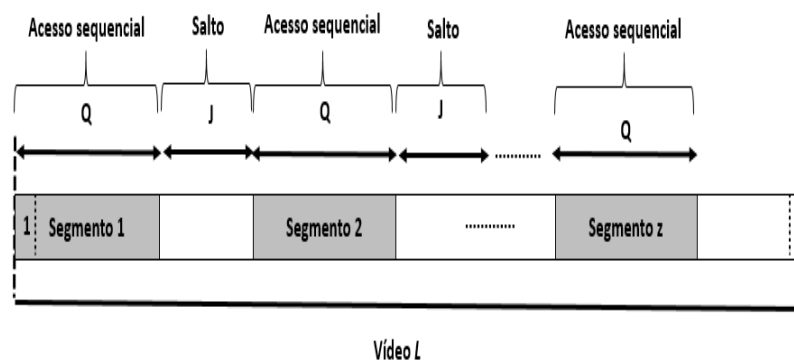


Figura 2: Acesso sequencial particionado sincronizado (SPS).

um intervalo de confiança de 95% que estão dentro do limite de 5% dos valores médios reportados, tendo sido realizadas 20 execuções (rodadas).

As métricas de desempenho estão definidas a seguir. São sete métricas no total: cinco métricas base, e duas métricas derivadas. As métricas base (a saber, *DO*, *ET*, *EB*, *PT*, e *PB*) permitem realizar uma análise competitiva entre propostas. Por sua vez, as métricas derivadas (a saber, *RE* e *RP*) são razões numéricas obtidas entre métricas bases, possuindo valores entre zero e um, e servem para permitir uma análise específica das políticas de seleção de pedaços das propostas: quanto mais próximo de zero (um), menos (mais) eficiente é a política de seleção.

1. *Download* Operacional (*DO*), medido em kB/seg: corresponde à taxa média com que o usuário realiza a recuperação (*download*) de todos os pedaços solicitados, os quais podem ser exibidos ou não localmente.
2. Espera Total (*ET*), medida em segundos: corresponde ao tempo médio total de espera do usuário por pedaços solicitados, os quais podem ou não ser exibidos localmente. Neste intervalo de tempo, o usuário tem sua reprodução congelada, gerando descontinuidades na exibição. Note que a taxa de recuperação de pedaços (avaliada por *DO*) independe da taxa de reprodução local dos pedaços (determinada pela taxa de codifi-

cação do vídeo, i.e., 300 kbps).

3. Espera Boa (*EB*), medida em segundos: corresponde ao tempo médio total de espera do usuário por pedaços solicitados que são exibidos localmente.
4. Razão de Espera (*RE*), resulta entre 0,0 e 1,0: calculada pela razão entre *EB* e *ET*. Permite obter o percentual de *ET* referente a pedaços solicitados e exibidos.
5. Pedaço Total (*PT*), medido em número de pedaços: corresponde à média do total de pedaços que são recebidos, os quais podem ser exibidos ou não localmente.
6. Pedaço Bom (*PB*), medido em número de pedaços: corresponde à média do total de pedaços que são recebidos e exibidos localmente.
7. Razão de Pedaço (*RP*), resulta entre 0,0 e 1,0: calculada pela razão entre *PB* e *PT*. Permite obter o percentual de *PT* referente a pedaços recebidos e exibidos.

6.3 Experimentos, Resultados e Análises

Os experimentos são organizados em três diferentes cenários no domínio de ensino a distância (EaD), diferenciados em acordo com os parâmetros do padrão de acesso SPS.

Este domínio é escolhido pela sua intrínseca importância e facilidade de caracterização devido aos resultados e análises já disponíveis na literatura. A configuração arbitrada para cada um dos cenários se baseia nos trabalhos citados na Seção 2, considerando especialmente a consolidação das informações realizadas em [28, 27, 24].

No Cenário 1, tem-se que um usuário assiste à $\approx 50\%$ de uma aula curta, com uma interatividade alta (i.e., 10 saltos) se comparada com os outros dois cenários. O Cenário 2 corresponde a uma aula longa, na qual um usuário assiste à $\approx 60\%$ dela, com uma interatividade média (i.e., 5 saltos). Finalmente, no Cenário 3, a aula também é longa, mas o usuário assiste à $\approx 85\%$ dela, com uma interatividade baixa (i.e., 3 saltos). A Tabela 1 apresenta as configurações utilizadas, incluindo os valores dos parâmetros v , w , δt , y_1 , e y_2 do algoritmo IB-A (vide Seção 3).

Nos três cenários é analisado o desempenho do algoritmo IB-A com sua política de seleção de pedaços original (vide Seção 3), bem como sua modificação considerando o emprego das variantes da *Política Esperta* (vide Seção 4) como política de seleção de pedaços. Por objetividade, apenas os resultados mais relevantes e atinentes às métricas DO , ET , RE e RP são apresentados neste trabalho. Ademais, para maior facilidade de discussão, doravante assumo as denominações de propostas a seguir.

- *Original*: IB-A com sua política de seleção de pedaços original.
- *EspertaSequencial k3*: IB-A com a variante *EspertaSequencial*, sendo $k = 3$.
- *EspertaSequencial k4*: IB-A com a variante *EspertaSequencial*, sendo $k = 4$.
- *EspertaMaisRaro k3*: IB-A com a variante *EspertaMaisRaro*, sendo $k = 3$.
- *EspertaMaisRaro k4*: IB-A com a variante *EspertaMaisRaro*, sendo $k = 4$.

6.3.1 Análise Competitiva entre Propostas

A análise competitiva desta subseção se baseia na avaliação das métricas DO e ET . Na Figura 3, tem-se o comportamento de DO em função de n nos Cenários 1, 2 e 3. Por sua vez, a Figura 4 apresenta o comportamento de ET em função de n nos Cenários 1, 2 e 3. A partir desses resultados, temos as constatações enumeradas a seguir.

1) *EspertaSequencial k3* e *EspertaSequencial k4* são as duas propostas de melhor desempenho. Esta superioridade indica que o particionamento de todo o restante do vídeo fora do *buffer* do *peer* (vide Figura 1) e a solicitação dos pedaços sequencialmente constituem, em conjunto, a melhor estratégia ante o acesso SPS. A pequena vantagem da primeira proposta sobre a segunda indica que existe um limite no número de partes a serem definidas, que nesta análise resulta empiricamente em $k = 3$. O estudo mais detalhado deste limite é deixado como trabalhos futuros.

2) Para uma percepção quantitativa da superioridade da *EspertaSequencial k3* em relação à *Original*, a Tabela 2 informa as otimizações (i.e., aumentos em DO e reduções em ET) alcançadas. Por exemplo, a otimização máxima em DO é de 24,3%, e em ET é de $\approx 100\%$. Os valores absolutos de ET muito próximos de zero implicam não haver descon-tinuidades na exibição. Isso acontece principalmente porque

a proposta: i) segue um comportamento semelhante ao de um acesso SPS; ii) evita que todos os *peers* requisitem um mesmo pedaço em simultâneo, pois as requisições ocorrem em diferentes partes do vídeo; e iii) provê um valor de DO acima da taxa de codificação, i.e., pedaços são recuperados mais rapidamente que exibidos.

3) *EspertaMaisRaro k3* e *EspertaMaisRaro k4* são as propostas de menor desempenho, sendo inclusive superadas pela *Original*. Isso indica que apenas o particionamento de toda a parte do vídeo que se encontra fora do *buffer* (vide Figura 1) não é suficiente para a otimização de desempenho. Fica assim evidenciado que uma política de seleção com base no pedaço mais raro não é a melhor estratégia. Esta política de seleção termina por aumentar a probabilidade de se baixar pedaços que não são exibidos no caso do acesso SPS, conforme discutido na análise da métrica RP mais adiante.

4) O desempenho das propostas diminui conforme n aumenta. Como todos os *peers* entram no sistema no mesmo instante, o único *peer* que recebe inicialmente todas as solicitações por pedaços é o *seed*. Assim, quanto maior é n , mais o *seed* é requisitado. Como o *seed* somente pode atender a um número limitado de *peers*, os demais têm de esperar. Consequentemente, ocorre um comprometimento de DO e de ET . Porém, todas as propostas, independente do cenário, permitem que o vídeo seja baixado a uma taxa maior que a de sua codificação (i.e., $DO > 300$ kbps), e apresentam esperas toleráveis (i.e., $ET < 200$ seg), redundando em adequados níveis de QoS e QoE [4].

5) Com respeito aos diferentes cenários, observa-se que DO é pouco afetado em seus valores máximo e mínimo, i.e., os correspondentes valores máximos e mínimos dos diferentes cenários são semelhantes. Deduz-se então que nem a duração do vídeo nem o nível de interatividade (i.e., número de saltos) impactam DO de forma significativa. Diferentemente, vê-se que a métrica ET é mais afetada. Isso acontece porque o número de acertos (i.e., número de pedaços de fato exibidos) com relação ao número total de pedaços que são baixados diminui, conforme aumentam-se a duração do vídeo e o nível de interatividade dos usuários. A discussão desses acertos é retomada na seção seguinte.

6.3.2 Análise das Políticas de Seleção de Pedaços

A análise desta subseção se baseia na avaliação das métricas RP e RE . Neste contexto, a Figura 5 apresenta o comportamento de RP em função de n nos Cenários 1, 2 e 3. De forma análoga, a Figura 6 traz o comportamento de RE em função de n nos Cenários 1, 2 e 3. A partir desses resultados, temos as constatações elencadas a seguir.

1) As políticas de seleção de pedaços de *EspertaSequencial k3* e *EspertaSequencial k4* são as mais eficientes dentre todas. Note que os correspondentes valores de RP e RE são os maiores em todos os cenários. Em específico, os valores de RP estão acima de 0,5, significando que mais da metade dos pedaços recebidos são exibidos localmente. Mas, ao examinarem-se os valores de RE , é possível notar valores abaixo de 0,5. Neste caso, mais da metade do tempo de espera é gasto com pedaços que não são exibidos localmente. Assim, apesar dos satisfatórios níveis de QoS e QoE observados na última subseção, ainda há espaço para otimização referente às políticas de seleção de pedaços.

2) As políticas *EspertaMaisRaro k3* e *EspertaMaisRaro k4* são em geral mais eficientes que aquela da *Original*, conforme pode ser observado pelos correspondentes valores de RP e

Tabela 1: Configurações dos cenários

Parâmetro	Unidade	Cenário 1	Cenário 2	Cenário 3
l	pedaço	322	604	611
Q	pedaço	5% de l	12% de l	28% de l
J	pedaço	5% de l	8% de l	5% de l
z	segmento	11	6	4
v	pedaço	30	56	57
w	pedaço	12	22	23
δt	segundo	200	200	200
y_1	slot	3	3	3
y_2	slot	1	1	1

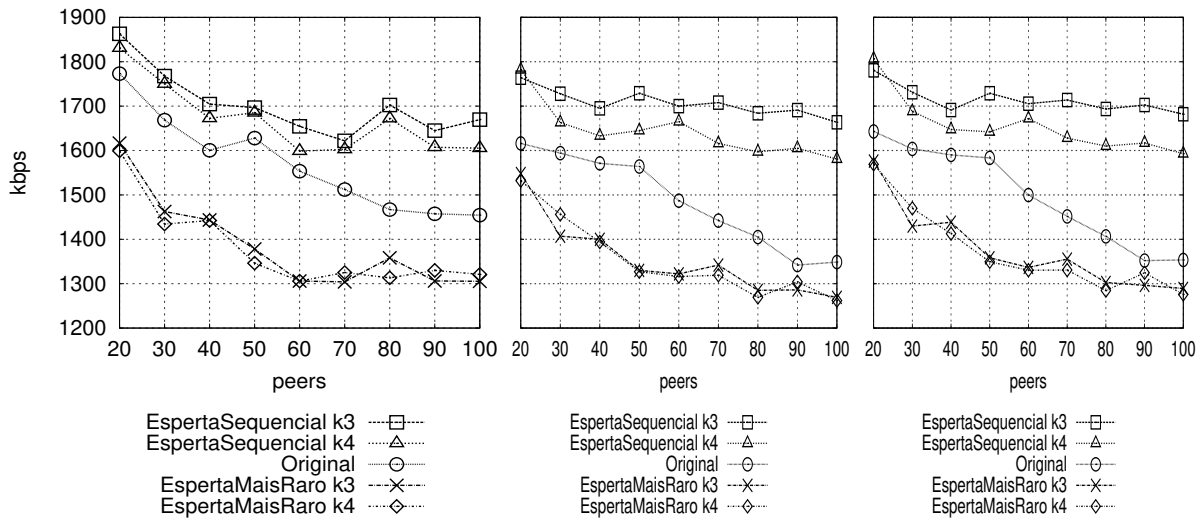


Figura 3: Download Operacional (DO): Cenário 1; Cenário 2; e Cenário 3.

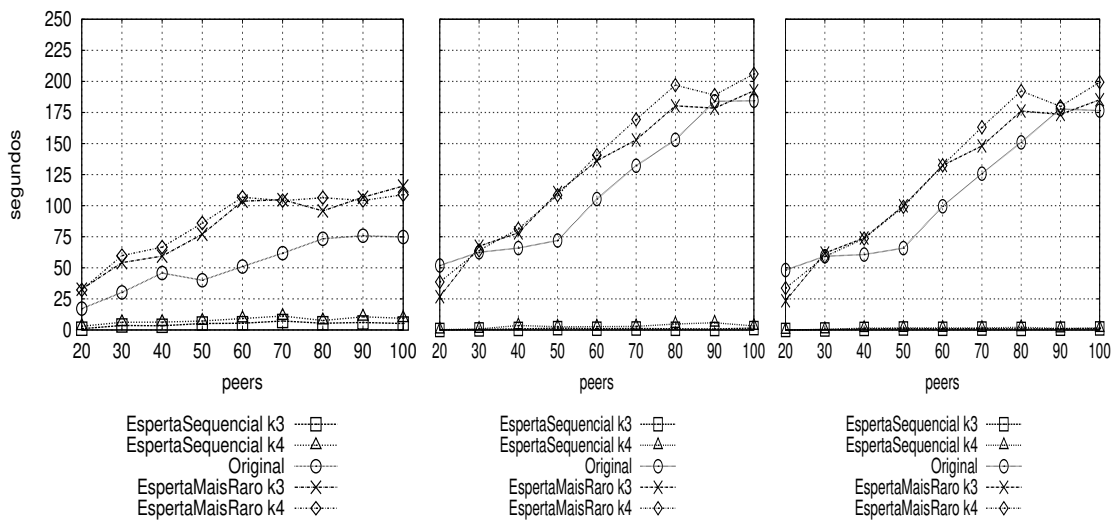


Figura 4: Espera Total (ET): Cenário 1; Cenário 2; e Cenário 3.

Tabela 2: Otimizações alcançadas pela *EspertaSequencial k3*

Métrica	<i>EspertaSequencial k3</i>				Original				Otimização			
	DO (kbps)		ET (seg)		DO (kbps)		ET (seg)		DO		ET	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
Cenário 1	1622	1863	1	7	1454	1773	17	76	11,6%	5,1%	94,1%	90,7%
Cenário 2	1663	1764	0	1	1343	1616	52	184	23,8%	9,2%	100%	99,5%
Cenário 3	1681	1780	0	1	1352	1643	48	177	24,3%	8,3%	100%	99,4%

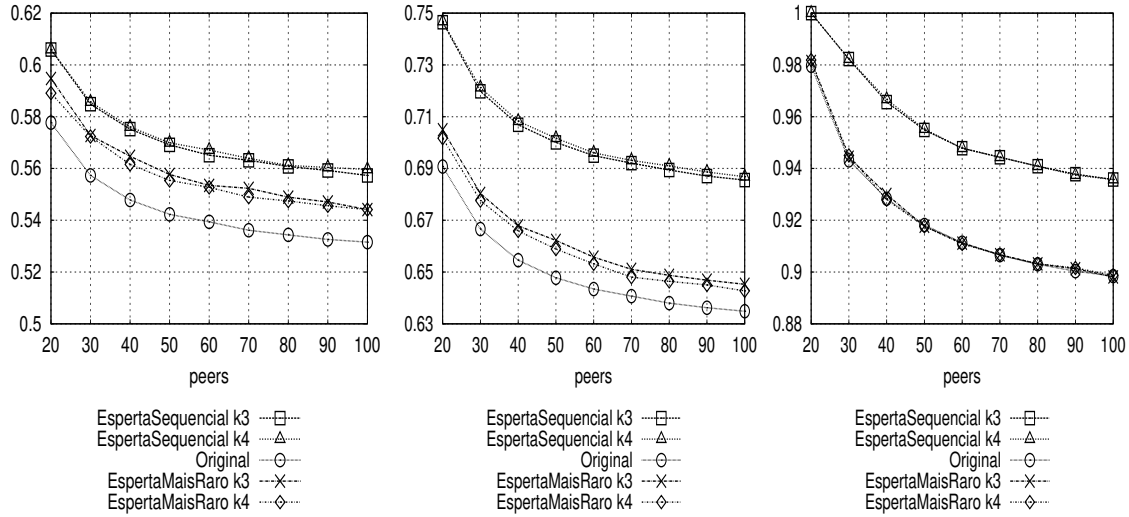


Figura 5: Razão de Peça (*RP*): Cenário 1; Cenário 2; e Cenário 3.

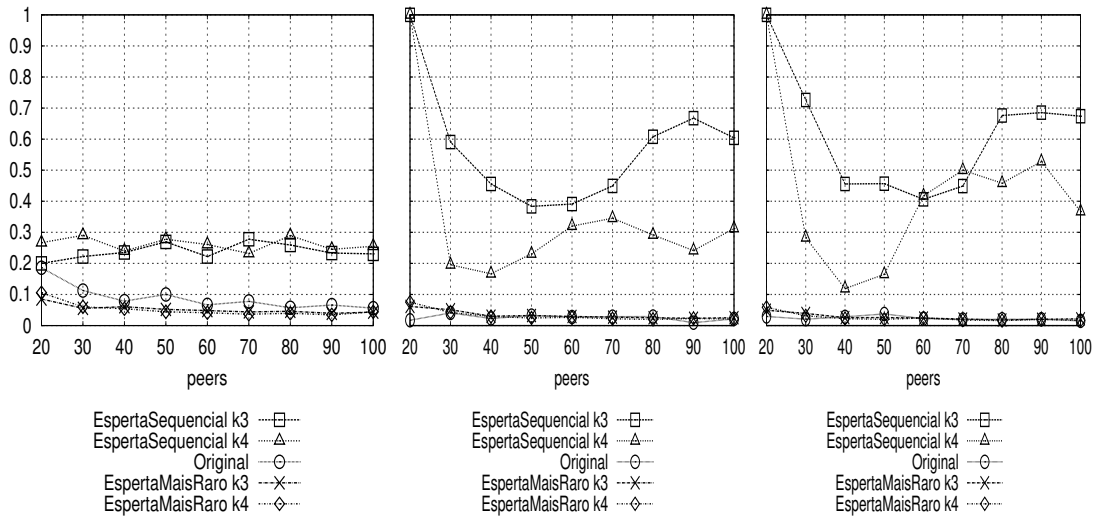


Figura 6: Razão de Espera (*RE*): Cenário 1; Cenário 2; e Cenário 3.

RE. Assim, embora a proposta *Original* tenha desempenho superior àqueles de *EspertaMaisRaro k3* e *EspertaMaisRaro k4*, conforme análise na última subseção, o mesmo não ocorre agora ao examinarem-se exclusivamente as correspondentes políticas de seleção de pedaços.

A justificativa para esse aparente paradoxo é que as métricas *RP* e *RE* avaliam especificamente a política de seleção de pedaços e não o desempenho global da proposta à qual elas se referem. Uma proposta pode então ter um valor de *DO* alto o suficiente (e.g., devido à capacidade de transferência de dados dos *peers*) para, mesmo com um baixo *RP* (i.e., muitos pedaços recebidos não são exibidos), propiciar níveis de QoS e QoE competitivos em relação a outra proposta de mais alto *RP*.

3) Para todas as propostas, os valores de *RP* do Cenário 1 são menores que aqueles do Cenário 2, os quais, por sua vez, são menores que aqueles do Cenário 3. Isso ocorre principalmente pela diferença da quantidade de saltos existentes entre os cenários. Mais precisamente, uma menor interatividade resulta em um melhor desempenho, pois mais pedaços baixados são exibidos localmente. No caso da métrica *RE*, existe de forma análoga uma tendência de melhoria de desempenho conforme se diminui a interatividade, com a peculiaridade de uma maior proximidade entre os correspondentes valores medidos para os Cenários 2 e 3, respectivamente.

7. CONCLUSÕES FINAIS

Com vistas ao serviço de *streaming* de VoD e um acesso de dados predominantemente sequencial, este artigo cumpriu dois objetivos. Primeiro, analisou o paradigma BitTorrent, considerando o algoritmo Interactive BitTorrent - Application (IB-A) [30]. Segundo, propôs e analisou uma nova política de seleção de pedaços do vídeo, denominada de *Política Esperta*. Para tanto, foram realizadas simulações em diferentes cenários, em que avaliaram-se sete métricas de desempenho.

Dentre os principais resultados alcançados, citam-se os seguintes. Primeiro, as propostas baseadas no paradigma BitTorrent podem ainda ser otimizadas em seu desempenho. Segundo, a *Política Esperta* se mostra eficiente em todos os cenários examinados, permitindo otimizações de até 24,3% na taxa de *download*, e de até $\approx 100\%$ no tempo de espera de dados, comparativamente a resultados da literatura. Finalmente, em que pese a melhoria de desempenho devido ao emprego da *Política Esperta*, foi possível verificar que ainda há espaço para atingir maior eficiência. Por exemplo, em certos cenários, mais da metade do tempo é gasto em baixar pedaços do vídeo que não são exibidos localmente.

Como trabalhos futuros e contemplando limitações desta pesquisa, consideramos o seguinte. Primeiro, a análise de políticas de seleção de pedaços para *live streaming* em redes cabeadas e móveis, ante acesso sequencial e não sequencial. Segundo, o desenvolvimento de geradores de cargas sintéticas para auxílio na tarefa de validação de novos protocolos de *streaming*. Finalmente, a consideração de outros domínios de aplicação diferentes do EaD (e.g., [23, 33, 31, 15, 22]).

8. REFERÊNCIAS

- [1] N. Anjum, D. Karamshuk, M. Shikh-Bahaei, and N. Sastry. Survey on peer-assisted content delivery networks. *Computer Networks*, 116:79–95, 2017.
- [2] M. Arantes, F. Figueiredo, and J. M. Almeida. Understanding Video-Ad Consumption on YouTube: A Measurement Study on User Behavior, Popularity, and Content Properties. In *Eighth ACM Conference on Web Science*, Hannover, Germany, 2016.
- [3] M. Arantes, F. Figueiredo, J. M. Almeida, and R. O. Prates. An Investigation of User Actions and Experiences When Exposed to YouTube Video Ads. In *XXIV Brazilian Symposium on Multimedia and the Web (WebMedia'18)*, Salvador, BA, Brasil, 2018.
- [4] 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)*. Springer International Publishing, 1st edition, 2020.
- [5] D. Carvalho et al. Understanding users-contents interaction in non-linear multimedia streaming services. In *XXIV Brazilian Symposium on Multimedia and the Web (WebMedia'18)*, Salvador, BA, Brasil, 2018.
- [6] D. Carvalho, N. Silva, A. Cardoso, E. Fazzion, A. Pereira, and L. Rocha. A Data Exploratory Methodology to Understand the Users' Interactions in Nonlinear Web Multimedia Services. In *XXV Brazilian Symposium on Multimedia and the Web (WebMedia'19)*, Rio de Janeiro, RJ, Brasil, 2019.
- [7] L. Chen, Y. Zhou, and D. M. Chiu. Video Browsing - A Study of User Behavior in Online VoD Services. In *22nd International Conference on Computer Communication and Networks (ICCCN)*, Nassau, Bahamas, 2013.
- [8] CISCO. Cisco Annual Internet Report (2018–2023). *Cisco White Paper*, 2020. [Online]. Available: <https://www.cisco.com/c/en/us/solutions/index.html>.
- [9] B. Cohen. Incentives build robustness in BitTorrent. In *First Workshop on Economics of Peer-to-Peer System*, Berkeley, USA, 2003.
- [10] L. D'Acunto, N. Chiluka, T. Vinkó, and H. Sips. BitTorrent-like P2P approaches for VoD: A comparative study. *Computer Networks*, 57(5):1253 – 1276, 2013.
- [11] P. Falkowski-Gilski and T. Uhl. Current trends in consumption of multimedia content using online streaming platforms: A user-centric survey. *Computer Science Review*, 37:100268, 2020.
- [12] A. Ghani, E. H. I. El haj, A. Hammouch, and A. Chaoub. Adaptation of Quality for Video Streaming in a Bittorrent P2P IMS Network. In *Third International Conference on Networking, Information Systems & Security*, Marrakech, Morocco, 2020.
- [13] K. P. Gummadi, S. Saroiu, and S. D. Gribble. King: Estimating Latency between Arbitrary Internet End Hosts. In *Second ACM SIGCOMM Workshop on Internet Measurement*, Marseille, France, 2002.
- [14] C. Hammami, A. Gazdar, I. Jemili, and A. Belghith. Study of VOD streaming on BitTorrent. In *International Symposium on Networks, Computers and Communications (ISNCC)*, Yasmine Hammamet, Tunisia, 2015.
- [15] S. Han, Y. Go, H. Noh, and H. Song. Cooperative Server-Client HTTP Adaptive Streaming System for Live Video Streaming. In *2019 International Conference on Information Networking (ICOIN)*,

Kuala Lumpur, Malaysia, 2019.

- [16] 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.
- [17] C. Lin and J. Lin. Streaming videos over BitTorrent by balancing playback continuity and distribution of piece rareness. In *2014 23rd International Conference on Computer Communication and Networks (ICCCN)*, Shanghai, China, 2014.
- [18] V. P. K. M, C. Timmerer, and H. Hellwagner. MiPSO: Multi-Period Per-Scene Optimization for HTTP Adaptive Streaming. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, London, UK, 2020.
- [19] S. Mantzouratos et al. Survey of cross-layer proposals for video streaming over Mobile Ad hoc Networks (MANETs). In *International Conference on Telecommunications and Multimedia (TEMU)*, Chania, Greece, 2012.
- [20] C. Moldovan, F. Wamser, and T. Hofffeld. User Behavior and Engagement of a Mobile Video Streaming User from Crowdsourced Measurements. In *11th International Conference on Quality of Multimedia Experience (QoMEX)*, Berlin, Germany, 2019.
- [21] A. Montresor and M. Jelasity. PeerSim: A scalable P2P simulator. In *IEEE Ninth International Conference on Peer-to-Peer Computing*, Washington, USA, 2009.
- [22] S. Petrangeli, T. Wauters, and F. D. Turck. QoE-Centric Network-Assisted Delivery of Adaptive Video Streaming Services. In *2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, Arlington, VA, USA, 2019.
- [23] I. Ribeiro, L. Castanheira, A. Schaeffer-Filho, W. Cordeiro, and V. Mota. Caracterização de mobilidade e detecção de comunidades baseadas em tópicos de interesse. In *XXXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, Rio de Janeiro, RJ, Brasil, 2020.
- [24] M. Rocha, M. Maia, I. Cunha, J. Almeida, and S. Campos. Scalable Media Streaming to Interactive Users. In *13th Annual ACM International Conference on Multimedia*, Hilton, Singapore, 2005.
- [25] M. V. M. Rocha and C. K. S. Rodrigues. On client's interactive behaviour to design peer selection policies for BitTorrent-like protocols. *International Journal of Computer Networks and Communications (IJCNC)*, 5(5):141–159, 2013.
- [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 (IJCNC)*, 6(5):39–58, 2014.
- [27] C. K. S. Rodrigues and R. M. M. Leão. Técnicas para Sistemas de Vídeo sob Demanda Escaláveis. In *XXIV Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, Curitiba, PR, Brasil, 2006.
- [28] C. K. S. Rodrigues and M. V. M. Rocha. A dispersão de dados como critério para a política de seleção de peers em uma rede BitTorrent para streaming sob demanda interativo. *Revista de Sistemas e Computação*, 7(1):36–43, 2017.
- [29] C. K. S. Rodrigues and V. Rocha. A Competitive Analysis of BitTorrent-like Algorithms for Interactive Video-on-Demand Streaming over MANETs. *IEEE Latin America Transactions*, 19(9):1451–1458, 2021.
- [30] C. K. S. Rodrigues and V. Rocha. Enhancing BitTorrent for efficient interactive video-on-demand streaming over MANETs. *Journal of Network and Computer Applications*, 174:102906, 2021.
- [31] W. E. Shabrina, D. Wisaksono Sudiharto, E. Ariyanto, and M. A. Makky. The QoS Improvement Using CDN for Live Video Streaming with HLS. In *2020 International Conference on Smart Technology and Applications (ICoSTA)*, Surabaya, Indonesia, 2020.
- [32] A. Shetty, S. Mhatre, N. Sinvhal, and K. K. Devadkar. Peer Assisted Parallel Downloading System. In *2019 International Conference on Communication and Signal Processing (ICCSIP)*, pages 0650–0655, 2019.
- [33] A. Streit, R. Leão, E. de Souza e Silva, and D. Menasche. Descobrimos perfis de tráfego de usuários: uma abordagem não supervisionada. In *XXXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, Gramado, RS, Brasil, 2019.
- [34] A. Streit and C. K. S. Rodrigues. Improving BitTorrent's Peer Selection For Multimedia Content On-Demand Delivery. *International Journal of Computer Networks and Communications (IJNCN)*, 7(6):111–124, 2015.
- [35] S. Takahashi, K. Yamagishi, P. Lebreton, and J. Okamoto. Impact of Quality Factors on Users' Viewing Behaviors in Adaptive Bitrate Streaming Services. In *11th International Conference on Quality of Multimedia Experience (QoMEX)*, Berlin, Germany, 2019.
- [36] Z. Tan, Y. Zhang, and W. Hu. Online prediction of video popularity in ovss: A video age-sensitive model with beyond views features. *IEEE Transactions on Broadcasting*, 66(2):241–250, 2020.
- [37] S. Vijaykumar and S. P. T. A Literature Survey on Various Streaming Protocol and Compression Techniques in Multimedia Transmission with Internet of Things. In *2021 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–6, 2021.
- [38] L. Wenjing, Y. Peng, W. Ruiyi, F. Lei, D. Ouzhou, and Q. Xuesong. Quality of experience evaluation of HTTP video streaming based on user interactive behaviors. *The Journal of China Universities of Posts and Telecommunications*, 24(3):24–32, 2017.
- [39] H. Yang, M. Liu, B. Li, and Z. Dong. A P2P Network Framework for Interactive Streaming Media. In *2019 11th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, volume 2, pages 288–292, Hangzhou, China, 2019.