Computers, Materials & Continua
DOI:10.32604/cmc.2021.014635
Article

Tech Science Press

Efficient Algorithms for Cache-Throughput Analysis in Cellular-D2D 5G Networks

Nasreen Anjum[1,*], Zhaohui Yang[1], Imran Khan[2], Mahreen Kiran[3], Falin Wu[4], Khaled Rabie[5] and Shikh Muhammad Bahaei[1]

[1]Department of Informatics, King's College London, London, UK
[2]Department of Electrical Engineering, University of Peshawar, Peshawar, Pakistan
[3]Department of Computer Science, Institute of Management Sciences, Peshawar, Pakistan
[4]School of Instrumentation and Optoelectronic Engineering, Beihang University, Beijing, 100191, China
[5]Department of Electrical and Electronic Engineering, Manchester Metropolitan University, Manchester, UK
[*]Corresponding Author: Nasreen Anjum. Email: nasreenanjum59@gmail.com
Received: 05 October 2020; Accepted: 10 November 2020

Abstract: In this paper, we propose a two-tiered segment-based Device-to-Device (S-D2D) caching approach to decrease the startup and playback delay experienced by Video-on-Demand (VoD) users in a cellular network. In the S-D2D caching approach cache space of each mobile device is divided into two cache-blocks. The first cache-block reserve for caching and delivering the beginning portion of the most popular video files and the second cache-block caches the latter portion of the requested video files 'fully or partially' depending on the users' video watching behaviour and popularity of videos. In this approach before caching, video is divided and grouped in a sequence of fixed-sized fragments called segments. To control the admission to both cache-blocks and improve the system throughput, we further propose and evaluate three cache admission control algorithms. We also propose a video segment access protocol to elaborate on how to cache and share the video segments in a segmentation based D2D caching architecture. We formulate an optimisation problem and find the optimal cache probability and beginning-segment size that maximise the cache-throughput probability of beginning-segments. To solve the non-convex cache-throughout maximisation problem, we derive an iterative algorithm, where the optimal solution is derived in each step. We used extensive simulations to evaluate the performance of our proposed S-D2D caching system.

Keywords: Device-to-Device (D2D); startup-delay; playback-delay; caching

1 Introduction

Mobile communication has gained tremendous popularity over the last decade due to the evolution toward higher generation (G) cellular networks from 1G to 5G. Specifically, the transition to 3G and the successful deployment of 4G technology was a boom in mobile data consumption. Due to the increase in mobile broadband speed, now millions of mobile users are watching videos

on their smart mobile devices. According to the latest Ericsson mobility report, mobile video traffic is forecast to account for 74% of all mobile data traffic in 2024 [1,2].

To confront growing mobile data traffic demands, many technologies have been explored and developed in the last decade, such as massive MIMO (deploying very dense and massive base station (BS) antennas) [3], cognitive radio (reuse of spectrum resources) [4], femtocells (deploying dense and small cells with caching ability) [5], and mm-wave bands (use of additional spectrum) [6]. However, in a real scenario, these methods sometimes can provide limited throughput gain [7] or tend to be very expensive due to the dependency on costly backhaul networks.

Recently, D2D caching network has become the centre of attention of both academia and industry research. D2D communication allows direct communication between proximal devices without traversal data through the BS or core network and without requiring additional infrastructure and maintenance cost. The D2D caching system works on the following mechanism: each device with storage capacity caches a subset of video files at random according to some popularity distribution. When a user requests video content, the individual might either find the desired video file in the local cache or will download from a proximity device through the direct D2D link [8]. In this way, a multitude of devices with heterogeneous and limited storage resources form a common virtual cache space that is capable of caching and sharing a dynamic and large number of video files [9].

1.1 Related Work

Many studies have shown that dissemination of popular video content through the D2D network can improve spectrum utilisation, video throughput, energy efficiency, and user's experience. For instance, in [9], the authors proposed a novel D2D caching architecture that can achieve the video throughput by two orders of magnitude. Similarly, a measurement study for cellular users of BBC iPlayer [10], a popular video-on-demand service in the UK, reported that in a realistic propagation environment D2D caching system has the potential to improve the network throughput by two orders of magnitude. To study the impact of scaling behaviour of D2D caching network on the throughput scale, the authors in [9] proposed a joint optimisation of caching and delivery strategy to maximise the number of D2D active links in a cell. The simulation results showed that the performance of D2D caching network in terms of the number of D2D active links is strictly dependent on the value of Zipf parameter. The authors also evaluate the optimal cache distribution and transmission distance for each user in a cell. The authors in [11] proposed the idea of D2D communication as an underlay and proved that D2D caching network has the potential to improve the spectral efficiency significantly. To achieve the guaranteed quality of experience (QoE) of video streaming applications in a cellular network, the authors in [12] proposed a user-centric video transmission mechanism based on D2D communication. In the proposed approach, the authors jointly considered the asynchronous content reuse feature of VoD applications, users' locations, willingness to share their storage and up-link resources, and QoE requirements. The simulation results showed that the proposed mechanism can improve the users' QoE up to 85%. The authors in [13] proposed an intelligent cache management scheme for the D2D cache network. This scheme guarantees the optimum number of D2D devices required to support the successful delivery of the most popular requested content.

1.2 Motivation and Contributions

Interestingly, Golrezaei et al. [9–13] have focused on caching techniques that exploit the asynchronous content reuse feature of VoD streaming applications, whereby the same video content

is requested repeatedly by a large number of users. In simple words, caching is optimised based on file popularity. However, the existing literature has ignored one of the most critical features of on-demand video content; namely, *user abandonment behavior*; *when a user abandons the video before completion after watching a few video chunks*.

The users abandoned the videos due to a variety of reasons such as lack of interest, a multitude of choices in video content, frequent re-buffering, length of video content, and long startup delays. For instance, authors in [14–16] reported some compelling findings regarding initial viewer abandonment due to long startup delays. According to their findings, users start to abandon videos after two-second startup delay, with 6% additional abandonment rate per second after that. Similarly, the authors in [17] reported the traces of 7000 YouTube video files that showed the higher average completion rate for the most popular video files. However, even for the most popular ones, on average, only 72% of each video is watched—furthermore, the average watch-time decreases as the duration of video increases.

Under such circumstances, due to heterogeneous and limited users' cache resources[1], all video files should be cached in a small and fixed-size portion called segment [19]. Second, long startup delays can frustrate users and make them leave the website forever. Therefore, to deal with the problem, sufficient starting portion of videos should always be cached internally. Third, only the most popular video files should be cached entirely. For a least popular file, a small portion or none of it should be cached.

Based on the observations mentioned above and different from the existing D2D caching algorithms [9–13], our contributions in this article are as follows:

i. We proposed a two-tiered S-D2D caching approach by taking into considerations video popularity and users' video abandonment behaviour. Our simulation results show that by employing S-D2D caching approach, the VoD users in a cellular network can experience a decrease in startup-time and playback-delay. We also propose a video segment access protocol to elaborate on how to cache and share the video segments in a segmentation based D2D caching architecture.
ii. To control the admission of video segments to both blocks of cache, we propose the (a) Beginning-Segments Cache Policy (BSCP), (b) Selective Partial Cache Policy (SPCP), and (c) Short Length Video Cache Policy (SLVCP). We propose these caching algorithms based on the findings of an extensive statistical studies which had been conducted to analyse the video viewing behaviour of millions of VoD users. To the best of our knowledge, none of the existing work has examined the effectiveness of the segmentation-based partial caching approach for large and small videos in a D2D communication scenario.
iii. We derive an optimisation approach in a stochastic D2D caching scenario to maximise the cache-throughput probability of the beginning-segments. The cache-throughput probability is the sum of successful requests served by the local caches (self-hit probability) of the requesting device and through the D2D link (cache-hit probability). We take into consideration the size of the beginning-segments and realistic network characteristics such as interference, shadowing, and success probability. To solve the non-convex cache-throughout

[1] The cache space of mobile devices in literature is handled as a rich resource. However, they possess heterogeneous and limited storage resources [18]. Additionally, the size and the popularity of the video files are highly dynamic. On the one hand, it may consume a whole cache space. On the other hand, the transient nature of the wireless devices will also prevent them from sharing the entire content.

maximisation problem, we derive an iterative algorithm, where the optimal solution found in each step.

iv. Finally, the admission control algorithms are evaluated and compared through numerical simulations in a realistic channel model, based on a practical indoor-hotspot WINNER-II propagation environment [20]. The simulation results prove that the caching algorithm for SLVCP outperforms all caching policies. For instance, 50% to 95% users can start the video with zero startup-time, and 47.5% users can download the remaining segments with zero playback-delay through the local-cache, and 31% of users can download the remaining segments of the desired short length videos from their neighbouring devices.

1.3 Paper Organization

The remainder of the article is structured as follows: Section 2 discusses the proposed two-tiered S-D2D caching approach in detail. Sections 3–5 elaborate the BSCP, SPCP and SLVCP. The system model is presented in Section 6. Section 7 presents the simulation results which illustrate the performance of the proposed caching policies in terms of the average cache-hit ratio, average self-hit ratio, and average cache-throughput ratio. Finally, Section 8 concludes the whole article.

2 S-D2D Caching System

In this section, we first discuss our proposed segmentation strategy for the video file and the user's cache space. Then, the video segment access protocol in an S-D2D caching network discussed in detail. Finally, we turn our discussion to the caching policies.

2.1 Segmentation Strategy

The segmentation strategy we use for the video files in an S-D2D caching architecture is illustrated in Fig. 1.

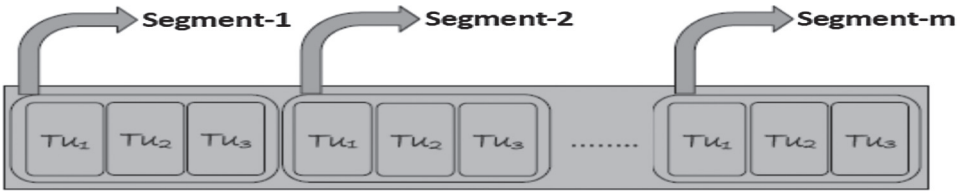

Figure 1: Segmentation strategy for the video files in a S-D2D caching architecture

The video file j is sliced into small pieces of equal size transmit units (TUs) before it transmitted to the end-user. The TU is the basic building block of information transmission in a communication system. Depending on the cache system requirements, multiple TUs grouped into equal-sized segments. The number of TUs grouped in each segment i is $s_{(i,j)}$, for $i \geq 1$. In simple words, a video content j is segmented uniformly in equal size lengths, i.e., $s_{(i,j)} = s_{(i,j)} =, \ldots, s_{(M,j)}$; where M is the total number of segments of a video content j. For the sake of tractability, we assume that the size of the video segment i should not exceed the size of a video file j and storage capacity C of each D2D device.

We divide the users' cache space into two blocks of different sizes, for example, see Figs. 2 and 3. The size of Block-1 is small and dedicated to caching only the beginning-segments of the desired video content. The Block-2 caches the subsequent segments of the video files 'partially or fully' depending on the users' video-watching behaviour.

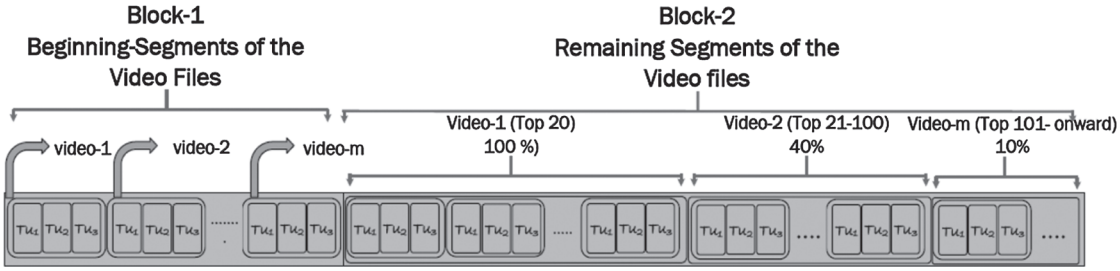

Figure 2: Segmentation strategy for a large size video file in a S-D2D caching architecture

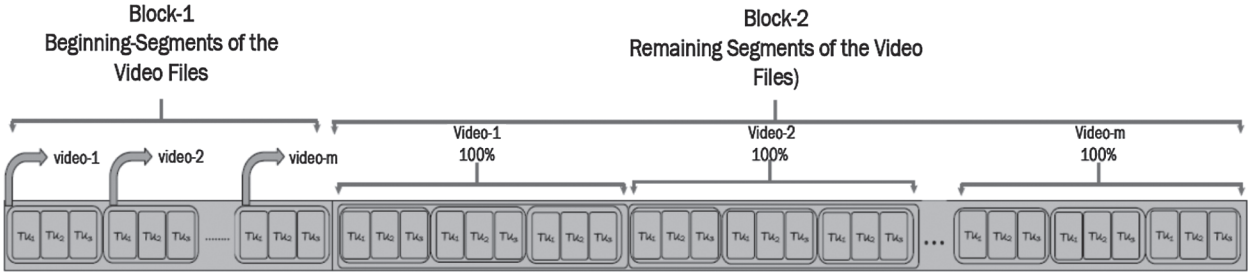

Figure 3: Segmentation strategy for a small size video file in a S-D2D caching architecture

2.2 Video Segment Access Protocol

Each D2D device in the S-D2D caching system has a storage capacity that enables two proximity devices to cache and share the video segments over a direct link. We assume that a cellular user, willing to distribute and receive the video segments via an S-D2D caching system, must first install special software—we named it "D2D service interface software (D-SIS)". The D-SIS implemented as a background application software that runs when the cellular user is logged into his/her system and requests video content. The main objective of the D-SIS is to make the cache placement, video segmentation and communication process between a pair of devices transparent and reliable. The proposed S-D2D caching system is operator/network controlled.

When a cellular user requests a video file, the D-SIS first performs a self-search process before placing the request to the BS. We termed it as a self-hit. For a fast startup, the D-SIS searches Block-1 of the user's cache space reserved for caching only the beginning-segments. The video will start with zero startup-time if the user has cached the beginning-segments of the desired content in its local storage. In this case, if the self-search process is unsuccessful, the D-SIS contacts the core network for the list of potential D2D (P-D2D) devices. The P-D2D device has the copy of the desired video segments in its local storage and located nearby of the requesting device. The core network filters the information stored in its central database[2] to find P-D2D devices. Once the P-D2D device or devices have been found, BS sends the list to the requesting device. We termed

[2] The central database keeps the identification information of each P-D2D device such as device-ID, video-ID and its geographic position. The device-ID is a sequence of unique bits that uniquely identify the devices. The video-ID is a string of bits that uniquely identify the video. The geographic position of the mobile device can be discovered by the Global Positioning System (GPS).

it as a cache-hit. After receiving the list of the P-D2D devices, we assume that the requesting device discovers and establishes a connection with one of the P-D2D devices using the discovery method described in [21].

Unluckily, suppose for some reason such as bad link quality, the requesting device is unable to download the segments from the neighbouring P-D2D, it will either start to download the segments from one of the P-D2D devices provided in a list, or the BS must serve the request. Similarly, in a case, the core network is unable to provide the desired list of P-D2D devices; it will forward the beginning-segments to the requesting device through the traditional cellular communication system. For service continuity and downloading the subsequent segments, the D-SIS will repeat the same procedure as discussed above and will explore the Block-2 of the user's cache space dedicated to caching the later segments of the video files.

3 Beginning-Segments Cache Policy (BSCP)

Beginning-segments always receive preferential treatment in the S-D2D caching approach, and video always starts with low startup latency. Algorithm 1 illustrates the BSCP for the admission as well as the replacement of the beginning-segments.

Algorithm 1: Caching algorithm for beginning-segment i of a video files j from a library of size L

Size-of-beginning-segment $= s_{(i,j)}$;
Size-of-Block $- 1 = \mathcal{B}_1$;
a cellular user requests for the beginning-segment i of a video file j from a library L.
if $\left(s_{(i,j)} \leq \mathcal{B}_1\right)$ **then**,
    cache the beginning-segment i in the Block-1 and play it.
$\mathcal{B}_1 \leftarrow \mathcal{B}_1 + s_{(i,j)}$; ▷ **update the cache Block-1**.
end if
if $\left(s_{(i,j)} \geq \mathcal{B}_1\right)$ **then**, ▷ **cache space in the Block-1 is not sufficient.**
find a replacement for an incoming beginning-segment and replace it with a beginning-segment that has the least access. Let beginning-segment i of a video file k fulfils the replacement criteria, then replace the beginning-segment of a video file j with the beginning-segment of a video file k.
$\mathcal{B}_1 \leftarrow \mathcal{B}_1 - s_{(i,j)}$; ▷ **update the cache block-1**.
end if
if (the beginning-segment i of a video file j has been successfully cached in a block-1 of a requesting user) **then**,
    invoke the Algorithm 2 for caching the subsequent segments of a video file 'j' in a block-2 of a requesting user.
end if

In general, cellular users request video content according to some popularity distribution. Many studies have reported the skewed distribution of users' interest toward a small fraction of top-ranked content. For instance, authors in [21,22] found that the first top 20% of the most popular video content accounts for up to 84% of the total video views on YouTube. Therefore, beginning-segments belonging to the most popular video files are permitted to cache on Block-1. The popularity of video files is measure by the number of viewers attracted to them. Many studies have reported Zipf-like distribution as a popular and well-established model to measure the popularity of video files [9].

We assume that a cellular user requests a segment i of a video file j from the library of size L. We also assume that the video and its segments share the same popularity distribution. The popularity of the requested segment 'i' of the video file 'j' is denoted by $p_{(i,j)}$ and is inversely proportional to its rank.

$$p_{(i,j)} = \frac{\frac{1}{j^{\gamma_r}}}{\sum_{k=1}^{L} \frac{1}{k^{\gamma_r}}}, \quad 1 \leq j \leq L, \tag{1}$$

where γ^r is a value of the exponent characterizing the distribution, i.e., larger the value of γ^r, more popular files are requested by the users. For the sake of admission control, we assume that each mobile device will cache the beginning-segments of the most popular video files according to the cache probability $q_{(i,j)}$. The caching probabilities of the beginning-segments in a video files j is denoted as $q = [q_1, q_2, \ldots, q_r]$. Because cache capacity is a limited resource. Therefore, we have $\sum_{i=1}^{M} \sum_{j=1}^{L} q_{(i,j)} \leq$ C. To create room for the incoming new beginning-segments, traditional least recently used (LRU) eviction policy will be used.

4 Selective Partial Cache Policy (SPCP)

One of the most straightforward approaches for caching the later segments of a large video file in Block-2, we keep on caching the subsequent segments immediately after the beginning-segments until the cache is full. However, to achieve higher throughput, this approach may not be feasible for the D2D caching system. For instance, a two-hour-long YouTube 1080p video file consumes 7.36 GB cache capacity. It is more likely that for a few popular video files such as top 20 movie content users will watch the content in its entirety. For less popular content a cellular user may abandon an ongoing video session before its completion. For instance, a comparison between the abandon rate of mobile and fixed-line users has conducted in a study of BBC iPlayer accesses. The results suggested that mobile users abandon sessions with a higher rate, i.e., only around 30% of mobile sessions last for longer than a half of a content's duration in comparison to around 50% for the fixed-line sessions [22]. The authors in [23] show that for an hour-long video session, only 40% of total videos completely downloaded, while 50% of videos abandoned after downloading only 60% portion. According to [17], users watch only 10% of the least popular video files. In this scenario, caching a whole video file based on only popularity may not work.

Based on the measurement studies [17,21–23] conducted to evaluate the users' video abandonment behaviour, we proposed Algorithm 2 for the Block-2. The most popular video files (Top 20) will always be cache entirely [21,22]. For less popular video files (Top 21–100) 40% subsequent portion will always be cache [17,22,23]. For the least popular video files (Top 101-onwards), only 10% subsequent portion of the video file will be cache [17]. Fig. 2 shows the example of the size distribution of the users' cache for the large size media content.

When cache space required for the incoming new video file, the video chunk of the least popular video file is evicted to make room for the video chunks of the most popular video files.

5 Short Length Video Cache Policy (SLVCP)

Many studies have reported that shorter videos have a longer viewing time and the abandonment rate increases as the video length increases. For instance, the authors in [24] reported that,

for the small size MSN videos, users generally opt to view the entire or most of the video clip, and only 20% of users watch 60% of video content with length greater than 30 min.

A similar observation is reported in a study [25]. For a video length between 3–5 min, the video abandonment rate was only 27.1%, while when the size of video increases by up to 10 min, the abandonment rate is raised by up to 62.5%. Interestingly, 40–50% of mobile users watch videos shorter than 3 min of duration [26]. Based on these findings, for short length videos, we consider the following admission cache rules:

- As video abandonment rate for small size videos is low in comparison to large size videos; therefore, for small size videos (3–5 min long), we propose to cache the whole video file. For clarity, Fig. 3 shows the cache space distribution for a small size video file.
- However, like the BSCP, only the subsequent segments of the most popular videos can cache on Block-2. The Zipf distribution model will be used to measure the popularity of short length multimedia files. In short, we follow the same cache policy as described in Algorithm-1 for caching and replacing the later segments of the small video files into the Block-2.

6 System Model and Problem Formulation

In this section, we derive optimal cache probability and optimal beginning-segment size for the BSCP that maximises the cache-throughput probability. First, we discuss the system model. Then, the optimisation problem is formulated, and a solution is derived.

6.1 System Model

In this section, we derive optimal cache probability and optimal beginning-segment size for the BSCP that maximises the cache-throughput probability. First, we discuss the system model. Then, the optimisation

Algorithm 2: Caching algorithm for the subsequent segments of a large video file j from a library of size L

1: Size-of-Block $-2 = \mathcal{B}_2 = 0$;
2: Size-of-video-file $-$ j $= Size_j$;
3: Size-of-video-file-k $= Size_k$;
4: Size-of-video-library $=$ L $= 1000$;
5: **for** each request of a video content j, **do**
6: **while** $\left(\mathcal{B}_2 \leq Size_j\right)$ **do**
7: **if** $(j \geq 1 \& j \leq 20)$**then,** ▷ **The video content j is the Top 20 content in a video library of size L.**
8: cache the all (100%) subsequent segments of a video content j into the cache Block-2 of a requesting device.
9: $\mathcal{B}_2 \leftarrow \mathcal{B}_2 + Size_j$; ▷ **update the cache Block-2.**
10: **end if**
11: **if** $(j \geq 21 \& j \leq 100)$ **then,** ▷ **The video content j is the Top 20-100 popular video content in a video library of size L.**

(Continued.)

12: cache the 40% subsequent portion of a video content j into the cache Block-2 of a requesting device.
13: $\mathcal{B}_2 \leftarrow \mathcal{B}_2 + Size_j$;
14: **end if**
15: **if** $(j \geq 101 \& j \leq L)$**then**, ▷ **The video content 'j' is least popular content (101-1000).**
16: cache the 10% subsequent portion of a video content 'j' into the cache Block-2 of a requesting device.
17: $\mathcal{B}_2 \leftarrow \mathcal{B}_2 + Size_j$;
18: **end if**
19: **end while**
20: **while** $\left(Size_j \leq \mathcal{B}_2\right)$**do**,
21: **if** (the cache space in a Block-2 is not sufficient) **then**,
22: find a replacement for an incoming video content j in a cache Block-2. Replace it with a least popular video content. Let video file k fulfills the replacement criteria, then evict the subsequent segments of video content k.
23: $\mathcal{B}_2 \leftarrow \mathcal{B}_2 - Size_k$;
24: **end if**
25: **end while**
26: **end for**

Each P-D2D has a probability $\xi \in [0, 1]$ that defines its status as an "**active**" or "**inactive**". A P-D2D device is said to be in an **active** state when it requests a segment of a video file, and it said to be in an **inactive** state when it serves the request for a segment of the desired video file. Based on this probability, the distribution of P-D2D transmitters follows homogeneous Poisson Point Process (PPP) $\phi_{\mathbf{u}}^{\mathbf{r}}$ with intensity $\xi\lambda_{\mathbf{u}}$ and, the distribution of P-D2D receivers follows homogeneous PPP $\phi_{\mathbf{u}}^{\mathbf{t}}$ with an intensity $(\mathbf{1}-\xi)\,\lambda_{\mathbf{u}}$. Each P-D2D device caches the video segments independently with cache probability $q_{(i,j)}$, therefore according to the thinning property of the PPP, the distribution of video segments follows a homogeneous PPP with intensity $\boldsymbol{q}_{(\boldsymbol{i},\boldsymbol{j})}\,(1-\xi)\,\lambda_u \boldsymbol{s}_{(\boldsymbol{i},\boldsymbol{j})}$.

Each P-D2D device in the system model can communicate with each other over a single direct link sing the cellular spectrum resources, as well as with the BSs through a traditional cellular communication system. Since the distance between P-D2D devices is typically small, multiple D2D active links can exist throughout the cellular region. Furthermore, we will also consider the interference received at each P-D2D receiver caused by the powerful signals transmitted by the BSs and other active D2D links within and outside the cell. For simplicity, we also assume that all P-D2D devices and BSs use the same transmission power. The transmission power determines the actual transmission range and can be optimised centrally. Typically, there is a trade-off between the transmission power and the probability of availability of the P-D2D devices caching the requested video segments. It indicates that higher transmission power leads to higher transmission coverage area and hence, increases the probability of finding the requested video-segments within the vicinity of the requesting users [27,28]. However, this consumes significant battery power resources of mobile devices. We also assume that the D2D communication does not interfere with the communication between the BS and the cellular users. We assume that P-D2D devices are operating on the orthogonal/dedicated spread spectrum resources. For the measurement of the D2D caching system throughput, we do not need to consider explicitly the cellular users and their associated communications.

Next, we derive the optimal beginning-segment size $s_{(i,j)}$ of the video file $j \in \{1,\ldots,L\}$ requested by the user $n \in N$. Each user 'n' requests the beginning-segment i of a video file j from a library of size L according to the request probability distribution $p_{(i,j)}$ as given in Eq. (1).

Self-Hit Probability: According to our segments access protocol; the requesting user first finds the beginning-segment $s_{(i,j)}$ of the desired video file in its local storage through the self-search process. If the requesting user finds the beginning-segment of the desire video file in its local cache, then self-hit probability will occur. In this case, no D2D communication will take place. We represent the self-hit probability by $p_{(self-hit)}$ for the request of $s_{(i,j)} \in j$.

$$p_{(self-hit)} = \sum_{i=1}^{L}\sum_{j=1}^{M} p_{(i,j)} q_{(i,j)} s_{(i,j)}, \tag{2}$$

Cache-Hit Probability: Now, we consider a second case, when the self-search process is unsuccessful, and the user sends a request to the BS for a list of the P-D2D devices. The probability of finding the $s_{(i,j)}$ of the video file j inside a particular area strongly depends on the popularity order of the video file, transmission range, the density of P-D2D devices and the size of $s_{(i,j)}$. The probability that the requesting user can find a P-D2D device within its transmission range ε_d is given by

$$p^{\varepsilon_d}_{(Cache-hit,i,j)} = 1 - e^{-\pi(1-\xi) q_{(i,j)} s_{(i,j)} \epsilon_d^2}, \tag{3}$$

Averaging over all the beginning-segments of the video files in a content library L, we have the D2D cache-hit probability as follows

$$p^{\varepsilon_d}_{(Cache-hit,i,j)} = \sum_{i=1}^{M}\sum_{j=1}^{N} p_{(i,j)} \left(1 - q_{(i,j)}\right) s_{(i,j)} p^{\varepsilon_d}_{(Cache-hit,i,j)}, \tag{4}$$

Thus,

$$p^{\varepsilon_d}_{(Cache-hit,i,j)} = \sum_{i=1}^{M}\sum_{j=1}^{L} p_{(i,j)} \left(1 - q_{(i,j)}\right) s_{(i,j)} \left(1 - e^{-\pi(1-\xi) q_{(i,j)} s_{(i,j)} \varepsilon_d^2}\right). \tag{5}$$

Cache-Throughput Probability: We define the total cache-throughput probability as the *sum of self-hit probability and the cache-hit probability (when the self-search process is unsuccessful)*. Mathematically, it is represented as $p_{Cache-throughput} = p_{Self-hit} + p^{\varepsilon_d}_{(Cache-hit,i,j)}$. After substituting the corresponding values, we have

$$p_{Cache-throughput} = \sum_{i=1}^{M}\sum_{j=1}^{L} p_{(i,j)} q_{(i,j)} s_{(i,j)} + \sum_{i=1}^{M}\sum_{j=1}^{L} p_{(i,j)} s_{(i,j)} \left(1 - q_{(i,j)}\right) e^{-\pi(1-\xi) q_{(i,j)} s_{(i,j)} \varepsilon_d^2}, \tag{6}$$

Here, we are mainly interested in optimising the size of the beginning-segment and the cache probability that increase the average number of requests that can be successfully and simultaneously handled by the P-D2D devices per unit area. In the self-hit probability case, the request automatically served with probability one. In the cache-hit probability case, the success probability

of content delivery depends on the received signal-to-interference-plus-noise ratio (SINR). Thus, we have the cache-throughput as follows:

$$p_{Cache-throughput} = \rho\lambda_u \left(\sum_{i=1}^{M}\sum_{j=1}^{N} p_{(i,j}s_{i,j} + \sum_{i=1}^{M}\sum_{j=1}^{N} p_{(i,j}(1-q_{(i,j)} \right) s_{(i,j)} p_{(Cache-hit,i,j)}^{\varepsilon_d} p_{(succ,i,j)}^{\varepsilon_d}, \tag{7}$$

where $\rho\lambda_u$ is the number of requests for the beginning-segments and $p_{(succ,i,j)}^{\varepsilon_d}$ indicates the expected probability of success in terms of good channel quality necessary to carry out the successful transmission between the P-D2D devices. Hence, the probability of successful reception of the beginning-segment i of the video content j' at a receiving node is given by

$$p_{(succ,i,j)}^{\varepsilon_d} = \mathrm{E}\left(\rho\left(\mathrm{SINR}_{(i,j)}^{\varepsilon_d} \geq \beta\right)\right). \tag{8}$$

The Eq. (8) states the condition that, if the SINR (Signal-to-Interference-Plus-Noise-ratio) at the receiver is greater than the predetermined SINR threshold β, then the P-D2D device caching the segment i of a video content j will be selected as a P-D2D transmitter from the list of P-D2D devices. The SINR denotes the ratio of the transmit power and the noise power spectral density. The SINR can be computed as follows:

$$SINR = \frac{P_t |h_{(t,n)}|^2 \varepsilon_{(d,t,n)}^{-\alpha}}{W_1 + W_2 + \delta_n^2}, \tag{9}$$

$$W_1 = \sum_{\substack{k \neq t \\ k \in \phi_n^{d_n}}} P_k |h_{(k,n)}|^2 \varepsilon_{(d,t,n)}^{-\alpha}, \tag{10}$$

$$W_2 = \sum_{\substack{m=1 \\ m \in \psi_n^{d_n}}} P_c |g_{(m,n)}|^2 \varepsilon_{(d,m,n)}^{-\alpha}, \tag{11}$$

where P_t is the transmission power of the P-D2D transmitter **t**, $|h_{(t,n)}|^2$ accounts for the small scale channel fading from the D2D transmitter **t** to the D2D receiver **n**, $\varepsilon_{(d,t,n)}$ is the distance between the requesting user **n** and the P-D2D transmitter **t**, $\boldsymbol{\alpha}$ is the path loss exponent, $\phi_n^{d_n}$ is the set of all active D2D pairs that are causing interference at the receiver **n**, P_k is the sum of transmission power of all the P-D2D transmitters, P_c is the sum of transmission power of all the BSs, and $\psi_n^{d_n}$ represents the set of all BSs that are causing interference at the receiver **n**.

As the transmission power of all P-D2D devices and the BSs fixed, therefore after simplifying the Eq. (8) we get

$$p^{\varepsilon_d}_{(succ,i,j)} = E\left[\rho\left(|h_{(t,n)}|^2 > \frac{\beta_{(t,n)}\varepsilon^{+\alpha}_{(d,t,n)}}{P_t}\left(\delta^2_n + \sum_{\substack{k\neq t\\ k\in\phi^{d_n}_n}} P_k|h_{(k,n)}|^2\varepsilon^{-\alpha}_{(d,k,n)} + \sum_{\substack{m=1\\ m\in\psi^{d_n}_n}} P_c|g_{(m,n)}|^2\varepsilon^{-\alpha}_{(d,m,n)}\right)\right)\right] \tag{12}$$

According to [29], the interference component of the success probability can be calculated by determining the Laplace transform of all interference powers received at node 'n', so the Laplace transform of the interference evaluated at 'n' is given as

$$p^{\varepsilon_d}_{(succ,i,j)} = \mathcal{L}I_{cd}\left(\beta_{(t,n)}\varepsilon^{+\alpha}_{(d,t,n)}\frac{P_c}{P_t}\right)\mathcal{L}I_{dd}\left(\beta_{(t,n)}\varepsilon^{+\alpha}_{(d,t,n)}\right)\exp\left(\frac{-\beta_{(t,n)}\varepsilon^{+\alpha}_{(d,t,n)}}{P_j}\delta^2_n\right), \tag{13}$$

$$\approx \exp\left[\frac{-\pi\varepsilon^2_{(d,t,n)}\beta^{\frac{2}{\alpha}}}{\sin\left(\frac{2}{\alpha}\right)}\left(\left(\frac{P_c}{P_t}\right)^{\frac{2}{\partial}}(\lambda_n+\lambda_d)\right)\right], \tag{14}$$

Considering the noise is negligible when comparing with $\mathcal{L}I_{cd}$ and $\mathcal{L}I_{dd}$, we get the interference from the BSs with normalised power as follows:

$$I_{cd} = \sum_{m\in\psi^{d_n}_n}|g_{(m,n)}|^2\varepsilon^{-\alpha}_{(d,m,n)}, \tag{15}$$

the interference from the P-D2D devices with normalised power is obtained as follows

$$I_{cd} = \sum_{k\in\phi^{d_n}_n}|h_{(k,n)}|^2\varepsilon^{-\alpha}_{(d,k,n)} \tag{16}$$

6.2 Cache-Throughput Optimization Problem

One of the key objectives of this article is to maximise the cache-throughput probability of the beginning-segments. Based on our analysis the problem can be formulated as

$$\underset{s_{(i,j)}q_{(i,j)}}{maximize}\sum_{i=1}^{M}\sum_{j=1}^{L}p_{(i,j)}q_{(i,j)}s_{(i,j)} + \sum_{i=1}^{M}\sum_{j=1}^{L}\left(p_{(i,j)}s_{(i,j)}\left(1-q_{(i,j)}\right)\left(1-e^{-\pi(1-\xi)q_{(i,j)}s_{(i,j)}\varepsilon^2_d}\right)p^{\varepsilon_d}_{(succ,i,j)}\right), \tag{17}$$

$$\text{Subject to } \sum_{i=1}^{M}\sum_{j=1}^{L}s_{(i,j)} \le f_j, \forall(i,j), \tag{18}$$

$$\sum_{i=1}^{M}\sum_{j=1}^{L} s_{(i,j)} \leq C, \forall (i,j), \tag{19}$$

$$0 \leq q_{(i,j)} \leq 1, \forall (i,j), \tag{20}$$

$$0 \leq s_{(i,j)} \leq f_i, \forall (i,j). \tag{21}$$

where, f_i is the size of the video file j. By employing the variable transformation: $T_{(i,j)} = q_{(i,j)} s_{(i,j)}$, our non-convex optimisation problem (17) is equivalent to

$$\underset{s_{(i,j)} T_{(i,j)}}{maximize} \sum_{i=1}^{M}\sum_{j=1}^{L} p_{(i,j)} T_{(i,j)} + \sum_{i=1}^{M}\sum_{j=1}^{L} \left(p_{(i,j)} \left(s_{(i,j)} - T_{(i,j)} \right) \left(1 - e^{-\pi (1-\xi) T_{(i,j)} \varepsilon_d^2} \right) p_{(succ,i,j)}^{\varepsilon_d} \right), \tag{22}$$

$$\text{Subject to } \sum_{i=1}^{M}\sum_{j=1}^{L} s_{(i,j)} \leq f_j, \forall (i,j), \tag{23}$$

$$\sum_{i=1}^{M}\sum_{j=1}^{L} s_{(i,j)} \leq C, \forall (i,j), \tag{24}$$

$$0 \leq T_{(i,j)} \leq s_{(i,j)}, \forall (i,j), \tag{25}$$

$$0 \leq s_{(i,j)} \leq f_i, \forall (i,j). \tag{26}$$

To solve the non-convex optimisation problem (22), an iterative algorithm is proposed. With fixed, problem (22) is a linear problem, which can be effectively solved. With fixed, problem (22) can be shown to be convex. To show this, we define g(x) such that,

$$g(x) = \left(s_{(i,j)} - x \right) \left(1 - e^{-\pi (1-\xi) \varepsilon_d^2 x} \right), \tag{27}$$

by partial derivative Eq. (27) with respect to 'x', we get

$$g'(x) = -1 + e^{-\pi (1-\xi) \varepsilon_d^2 x} + \pi (1-\xi) \varepsilon_d^2 \left(s_{(i,j)} - x \right) e^{-\pi (1-\xi) \varepsilon_d^2 x}, \tag{28}$$

by partial derivative Eq. (28) with respect to 'x', we get

$$g''(x) = -\pi (1-\xi) \varepsilon_d^2 e^{-\pi (1-\xi) \varepsilon_d^2 x} - \pi^2 (1-\xi)^2 \varepsilon_d^4 (s_{(i,j)} - x)) e^{-\pi (1-\xi) \varepsilon_d^2 x}, \tag{29}$$

$$-\pi^2 (1-\xi)^2 \varepsilon_d^4 \left(s_{(i,j)} - x \right) e^{-\pi (1-\xi) \varepsilon_d^2 x} \leq 0. \tag{30}$$

which shows that the objective function of problem (22) is concave. Since all the constraints of problem (22) is linear, problem (22) is convex, which can be effectively solved via the interior point method. To avoid the complexity of the computation, we solve our problem numerically using the Monte Carlo simulations. The Monte Carlo simulations asymptotically converge to the correct probability after 1000 Monte Carlo iterations.

7 Performance Evaluation

In this section, the performance of the S-D2D caching system evaluated using Monte-Carlo simulations. We first discuss the propagation environment and the channel model. Then simulation results are presented. The values of parameters we use in the simulation experiments are summarised in Tab. 1.

Table 1: Parameter values of the channel model

Parameters	Values
Building dimension	100 m^2
Street width	20 m^2
λ_b (BS density)	5 m
λ_u (UEs density)	200
UE Height	1.5 m
ε_d (Transmission distance)	100 m
B (D2D/Cellular transmission)	20 MHz
f_c (D2D Transmission)	2.1 GHz
f_c (Cellular Transmission)	2.45 GHz
p_t (D2D Transmitter Power)	20 dBm
p_c (BS Transmission)	43 dBm
G_t (D2D Transmission)	12 dBm
G_t (Cellular Transmission)	12 dBm
G_r (D2D/Cellular Transmission)	0
A_1, A_2, A_3	37.8, 36.5, 23
Floors	4
$5n_w$	light wall loss parameter for n = 1
β (SINR threshold)	[−20 20]
F_N (Noise power density)	6
$\sigma\mathcal{L}_s$(Body shadowing loss)	4.2
Ψ_σ (Shadowing parameter)	log-normal distribution (mean = 0, σ = 6)

7.1 Propagation Environment

We consider a typical and isolated urban-macro cell of dimension (1000×1000-m^2). We perform our experiments for—indoor-hotspots—that typically describe big shopping malls, factories, or airports' halls. Each building consists of multiple floors which further may consist of small rooms such as shops and counters. We assume that the cell filled with square dimensional buildings on a grid street of width 20 m. The side length of each building is 100 m. A total of 200 UEs are distributed randomly inside the buildings (indoor), and Base stations distributed with an intensity 0.2 on the rooftop of surrounding buildings. We will focus on NLOS as our typical urban propagation condition [30]. The distance between BS and UE is 200 m, and the maximum transmission distance between a pair of the D2D devices is 100 m[3]. We also assume that the S-D2D communication system operates at 2.4 GHz frequency and, communication from the BS to the UE can be carried out at 2.1 GHz carrier frequency.

[3] This transmission rang has been justified in literature suitable for the device discovery [21].

7.2 Channel Model

The propagation channel between a pair of D2D devices is not as same as it is between the BS and the UE. The height of the antenna and the transmission power of the UE is very low as compared to the eNodeB, which limits the area coverage of the D2D communication. Therefore, we cannot directly use the channel models designed for the cellular system for the D2D communication system [29]. We use the WINNER II path loss channel model designed to carry out the D2D communication in the indoor-hotspot environment [29,31] as follows:

$$PL(\varepsilon_d)dB = A1\log_{10}(\varepsilon) + A2 + A3\log_{10}(f_c[GHz]/5 + \boldsymbol{\mathcal{V}} + \Psi_\sigma + \sigma_{\boldsymbol{\mathcal{L}}_s}, \tag{31}$$

where f_c indicates the carrier frequency. A1 includes the path loss exponent. A2 represents the intercept, and A3 shows the path loss frequency dependency. $\boldsymbol{\mathcal{V}} = 5n_w$ is the (light) wall penetration parameter, where n_w is the number of walls between the transmitter and receiver. Ψ_σ indicates the shadowing parameter. Its value based on log-normal distribution with mean zero and standard deviation $\sigma = 6$. $\sigma_{\boldsymbol{\mathcal{L}}_s} = 4.2$ is the body shadowing loss [31]. The signal strength at the receiver is measured by considering the transmit antenna gain as follows:

$$P_{signal,dB} = P_t + G_t + G_r - PL(\varepsilon_d)dB, \tag{32}$$

where P_t is the transmit power of a transmitting device, G_t indicates the transmit antenna gain and Gr indicates the receive antenna gain. The noise power on a dB scale is calculated as follows

$$P_{signal,dB} = 10\log_{10}(\boldsymbol{\mathcal{K}}_B\boldsymbol{\mathcal{T}}_e) + 10\log_{10}\beta + \boldsymbol{\mathcal{F}}_N, \tag{33}$$

where $\boldsymbol{\mathcal{K}}_B\boldsymbol{\mathcal{T}}_e = 174$ dBm/Hz is the noise power spectral density and $\boldsymbol{\mathcal{F}}_N = 6$ dB represents a noise figure of the receiving device.

7.3 Video Caching Setup

For video streaming, we assume that each device can store 30–90 min (1800 seconds to 5400 seconds) long 1080 p YouTube video. We use the default recommended setting to measure the size of the segments and the video.

7.4 Performance Metrics

According to the default settings, the maximum video file size is 5.4 Gigabytes (GB)[4]. We vary the segment playtime from 1 second to 10 seconds to measure the optimal beginning-segment size. Each device requests for the video segments from a video library that comprises of 1000 distinct video files. We vary the video caching probability γ_c from 0.2 to 1.2 and request distribution probability γ_r from 0.6 to 1. In the S-D2D caching system, each device will cache the beginning-segments and the subsequent segments of the unique video files[5] based on a given value of Zipf exponent γ_c from a video library of size L.

In the simulations first: (i) We distribute the devices randomly in a cell area then, (ii) We assign the beginning-segments of the desired video files in a Block-1 and the subsequent segments in a Block-2 according to the Algorithm 1 and Algorithm 2 respectively. (iii) We generated 200 requests for beginning-segments, and subsequent segments for the large and small videos according

[4] The average storage capacity of smartphones varies from 32 GB to 64 GB. Therefore, we assume that each UE in our simulation is capable of caching video files equivalent to one-hour long 1080p YouTube video.

[5] In our simulations, we make sure that each mobile device is caching no duplicates of the video content.

to the specified values of γ_r. (iv) Finally, the requesting devices find the list of P-D2D devices according to the segments access protocol described in a subsection.

We compare the proposed two-tiered caching approach with two traditional full video caching schemes: the most popular content only (MPCO) caching scheme [32,33] that always stores in the cache the most popular video files; and the optimal cache policy (OCP) [32,34] that does not exploit any knowledge of the users' video abandonment behavior and always caches the unique video files on the users' cache. To assess the performance of our proposed caching approaches, we evaluate three important performance parameters: (i) Cache-hit ratio, (ii) Self-hit ratio and (iii) Cache-throughput ratio. The cache-hit ratio measures the total number of requests satisfied, over the total number of requests generated for the video content. The self-hit ratio is the total number of requests served through the requested users' local cache, over the total number of requests generated for the video files. The cache-throughput ratio measures the total number of self-hits and cache-hits (when the self-search process is unsuccessful), over the total number of requests generated for the videos.

Next, first, in Fig. 4 we focus on the Block-1 of users' cache space that evaluates the impact of different values of γ_c and γ_r, on the size of beginning-segments for the indoor-hotspot D2D communication scenario discussed in Section 6.1. Then, in Figs. 5–7, we assessed the performance of our proposed caching schemes and compared them with MPCO and OCP.

7.5 Simulation Results

Figs. 4a–4c evaluate the impact of skew in video popularity and size of beginning-segments on the average cache-hit ratio, average self-hit ratio, and average cache-throughput ratio respectively. We dedicate 5% of the total users' cache capacity to the 'Block-1' that is approximately equal to 300 seconds video clip.

It is observed from the figures that the average cache-hit ratio, average self-hit ratio and average cache-throughput ratio increases as the size of the beginning-segments decreases. For instance, we can observe that, when each mobile device is caching 10 seconds beginning video clip, we obtained the average cache-hit ratio 29.9–60.3%, average self-hit ratio 10–43% and, average cache-throughput ratio 30.1–62.4% for different values of γ_c and γ_r. When the size of the beginning-segments is decreased from 10 seconds to 1 second, we observed a significant increase in the performance ratio. The average cache-hit ratio increases to 84.2–91.3%, average self-hit ratio increases to 50–79%, and average throughput-ratio increases to 83.9–93.6%. It shows that 79% of users can start the video with zero startup-time through the self-search process, while 13% of requests can be satisfied through the cache-hit. This performance improvement is due to the smaller segment size that creates a large and diverse pool of a virtual cache of beginning-segments.

Consequently, the requesting users are more likely to find the beginning-segments of the desired video files in their local cache as well as through the D2D link. This result proves the effectiveness of delivering beginning-segments through our BSCP. We can also observe from the figure, that the average performance ratio of the BSCP increases as we increase the value of γ_c and γ_r. In general, the more skewed the popularity distribution is, most viewers are interested in a few and the most popular video content, that leads to the overall improvement in the BSCP performance. Thus, we selected $\gamma_c = 1.2$ as our $\gamma_{(c,opt)}$ and 1–3 seconds video clip as a possible optimal beginning-segment size. To favor the most popular and less popular video content, we will use $\gamma_r = 0.6$ as our optimal request distribution exponent.

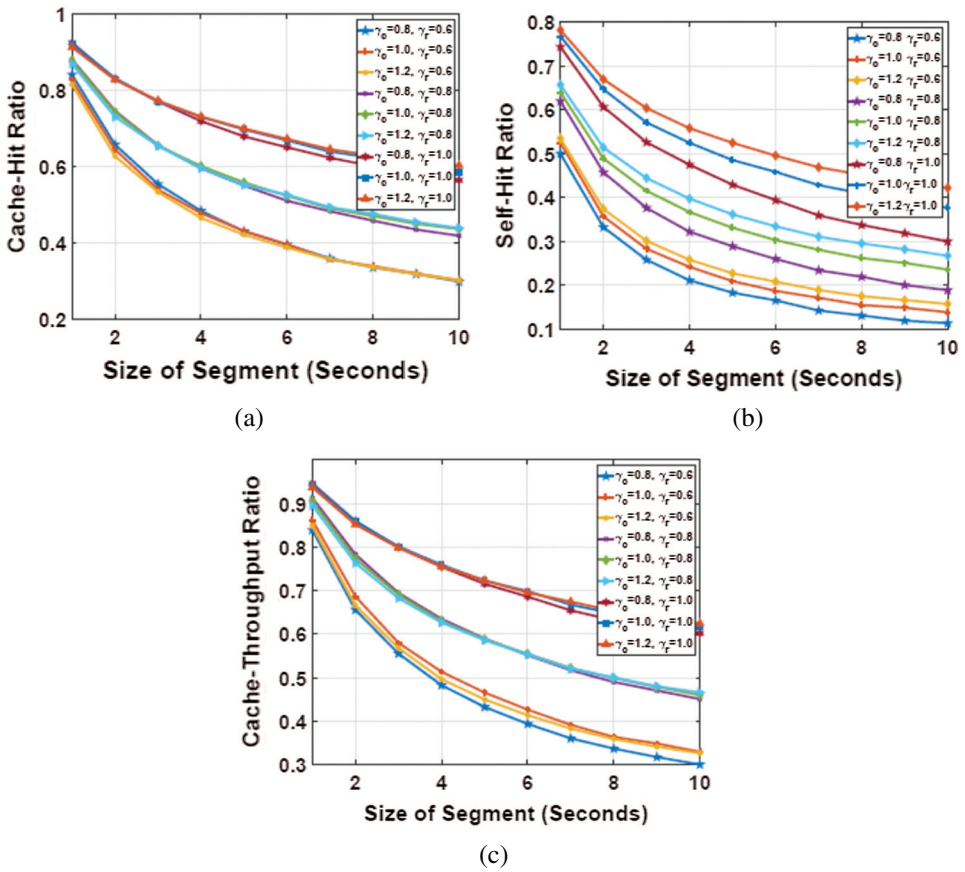

Figure 4: Evaluation of $\gamma_{(c,opt)}$ and optimal $s_{(i,j)}$ for different values of γ_c and γ_r. (a) Cache-hit ratio *vs.* size of segments (seconds) (b) Self-hit ratio *vs.* size of segments (seconds) (c) Cache-throughput ratio *vs.* size of segments (seconds)

Fig. 5 compares the performance of proposed cache policies for different cache sizes. We varied the size of cache capacity from 5.4 GB to 64 GB. Figs. 5a–5c show the impact of cache sizes on the average cache-hit ratio, average self-hit ratio and, average cache-throughput ratio respectively. Surprisingly, a small increase in a percentage of cache capacity dedicated to the BSCP can contribute significantly to the cellular network in terms of delivering the beginning-segments of the desired video files with zero startup-time. For example, consider the case, when the cache size of the Block-1 is increased from 5% to 30%. We can observe clearly from the figure that 50% to 95% of users can start the video with zero startup-time through the self-hit. The reason for this improvement is that, when the percentage of cache size (Block-1) increases, it also increases the cache space to accommodate more dynamic beginning-segments of the most popular as well as the less popular video files. Thus, the self-hit ratio improvement is more beneficial for starting the video with zero startup-time than increasing the total cache-hit ratio. However, no more benefits can be obtained once the percentage of the cache Block-1 for storing the beginning-segments increases beyond 30%.

When the performance of SLVCP and SPCP is assessed, we observed that both caching policies follow the same trend when the percentage of cache capacity for the Block-2 is increased.

However, for large cache space, SLVCP policy has a higher cache performance ratio. For instance, the users of short length videos can not only start the video with zero startup-time, but up to 47.5% of users can enjoy continuous streaming delivery with zero playback-delay. While the 31% of users can download the remaining segments of the desired short length videos from their neighboring devices with low playback-delay. We can also observe that the performance of the SPCP is better than the MPCO strategy. The reason behind that is that SPCP is capable of caching more dynamic and large number of content (fully and partially) in comparison to the MPCO strategy, which concentrates only on highly skewed popular content γ_c and caches the whole video files. The performance of the OCP is far from better. However, it was the expected result because OCP does not exploit any knowledge of users' priorities for watching the videos.

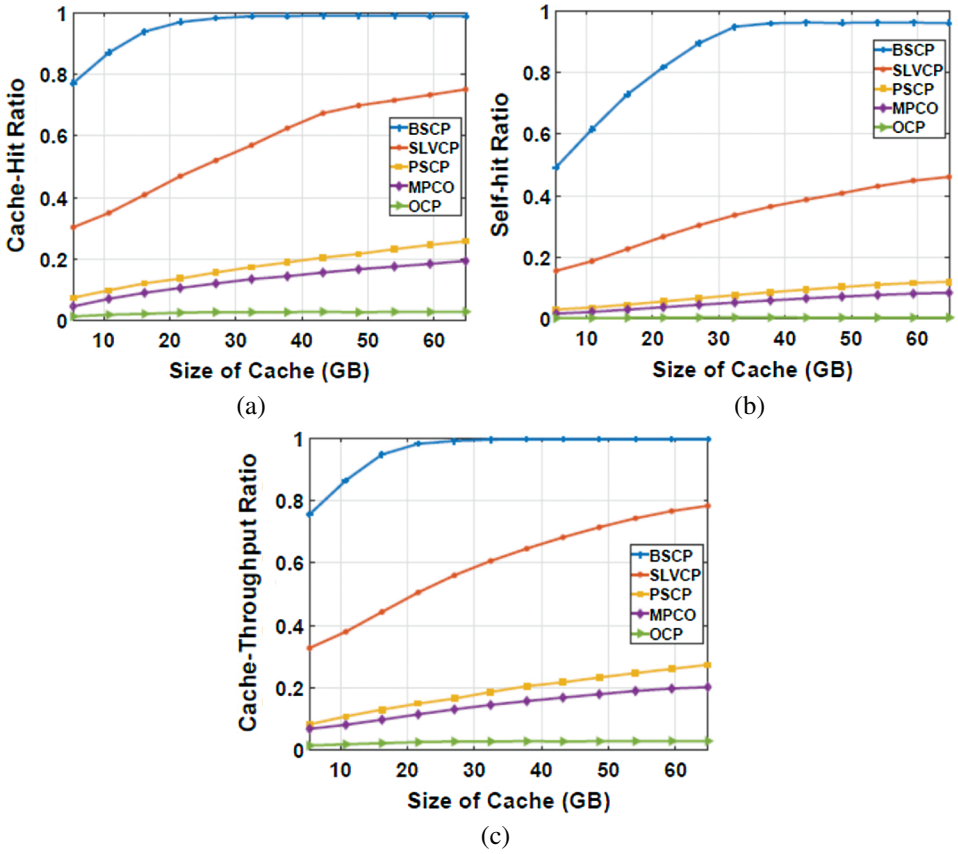

Figure 5: Evaluation of the impact of different sizes of cache capacity on the average cache-hit ratio, average self-hit ratio and, the average cache-throughput ratio on the BSCP, SLVCP, SPCP, MPCO and OCP. (a) Cache-hit ratio vs. size of cache (GB) (b) Self-hit ratio vs. size of cache (GB) (c) Cache-Throughput ratio vs. size of cache (GB)

Figs. 6a–6c evaluate the impact of different values of γ_r on the performance of the BSCP, SLVCP, and SPCP. We observe that, the average cache-hit ratio, average self-hit ratio and average throughput-ratio increases as we increase the value of γ_r. The implication is that for the small value of γ_r the probability of finding the less popular files among a virtual cache pool of most popular video files is much smaller. However, as we increase the value of γ_r the average cache-hit

ratio, average self-hit ratio and average throughput-ratio also increases. Which also increases the probability of finding the beginning-segments and later segments of the desired video content within the vicinity of the requesting cellular users. Not surprisingly, our proposed cache policies perform better than MPCO and OCP.

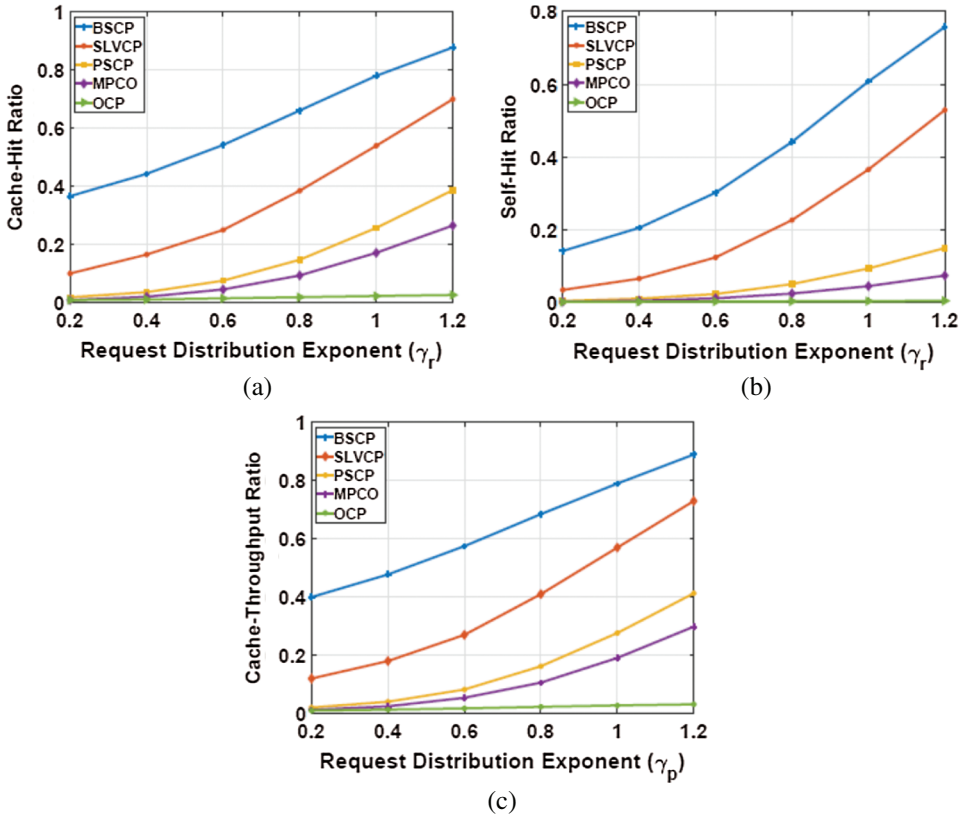

Figure 6: Evaluation of the impact of different values of γ_r on the average cache-hit ratio, average self-hit ratio and, average cache-throughput ratio on the BSCP, SLVP, SPCP, MPCO and OCP. (a) Cache-hit ratio vs. request distribution exponent γ_r (b) Self-hit ratio vs. request distribution exponent γ_r (c) Cache-throughput ratio vs. request distribution exponent

In Figs. 7a and 7b, we compare the performance of proposed caching schemes from sparse to very dense environments. Fig. 7a illustrates the impact of density of P-D2D devices on the average cache-hit ratio and Fig. 7b shows the effect of density of P-D2D devices on the average cache-throughput ratio. Interestingly, the average cache-hit ratio and the average-cache-throughput ratio for all caching policies are very close. They show the same trend when the density of P-D2D devices is increased. The intuition behind this is that, as we increase the density of the P-D2D devices in a cell, users get the opportunity to satisfy their random requests for the beginning-segments and the subsequent segments from a very large aggregated virtual cache pool. Although the dense user environment may lead to very intense D2D interference, users can find the video segments from the devices in very close proximity. The chances of D2D success probability also become higher. As expected, the performance of SLVCP and SPCP is even better than MPCO and OCP. The remarks for this observation is that, as the SPCP can cache the dynamic and large

numbers of content in the users' cache space, the increase in the density of the P-D2D device also linearly increases the size of the virtual cache. Thus, it increases the probability of finding the desired segments in the proximity of the requesting device.

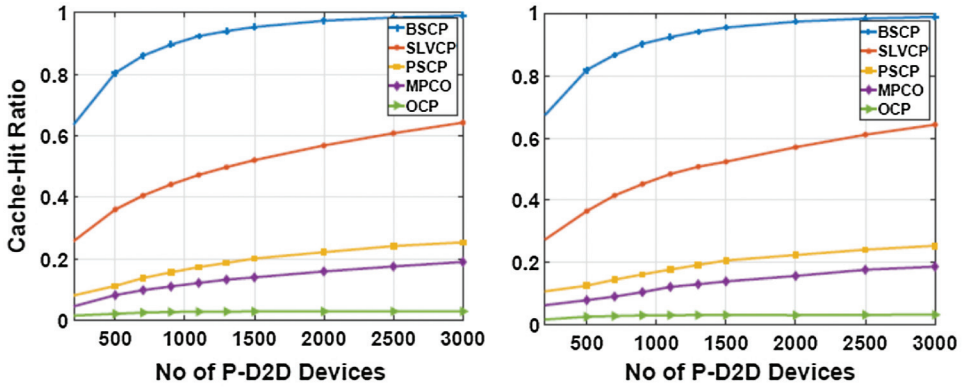

Figure 7: The impact of P-D2D density on average cache-hit ratio and average cache-throughput ratio for beginning-segments, SLVP, SCP, MPCO and OCP, when $\gamma_r = 1.2$ and $\gamma_r = 0.6$. (a) Cache-hit ratio vs. the number of P-D2D devices (b) Cache-throughput vs. the number of P-D2D devices

8 Conclusion

In this paper, we proposed a two-tiered S-D2D caching approach by taking into consideration one of the important features of on-demand video streaming applications, namely, User Abandonment Behavior. Which is when users stop watching videos before their completion and after watching only a few video chunks of the video. The S-D2D approach divides the cache space of each D2D device into two blocks of different sizes. The first small block of the user's cache is reserved for storing and delivering only the beginning portion. The second block caches the latter portion of the requested video file' fully or partially' depending on the users' video abandonment behaviour and popularity of the video. We also proposed a segment access control protocol, describing how the video segments are cached and shared in an S-D2D caching system. To control the admission of segments into the user's cache and improve the system throughput, we further proposed and evaluated three caching algorithms, i.e., BSCP, SPCP and SLVCP. Our simulation results showed that the BSCP achieved the average throughput-ratio from 83.9%–93.6%, among which 79% of users can start the video with zero startup-time through the self-hit, while 13% of requests can be satisfied through the cache-hit. Our simulation results also proved that the SLVCP outperforms all caching policies. We also proved in our simulations that, the SPCP performs better than the MPCO and OCP, even if the caching conditions are not favourable. Our simulation results also proved that the SLVCP outperforms all caching policies.

Acknowledgement: The Corresponding author K.R would like to express his gratitude to Manchester University for support. The author A. A is thankful to Prince Sattam Bin Abduaziz University for their support.

Funding Statement: The author F.W. would like to express their gratitude to the Baihang university, Beijing, China for their financial and technical support under Code No. BU/IFC/INT/01/008.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Ericsson Mobility Report. 2018. [Online]. Available: https://www.ericsson.com/en/mobility-report/reports.

[2] J. Sahoo, M. A. Salahuddin, R. Glitho, H. Elbiaze and W. Ajib, "A survey on replica server placement algorithms for content delivery networks," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 1002–1026, 2017.

[3] Y. Li, C. Tao, G. Seco-Granados, A. Mezghani, A. L. Swindlehurst *et al.,* "Channel estimation and performance analysis of one-bit massive MIMO systems," *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 4075–4089, 2017.

[4] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 116–130, 2009.

[5] J. Sung, M. Kim, K. Lim and J. K. Rhee, "Efficient cache placement strategy in two-tier wireless content delivery network," *IEEE Transactions on Multimedia*, vol. 18, no. 6, pp. 1163–1174, 2016.

[6] S. Sun, T. S. Rappaport, M. Shafi, P. Tang, J. Zhang *et al.,* "Propagation models and performance evaluation for 5G millimeter-wave bands," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8422–8439, 2018.

[7] L. Li, G. Zhao and R. S. Blum, "A survey of caching techniques in cellular networks: Research issues and challenges in content placement and delivery strategies," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 1710–1732, 2018.

[8] J. Yan, D. Wu and R. Wang, "Socially aware trust framework for multimedia delivery in D2D cooperative communication," *IEEE Transactions on Multimedia*, vol. 21, no. 3, pp. 625–635, 2019.

[9] N. Golrezaei, A. G. Dimakis and A. F. Molisch, "Scaling behavior for device-to-device communications with distributed caching," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4286–4298, 2014.

[10] M. Lee, M. Ji, A. F. Molisch and N. Sastry, "Performance of caching-based D2D video distribution with measured popularity distributions," in *2019 IEEE Global Communications Conf. (GLOBECOM)*, Waikoloa, HI, USA, pp. 1–6, 2019.

[11] K. Doppler, M. Rinne, C. Wijting, C. B. Ribeiro and K. Hugl, "Device-to-device communication as an underlay to LTE-advanced networks," *IEEE Communications Magazine*, vol. 47, no. 12, pp. 42–49, 2009.

[12] D. Wu, Q. Liu, H. Wang, Q. Yang and R. Wang, "Cache less for more: exploiting cooperative video caching and delivery in D2D communications," *IEEE Transactions on Multimedia*, vol. 21, no. 7, pp. 1788–1798, 2019.

[13] Y. Xu and F. Liu, "QoS provisionings for device-to-device content delivery in cellular networks," *IEEE Transactions on Multimedia*, vol. 19, no. 11, pp. 2597–2608, 2017.

[14] S. S. Krishnan and R. K. Sitaraman, "Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs," *IEEE/ACM Transactions on Networking*, vol. 21, no. 6, pp. 2001–2014, 2013.

[15] Maximising audience engagement: How online video performance impacts viewer behavior akamai white paper, 2015. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.pdf.

[16] N. Anjum, Z. Yang, H. Saki, M. Kiran and M. Shikh-Bahaei, "Device-to-Device (D2D) communication as a bootstrapping system in a wireless cellular network," *IEEE Access*, vol. 7, pp. 6661–6678, 2019.

[17] L. Maggi, L. Gkatzikis, G. Paschos and J. Leguay, "Adapting caching to audience retention rate," *Computer Communications*, vol. 116, pp. 159–171, 2018.

[18] N. Anjum, D. Karamshuk, M. Shikh-Bahaei and N. Sastry, "Survey on peer-assisted content delivery networks," *ELSEVIER Computer Networks*, vol. 116, pp. 79–95, 2017.

[19] J. Summers, T. Brecht, D. Eager and B. Wong, "To chunk or not to chunk: Implications for HTTP streaming video server performance," in *Proc. of the 22nd Int. Workshop on Network and Operating System Support for Digital Audio and Video*, pp. 15–20, 2012.

[20] WINNER II channel models, 2017. [Online]. Available: https://cept.org/file/8339/winner2%20-%20final%20report.

[21] M. Naslcheraghi, S. A. Ghorashi and M. Shikh-Bahaei, "FD device-to-device communication for wireless video distribution," *IET Communications*, vol. 11, no. 7, pp. 1074–1081, 2017.

[22] D. Karamshuk, N. Sastry, A. Secker and J. Chandaria, "On factors affecting the usage and adoption of a nation-wide TV streaming service," in *2015 IEEE Conf. on Computer Communications (INFOCOM)*, Kowloon, pp. 837–845, 2015.

[23] J. Erman, A. Gerber, K. K. Ramadrishnan, S. Sen and O. Spatscheck, "Over the top video: The gorilla in cellular networks," in *Proc. of the 2011 ACM SIGCOMM Conf. on Internet Measurement Conf.*, pp. 127–136, 2011.

[24] H. Nam and H. Schulzrinne, "Youslow: What influences user abandonment behavior for internet video?," Columbia University, Tech. Rcp., 2016.

[25] A. Finamore, M. Mellia, M. M. Munafò, R. Torres and G. Rao, "Youtube everywhere: Impact of device and infrastructure synergies on user experience," in *Proc. of the 2011 ACM SIGCOMM Conf. on Internet Measurement Conf.*, pp. 345–360, 2011.

[26] C. Huang, J. Li and K. W. Ross, "Can internet video-on-demand be profitable?," *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 4, pp. 133–144, 2007.

[27] A. F. Molisch and F. Tufvesson, "Propagation channel models for next-generation wireless communications systems," *IEICE Transactions on Communications*, vol. 97, no. 10, pp. 2022–2034, 2014.

[28] N. Anjum and M. Shikh-Bahaei, "Evaluation of availability of initial-segments of video files in Device-to-Device (D2D) network," in *2017 Int. Conf. on Wireless Networks and Mobile Communications (WINCOM)*, Rabat, pp. 1–7, 2017.

[29] M. Haenggi and R. K. Ganti, "Interference in large wireless networks, " in *Interference in Large Wireless Networks*, Now Foundations and Trends, 2009.

[30] I. Khan, I. S. Henna, N. Anjum, A. Sali, J. Rodrigues *et al.,* "An efficient precoding algorithm for mmWave massive MIMO systems," *Symmetry*, vol. 11, pp. 1099, 2019.

[31] M. Ji, G. Caire and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 1, pp. 176–189, 2016.

[32] N. Golrezaei, A. F. Molisch and A. G. Dimakis, "Base-station assisted device-to-device communications for high-throughput wireless video networks," in *2012 IEEE Int. Conf. on Communications (ICC)*, Ottawa, ON, pp. 7077–7081, 2012.

[33] Z. Chen, N. Pappas and M. Kountouris, "Probabilistic caching in wireless D2D networks: Cache hit optimal versus throughput optimal," *IEEE Communications Letters*, vol. 21, no. 3, pp. 584–587, 2017.

[34] M. Gregori, J. Gómez-Vilardebó, J. Matamoros and D. Gündüz, "Wireless content caching for small cell and D2D networks," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1222–1234, 2016.