Multidimensional Content Modeling and Caching in D2D Edge Networks

S. Sinem Kafıloğlu, Gürkan Gür† and Fatih Alagöz
Department of Computer Engineering, Bogazici University
34342, Istanbul, Turkey
†Institute of Applied Information Technology (InIT)
Zurich University of Applied Sciences (ZHAW)
8401, Winterthur, Switzerland
Email: {sinem.kafiloglu, fatih.alagoz}@boun.edu.tr, gueu@zhaw.ch

***Abstract*—Future Internet is going to be shaped by networked multimedia services with exploding video traffic becoming the dominant payload. That evolution requires a remedial shift from the connection-oriented architecture to a content-centric one. Another technique to address this capacity crunch is to improve spectral utilization through new networking paradigms at the wireless network edge. To this end, Device-to-Device (D2D) communications has the potential for boosting the capacity and energy efficiency for content-centric networking. To design and implement efficient content-centric D2D networks, rigorous content modeling and in-network caching mechanisms based on such models are crucial. In this work, we develop a multidimensional content model based on popularity, chunking and layering, and devise caching schemes through this model. Our main motivation is to improve the system performance via our caching strategies. The numerical analysis shows the interplay among different system parameters and performance metrics: while our schemes perform slightly poorer in terms of system goodput, they also decrease the system energy expenditure. Overall, this improvement dominates the loss in the goodput, leading to greater energy efficiency compared to the commonly-used caching technique Least Recently Used (LRU).**

I. Introduction

The drastically increasing video content consumption in the Internet leads to burgeoning data traffic in wireless networks [1]. This challenge requires the improvement of network capacity while increasing energy efficiency for sustainable systems. Content-centric design of networks is promising to realize these goals. Accordingly, the classical networking techniques are revised for improving the content dissemination in terms of capacity and energy efficiency. These solutions inherently employ in-network caching, and thus should consider prospective content requests and cache videos according to request patterns and content characteristics. Apart from pervasive caching, Device-to-Device (D2D) communications is also a promising enabler for boosting the system capacity via short distance transmissions and better spectrum utilization. This paradigm is especially instrumental in mobile edge networks with localized computation and communications for multimedia-intensive services such as Augmented Reality (AR) or massive content streaming. Therefore, how D2D paradigm and content model driven caching for video traffic can be jointly exploited is a key research topic for future wireless networks.

In this work, we develop a video content model for content-centric edge networks with D2D communications. In-network caching is a fundamental capability for minimizing the energy consumption and increasing the capacity in that setting. Thereof, we focus on that aspect and devise novel caching strategies relying on our proposed model. Our developed caching schemes are particularly built upon the *popularity*, *chunking* and *layering* attributes. They are studied in terms of their energy consumption, system goodput and energy efficiency performance. The improvement in energy efficiency and the trade-off between the energy consumption and goodput are revealed through simulation-based experiments.

There is a plethora of research works that utilize tuples of *i) popularity, ii) chunking* and *iii) layering* in their content models. The popularity attribute is used to grasp an elemental insight into the video consumption preferences of users. In [2], Hachem et al. make use of the popularity characteristic of contents in their study in that regard. Besides, the chunking dimension that determines how videos are partitioned into segments is an enabler for improving system efficiency. For instance, chunking is used for its beneficial impact on bandwidth utilization in [3]. Layering is another dimension in content modeling that is utilized for providing scalability to the content dissemination in networks. Chau et al. make use of the concept of layering in their developed content model in [4]. There are also studies that contain couples of these content model dimensions. For instance, in [5], the popularity and layering are used for content modeling while Xu et al. utilize popularity and chunking dimensions in [6]. Moreover, there exist some studies that take advantage of content popularities and also focus on layered content caching such as [7] and [8]. In [9], Ramzan et. al. exploit both layering and chunking aspects of content modeling for video streaming.

In-network caching has also been extensively investigated in the literature from *i) content* and *ii) D2D* caching viewpoints. For instance, [8] and [10] are content-based caching works which primarily utilize inherent content features in their caching decisions. In [8], Zhan et. al. propose a heuristic caching algorithm for minimizing the latency of layered

content dissemination in heterogeneous networks (HetNets). However, these caching algorithms are not studied in the parlour of D2D domain. In contrast, there are some D2D caching studies exploiting content-specific attributes in their caching decisions [7], [11], [12]. Gregori et al. study the joint caching and resource allocation problem in small cells and D2D network where the popularity of contents follows the Zipf distribution in [11]. [12] makes use of content popularities for caching in HetNets with D2D and cognitive communications. In [13], Bok et al. focuses on a P2P mobile network architecture and proposes a cooperative caching technique for improving cache hit ratio, cache replacement time and power efficiency performance. However, none of these studies utilizes all of the content dimensions *i) popularity*, *ii) chunking* and *iii) layering* in their investigations.

As a key contribution in this work, we develop a popularity, chunking and layering based video content model for content-centric D2D edge networks. To the best of our knowledge, our work is the first proposal that models video contents according to all these dimensions — especially from the perspective of caching in D2D networks. Additionally, based on our novel content model, we propose two caching algorithms via prioritization on content attributes in such systems. We also investigate the impact of caching on the energy consumption, goodput and energy efficiency.

II. System Model

In our system, we consider the wireless nodes in the network edge exchanging content via D2D communications in an infrastructure-independent manner [14]. This architectural layout refers to emerging mobile edge computing scenarios such as AR and edge-accelerated content streaming. Devices in this network setting need to be protected against excessive energy consumption due to video traffic while enjoying very high bitrates. In that regard, we focus on video content modeling and caching in these ad hoc D2D networks.

In our system, users are dispersed in the spatial domain without access to a base station for content delivery. For modeling the user locations, Poisson Point Process (PPP) is a commonly utilized spatial distribution [15]. In our network, users are distributed according to PPP with mean density λ_{users}. They have devices with storage that is capable of storing contents. These devices can exchange video content with each other via D2D communications. When a content is requested, first the requester will check its local cache. If the content is not found, it will try to use D2D transmissions. It will fetch the requested content from the closest accessible device that stores that content. All users have equal priority while accessing the wireless medium. For the D2D wireless channel, the employed pathloss model for a given distance d is:

$$P_r(d) = P_{D2D} - 20 \cdot log_{10}(\frac{4\pi f d_0}{c}) - 10 \cdot log_{10}(\frac{d}{d_0})^n \quad (1)$$

where P_{D2D} is the transmit power of a device and $P_r(d)$ received power, n is the path loss exponent and d_0 is a reference distance of the device antenna. The D2D channel capacity is calculated by $C = B \cdot log_2(1 + \frac{P_r(d)}{B \cdot N_0})$ where B is the bandwidth and N_0 is the noise power density.

TABLE I: Video sequences [20].

Video	**Genre**
Citizen Kane	Drama
Silence of the Lambs	Drama
Jurassic Park I	Action
Die Hard I	Action
The Terminator I	Action
Total Recall	Action
Star Wars IV	Sci-fi
Star Wars V	Sci-fi
Aladdin	Cartoon
Cinderella	Cartoon
The Firm	Drama
Tonight Show	Late Night Show
Baseball	Game 7 of the 2001 World Series
Snowboarding	Snowboarding Competition

A. Video Content Model

To explain the rationale behind our three dimensional (popularity, chunking and layering) video content model, we first describe these dimensions:

- ***Popularity:*** Popularity is a key content attribute that is used to optimize caching according to content request characteristics. The emergence of content-centric networking requires popularity profiling of contents. In the literature, the Zipf distribution $Zipf(s, N)$ is widely used for generic modeling of content requests [5], [16]. Here, N stands for the total number of contents in the system while s determines the skewness of the distribution.
- ***Chunking:*** The partitioning of contents into chunks leverages the caching gain [17]. It is also a practical strategy for designing simpler caching schemes and enabling differentiation among different parts of a content. In that regard, it is beneficial to be utilized in the content model.
- ***Layering:*** In scalable coding, the base layer is the standard quality (SQ) video segment while enhancement layers improve the video quality [18]. The upper layers require low quality layer portions for successful decoding. Scalable video coding provides adaptability for different network conditions [19] such as congestion or packet loss. Thus, it is integrated into our content model.

For empirically determining the video characteristics in our content model, we utilize 60 minutes long QCIF formatted temporal scalable encoded videos [20] listed in Table I. $IBBPBBPBBPBBPBB...$ is the group of picture (GoP) structure of these videos with frame rate 30 fps. In [20], layering dimension is used where the trace statistics of temporal scalable encoded videos are provided. I and P frames constitute the base layer while B frames form the enhancement layer. The calculation of mean video frame sizes of base and

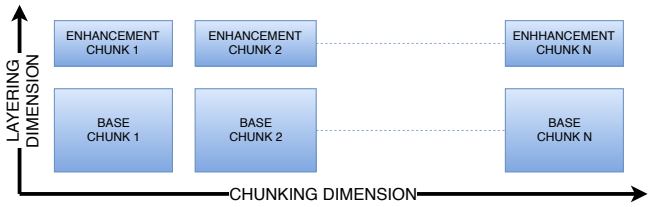


Fig. 1: Layer and chunk dimensions of the video content model.

enhancement layer $\overline{X^{\gamma}}$ is given in (2) [21]. X_n is the size of the n^{th} frame for $n = 0, 1, \cdots, N-1$ while X_n^b=X_n for I and P (base layer) frames and X_n^b is zero for B (enhancement layer) frames. On the contrary, X_n^e is zero for base layer frames and X_n^e=X_n for enhancement layer frames.

$$\overline{X^{\gamma}} = \frac{1}{N} \sum_{n=0}^{N-1} X_n^{\gamma} \quad , \qquad \gamma \in \{b, e\}. \tag{2}$$

For video coding, our sample videos consist of frames partitioned into 8×8 sample blocks of luminance, hue and intensity and all of them are mapped to 8×8 transform coefficient blocks via DCT. These blocks are quantized based on a quantization scale where low scale means higher quality and high scale entails lower quality in [20]. In our work, we consider 10, 14 and 16 as quantization scales for I, P and B frames, respectively. For the given quantization scale, the mean base frame size $\overline{X^b}$ is 0.3727 kB while the mean enhancement frame size $\overline{X^e}$ is 0.176 kB [20]. The average size of 60 minutes long SQ videos of frame rate 30 Hz $\overline{s_{SQ}}$ is then calculated as:

$$\overline{s_{SQ}} := 3600 \ s \cdot 30 \text{ Hz} \cdot \overline{X^b} \ kB \cdot 8 \cdot 10^3 \ \frac{bits}{kB} \tag{3}$$

Accordingly the average size of HQ videos (60 minutes long, frame rate 30 Hz) $\overline{s_{HQ}}$ is calculated using the value $\overline{s_{SQ}}$ and the additional enhancement layer contribution as shown below:

$$\overline{s_{HQ}} := \overline{s_{SQ}} + (3600 \ s \cdot 30 \text{ Hz} \cdot \overline{X^e} \ kB \cdot 8 \cdot 10^3 \ \frac{bits}{kB}) \tag{4}$$

Then the average sizes according to employed video sequences are $\overline{s_{SQ}}$ = 322 Mb and $\overline{s_{HQ}}$ = 474 Mb.

As the chunking technique, we employ equipartioning, i.e. partition the contents into equal sizes [10], [17] and take the base chunk size as 16 Mbits. The two layers in our video content model with chunking are shown in Fig. 1. We assume two video layers with equal sized chunks in these layers.

III. Multidimensional Caching Schemes For D2D Edge Networks

The benefits of content-centric networking are realized through the utilization of content popularity differentiation in caching. For instance, Suksomboon et. al. propose *PopCache* in content-centric networks which stores popular contents close to the requesters [10]. However, apart from the inter-content patterns such as relative popularity, intra-content features are promising as some content portions are more beneficial in terms of caching gains. For instance, from the layering aspect, the base is more critical compared to enhancement layer(s) since videos cannot be rendered when it is unavailable [19]. From the perspective of chunking factor, the initial chunks are more worthy owing to the inter-chunk request phenomenon where the initial chunks are demanded more frequently (e.g., the beginning segments of a video compared to the end ones) [22]. Based on these content related observations, we develop our prioritized content caching techniques.

Algorithm 1 LPPC/CPPC Caching Algorithms

CACHE(S_c, c_x, C,**TYPE**){

$Cur_{Cap} = Capacity(S_c)$
if $(Cur_{Cap} + size(c_x) \leq C)$ **then**
  return $S_c \cup \{c_x\}$;
else
  $S_{sorted} \leftarrow$ sort(S_c, POP);
  if (TYPE == $LPPC$**) then**
    $S_{sorted} \leftarrow$ sort(S_{sorted}, LAYER);
    $S_{sorted} \leftarrow$ sort(S_{sorted}, CHUNK);
  else if (TYPE == $CPPC$**) then**
    $S_{sorted} \leftarrow$ sort(S_{sorted}, CHUNK);
    $S_{sorted} \leftarrow$ sort(S_{sorted}, LAYER);
  end if
  //$S_{sorted} = \{s_1, \ s_2, \ \ldots \ , \ s_k\}$ ordered from s_1 to s_k
  j = k;
  while $(j \geq 1)$ **do**
    $S_{sorted} \leftarrow S_{sorted} \setminus \{s_j, s_{j+1}, \ldots s_k\}$;
    if $(Cur_{Cap} + size(c_x) - \sum_{\theta=j}^{k} size(s_\theta) \leq C)$ **then**
      return $S_{sorted} \cup \{c_x\}$;
    end if
    $j \leftarrow j - 1$;
  end while
end if
}

Our proposals, namely *Layer Prioritized Popularity Based Caching (LPPC)* and *Chunk Prioritized Popularity Based Caching (CPPC)*, are constructed with the aim of preserving "important" content segments in caches for improving the system performance. In both schemes, listed in Algorithm 1, the content units in the cache S_c are first sorted on the popularity dimension in descending order. In this way, the highest priority is given to the popularity attribute. Next, the caching mechanisms focus either on the chunk or layer order. $LPPC$ first sorts on the layering dimension of a given content. For breaking ties among the same layer of the same content, it sorts on the chunk order. Thus, the layering dimension dominates the chunking dimension in $LPPC$. On the contrary, $CPPC$ gives greater importance to the chunking dimension compared to layering and does the opposite sorting, i.e, first according to chunking, then layering. If the to-be-cached content c_x cannot fit into the the cache of capacity C, the content units are discarded from the cache starting from the

lowest order until the free cache capacity suffices for the c_x.

A. Complexity

The complexity of any caching algorithm is important for practical purposes. Therefore, we also investigate the complexity of our proposed algorithms. In our system, let N_c be the number of contents, N_l the number of layers and N_{ch} the maximum number of chunks of a content. Both algorithms first sort contents on popularity with time complexity $O(N_c \log N_c)$. Next, $LPPC$ sorts on the layering dimension on each content with complexity $O(N_c(N_l \log N_l))$. Finally, $LPPC$ sorts on the chunking dimension for each layer of all contents with complexity $O(N_c N_l(N_{ch} \log N_{ch}))$. Overall, the time complexity of $LPPC$ algorithm is $O(N_c \log N_c + N_c(N_l \log N_l) + N_c\ N_l(N_{ch} \log N_{ch}))$.

The sorting order of layering and chunking dimensions of $CPPC$ are the direct opposite of $LPPC$ with time complexity $O(N_c \log N_c + N_c(N_{ch} \log N_{ch}) + N_c N_{ch}(N_l \log N_l))$. Consequently, both $CPPC$ and $LPPC$ operate in polynomial time.

IV. Performance Metrics

We investigate our proposed caching schemes in terms of the performance metrics *(i) energy consumption*, *(ii) goodput*, and *(iii) energy efficiency*. The system parameters are listed in Table II.

A. Energy

One of the energy consumption components is due to the local cache hits of requested content units E_{loc}. In that regard, $P_{loc} \cdot \frac{|s_u|}{C_{loc}}$ is the energy consumption of each local service for some content unit u ($r_u \in S_{(n,n)}$). The summation of local services for all content units and devices in the analyzed network region provides E_{loc}:

$$E_{loc} := \sum_{u \in U} \sum_{n \in N} \sum_{r_u \in S_{(n,n)}} P_{loc} \cdot \frac{|s_u|}{C_{loc}} \tag{5}$$

The aggregate transmission energy of devices operating in the D2D mode E_{D2D} is another element for the system energy usage. $P_{D2D} \cdot \frac{|s_u|}{C_{D2D}(n,m)}$ is the energy consumption of the D2D service for some content unit u transmitted between devices n and m. Then, E_{D2D} refers to the aggregate of all D2D utilizing services for all content units from all device pairs:

$$E_{D2D} := \sum_{u \in U} \sum_{\substack{n,m \in N \\ n \neq m}} \sum_{r_u \in S_{(n,m)}} P_{D2D} \cdot \frac{|s_u|}{C_{D2D}(n,m)} \tag{6}$$

Some content unit requests are blocked due to the limitation of the network and cache capacities. However, the devices have to switch from the sleep to the idling state for these unsuccessful attempts. In that regard, the activation energy of devices for content units $E_{block}(s_u)$ is aggregated to calculate the total blocking energy consumption:

$$E_{block} := \sum_{u \in U} \sum_{n \in N} E_{block}(s_u) \tag{7}$$

Then, the overall energy consumption of our system E_{all} is

$$E_{all} := E_{loc} + E_{D2D} + E_{block} \tag{8}$$

B. Goodput

G_{loc} is the aggregate number of received bits via local hits of requested content units over the course of the experiment. For successful reception, any content unit request r_u should be in the $Comp$ set. This is because none of the corresponding content units of a given content request has contribution to the goodput if that given request has incomplete base chunk(s).

$$G_{loc} := \frac{\sum_{u \in U} \sum_{n \in N} \sum_{r_u \in (Comp \cap S_{(n,n)})} |s_u|}{T_{sim}} \tag{9}$$

The overall goodput provided by the network through D2D mode is G_{D2D}. For contributing to the D2D goodput, any request should be in the $Comp$ set by the same reasoning explained above. Such a request should also be a member of $\overline{Fail}$ set (i.e., its transmission should not fail). Please note that the $Comp$ and $\overline{Fail}$ set are not necessarily the same. A content request can have all of its base chunks successfully transmitted, i.e., $r_u \in Comp$. However, some enhancement chunk unit u might have failed for that content and then $r_u \in Fail$, thus not contributing to the goodput via D2D communications.

$$G_{D2D} := \frac{\sum_{u \in U} \sum_{\substack{n,m \in N \\ n \neq m}} \sum_{\substack{r_u \in (Comp \cap S_{(n,m)}) \\ r_u \in \overline{Fail}}} |s_u|}{T_{sim}} \tag{10}$$

By summing all of these contributions, we get the overall network goodput G_{all} as shown in (11):

$$G_{all} := G_{loc} + G_{D2D} \tag{11}$$

TABLE II: System parameters.

Parameter	Explanation
P_{loc}	The power consumption of local content unit retrieval
P_{D2D}	The transmission power consumption of a device
C_{loc}	The service capacity of content
$C_{D2D}(n,m)$	The channel capacity between the n^{th} and m^{th} devices
E_{block}	The activation energy of devices from the sleeping to the idling state
N	The total number of devices
U	The set of content units uniquely identifiable by content, chunk, and layer id
r_u	The request for the content unit u
s_u	The size of the content unit u
$S_{(n,m)}$	The set of services from the n^{th} device to m^{th} device
$Comp$	The set of requests for a content where all the base chunks are transmitted successfully (service completed successfully)
$Fail$	The set of requests for content units that have failed

TABLE III: Simulation parameters.

Parameter	Explanation	Value
T_{sim}	The simulation duration	1200 s
s	The Zipf distribution exponent	0.8
λ	The Weibull distribution scale parameter	1
k	The Weibull distribution shape parameter	0.6
p_{HQ}	The ratio of high quality content consumers	1
λ_{users}	The mean density of user distribution in PPP	0.0015 user/m^2
R_{BS}	The radius of the investigation zone	330 m
R_{D2D}	The radius of interference-free D2D transmission zone	120 m
C	The cache capacity of devices	47.1 Mbits
P_{D2D}	The transmission power of a device	80 mW
d_0	The reference distance of device antenna	1 m
n	The path loss exponent of D2D transmission	3
B	The bandwidth of the terrestrial channel	2 MHz
N_0	The noise power density	-95 dBm

C. Energy Efficiency (EE)

The division of the overall system energy consumption over the total number of transmitted bits in the system gives the energy efficiency metric EE as

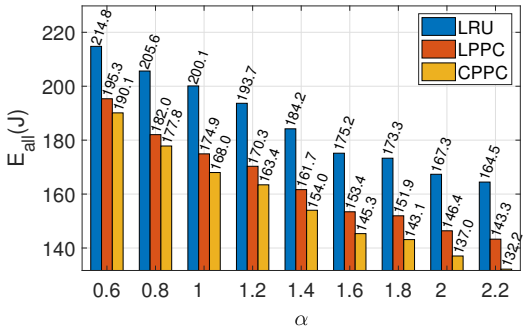
$$EE := \frac{E_{all}}{G_{all} \cdot T_{sim}} \tag{12}$$

in energy spent per bit (joule per bit - jpb) units which has to be minimized to improve energy efficiency.

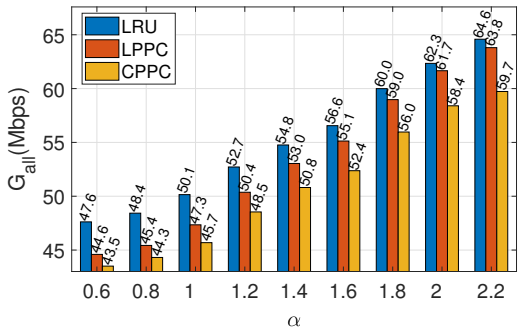
V. Performance Evaluation

For performance evaluation, we inspect how our system performs with varying Zipf distribution parameter α for the performance metrics *(i) energy*, *(ii) goodput* and *(iii) energy efficiency*. We compare our strategies to the baseline *Least Recently Used (LRU)* scheme. LRU replaces the chunk(s) least recently accessed from the cache in the sake of a new request when the cache capacity is not sufficient to store the newly requested one. It is a common algorithm extensively employed in hardware and software-based caches. The simulation parameters and their values are listed in Table III.

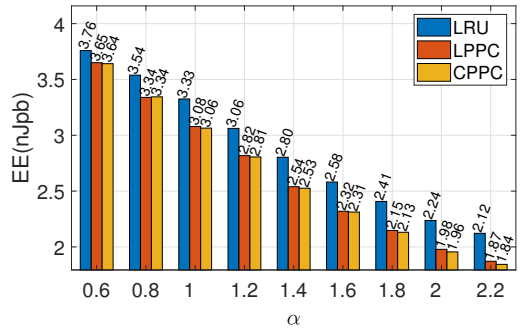
In Fig. 2(a), it is observed that with increasing α values the total system energy consumption decreases for all caching mechanisms. This is intuitive since for larger α, the popularity gap between contents increases and the popularity concentrates on fewer contents. Then the most popular contents are stored in devices more often and the overall local hit rates are improved. Therefore, increasing α benefits energy due to two factors. First, the power consumption P_{loc} for a local hit is less than the device transmission power P_{D2D}. Second, local hits attain larger local service capacity C_{loc} compared to wireless transmissions. Specifically, $CPPC$ and $LPPC$ algorithms



(a) Energy consumption.



(b) Goodput.



(c) Energy efficiency.

Fig. 2: Performance of caching mechanisms for different α values - (a) energy consumption, (b) goodput, (c) EE.

deplete less energy than the LRU algorithm for any fixed α as seen in Fig. 2(a). The energy consumption of $CPPC$ ($LPPC$) has an improved performance gap with the classical LRU ranging from 11.49% (9.05%) to 19.63% (12.88%). Our proposed techniques are better at keeping important content portions corresponding to prospective user requests in local caches. That advantage reduces network traffic and thus results in less overall energy consumption. With increasing α, the gap between content popularities increases: the maximal improvement of $CPPC$ ($LPPC$) over LRU is attained at the largest α = 2.2 value with 19.63% improvement from 164.5 Joule to 132.2 Joule (12.88% improvement from 164.5 Joule to 143.3 Joule). $CPPC$ is more beneficial for energy

consumption compared to $LPPC$ especially for larger α values. This phenomenon means that chunking is dominant over layering dimension of the video model for large α's regarding the minimization of total energy consumption.

According to the experimental results, the system goodput also improves with increasing α in all caching strategies as shown in Fig. 2(b). This is again due to the increased caching benefits with high popularity differentiation among content units in large α regime. In Fig. 2(b), $CPPC$ and $LPPC$ schemes achieve poorer system goodput in contrast to LRU. The worst degradation compared to LRU in the goodput is observed for $CPPC$ scheme with 8.9% reduction from 50.1 to 45.7 Mbps at α=1. For the entire α range, $LPPC$ is substantially more beneficial than $CPPC$ in terms of system goodput, especially for larger α's. Unlike the energy consumption case, for this case we deduce that layering has greater impact than chunking on the service quality.

To analyze the system EE, we inspect the results in Fig. 2(c). Evidently, our proposed schemes perform better compared to LRU with increasing α. For the largest value 2.2 in our α range, the improvement of $CPPC$ ($LPPC$) over the LRU in terms of EE is 13.10% from 2.12 to 1.84 nJpb (11.81% from 2.12 to 1.87 nJpb). Despite the system goodput degradation in our caching techniques, they provide gains in the energy consumption to a greater extent. Thereof, our schemes are overall more energy efficient. No apparent EE difference between our techniques is observed. Hence, the different prioritization order in our proposed schemes do not significantly alter the network EE characteristics in this case.

VI. Conclusions

In this work, we develop a video content model based on the popularity, chunking and layering dimensions for content-centric and D2D edge networks. Moreover, we propose priority based caching schemes that utilize our video content model for caching decisions. Our caching techniques yield varying prioritization of chunking and layering dimensions. The chunking dimension compared to the layering has greater effect on the reduction of the system energy consumption while the layering dimension in contrast has greater effect on the improvement of the system goodput. Our numerical results show that our caching schemes outperform the standard LRU algorithm and provide higher EE due to dominant energy consumption reduction in spite of slightly worse system goodput. As future work, we plan to formulate the caching gain in terms of energy consumption in the D2D network as a rigorous optimization problem and solve using different techniques.

Acknowledgment

This work was supported by the Scientific and Technical Research Council of Turkey (TUBITAK) under grant number 116E245.

References

[1] C. W. Paper, "Cisco visual networking index: Global mobile data traffic forecast update, 2016-2021," Feb. 2017.

[2] J. Hachem, N. Karamchandani, S. Moharir, and S. Diggavi, "Caching with partial matching under Zipf demands," in *2017 IEEE Information Theory Workshop (ITW)*, Nov. 2017, pp. 61–65.

[3] K. W. Hwang, D. Applegate, A. Archer, V. Gopalakrishnan, S. Lee, V. Misra, K. K. Ramakrishnan, and D. F. Swayne, "Leveraging video viewing patterns for optimal content placement," in *NETWORKING 2012*, R. Bestak, L. Kencl, L. E. Li, J. Widmer, and H. Yin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 44–58.

[4] P. Chau, Y. Lee, T. D. Bui, J. Shin, and J. P. Jeong, "An efficient resource allocation scheme for scalable video multicast in LTE-Advanced networks," in *2017 11th International Conference on Ubiquitous Information Management and Communication*, Jan. 2017, pp. 1–8.

[5] G. Gür and S. Kafiloğlu, "Layered content delivery over satellite integrated cognitive radio networks," *IEEE Wireless Commun. Lett.*, vol. 6, no. 3, pp. 390–393, June 2017.

[6] C. Xu, M. Wang, X. Chen, L. Zhong, and L. A. Grieco, "Optimal information centric caching in 5G device-to-device communications," *IEEE Trans. Mobile Comput.*, vol. 17, no. 9, pp. 2114–2126, Sept. 2018.

[7] C. Zhan and G. Yao, "SVC-based caching and transmission strategy in wireless device-to-device networks," in *2018 16th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, May 2018, pp. 1–6.

[8] C. Zhan and Z. Wen, "Content cache placement for scalable video in heterogeneous wireless network," *IEEE Commun. Lett.*, vol. 21, no. 12, pp. 2714–2717, Dec. 2017.

[9] N. Ramzan, E. Quacchio, T. Zgaljic, S. Asioli, L. Celetto, E. Izquierdo, and F. Rovati, "Peer-to-peer streaming of scalable video in future internet applications," *IEEE Commun. Mag.*, vol. 49, no. 3, pp. 128–135, March 2011.

[10] K. Suksomboon, S. Tarnoi, Y. Ji, M. Koibuchi, K. Fukuda, S. Abe, N. Motonori, M. Aoki, S. Urushidani, and S. Yamada, "PopCache: Cache more or less based on content popularity for information-centric networking," in *38th Annual IEEE Conference on Local Computer Networks*, Oct. 2013, pp. 236–243.

[11] M. Gregori, J. Gmez-Vilardeb, J. Matamoros, and D. Gündüz, "Wireless content caching for small cell and D2D networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1222–1234, May 2016.

[12] S. Sinem Kafıloğlu, G. Gür, and F. Alagöz, "Analysis of content-oriented heterogeneous networks with D2D and cognitive communications," *arXiv*, 2018.

[13] K. Bok, J. Kim, and J. Yoo, "Cooperative caching for multimedia data in mobile p2p networks," *Multimedia Tools and Applications*, vol. 78, pp. 1–24, 06 2017.

[14] M. Ji, G. Caire, and A. F. Molisch, "Fundamental limits of caching in wireless D2D networks," *IEEE Trans. Inf. Theory*, vol. 62, no. 2, pp. 849–869, Feb. 2016.

[15] S. Kusaladharma and C. Tellambura, "Performance characterization of spatially random energy harvesting underlay D2D networks with primary user power control," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–7.

[16] M. Emara, H. ElSawy, S. Sorour, S. Al-Ghadhban, M. Alouini, and T. Y. Al-Naffouri, "Optimal caching in multicast 5G networks with opportunistic spectrum access," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, Dec. 2017, pp. 1–7.

[17] L. Wang, S. Bayhan, and J. Kangasharju, "Optimal chunking and partial caching in information-centric networks," *Computer Communications*, vol. 61, pp. 48 – 57, May 2015.

[18] M. Siekkinen, E. Masala, and J. K. Nurminen, "Optimized upload strategies for live scalable video transmission from mobile devices," *IEEE Trans. Mobile Comput.*, vol. 16, no. 4, pp. 1059–1072, April 2017.

[19] J.-R. Ohm, "Advances in scalable video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 42–56, Jan. 2005.

[20] P. Seeling, M. Reisslein, and B. Kulapala, "Network performance evaluation using frame size and quality traces of single-layer and two-layer video: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 6, no. 3, pp. 58–78, Third 2004.

[21] M. Reisslein, J. Lassetter, S. Ratnam, O. Lotfallah, F. Fitzek, and S. Panchanathan, "Traffic and quality characterization of scalable encoded video: a large-scale trace-based study, part 1: Overview and definitions," *Arizona State Uni. Telecommunications Research Center, Tech. Rep*, 2002.

[22] S. Lim, Y. Ko, G. Jung, J. Kim, and M. Jang, "Inter-chunk popularity-based edge-first caching in content-centric networking," *IEEE Commun. Lett.*, vol. 18, no. 8, pp. 1331–1334, Aug. 2014.