SVC-based Caching and Transmission Strategy in Wireless Device-to-Device Networks

Cheng Zhan and Guo Yao
School of Computer and Information Science, Southwest University, ChongQing, 400715 P.R.China.
Email: zhanc@swu.edu.cn, yguoswu@gmail.com

***Abstract*—To address the explosively growing demand for mobile traffic, wireless device-to-device (D2D) networks have been introduced, where caching at user devices can be exploited to alleviate the burden on base stations. In this paper, we consider joint caching and transmission of scalable video coding (SVC) streaming over wireless D2D networks. We formulate a joint caching and transmission problem using integer linear programming to minimize the average download time for user, and prove that finding the optimal solution is NP-hard. A heuristic solution is proposed based on a two-phase sub-optimization problem focusing on caching and transmission decisions, which is solved by using relaxed linear programming. Simulation results show that the proposed scheme can significantly reduce the average download time in comparison with existing caching strategies.**

***Index Terms*—Device-to-device network, scalable video streaming, caching, transmission schedule**

I. Introduction

Wireless data consumption has been growing at exponential rates recently. As reported by Cisco [1], mobile data traffic has grown 18-fold over the past five years, and over 78% of the mobile data traffic will be video by 2021. The rapidly growing video streaming traffic has created significant challenges in cellular networks due to the limited capacity of the backhaul links and the scarce radio resources. For example, if a requested video is retrieved from a remote video server, user experience will decrease greatly due to congestion and the increased latency. The increasing video demand requires efficient techniques to achieve the desired user quality of experience (QoE). A variety of solutions have been proposed, such as increased spectral efficiency [2], femto-caching [3], device-to-device caching [4], [5], etc.

Recently, wireless caching has attracted a lot of attentions for the advantages of fast response and it does not rely on the backhaul heavily [6], [7]. Different from the commonly considered femto-caching, a device-to-device caching network, as shown in Fig. 1, allows caching at user devices. Therefore, the caching capacity grows with the number of devices, and users can directly acquire requested data from nearby caching devices via D2D links. Recently, several studies on cache placement policies over wireless caching networks were conducted [8]–[10]. A novel proactive caching algorithm based on the online learning of content popularity was proposed in [8]. The work in [9] proposed optimal cache placement to maximize the successful offloading probability in a cache-enabled wireless heterogeneous network. The work in [10] studied the cache placement in multiple-level hierarchical caching networks to reduce the server load by serving as many requests as possible. However, the above work did not consider the quality of experience (QoE) of mobile video caching, which has layered structures.

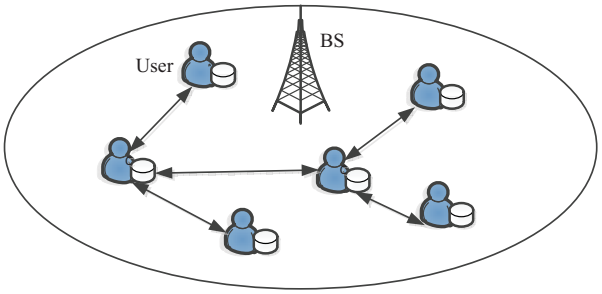


Fig. 1. System model for a cache-enabled D2D transmission.

Considering user requirements for a mobile video, it is common that different users may have different QoE due to different processing power and network bandwidth. This can be addressed by using scalable video coding (SVC) [11], where videos are encoded into a base layer and one or more enhancement layers. The base layer contains the necessary data for decoding and is required to watch the video, but the enhancement layers augment the quality of the video. The decoding of a higher enhancement layer requires the base layer and all its lower enhancement layers of the video [12]. In other words, the lth layer data cannot be decoded unless all the kth layer data are available, $k < l$. Therefore, users can enjoy high video quality if they download more enhancement layers, but high-layer data are useless unless receiving lower layer data.

When considering caching SVC videos over wireless D2D networks, the caching placement problem becomes more challenging. It is easy to see that a better QoE is obtained by caching a video with more layers, however, more cache space will be consumed and there will be no space for caching other videos. Caching policies for layered encoded videos in content delivery networks were investigated in [13]. However, they did not consider caching at user devices and transmission scheduling for user QoE requirements. Caching at user devices can promote D2D communications, where nearby mobile devices can communicate directly, and it is not necessary to to communicate through the base station. On the other hand, if a video is cached at a device with small transmission capacity, it will be difficult for other users to download. When

allocating cache and transmission resources for SVC videos over wireless D2D networks, a series of interesting problems arise: 1) which videos should be cached, 2) which layers of each video should be cached, and 3) how to transmit based on the cache placement for all possible requests, so as to satisfy QoE requirements of users. In this paper, we study a joint caching and transmission problem for SVC videos in wireless D2D networks, where the objective is to minimize the average download time while satisfying the QoE requirements for all users. The main contributions of this paper are summarized as follows:

- We investigate joint caching and transmission scheduling problem for SVC videos in a wireless D2D network, and formulate the problem using integer linear programming.
- We prove that solving the joint caching and transmission scheduling problem is NP-hard, and propose a heuristic solution based on a two-phase sub-optimization strategy using a relaxation of linear programming.
- Simulation results show that the proposed scheme can achieve significant caching gains in terms of the average download time for all users while satisfying the QoE requirements.

II. System Model and Problem Statement

We consider a wireless D2D cache system as shown in Fig. 1, where there is a base station that is connected to a video server and n user devices placed uniformly in the cell, $U = \{u_1, u_2, \ldots, u_n\}$. Set $N(u_i)$ is used to denote the neighbor set of user u_i, which is the set of users that can directly connect to u_i by using a D2D link. In other words, u_i can communicate with the users in $N(u_i)$ directly using high data rate D2D communication in a dedicated frequency band for D2D transmission. We assume that the base station has enough space to store the entire video content. Users will not download all the required videos from the base station due to the costly transmission to users. Therefore, we assume that each user u_i is equipped with a cache of capacity C_i. Each user can download data from either the base station or other users that are in the neighbor set. Specifically, the requested video data can only be downloaded from the base station as long as it cannot be served from the neighbor users. Therefore, all user requests can be satisfied.

Suppose that the video library contains a set of $V = \{v_1, v_2, \ldots, v_m\}$ videos with cardinality m. Each user $u_i \in U$ makes a request for video $v_j \in V$ in an independent manner according to a given request probability mass function. The popularity of videos can be modeled by the Zipf distribution [14]. In the Zipf distribution, the probability of a user's demand for video v_j is given by

$$f_j = \frac{1/j^{\theta}}{\sum_{k=1}^{m} (1/k^{\theta})}, \tag{1}$$

where $1 \leq j \leq m$, and $\theta \geq 0$ is a constant that reflects the skew of the popularity distribution. Note that different users may request the same video, but the required quality may be different, which is a characteristics of SVC video.

We assume that each video can be encoded with L layers, where the data size of the lth layer of v_j is denoted as e_{jl}. If u_k does not already cache the l-th layer of v_j, it will issue a request. Different mobile users in the system may require different videos in V, and even for the same required video, different users may require for different qualities. Assume that each layer of a video has equal probability of being requested, where p_{kjl} is the probability that user u_k requests lth layer of video v_j, so that $p_{kjl} = \frac{f_j}{L}$. We assume that r_{0k} is the download rate of u_k from the base station, and the download rate of u_k from u_i is denoted as r_{ik}, $u_i \in N(u_k)$, $r_{ik} > r_{0k}$. It is not difficult to find that user u_i cannot download data from other users which have no connection to it. Thus, if $u_i \notin N(u_k)$, then $r_{ik} = 0$. Therefore, we can obtain the download latency of user u_k downloading the lth layer of v_j from $u_i \in N(u_k)$ and the base station as $\frac{e_{jl}}{r_{ik}}$ and $\frac{e_{jl}}{r_{0k}}$, respectively. Considering about the QoE requirement of users, if user u_i require video v_j with rth quality level, then all the lth layers of video v_j need to be delivered to u_i, $1 \leq l \leq r$. Since every user cannot serve an unlimited number of requests at the same time due to its limited upload capacity, we assume that the upload capacity of u_i is B_i, which means that the sum download rate of the neighbor users from u_i should not exceed B_i.

In general, due to cache capacity limitations, each user may only partially cache the data of a video, or of some layers of the video, in detail. On the other hand, due to transmission capacity limitations, even some users cached the whole video data, other users may only download partial data from the cached user. The caching placement strategy and the transmission schedule need to be appropriately designed to satisfy the different QoE requirements of users for SVC videos. For a given wireless D2D network topology, content request probability of users, user cache capacities and transmission capacity, we should determine how to place the data content of videos in the caches of users and how to schedule the data transmissions in order, to minimize the average download time for all users. Such a joint cache placement and transmission schedule optimization problem for SVC videos in wireless D2D networks is referred to as the *Joint Video Caching and Transmission* (JVCT) optimization problem.

III. Joint Video Caching and Transmission Schedule Optimization Problem

In this section, we will present an integer linear programming formulation of the JVCT problem and characterize its complexity.

We define x_{ijl} as the binary caching decision variable, which indicates whether Layer l of video v_j is placed at the cache of user u_i or not, $1 \leq i \leq n, 1 \leq j \leq m, 1 \leq l \leq L$. If Layer l of v_j is cached at u_i, then $x_{ijl} = 1$, otherwise $x_{ijl} = 0$. In order to minimize the average video download time for all users, the caching decision should be determined based on the user video requests. On the other hand, although some data content is cached at nearby devices, if the transmission scheduling strategy does not include theses devices, it will

be no help for decreasing the download time. Therefore, the transmission schedule needs to be considered jointly with the cache placement for all possible requests to minimize the download time. To this end, we define a binary transmission variable y_{ik}^{jl} to indicate the transmission decision of whether or not u_i is chosen to transmit the lth layer of v_j to u_k. If u_i is chosen to transmit the lth layer of v_j to u_k, then $y_{ik}^{jl} = 1$, otherwise $y_{ik}^{jl} = 0$. We also define $y_{ii}^{jl} = 0, 1 \leq i \leq n$, and y_{0k}^{jl} is defined to indicate whether or not the base station is chosen to transmit the lth layer of v_j to u_k.

Given the network topology of user devices, and estimated video requests from all users, the JVCT problem determines 1) which video every user should cache, 2) which layer of the video should each user cache, and 3) how to transmit each layer of a requested video from a cached device to requesting users. These decisions are made under the resource constraints of the wireless D2D cache network, including limited cache capacity and bandwidth constraints at each user device. Let $\mathbf{X}, \mathbf{Y}$ be the decision matrices of the caching decision and transmission decisions, respectively, so that the problem of optimal joint caching and transmission scheduling problem can be formulated as the following integer linear programming.

$$\min_{\mathbf{X},\mathbf{Y}} \quad \sum_{k=1}^{n}\sum_{j=1}^{m}\sum_{l=1}^{L} p_{kjl}\left(\frac{y_{0k}^{jl}e_{jl}}{r_{0k}} + \sum_{i=1}^{n}\frac{y_{ik}^{jl}e_{jl}}{r_{ik}}\right)$$

$$s.t. \quad \sum_{j=1}^{m}\sum_{l=1}^{L} x_{ijl}e_{jl} \leq C_i, \qquad \forall i \tag{2}$$

$$\sum_{k=1}^{n}\sum_{j=1}^{m}\sum_{l=1}^{L} y_{ik}^{jl}r_{ik} \leq B_i, \qquad \forall i \tag{3}$$

$$y_{ik}^{jl} \leq x_{ijl} \qquad \forall i,j,l,\forall u_k \in N(u_i) \tag{4}$$

$$\sum_{i=0}^{n} y_{ik}^{jl} \leq 1, \qquad \forall k,j,l \tag{5}$$

$$\sum_{i=0}^{n} y_{ik}^{jl} \leq \sum_{i=0}^{n} y_{ik}^{j(l-1)}, \qquad \forall j,k,2 \leq l \leq L \tag{6}$$

$$x_{ijl} \in \{0,1\}, \qquad \forall i,j,l \tag{7}$$

$$y_{ik}^{jl} \in \{0,1\}, \qquad \forall i,j,l,\forall u_k \in N(u_i). \tag{8}$$

The objective is to minimize the average download time for all users. Consider the case that user u_k requests the lth layer of video v_j, so that $p_{kjl} = 1$. If the request can be satisfied by other user devices, then the download delay is $\sum_{i=1}^{n}\frac{y_{ik}^{jl}e_{jl}}{r_{ik}}$. Otherwise, the request can only be satisfied by the base station, which means that $\sum_{i=1}^{n} y_{ik}^{jl} = 0$ and $y_{0k}^{jl} = 1$, so that the download delay is $\frac{e_{jl}}{r_{0k}}$.

Among the constraints of the formulation, (2), (3), and (4) focus on resources constraints at each user device. In constraint (2), we ensure that the total size of the cached data at u_i should not be larger than the storage capacity C_i. In constraint (3), we make sure that the sum of upload transmission rates of u_i cannot exceed the upload capacity. Constraint (4) means that if a video layer is scheduled to be downloaded from u_i, then it should be first cached at u_i. Constraint (5) indicates that each video layer can be downloaded from only one user device or base station. Constraint (6) is the layering constraint, which guarantees that if the lth layer of video v_j has been received by the user u_k, then all the tth layers of v_j need to be received by u_k, $t < l$, which avoids receiving only higher-layer data without lower-layer data. However, with this integer linear programming, it is not feasible to obtain the optimal solution of this problem in polynomial time. From the following theorem, we obtain the result that solving the JVCT problem is NP-hard.

Theorem 1: Solving the joint video caching and transmission (JVCT) optimization problem is NP-hard.

Proof: The methodology of this proof is that we give a reduction from a well known NP-completeness problem to a special case of JVCT problem with polynomial time. According to constraint (5), $y_{0k}^{jl} \leq 1 - \sum_{k=1}^{n} y_{ik}^{jl}$, the objective function is equivalent to

$$\min_{\mathbf{X},\mathbf{Y}} \quad \sum_{k=1}^{n}\sum_{j=1}^{m}\sum_{l=1}^{L} p_{kjl}\left[\frac{e_{jl}}{r_{0k}} - \sum_{i=1}^{n} y_{ik}^{jl}\left(\frac{e_{jl}}{r_{0k}} - \frac{e_{jl}}{r_{ik}}\right)\right].$$

We can transform the minimum download time problem to the following maximum download time reduction problem.

$$\max_{\mathbf{X},\mathbf{Y}} \quad \sum_{k=1}^{n}\sum_{j=1}^{m}\sum_{l=1}^{L}\sum_{i=1}^{n} p_{kjl}y_{ik}^{jl}\left(\frac{e_{jl}}{r_{0k}} - \frac{e_{jl}}{r_{ik}}\right).$$

The maximum optimization problem still seems to be complicated since it has two types of variables, x_{ijl} and y_{ik}^{jl}, and too many constraints. Consider a special case of the JVCT problem that there is only one video ($m = 1$) and only one layer for the video ($L = 1$), so that each user has enough cache capacity, and any two devices can connect to each other. In other words, each user can cache all video data, $\sum_{j=1}^{m}\sum_{l=1}^{L} e_{jl} \leq C_i$. Therefore, constraints (2),(4),(6), and (7) can be removed from this special case of the JVCT problem. Since the optimization problem is transformed to a maximum optimization problem, constraint (5) can be written as $\sum_{i=0}^{n} y_{ik}^{jl} = 1$. Let $p'_{ik} = p_{kjl}(\frac{e_{jl}}{r_{0k}} - \frac{e_{jl}}{r_{ik}})$, and $y'_{ik} = y_{ik}^{jl}$, the special case of the JVCT problem can be transformed as

$$\max_{Y} \quad \sum_{i=0}^{n}\sum_{k=1}^{n} p'_{ik}y'_{ik}$$

$$s.t. \quad \sum_{k=1}^{n} r_{ik}y'_{ik} \leq B_i, \qquad \forall i$$

$$\sum_{i=0}^{n} y'_{ik} = 1, \qquad \forall k$$

$$y'_{ik} \in \{0,1\}, \qquad \forall i,k.$$

We can construct $n+1$ kinds of items, a_0 through a_n and n kinds of bins b_1 through b_n. Each bin b_i is associated with a budget B_i. For Bin b_i, each item a_k has a profit p'_{ik} and a weight r_{ik}. For each bin b_i the total weight of assigned items is at most B_i. The solution profit is the sum of profits for

each item-bin assignment. The goal is to find a maximum-profit feasible solution, which is an assignment of items to bins. This problem is equivalent to the general assignment problem [15], which is known to be NP-hard. Therefore, we find a reduction from the general assignment problem to a special case of the JVCT problem, which shows that solving the JVCT problem is NP-Hard. ■

IV. Solution of the JVCT Problem

Since solving the JVCT problem is NP-hard, it is not feasible to find the optimal solution in polynomial time. Therefore, we seek to find an effective heuristic solution in polynomial time.

We propose an efficient two-stage heuristic solution. In the first stage, the cache placement algorithm generates the caching decision based on user demand for the videos. Based on the caching decision of the first stage, the transmission scheduling algorithm uses the updated linear programming formulation to decide the transmission schedule. The main idea of the first stage is based on relaxation of the integer linear programming formulated in Section. III. Firstly, the integer constraints (7) and (8) are replaced by

$$0 \leq x_{ijl} \leq 1, \forall i, j, l$$
$$0 \leq y_{ik}^{jl} \leq 1, \forall i, j, l, \forall u_k \in N(u_i).$$

The relaxation of the integer linear programming with constraints (7) and (8) is a linear programming, which can be solved by standard linear optimization techniques in polynomial time. However, since the solution obtained by linear programming violates constraints (7) and (8) of the original problem, it is not a feasible solution of the original problem. Therefore, given the output of the relaxation problem, we propose a greedy bounded algorithm in order to get a feasible solution of the original problem. We refer to the proposed algorithm as the cache placement algorithm, where the details are presented as in Algorithm 1.

Algorithm 1 Cache Placement Algorithm

1: **for** $\forall i, j, l$ **do**
2: Initialize $x_{ijl} \leftarrow 0$;
3: **end for**
4: Obtain $\tilde{x}_{ijl}$ and $\tilde{y}_{ik}^{jl}$ by solving the relaxed linear programming problem with standard methods;
5: **for** $\forall i, j, l$ **do**
6: $\alpha_{ijl} \leftarrow \sum_{k=1}^{n} p_{kjl} \tilde{y}_{ik}^{jl} (\frac{e_{jl}}{r_{0k}} - \frac{e_{jl}}{r_{ik}})$;
7: **end for**
8: Initialize $c_i \leftarrow C_i$ for each user u_i;
9: Sort all variables $\tilde{x}_{ijl}$ in decreasing order of α_{ijl};
10: **for** each $\tilde{x}_{ijl}$ in sorted order **do**
11: **if** $c_i \geq e_{jl}$ **then**
12: $x_{ijl} \leftarrow 1$;
13: $c_i \leftarrow c_i - e_{jl}$;
14: **end if**
15: **end for**

Suppose that $\tilde{x}_{ijl}$, $\tilde{y}_{ik}^{jl}$ is the result obtained by solving the linear programming problem. Define the cost α_{ijl} of variable x_{ijl} to measure the effect on download time by caching lth layer of v_j at u_i, $\alpha_{ijl} = \sum_{k=1}^{n} p_{kjl} \tilde{y}_{ik}^{jl} (\frac{e_{jl}}{r_{0k}} - \frac{e_{jl}}{r_{ik}})$. If the cost is larger, then it has a great impact on the optimization objective function. Therefore, we first sort α_{ijl} in descending order, and then assign the corresponding $x_{ijl} = 1$ unless the caching capacity constraint is violated. The process continues until all sorted caching variables have been investigated, which results in a feasible solution of the caching decision. At the end of the above cache placement algorithm, we can easily check that x_{ijl} is a feasible solution of the original problem: after the algorithm is completed, for each device u_i, the total size of cached data cannot be larger than the cache capacity of u_i. In addition, x_{ijk} is either 0 or 1, which satisfies the constraint (7). The computational complexity of the cache placement algorithm is $O(n^2 m^2 L^2)$, since the main step of this algorithm is solving the relaxed linear programming and the sorting operation.

Although we can also obtain $\tilde{y}_{ik}^{jl}$ from Algorithm 1, we do not use $\tilde{y}_{ik}^{jl}$ in the second stage, because $\tilde{y}_{ik}^{jl}$ is obtained when we set all $x_{ijl} = 0$ initially. The main idea of the second stage is that taking the caching decision result obtained from the placement algorithm as input, and obtaining a feasible transmission scheduling strategy can serve the requesting users with as many layers as possible. Initially, we set x_{ijl} as the output of Algorithm 1, and then we solve the above relaxed linear programming problem based on the caching solution x_{ijl}, obtaining the output $\hat{y}_{ki}^{jl}$. The resulting fractional solution is then greedily rounded to obtain a feasible transmission scheduling solution. In the rounding procedure, we try not to violate the transmission bandwidth constraint (3) and layering constraint (6). The algorithm is summarized in detail as shown in Algorithm 2.

To fully utilize the transmission bandwidth, we allocate the transmission resources from the lowest layer to the highest layer for all videos, and try to serve as many users as possible under the bandwidth constraint for each layer. Note that we define a new cost value β_{ik}^{jl} of variable y_{ik}^{jl} for transmission of video v_j layer l from user u_i to user u_k, which can be helpful for decreasing the download time, $\beta_{ik}^{jl} = p_{kjl}(\frac{e_{jl}}{r_{0k}} - \frac{e_{jl}}{r_{ik}})$. Similar to the cache placement stage, we sort all variables $\hat{y}_{ik}^{jl}$ in descending order of β_{ik}^{jl}, and assign $y_{ki}^{jl} = 1$ as long as the bandwidth and layering constraint are satisfied, and 0 otherwise. The residual resources are updated, and the process continues until all variables are assigned, which results in a feasible transmission schedule solution. The computational complexity of the transmission scheduling algorithm is $O(n^2 m^2 L^2)$.

V. Simulations

In the following, we will conduct several simulations to show the performance of our proposed joint caching and transmission schedule scheme, where we use the average download time experienced by all users as a metric.

Algorithm 2 Transmission Scheduling Algorithm

for $\forall i, j, l$ **do**
  Initialize x_{ijl} as the output of Algorithm 1;
end for
Obtain $\hat{y}_{ki}^{jl}$ by solving the relaxed linear programming problem with standard methods;
for $\forall i, k, j, l$ **do**
  $\beta_{ik}^{jl} \leftarrow p_{kjl}(\frac{e_{jl}}{r_{0k}} - \frac{e_{jl}}{r_{ik}})$;
  $b_i \leftarrow B_i$;
end for
for $l \leftarrow 1$ to L **do**
  for $\forall i, k, j$ **do**
    Sort all variables $\hat{y}_{ik}^{jl}$ in decreasing order of β_{ik}^{jl};
  end for
  for each $\hat{y}_{ik}^{jl}$ in sorted order **do**
    if $b_i \geq r_{ik}$ **then**
      if $y_{ik}^{j(l-1)} = 1$ or $l = 1$ **then**
        $y_{ik}^{jl} \leftarrow 1$;
        $b_i \leftarrow b_i - r_{ik}$;
      end if
    end if
  end for
end for

We compare our proposed scheme with the following schemes 1) optimal dual-solution based caching algorithm for D2D caching [16], where the content request can be fulfilled by other users via D2D links, and 2) cooperative layered video caching [13], which also considers caching for SVC video in content delivery networks. We use a randomly generated wireless D2D network environment for evaluation, and the parameter settings of the simulation are summarized in Table I.

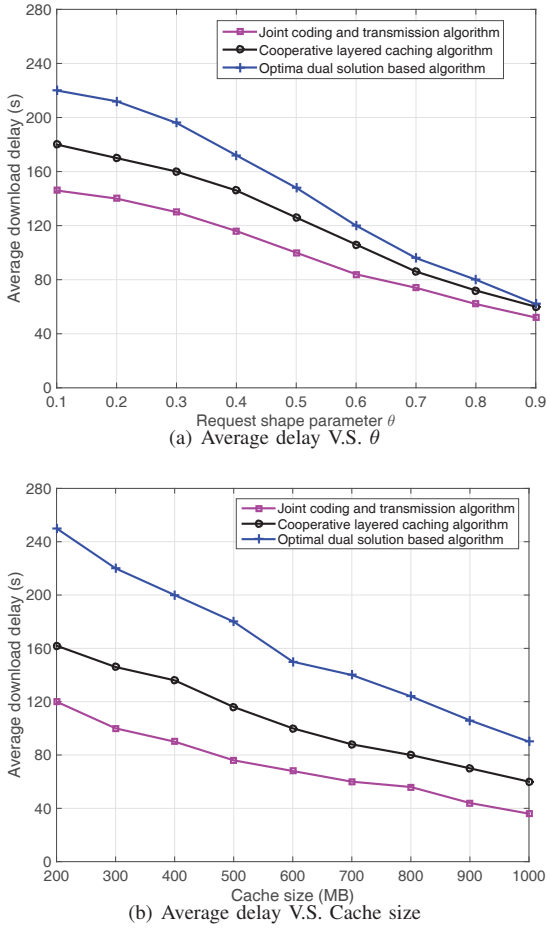
TABLE I
SIMULATION PARAMETERS

Parameters	Value Range	Default
Number of users	$50 - 200$	100
Number of videos	$20 - 200$	100
Number of video layers	$2 - 8$	5
Size of video	50 MB−1 GB	500 MB
Cache capacity of user device	200 MB−2 GB	1 GB
User upload capacity	$100 \sim 1000$ Mbps	200 Mbps
Download rate from other user device	$2 - 10$ Mbps	5 Mbps
Request shape parameter θ	$0 - 1$	0.6

Each user has a cache capacity C_i, uniformly generated over $[200, 2000]$ (MB), and upload capacity B_i uniformly generated over $[100, 1000]$ (Mbps). The number of available videos is set to 500, with popularity following a Zipf distribution with default parameter $\theta = 0.6$. The size of each video is generated uniformly over $[50, 1000]$ (MB), and the download rate is generated uniformly over $[2, 10]$ (Mbps). The download rate from the base station is set to be 1 Mbps. Each video has $L = 5$ layers, and the probability of layers requested by users follows a uniform distribution among the L layers.

First, let us examine the average download time with different value of the shape parameter θ in the Zipf distribution. From Fig. 2(a), we observe that when θ increases, the average download time decreases for all three caching schemes. The reason is that, when θ increases, there exist some videos with high popularity, and then downloading these videos in every user cache will be more effective. Fig. 2(a) also shows that, with increasing θ, the difference between the performance among the three schemes becomes smaller. This is because when θ is large, the output of all three caching schemes will be caching the most popular videos, thus all provide the similar benefits.

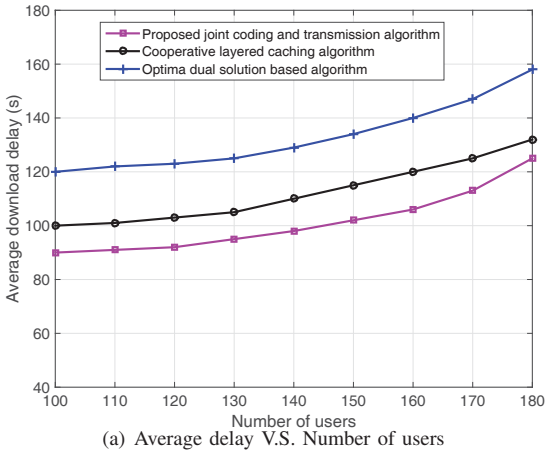
In Fig. 2(b) , we compare the average download time when the cache capacity of users varies. From Fig. 2(b) we can find that, if the user cache capacity becomes larger, then the average download time decreases for all threes schemes since more video content can be cached in user devices. When the cache capability of users is small, the gap between using and not using layered-video-based caching increases, because a layered video based cache can serve more users with higher QoE when data are congested with limited transmission bandwidth. Moreover, the joint transmission and layered caching scheme results in the lowest average download time, because a proper transmission schedule can fetch more layers from other users rather than from the base station.



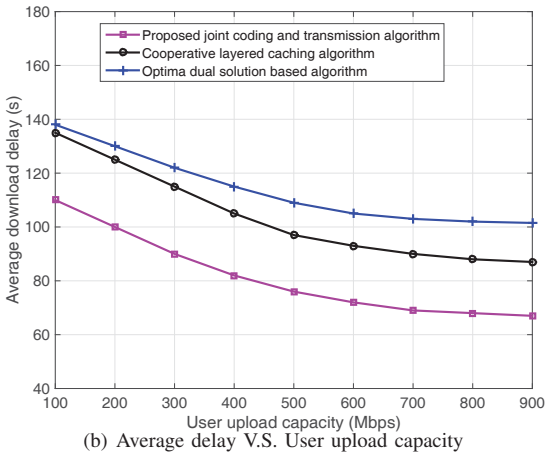
(a) Average delay V.S. θ
(b) Average delay V.S. Cache size

Fig. 2. Average dealy versus request shape parameter θ or cache size.

Fig. 3(a) and Fig. 3(b) show the average download time versus the number of users and the upload capacity, respectively. In Fig. 3(a), it is observed that the average download time increases with the increase of the number of users for all the three schemes. The reason is that the communication bandwidth capacity are limited in this area and shared by all users, and more users competing for the constrained resources will result in larger download time. Fig. 3(b), we compare the average download time when the user upload capacity varies. As expected, the average download time decreases with increasing of user upload capacity for the three schemes, and the reason is that each user can serve more video requests with larger upload capacity. It is observed that the average download time of our proposed scheme is significantly lower than that of the other two schemes, and this performance gain comes mainly from the cooperative layered caching for users and the transmission optimization on layered video in our scheme.



(a) Average delay V.S. Number of users



(b) Average delay V.S. User upload capacity

Fig. 3. Average dealy versus number of users or user upload capacity.

VI. Conclusion

In this paper, we investigated the joint caching and transmission strategy for reducing the average download time of SVC videos over wireless D2D networks. We have formulated the joint caching and transmission scheduling as an integer linear optimization problem, given the distribution of video popularity along with caching and bandwidth constraints. We prove that finding the optimal solution is NP-hard, and derive a heuristic solution based on a two-stage relaxed linear programming based algorithm. The proposed algorithm determines the cache configuration details of each video and the transmission strategy. Simulation results indicate that with our proposed joint caching and transmission scheme, significant caching gain in terms of the average download time can be obtained in wireless D2D networks.

Acknowledgement

This work was supported by National Natural Science Foundation of China (No. 61702426).

References

[1] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update (2016-2021), Cisco, San Jose, CA, USA, Mar. 28, 2017.

[2] C. Li, J. Zhang, J. G. Andrews, and K. B. Letaief, "Success probability and area spectral efficiency in multiuser MIMO HetNets," *IEEE Trans. Commun.*, vol. 64, no. 4, pp. 1544-1556, Apr. 2016.

[3] K. Shanmugam, N. Golrezaei, A. Dimakis, A. Molisch, and G. Caire, "Femtocaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inform. Theory*, vol. 59, no. 12, pp. 8402-8413, Dec. 2013.

[4] M. Ji, G. Caire and A. F. Molisch, "Wireless device-to-device caching networks: basic principles and system performance," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 176-189, Jan. 2016.

[5] S. W. Jeon, S. N. Hong, M. Ji, G. Caire, and A. F. Molisch, "Wireless multihop device-to-device caching networks," *IEEE Trans. Inform. Theory*, vol. 63, no. 3, pp. 1662-1676, Mar. 2017.

[6] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," *IEEE Commun. Magazine*, vol. 52, no. 8, pp. 82-89, Aug. 2014.

[7] M. Ji, G. Caire and A. F. Molisch, "Fundamental Limits of Caching in Wireless D2D Networks," *IEEE Trans. Inform. Theory*, vol. 62, no. 2, pp. 849-869, Feb. 2016.

[8] S. Mller, O. Atan, M. van der Schaar, and A. Klein, "Context-aware proactive content caching with service differentiation in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 2, pp. 1024-1036, Feb. 2017.

[9] D. Liu and C. Yang, "Optimal content placement for offloading in cache-enabled heterogeneous wireless networks," in *Proc. IEEE Global Commun. Conf.* (GLOBECOM), Washington, DC, 2016, pp. 1-6.

[10] K. Poularakis and L. Tassiulas, "On the complexity of optimal content placement in hierarchical caching networks," *IEEE Trans. Commun.*, vol. 64, no. 5, pp. 2092-2103, May 2016.

[11] H. Schwartz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard", *IEEE Trans. on Circ. and Sys. for Video Tech.*, vol. 17, issue 9, pp. 1103-1120, 2007.

[12] S. A. Hosseini, Z. Lu, G. de Veciana, and S. S. Panwar, "SVC-Based Multi-User Streamloading for Wireless Networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 8, pp. 2185-2197, Aug. 2016.

[13] K. Poularakis, G. Iosifidis, A. Argyriou, I. Koutsopoulos and L. Tassiulas, "Caching and operator cooperation policies for layered video content delivery," in *Proc. IEEE Int. Conf. Comput. Commun.* (INFOCOM), San Francisco, CA, pp. 1-9, 2016.

[14] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: evidence and implications," in *Proc. 18th Annu. Joint Conf. IEEE Comput. Commun. Societies*, 1999, vol, 1, pp. 126-134.

[15] Ö. Lale, B. Adil, and T. Pmar, "Bees algorithm for generalized assignment problem". *Applied Mathematics and Computation*,215:3782-3795, 2010.

[16] H. J. Kang, K. Y. Park, K. Cho and C. G. Kang, "Mobile caching policies for device-to-device (D2D) content delivery networking," in *Proc. IEEE Int. Conf. Comput. Commun. Workshops* (INFOCOM WKSHPS), Toronto, ON, pp. 299-304, 2014.