

Service availability of a peer with dynamic buffering for multiple description coded videos on multi-source streaming networks

Chow-Sing Lin · Jhe-Wei Lin

Received: 18 October 2013 / Revised: 23 October 2014 / Accepted: 24 October 2014 /
Published online: 14 November 2014
© Springer Science+Business Media New York 2014

Abstract In multi-source streaming systems, such as peer-to-peer (P2P), because the buffer space of each peer is limited, most systems employ the cache-and-relay schemes that require each peer to cache the most recent video blocks it receives. As long as a peer keeps the initial part of the video stream in the buffer, it can relay the cached blocks to new arriving peers in a pipelining fashion and then reduce the loading of a server. In our previous research work, we propose a novel caching scheme for peer-to-peer on-demand video streaming, called *Dynamic Buffering*, which relies on the feature of *Multiple Description Coding* to gradually reduce the number of cached descriptions in a peer once the buffer is full in order to prolong the service availability of a peer. In this paper we study the service availability of a peer with dynamic buffering for various numbers of different forwarded descriptions, and provide detailed analysis on how the number of different forwarded descriptions affects the average service availability of a peer. In addition, we derive the mathematical formulas of the reduction of the average service availability of a peer for various numbers of different forwarded descriptions, compared to the best case of the service availability of a peer, and the gain of the average service availability of a peer by releasing the overlapped buffer space with child peers. Our experimental results show that the reduction of the average service availability of a peer is only related to the number of different forwarded descriptions. Besides, regardless of arrival rates, most peers forwarding only one description would possess the highest average service availability for various numbers of different forwarding descriptions, which matches the criteria of the previous work *Splitstream*.

Keywords Dynamic buffering · Service availability · Multiple description coding · Peer-to-peer · Streaming

This work was partially supported by National Science Council of Taiwan under contracts NSC 100-2221-E-024-006.

C.-S. Lin (✉) · J.-W. Lin
Department of Computer Science and Information Engineering, National University of Tainan, No. 33,
Sec. 2, Su-Lin St., Tainan, 70005, Taiwan
e-mail: mikelin@mail.nutn.edu.tw

J.-W. Lin
e-mail: sh900010@yahoo.com.tw

1 Introduction

With the rapid advances of Internet and networking technologies, there has been a dramatic proliferation of the demand from users for various multimedia applications, especially for on-demand media streaming services. In on-demand streaming systems, the multimedia contents (such as movies) are pre-recorded, and a user is allowed to issue a request to select any one of the provided videos to watch. Upon receiving a user's request, on-demand streaming systems must ensure he/she receive a continuous and complete playback of the requested video. In general, providing such on-demand media streaming services requires a streaming server to dedicate non-trivial amount of forwarding bandwidth to each user for a long period of time. Such a resource-consuming nature of media streaming drives most service providers to shift the design of media-on-demand (MoD) systems from traditional Client-Server infrastructure to multi-source such as peer-to-peer (P2P) networks [12, 24, 26]. In a P2P system, a peer (or a user) not only consumes the resources from the system, but also contributes its own resources such storage and bandwidth to the system in return [11, 25, 32]. With the aid of peers, the scalability of an on-demand streaming system can be effectively improved, and more users can be concurrently served to solve the problem of flash crowds [6, 20, 30].

In the past, there has been a great deal of work done on providing on-demand streaming services in a P2P manner [4, 5, 8, 10, 17, 18, 22, 23]. Most of these systems employ the cache-and-relay schemes which require each peer to cache the most recent video contents it receives. In general, a user requesting an on-demand video service normally intends to watch the video from the very beginning. Therefore, as long as a peer still keeps the initial part of a video stream in its buffer, it can then relay the cached video blocks to new coming peers requesting the same video in a pipelining fashion. The ability of a peer to contribute to the P2P MoD system by serving new coming peers depends on the amount of available network bandwidth and the existence of the initial part of a video stream in its buffer. A peer cannot serve any new coming peers if the initial part of a video is no longer available in its buffer even if it still has available uploading bandwidth left. Since the downloading and uploading capacity of a peer is determined when it is initialized, the key challenge of maximizing the service availability of a peer for stream relay is how to keep the initial part of a video cached in its buffer as long as possible.

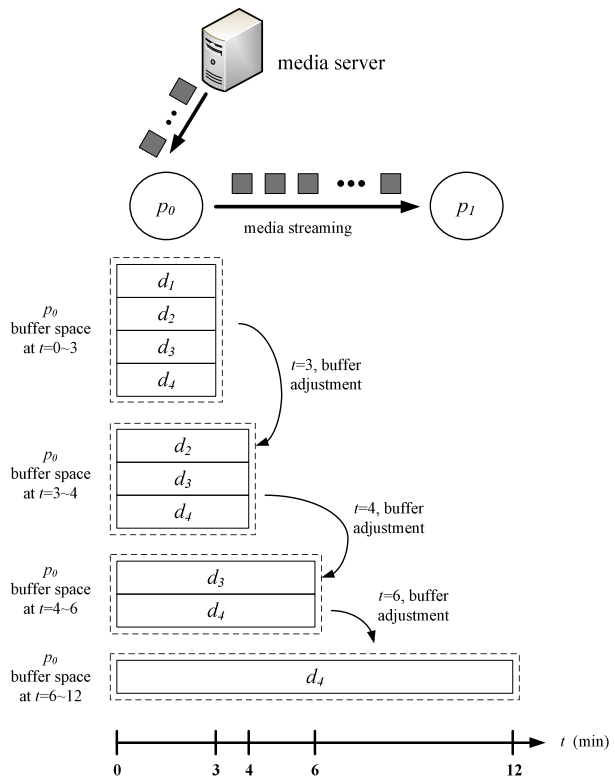
In addition, to address the issues of delivering data on lossy networks and also provide scalable quality of services for heterogeneous clients, the adoption of scalable video coding techniques, such as Multiple Description Coding (MDC), [9, 27] has been proven as a feasible resolution by a lot of research [4, 7, 13, 23, 28, 29]. The MDC is a flexible encoding technique capable of encoding a video into multiple sub-streams, called *descriptions*. Any subset of descriptions can be combined and decoded to reconstruct the original video content with fidelity commensurate to the number of descriptions they receive. The more descriptions are received, the higher the quality of the reconstructed video. There has been a lot of research work for the realization of MDC with existing coding standards. In particular, a great deal of effort has been put into developing an H.264/AVC standard-compliant multiple description (MD) encoder [1–3, 16, 31].

In our previous work, we proposed a novel caching scheme of a peer for peer-to-peer on-demand streaming, called *Dynamic Buffering* [14, 15, 19]. The dynamic buffering relies on the feature of MDC to gradually reduce the number of cached descriptions in a peer once its buffer is full. Preserving as many initial parts of descriptions in the buffer as pos-

sible, instead of losing them all at one time, would significantly extend the service time of a peer and hence increase the service availability of a peer. In dynamic buffering, a cached description in a peer is selected to discard and its buffer space is reallocated to other descriptions to prolong their availability of initial blocks once the buffer space of the peer gets full. As an example shown in Fig. 1, the 12-minute video is evenly encoded into 4 descriptions $d_1, d_2, d_3,$ and d_4 . The first peer, p_0 , arrives at the 0th minute and receives 4 descriptions from the server with the buffer length of each description equal to 3 minutes. Each description is downloaded at the rate of one block per minute. At the 3th minute, p_0 's buffer is filled up with 4 descriptions. Instead of losing the initial parts of all 4 descriptions at one time as the conventional cache-and-relay scheme does, p_0 chooses one description, says d_1 , to discard and then evenly distributes the released buffer space to other descriptions. In this way, now the buffer length of the remaining descriptions is 4 minutes, providing one extra minute of availability for servicing new peers. The p_0 would repeat the above process every time its buffer is filled up until there is only one description left in its buffer. As shown in Fig. 1, during the time 3 to 4, time 4 to 6, and time 6 to 12, p_0 would only have 3, 2, and 1 descriptions available to forward, respectively. If peer p_1 arrives after time 12, since p_0 has no initial part of any description to offer, p_1 will request the desired descriptions from the server.

It is noted that the proposed dynamic buffering, which is known as the buffer management scheme of a peer, can be applied on top of any multi-source streaming networks,

Fig. 1 An example of dynamic buffering with 12-minute buffer



such as tree-based and mesh-based P2P streaming networks [21]. The dynamic buffering only concerns how to preserve the initial parts of received descriptions in a peer as long as possible by dynamically adjusting the buffer space allocated to each cached descriptions. Apparently, how to determine the receiving descriptions, select parent peers, and recover from a peer failure which are mainly relied on the adopted specific P2P streaming network model are independent from the buffer management scheme of a peer such as dynamic buffering.

In general, cached descriptions in dynamic buffering can be classified into *non-forwarded descriptions* if they are not yet forwarded to child peers and *forwarded descriptions* if they are already forwarded to child peers. In our previous analysis [19], we had studied *the best case* of service availability of a peer with dynamic buffering where a peer was assumed to receive descriptions without forwarding any descriptions. However, in practical, generally a peer would probably forward cached descriptions to child peers on P2P streaming networks. When a peer's buffer is filled up, in addition to selecting a non-forwarded description to drop for extending its service time, we may also select a forwarded description to drop. In this case, only the buffer space overlapping with its child peer, i.e., only the partial buffer space allocated to a description, would be released to preserve the continuity of streaming. Once any description of a peer is forwarded to a child peer, compared to the case with only non-forwarded descriptions, its service availability would be reduced.

In this paper, we discuss and provide detailed analysis on how the number of different forwarded descriptions in a peer with dynamic buffering could affect the service availability. In addition, the mathematical formulas of the reduction of the service availability for forwarding various numbers of different descriptions in a peer are derived. Note that we assume the uploading and downloading bandwidths of a peer are equal and all peers are altruistic without causing the problem of free riders. The experimental results confirm that compared to the best case of service availability of a peer, the reduction of service availability was only related to the number of different forwarded, not cached, descriptions. And, the average service availability of a peer is the same when it forwards one kind of description or none to child peers. Moreover, the experimental results also show that a peer forwarding only one kind of description would have the most average service availability. Interestingly, this result is in agreement with the Splitstream [4] where a peer is limited to forward only one kind of description in a P2P streaming network. In conclusion, the contribution of this paper can be summarized as follows.

1. We confirmed the best dropping policy for dynamic buffering, which would be first applying non-forwarded descriptions dropped and then releasing overlapped buffer of forwarded descriptions, which is called dynamic buffering with dropping non-forwarded description first.
2. We derived the mathematical formulas of the average service availability of a peer with dynamic buffering applying dropping non-forwarded description first.
3. We found that a peer would have the most average service availability of a peer when it only forwards one kind of description.

The remainder of the paper is organized as follows. In Section 2 we describe the best and general cases for service availability of a peer with dynamic buffering. In Section 3 we derive the general formula of the average service availability of a peer. The experimental results and analysis are presented in Section 4, and finally conclusions are drawn in Section 5.

2 Service availability of a peer with dynamic buffering

2.1 The best case

In our previous work [19], we have established the formula of service availability of a peer with dynamic buffering for the best case where a peer only receives descriptions without forwarding (aka. non-forwarding case). Assume that a peer with the buffer space of b receives n descriptions of equal bit rate r at time $T = 0$. Therefore, b/nr buffer space is allocated to each description. To simplify the analysis, assume that r is equal to the bit rate of one description. Figure 2 shows the diagram of service availability for a peer with dynamic buffering. Based on our previously analysis on dynamic buffering, the $n, n - 1, n - 2, \dots, 2$, and 1 available numbers of descriptions in will be expired at times $b/n, b/(n - 1), b/(n - 2), \dots, b/2$, and b , respectively. The available service duration of a peer for servicing x descriptions, where $1 \leq x \leq n$, can be computed as the difference of the expiration time for providing $x - 1$ descriptions and that for providing x descriptions. For example, the available service duration for providing $n - 1$ descriptions is $b/(n - 1) - b/n = b/n(n - 1)$. The ratios of available service durations for providing $n, n - 1, n - 2, \dots, 2$, and 1 descriptions are $1/n, 1/n(n - 1), 1/(n - 1)(n - 2), \dots, 1/(3 \times 2)$, and $1/2$, respectively. It is worthy noted that the service duration for providing *one* description is one half of the whole service duration of a peer. In contrast, the ratio of the service available duration of providing n descriptions is $1/n$, and its significance on service availability of a peer depends on the value of n .

Depending on the arrival time of a child peer, the number of selectable descriptions to forward for a parent peer would be ranged from one to n . The average number of selectable descriptions can be computed by the summation of the product of each possible number of available descriptions and its ratio of available service duration,

$$\begin{aligned}
 & n \times \frac{1}{n} + (n - 1) \times \frac{1}{n(n - 1)} + (n - 2) \times \frac{1}{(n - 1)(n - 2)} + \dots + 2 \times \frac{1}{2 \times 3} + 1 \times \frac{1}{1 \times 2} \\
 &= 1 + \frac{1}{n} + \frac{1}{n - 1} + \frac{1}{n - 2} + \dots + \frac{1}{3} + \frac{1}{2} \\
 &= \sum_{i=1}^n \frac{1}{i} = \ln n + \gamma.
 \end{aligned}$$

It is interesting to note that the average number of selectable descriptions in a peer when a request of stream relay arrives is n -th harmonic number, and its approximate value is well-known as $\ln n + \gamma$ where γ is *Euler’s constant* (~ 0.57721).

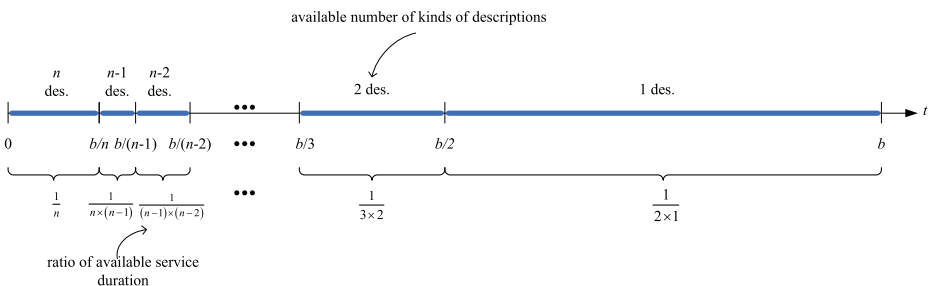


Fig. 2 Diagram of service availability of a peer with dynamic buffering

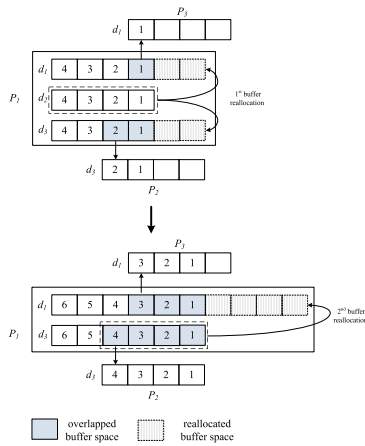
It is noted that if a peer only forwards one kind of description to child peers, there is no reduction on its service availability because it would gradually drop all of other non-forwarded descriptions to extend the service time of the description. Therefore, the cases of not forwarding any descriptions and only forwarding one kind of description to child peers are the best cases for service availability of a peer.

2.2 The general case

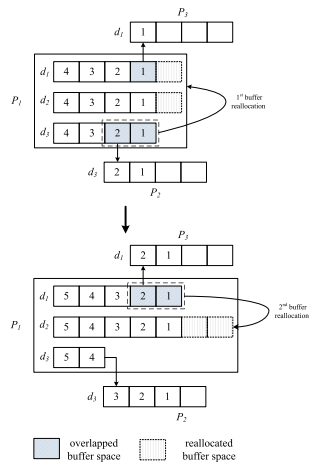
In general, a peer would forward available descriptions to its child peers on a practical P2P streaming network. At any given time, there are some descriptions which are selected to be forwarded to child peers. In this general case, once the buffer space of a peer is filled up, which description is selected to be dropped and release its buffer space to extend the availability of other descriptions? a non-forwarded description? or a forwarded description? To analyze the service availability of this general case, there are two policies for releasing buffer space of descriptions, called *dropping non-forwarded description first* and *dropping forwarded description first*. For dropping non-forwarded description first policy, once the peer's buffer is filled up, it would first drop the non-forwarded descriptions. If there are multiple non-forwarded descriptions, one is arbitrarily selected to release its buffer space. Until there are no non-forwarded descriptions left in the peer's buffer, the peer would then start to release the overlapped buffer space of forwarded descriptions. The forwarded descriptions are selected to drop based on the decreasing order of the size of the overlapping buffer space. Again, a tie is broken arbitrarily. In contrast, the dropping forwarded description first policy just does the opposite.

Figure 3 shows the examples of the two description dropping policies. Assume a peer p_1 with the buffer space of 12 blocks receives 3 descriptions. Peer p_1 has forwarded the description d_3 to p_2 and d_1 to p_3 . Figure 3a shows the a case of dropping non-forwarded description first policy. When the first time p_1 's buffer is filled up, p_1 completely drops the non-forwarded description d_2 and then evenly reallocate the buffer of d_2 to the other two descriptions d_1 and d_3 . When the second time p_1 's buffer is filled up, there are two forwarded descriptions in the buffer. The buffer space of d_3 has the largest overlap with the child peer, therefore d_3 is selected to release its overlap buffer space to the description d_1 . As shown in the figure, this policy could prolong the service time of peer p_1 with only one description of d_1 left up to the 10th time slot. Figure 3b shows a case of dropping forwarded description first policy. When the first time p_1 's buffer is filled up, peer p_1 selects the forwarded description d_3 which has the largest overlapped buffer space, the blocks 1 and 2, with the child peer and evenly reallocates these two blocks to the other two descriptions d_1 and d_2 . When the second time p_1 's buffer is filled up, similarly, p_1 releases the buffer of d_1 and it is reallocated to the description d_2 . The non-forwarded descriptions would be kept until there are no forwarded descriptions in p_1 's buffer. As shown in the figure, this policy only prolongs the service time of p_1 with the description d_2 left up to the 6th time slot. Clearly, as far as the service availability of a peer is concerned, the dropping non-forwarded description first is better than the dropping forwarded description first. Based on this conclusion, in this paper we use dropping non-forwarded description policy for description selection. Note that if p_1 does not forward no descriptions to other peers in Fig. 3, i.e., all cached descriptions are non-forwarded ones, the service time of p_1 would last up to the 12th time slot, which represents the best case of service availability of p_1 .

Compared to the best case of dynamic buffering, the reduction of service availability of a peer depends on the amount of the overlapped buffer space with child nodes, which is



(a) Dropping non-forwarded descriptions first.



(b) Dropping forwarded descriptions first.

Fig. 3 Two dropping policies for dynamic buffering

determined by the number of different forwarded descriptions and the latest forwarding time of a description. Note that given a peer forwarding n descriptions to n distinct child nodes, the number of different descriptions may be ranged from one to n . Apparently, the more the number of different forwarded descriptions, the less the number of different non-forwarded descriptions and then the more reduction on the service availability of a peer. In addition, when a peer forwards a description to multiple child peers arriving at different times, the releasable buffer space of the description for reallocation could be determined by the latest child peer. Figure 4 shows an example of determining the largest releasable buffer space for reallocation. Peer p_j arrives 3 time slots later than peer p_i , and they are receiving the same description, d_k , from a parent peer. As shown in the figure, only the blocks 1, 2, and 3 can be released and reallocated to other descriptions for dynamic buffering. In conclusion, in this paper we study the impacts of the number of different forwarded descriptions and the arriving time of the latest child peers of a description on the service availability of a peer

with the dropping non-forwarded description first policy, which is considered as the general case of dynamic buffering.

3 Analysis for service availability of the general case

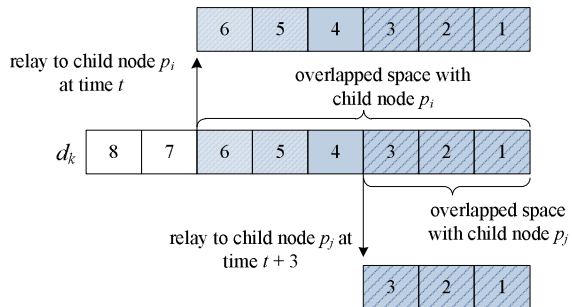
As mentioned previously, the dropping non-forwarded description first policy of dynamic buffering consists of two phases, first sequentially dropping the buffer space of non-forwarded descriptions and then releasing the overlapped buffer space of forwarded descriptions if no non-forwarding description left in the buffer. To study the service availability of a peer for the general case, compared to the service availability of the best case, we first consider the reduction of service availability of dropping non-forwarded descriptions and then derive the reduction of service availability of releasing the overlapped buffer space of forwarded descriptions, which are described in the following sections.

3.1 Service availability by dropping non-forwarded descriptions

In this section, we discuss the effect of the number of different forwarded descriptions in a peer on its service availability with Dynamic Buffering. In dynamic buffering, the number of disposable non-forwarded descriptions are limited by the forwarded descriptions in a peer. Compared to the best case, the service availability of a peer would be reduced if a peer forwards any cached descriptions to child peers, and its service availability would be decreased along with the increase of the number of different forwarded descriptions. The question is, how much that would be? Assume that a peer forwards f different descriptions, in the end there are f forwarded descriptions which would not be selected to drop during the first phases of dynamic buffering, and there are no buffer adjustments after time b/f . Therefore, we would lose the capability of buffer adjustment during the service availability of $f - 1, f - 2, \dots, 2,$ and 1 non-forwarded descriptions. By observing Fig. 2, the average reduction of service availability for forwarding f descriptions would be, $\frac{1}{f \times (f-1)} \times (f - 1) + \frac{1}{(f-1) \times (f-2)} \times (f - 2) + \dots + \frac{1}{3 \times 2} \times 2 + \frac{1}{2 \times 1} \times 1 = \frac{1}{f} + \frac{1}{f-1} + \dots + \frac{1}{3} + \frac{1}{2} = \sum_{i=2}^f \frac{1}{i}$.

Figure 5 shows an example of a peer with the buffer space of 12 receiving 4 descriptions. Before 1st buffer adjustment $t = 3$, p_0 has forwarded three descriptions, says, $d_2, d_3,$ and d_4 to $p_1, p_2,$ and $p_3,$ respectively. There are no buffer adjustments after time $12/3 = 4$. Figure 6 shows the diagram of the service availability of this example. Compared to the best case, the reduction of average service availability of this case is $1 \times \frac{1}{2} + 2 \times \frac{1}{2 \times 3} = \frac{1}{2} + \frac{1}{3}$.

Fig. 4 An example of determining the largest releasable buffer space for reallocation



Lemma 1 Compared to the best case of dynamic buffering, the reduction of average service availability by only dropping non-forwarded descriptions for a peer with f forwarded descriptions is,

$$\ln f + \gamma - 1, \text{ where } f > 1.$$

Proof

$$\sum_{i=2}^f \frac{1}{i} = \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{f} = \left(1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{f}\right) - 1 \tag{1}$$

$$= \ln f + \gamma - 1. \tag{2}$$

□

In Lemma 1, we can find a clear relationship between the reduction of average service availability and the number of different forwarded descriptions f . The reduction of average service availability is only related to the number of different forwarded descriptions, f , and it is independent from the number of received descriptions, n . It is noted that the best cases of average service availability occur at $f = 0$ and $f = 1$ which do not cause any reduction on service availability, so these cases are not applicable to Lemma 1.

3.2 Service availability by releasing the overlapped buffer space of forwarded descriptions

When there are only forwarded descriptions left in a peer’s buffer after dropping all the buffer space of non-forwarded descriptions, to further extend the service availability, the peer, next, would select a forwarded description one by one to release the buffer space overlapped with a child peer until there is one forwarded description left. The selection would be based on the lengths of overlapped buffer space of forwarded descriptions, and the forwarded description with the longest overlapping buffer space would be selected first.

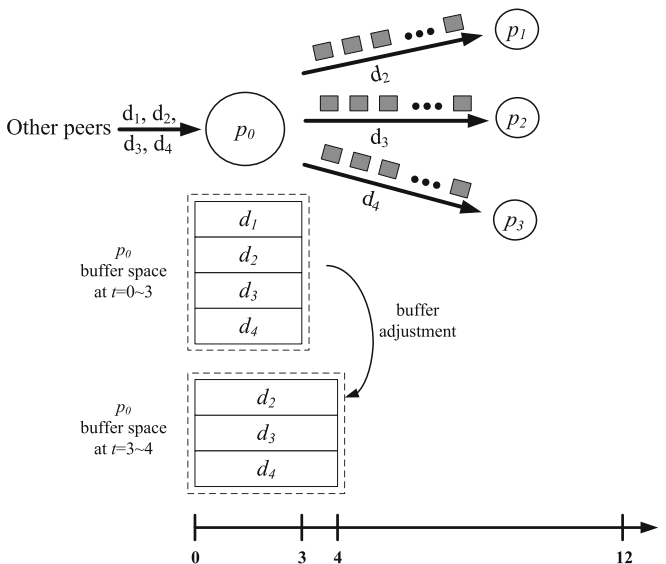


Fig. 5 An example of dynamic buffering with 12-minute buffer and forwarded descriptions $f = 3$

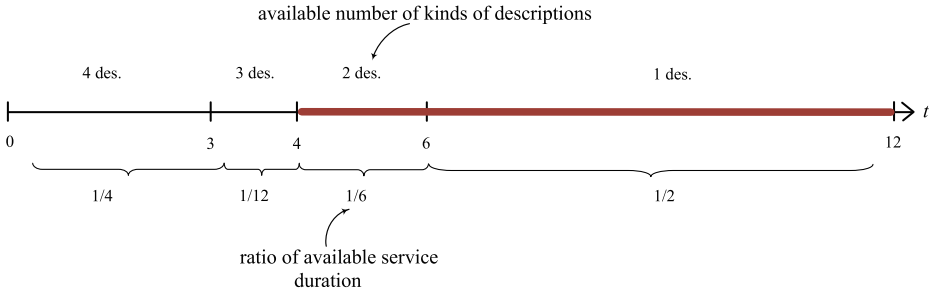


Fig. 6 Diagram of the service availability for Fig. 5

Releasing the overlapped buffer space of forwarded descriptions can additionally extend service time of a peer. In this section, we would like to infer the mathematical formulas of additional average service availability of a peer gained by releasing the overlapped buffer space of forwarded descriptions.

Figure 7 shows the general case of releasing overlapped buffer space for f forwarded descriptions. Assume that peers p_1, p_2, \dots, p_f denote the latest peers requesting descriptions $d_{n-f}, d_{n-f+1}, \dots, d_n$, at the times t_1, t_2, \dots, t_f , respectively. Without losing the generality, we assume $t_1 \geq t_2 \geq \dots \geq t_f$. S_1, S_2, \dots, S_f represents the additional average service availability by releasing the overlapped buffer space of descriptions d_1, d_2, \dots, d_f , respectively. As shown in the figure, at the 1st buffer adjustment, the description d_1 would release its overlapped buffer space from time t_1 to b/f . That is, $S_1 \times b = b/f - t_1 \Rightarrow S_1 = 1/f - t_1/b$. This released buffer space would be evenly distributed to the other $f - 1$ descriptions, and each would gain additional $\frac{S_1 \times b}{f-1}$ buffer time. Therefore, at the 2nd buffer adjustment, the description d_2 would release its overlapped buffer space from time t_2 to b/f plus the additional $\frac{S_1 \times b}{f-1}$. That is, $S_2 \times b = (b/f - t_2) + \frac{S_1 \times b}{f-1} \Rightarrow S_2 = 1/f - t_2/b + \frac{S_1}{f-1}$. Again, this released buffer space would be evenly distributed to the other $f - 2$ descriptions, and each would gain additional $\frac{S_2 \times b}{f-2}$ buffer time. Therefore, $S_3 \times b = (b/f - t_3) + \frac{S_1 \times b}{f-1} + \frac{S_2 \times b}{f-2} \Rightarrow S_3 = 1/f - t_3/b + \frac{S_1}{f-1} + \frac{S_2}{f-2} = 1/f - t_3/b + \sum_{j=1}^2 \frac{S_j}{f-j}$. From above derivation, we can infer the general formula of S_i ,

$$S_i = \frac{1}{f} - \frac{t_i}{b} + \sum_{j=1}^{i-1} \frac{S_j}{f-j}, \tag{3}$$

where $f \geq 2$ and $1 \leq i \leq f - 1$. The $1/f - t_i/b$ represents average service availability gained by releasing the overlapped buffer of description d_i , and $\sum_{j=1}^{i-1} \frac{S_j}{f-j}$ represents the additional average service availability gained by releasing the overlapped buffer of descriptions d_1 to d_{i-1} . It is noted that the average service availability gained by releasing the overlapped buffer space of $f - 1$ forwarded descriptions is, $\sum_{i=1}^{f-1} S_i$.

To solve the recursion of (3), we list $S_1, S_2, S_3,$ and S_4 to observe the occurrence.

$$S_1 = \frac{1}{f} - \frac{t_1}{b},$$

$$\begin{aligned} S_2 &= \frac{1}{f} - \frac{t_2}{b} + \frac{S_1}{f-1} \\ &= \frac{1}{f-1} - \frac{t_2}{b} - \frac{1}{f-1} \times \frac{t_1}{b} \\ &= \frac{1}{f-1} \left(1 - \frac{t_1}{b} \right) - \frac{t_2}{b}, \end{aligned}$$

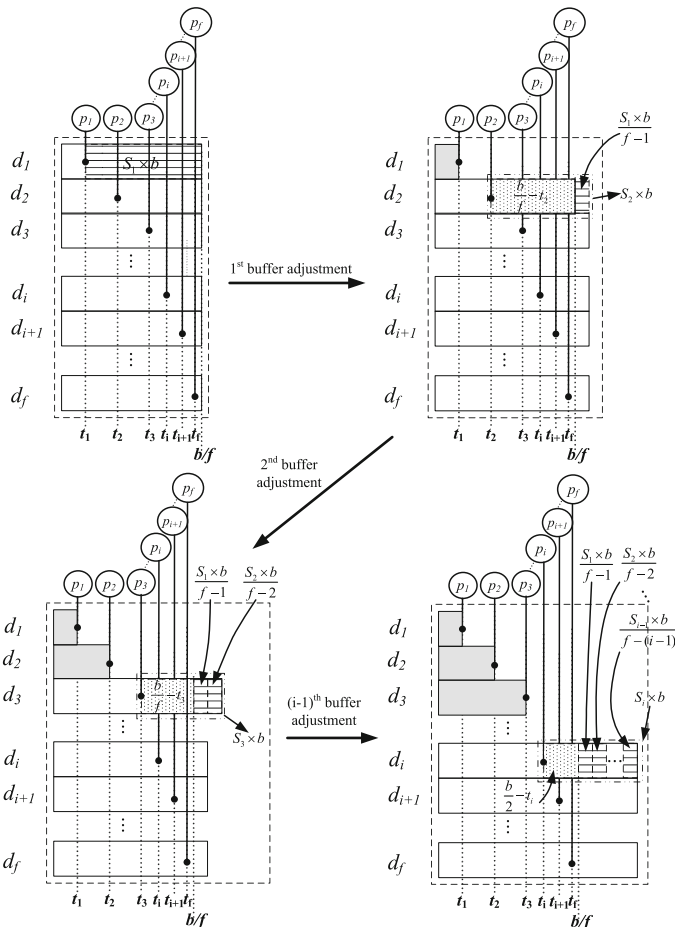


Fig. 7 The general case of releasing overlapped buffer space for f forwarded descriptions

$$\begin{aligned}
 S_3 &= \frac{1}{f} - \frac{t_3}{b} + \frac{S_1}{f-1} + \frac{S_2}{f-2} \\
 &= \frac{1}{f-2} - \frac{t_3}{b} - \frac{1}{f-2} \times \frac{t_1}{b} - \frac{1}{f-2} \times \frac{t_2}{b} \\
 &= \frac{1}{f-2} \left(1 - \frac{t_1}{b} - \frac{t_2}{b} \right) - \frac{t_3}{b}, \text{ and}
 \end{aligned}$$

$$\begin{aligned}
 S_4 &= \frac{1}{f} - \frac{t_4}{b} + \frac{S_1}{f-1} + \frac{S_2}{f-2} + \frac{S_3}{f-3} \\
 &= \frac{1}{f-3} - \frac{t_4}{b} - \frac{1}{f-3} \times \frac{t_1}{b} - \frac{1}{f-3} \times \frac{t_2}{b} - \frac{1}{f-3} \times \frac{t_3}{b} \\
 &= \frac{1}{f-3} \left(1 - \frac{t_1}{b} - \frac{t_2}{b} - \frac{t_3}{b} \right) - \frac{t_4}{b}.
 \end{aligned}$$

From the above statements, (3) now can be rewritten as the general equation,

$$S_i = \frac{1}{f} - \frac{t_i}{b} + \sum_{j=1}^{i-1} \frac{S_j}{f-j} = \frac{1}{f-i+1} \left(1 - \frac{1}{b} \sum_{j=1}^{i-1} t_j \right) - \frac{t_i}{b}. \tag{4}$$

Proof Here we are dealing with the statement,

$$F(i) : \frac{1}{f} - \frac{t_i}{b} + \sum_{j=1}^{i-1} \frac{S_j}{f-j} = \frac{1}{f-i+1} \left(1 - \frac{1}{b} \sum_{j=1}^{i-1} t_j \right) - \frac{t_i}{b}, \text{ where } i \geq 1.$$

Base case: When $i = 1$,

$$F(1) : \frac{1}{f} - \frac{t_1}{b} = \frac{1}{f}(1) - \frac{t_1}{b}.$$

The both sides of (4) are equal and (4) is true for $i = 1$. That is, $F(1)$ holds.

Induction hypothesis: Assume $F(k)$ holds for $i = k$. That is,

$$S_k = \frac{1}{f} - \frac{t_k}{b} + \sum_{j=1}^{k-1} \frac{S_j}{f-j} = \frac{1}{f-k+1} \left(1 - \frac{1}{b} \sum_{j=1}^{k-1} t_j \right) - \frac{t_k}{b}.$$

Induction step: Then,

$$\begin{aligned}
 F(k+1) &: \frac{1}{f} - \frac{t_{k+1}}{b} + \sum_{j=1}^k \frac{S_j}{f-j} \\
 &= \frac{1}{f} - \frac{t_{k+1}}{b} + \sum_{j=1}^{k-1} \frac{S_j}{f-j} + \frac{S_k}{f-k}
 \end{aligned}$$

From (3), we get $\frac{1}{f} + \sum_{j=1}^{k-1} \frac{S_j}{f-j} = S_k + \frac{t_k}{b}$. So,

$$\begin{aligned}
 F(k+1) &: S_k + \frac{t_k}{b} - \frac{t_{k+1}}{b} + \frac{S_k}{f-k} \\
 &= \left(1 + \frac{1}{f-k} \right) S_k + \frac{t_k}{b} - \frac{t_{k+1}}{b}
 \end{aligned}$$

Next, assign $S_k = \frac{1}{f-k+1} \left(1 - \frac{1}{b} \sum_{j=1}^{k-1} t_j\right) - \frac{t_k}{b}$. Then,

$$\begin{aligned}
 F(k+1) &: \left(\frac{f-k+1}{f-k}\right) \left(\frac{1}{f-k+1} \left(1 - \frac{1}{b} \sum_{j=1}^{k-1} t_j\right) - \frac{t_k}{b}\right) + \frac{t_k}{b} - \frac{t_{k+1}}{b} \\
 &= \frac{1}{f-k} \left(1 - \frac{1}{b} \sum_{j=1}^{k-1} t_j\right) - \frac{t_k}{b} \left(\frac{f-k+1}{f-k} - 1\right) - \frac{t_{k+1}}{b} \\
 &= \frac{1}{f-k} \left(1 - \frac{1}{b} \sum_{j=1}^{k-1} t_j\right) - \frac{1}{f-k} \times \frac{t_k}{b} - \frac{t_{k+1}}{b} \\
 &= \frac{1}{f-k} \left(b - \sum_{j=1}^{k-1} t_j - t_k\right) - t_{k+1} \\
 &= \frac{1}{f-k} \left(1 - \frac{1}{b} \sum_{j=1}^k t_j\right) - \frac{t_{k+1}}{b}.
 \end{aligned}$$

Thus, (4) holds for $i = k + 1$, and the proof of the induction step is complete.

Conclusion: By the principle of induction, (4) is true for $i \geq 1$.

□

Lemma 2 *By dynamic buffering with dropping non-forwarded description first, additional average service availability gained by releasing the overlapped buffer space of the i -th forwarded description for a peer with f forwarded descriptions is,*

$$S_i = \frac{1}{f} - \frac{t_i}{b} + \sum_{j=1}^{i-1} \frac{S_j}{f-j} = \frac{1}{f-i+1} \left(1 - \frac{1}{b} \sum_{j=1}^{i-1} t_j\right) - \frac{t_i}{b}, \text{ where } f \geq 2 \text{ and } 1 \leq i \leq f-1.$$

By Lemma 2, instead of using recursive computation as (3), we now can directly estimate the S_i by the latest relay time of i forwarded descriptions, the number of different forwarded descriptions, f , and the buffer space of the peer, b , which would greatly reduce the computational complexity.

Lemma 3 *Given a peer buffering n descriptions and forwarding f different descriptions, its average service availability by dynamic buffering with dropping non-forwarded description first is, $SA_f^n = (\ln \frac{n}{f} + 1) + \sum_{i=1}^{f-1} S_i$, where $1 \leq i \leq n$ and $1 \leq f \leq n$.*

Proof By Lemma 1 and 2, SA_f^n would be the average service availability of the best case subtracting the reduction of average service availability caused by forwarded descriptions defined in Lemma 1, and then adding the additional average service availability owing to releasing the overlapped buffer spaces of forwarded descriptions to extend their service

times defined in Lemma 2. That is,

$$SA_f^n = [(\ln n + \gamma) - (\ln f + \gamma - 1)] + \sum_{i=1}^{f-1} S_i = \left(\ln \frac{n}{f} + 1\right) + \sum_{i=1}^{f-1} S_i.$$

□

4 Experimental results

In this section, we present the experimental results of the service availability for various number of forwarded descriptions with respect to various numbers of buffered descriptions. As we stated previously, the dynamic buffering could be applied on top of any P2P streaming networks. Therefore, in our experiments we assumed that no network topology was involved. All peers were assumed to request the same video. The uploading and downloading bandwidths of a peer were equal, and all peers were assumed to be altruistic without causing the problem of free riders. Furthermore, since the buffer management of a peer is independent from the process of the recovery of source peer failure caused by either the leave of parent peers or the malfunction of networks, we also assumed that there was no peer failure in our experiments. The experimental environment was simulated in Matlab. Besides, in [14] we already showed the merits of dynamic buffering over existing work such as CoopNet, in this paper we further study the characteristics of dynamic buffering applied to delivering MDC streams on P2P networks, specially focusing on how the number of different forwarded descriptions affecting the average service availability of a peer by mathematical models.

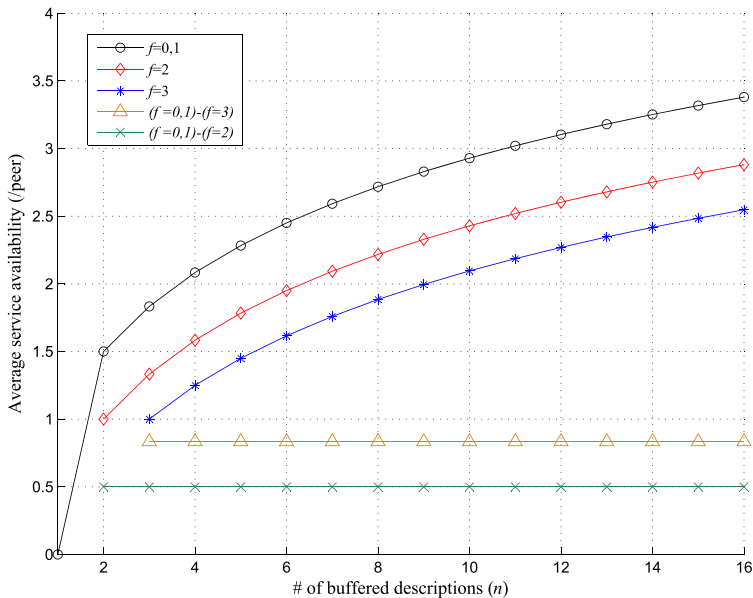


Fig. 8 The average service availability for the cases of $f = 0, 1, 2,$ and 3 without releasing overlapped buffer space of forwarded descriptions

Figure 8 shows the average service availability for the cases of forwarded descriptions $f = 0, 1, 2,$ and 3 with the buffer space of 12. By Lemma 1, the average service availability for f forwarded descriptions can be derived by $\ln n + \gamma - (\ln f + \gamma - 1) = \ln \frac{n}{f} + 1$. Note that the average service availability of $f = 1$ is the same as that of $f = 0$. In this experiment we were not concerned about the additional service availability gained by releasing overlapped buffer space of forwarded descriptions. As shown in the figure, the average service availability for $f = 0, 1, 2,$ and 3 increases along with the number of buffered descriptions. With the number of buffered descriptions $n = 4, 8,$ and 16 , the average service availability of $f = 2$ is 24 %, 18.3 %, and 14.7 % less than that of the best case, and the average service availability of $f = 3$ is 40 %, 30.6 %, and 24.6 % less than that of the best case, respectively. It is interesting to see that given a fixed f forward descriptions, when the number of buffered descriptions is increased, the percentage of the reduction of average service availability would be decreased as well. When n gets very large, such a percentage reduction would be negligible and the average service availability of f forward descriptions would be close to that of the best case. In Fig. 8 we also shows the difference between the average service availability of $f = 2$ and that of the best case $f = 0$ with respect to n , denoted as $(f = 0, 1) - (f = 2)$. It is the constant value of 0.5 regardless of the value of n . Similarly, the $(f = 0, 1) - (f = 3)$ is always to be 0.83 regardless of the value of n . Those facts show that compared to the best case, the reduction of average service availability is only related to the number of different forwarded descriptions f , not the number of buffered descriptions n .

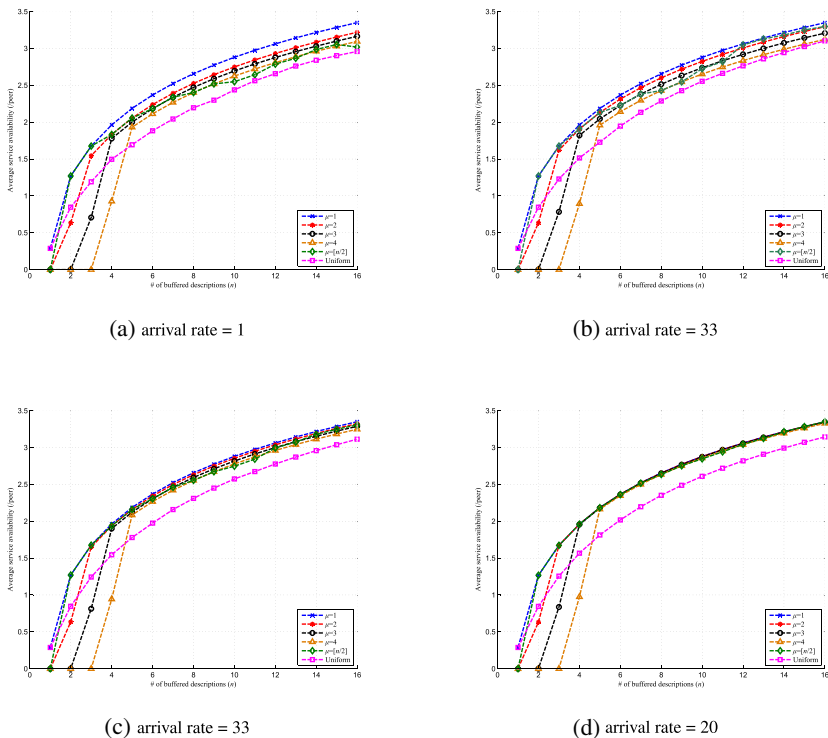


Fig. 9 The average service availability for various number of buffered receiving descriptions

To evaluate average service availability of a peer forwarding various numbers of descriptions, In the next experiment, we define the probability of a peer forwarding f descriptions as $p(f)$, $0 \leq p(f) \leq 1$ and $1 \leq f \leq n$, where n is the number of buffered descriptions, $1 \leq n \leq 16$. Two distributions of $p(f)$ are discussed which are the uniform distribution ($p(f) = \frac{1}{n+1}$) and the normal distribution of $p(f)$ with the mean $u = 1, 2, 3, 4$, and $\lfloor n/2 \rfloor$. In addition, the arrival rates of peers followed the Poisson distribution, which were equal to 1, 2, 5, and 20 (peers/second). Based on above assumptions, overall the average service availability of a peer would be, $SA_f^n \times p(f)$.

Figures 9a–d shows the average service availability with respect to various numbers of buffered descriptions when the arrival rate equal to 1, 2, 5, 20, respectively. In general, the average service availability decreases along with the increase of the number of buffered descriptions. In our experiment, as shown in the figures, assuming $p(f)$ in normal distribution generally generates higher average service availability with respect to various numbers of buffered descriptions than that in uniform distribution. Through the observation of Fig. 9a–d, we could draw three important conclusions. First, most peers forwarding only one description (i.e., $u = 1$) would possess the highest average service availability among different values of u , regardless of arrival rates. This matches the conclusion in Section 2.1 stating that a peer forwarding none ($u = 0$) and one ($u = 1$) description possess the best average service availability. Second, as the arrival rate increases, the gaps of average service availability between $u = 1$ and $u = 2, 3, 4, \lfloor n/2 \rfloor$ become smaller. When the arrival rate equals 20 peers/second, they are almost identical as shown in Fig. 9d. This is because when the arrival rate increases, the chance of child peers coming in earlier may be increased and hence the average service availability would become larger. Finally, most importantly, based on the experimental results of these four figures, a peer forwards only one description would have the most average service availability.

5 Conclusion

In this paper we discuss the average service availability of a peer with dynamic buffering for the dropping non-forwarded description first, and provide detailed analysis on how the number of different forwarded descriptions affects the average service availability of a peer. In addition, we derive the mathematical formulas of the reduction of average service availability for various numbers of different forwarded descriptions and the gain of the average service availability of a peer by releasing the overlapped buffer space with child peers. Our experimental results show that the reduction of the average service availability of a peer is only related to the number of different forwarded descriptions. Besides, regardless of arrival rates, most peers forwarding only one description would possess the highest average service availability among different probability distributions for various numbers of different forwarding descriptions, which matches the criteria of the previous work, Splitstream. In the future we plan to apply the dynamic buffering with dropping non-forwarded description first on practical P2P streaming systems to observe its performance. In addition, whether the optimization of average service availability of a peer might lead to the optimum global average service availability on P2P streaming networks will be further explored.

Acknowledgments The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper.

References

1. Apostolopoulos J (2001) Reliable video communication over lossy packet networks using multiple state encoding and path diversity. In: Proc. of visual communications: image processing, pp 392–409
2. Bernardini R, Durigon M, Rinaldo R, Celetto L, Vitali A (2004) Polyphase spatial subsampling multiple description coding of video streams with h264. In: Proc. of IEEE international conference on image processing, vol 5, pp 3213–3216
3. Campana O, Milani S (2004) A multiple description coding scheme for the h.264/avc coder. In: Proceedings of the international conf. on telecommunication and computer networks, pp 191–195
4. Castro M, Druschel P, Kermarrec AM, Nandi A, Rowstron AIT, Singh A (2003) Splitstream: high-bandwidth multicast in cooperative environments. In: SOSIP, pp 298–313
5. Chadagorn A, Khalil I, Cameron C, Tari Z (2014) Pilecast: multiple bit rate live video streaming over bittorrent. *J Netw Comput Appl* 39(0):167–178
6. Chen Y, Zhang B, Chen C, Chiu DM (2014) Performance modeling and evaluation of peer-to-peer live streaming systems under flash crowds. *IEEE/ACM Trans Netw* 22(4):1106–1120
7. Correia P, Assuncao PA, Silva V (2012) Multiple description of coded video for path diversity streaming adaptation. *IEEE Trans Multimed* 14(3):923–935
8. Do TT, Hua KA, Tantaoui MA (2004) P2VoD : providing fault tolerant video-on-demand streaming in peer-to-peer environment. In: Proc. of IEEE communications, pp 1467–1472
9. Goyal VK (2001) Multiple description coding: compression meets the network. *IEEE Signal Process Mag* 18(5):74–93
10. Guo Y, Suh K, Kurose J, Towsley D (2003) P2Cast: peer-to-peer patching scheme for VoD service. In: Proc. of ACM WWW, pp 301–309
11. Hu C, Chen M, Xing C, Zhang G (2014) Exploring the optimal substream scheduling and distribution mechanism for data-driven {P2P} media streaming. *Comput Commun* 44(0):14–25
12. Li B, Wang Z, Liu J, Zhu W (2013) Two decades of internet video streaming: a retrospective view. *ACM Trans Multimedia Comput Commun Appl* 9(1s):33:1–33:20
13. Liao Y, Gibson JD (2011) Routing-aware multiple description video coding over mobile ad-hoc networks. *IEEE Trans Multimed* 13(1):132–142
14. Lin CS (2010) Improving the availability of scalable on-demand streams by dynamic buffering on P2P networks. *KSII Trans Int Inf Syst (TIIS)* 4(4):491–508
15. Lin CS (2013) Balanced dynamic buffering for scalable video-on-demand streaming on peer-to-peer networks. *Multimed Tools Appl* 62(3):701–718
16. Lin CS, Chang RH, Lin JW (2014) Tubeck: a novel peer-to-peer streaming system with loopback-mdc for scalable h.264/avc videos. *Multimed Appl* 72(2):1653–1679
17. Lin CS, Cheng YC (2007) P2mcmd: a scalable approach to vod service over peer-to-peer networks. *J Parallel Distrib Comput* 67(8):903–921
18. Lin CS, Lee I et al (2010) Applying multiple description coding to enhance the streaming scalability on cdn-p2p network. *Int J Commun Syst* 23(5):553–568
19. Lin CS, Yan MJ (2013) Dynamic peer buffer adjustment to improve service availability on peer-to-peer on-demand streaming networks. *Peer-to-Peer Netw Appl* 7(1):1–15
20. Liu F, Li B, Zhong L, Li B, Jin H, Liao X (2012) Flash crowd in p2p live streaming systems: fundamental characteristics and design implications. *IEEE Trans Parallel Distrib Syst* 23(7):1227–1239
21. Magharei N, Rejaie R, Guo Y (2007) Mesh or multiple-tree: a comparative study of live p2p streaming approaches. In: 26th IEEE international conference on computer communications (INFOCOM 2007), pp 1424–1432
22. Magharei N, Rejaie R, Rimac I, Hilt V, Hofmann M (2014) Isp-friendly live p2p streaming. *IEEE/ACM Trans Netw* 22(1):244–256
23. Padmanabhan VN, Wang HJ, Chou PA, Sripanidkulchai K (2002) Distributing streaming media content using cooperative networking. In: ACM conference on NOSSDAV, pp 177–186
24. Ramzan N, Park H, Izquierdo E (2012) Video streaming over p2p networks: challenges and opportunities. *Signal Process Image Commun* 27(5):401–411
25. Schollmeier R (2001) A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In: 2001 Proceedings first international conference on peer-to-peer computing, pp 101–102
26. Stoica I, Morris R, Liben-Nowell D, Karger D, Kaashoek M, Dabek F, Balakrishnan H (2003) Chord: a scalable peer-to-peer lookup protocol for internet applications. *IEEE/ACM Netw* 11(1):17–32

27. Wang Y, Reibman AR, Lin S (2005) Multiple description coding for video delivery. *Proc IEEE* 93(1):57–70
28. Xia P, Chan SH, Jin X (2011) Optimal bandwidth assignment for multiple-description-coded video. *IEEE Trans Multimed* 13(2):366–375
29. Xu Y, Zhu C, Zeng W, Li XJ (2012) Multiple description coded video streaming in peer-to-peer networks. *Signal Process Image Commun* 27(5):412–429
30. Yuan X, Min G, Ding Y, Liu Q, Liu J, Yin H, Fang Q (2013) Adaptive resource management for p2p live streaming systems. *Future Gener Comput Syst* 29(6):1573–1582
31. Zandon'a N, Milani S, Giusti AD (2005) Motion-compensated multiple description video coding for the h.264/avc standard. In: *Proc. of IADAT international conf. on multimedia, image processing and computer vision*, pp 290–294
32. Zhang J, Xing W, Wang Y, Lu D (2014) Modeling and performance analysis of pull-based live streaming schemes in peer-to-peer network. *Comp Commun* 40:22–32



Chow-Sing Lin received the Ph.D. degree in Computer Engineering from the University of Central Florida, Florida, USA, in 2000. He is currently a Professor in the Department of Computer Science and Information Engineering, National University of Tainan, Taiwan. His research interests include wired/wireless media delivery, peer-to-peer media streaming, distributed multimedia systems, mobile computing, media content analysis, and sensor networks.



Jhe-Wei Lin received his B.S.Eng. degree in Computer Science and Information Engineering from National University of Tainan, Taiwan, in 2011. He is currently a M.S student in the department of Computer Science and Information Engineering at National University of Tainan, Taiwan. His research focuses on BitTorrent with real-time video service, peer-to-peer media streaming, and multiple description coding over P2P networks.