RESEARCH   **Open Access**

Fast IPTV channel switching using hot-view and personalized channel preload over IEEE 802.16e

Sheng-Tzong Cheng, Chih-Lun Chou, Gwo-Jiun Horng[*] and Tun-Yu Chang

Abstract

The Internet protocol television (IPTV) is emerging as one of the most promising applications over next generation networks. The recently released IEEE 802.16d/e is capable of ensuring high bandwidths and low latency, making it suitable for delivering multimedia services. In addition, it also provides wide area coverage, mobility support, and non-line-of-sight operation. In this article, we deliver IPTV streaming over 802.16 wireless systems and propose a simple but effective IPTV channel-switching algorithm to keep the channel zapping time in a tolerable range. In addition, we discuss how to allocate channels in the limited bandwidth over wireless networks, such as 802.16. The proposed algorithm is based on hot-view channel and personal favorite channel preloading to reduce the network delay and achieve the goal of fast channel switching. Finally, the experimental results show the performance of the proposed algorithm.

Keywords: IPTV, channel-switching, IEEE802.16e, hot-view

Introduction

With the development of network services, transferring materials have evolved from data to multimedia. Recently, the idea of triple-play, which consists of data, audio, and video, is the main issue. The Internet protocol television (IPTV) service is derived from this idea.

IPTV offers service over IP networks, and offers viewers interactive multimedia services such as program voting and advertisements. Overall, IPTV depends on the IP network and whether it can support unicast, multicast, broadcast multimedia, games, VOIP, etc. [1-3].

In the wired IPTV service, the service terminals consist of a set-top box and television in the home. But for mobile IPTV service, the terminals are wireless networks, such as 3G, Wi-Fi, or WiMAX. When comparing wired and mobile IPTV services, the former has sufficient bandwidth to permit system operators to directly broadcast all channels to viewers. Viewers only need to tune antenna frequency for channel changing. On the other hand, for mobile IPTV there is the issue of insufficient bandwidth, forcing system operators to broadcast

partial channels. These issues prompt discussion, such as how to effectively allocate channels to increase the channel's utility rates.

Consider this scenario, a user is watching channel #1 and after a while he wants to change to channel #2. Between pressing the switching button on the remote controller and the time it takes for the monitor to display the program on channel #2, the following events occur: [4-6].

(1) Streaming encoding and decoding
(2) Channel zapping time
(3) Channel switching algorithm
(4) Quality of experience (QoE)

In this article, we put emphasis on the first and second sections, and propose an algorithm to handle the channel switching. Lastly, we propose the method that must meet the QoE for viewers.

Related study

Channel zapping time is one of the import factors for benchmark the QoE. For a viewer, even a 1-second delay can be unbearable. Therefore, how to reduce the channel zapping time becomes the first challenge to overcome.

* Correspondence: grojium@gmail.com
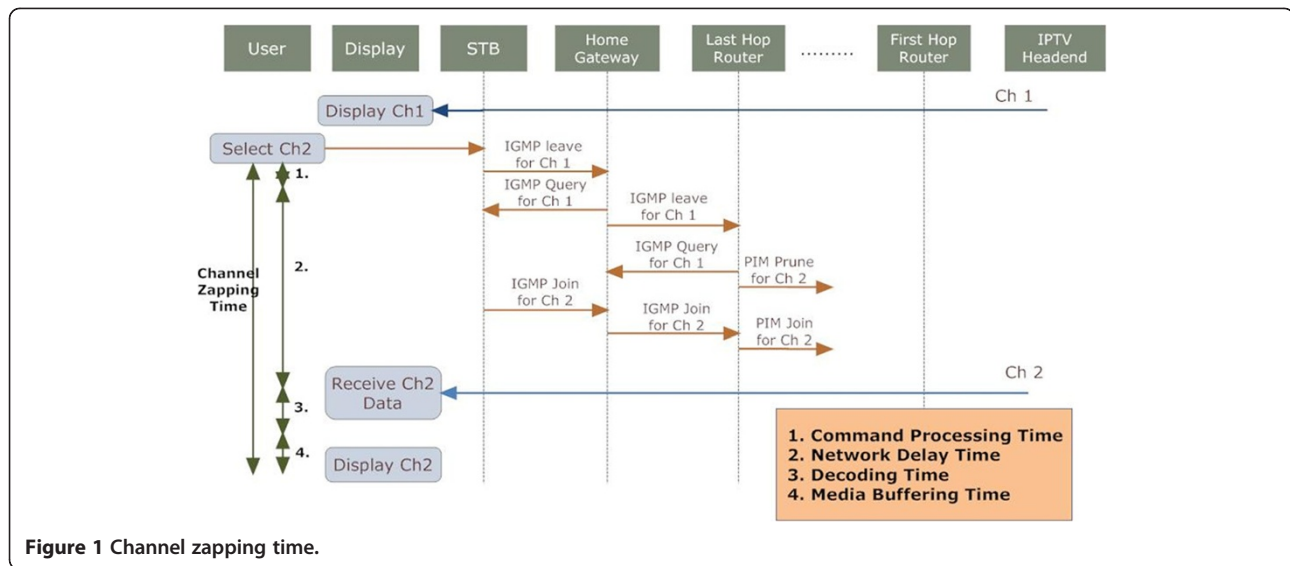Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan

Figure 1 Channel zapping time.

Channel zapping time consists of four major time durations as shown in Figure 1:

1 Command processing time
2 Network delay time
3 Decoding time
4 Media buffering time

The decoding time and media buffering time cannot directly improve the channel zapping time; therefore, we will not discuss these parts in this article [7,8].

For reducing IPTV channel zapping time, the main ideas are IPTV channel zapping pre-load and fast streaming encoding/decoding. Chunglae et al. [9] pre-load channels adjacent to the servicing channel, n. For example, when viewers request channel n, channels $n − 2$, $n − 1$, $n + 1$, $n + 2$ will be preloaded at the same time. While viewers do up/down operations of changing IPTV channels, the predicted channels will be caught and transferred to the viewer immediately. However, if viewers switch channels randomly, the average channel zapping time will increase.

In [10], the same methods are used as in [9]; the authors pre-load adjacent channels of the cable TV network. The difference is that the authors set the commands of channel changing as UGS data flow type. In this way, these commands can be parsed and treated as the highest priority to reduce packet-scheduling delay and furthermore reduce channel-zapping time. Because of base station periodically allocates fixed bandwidth for UGS data flow. If there is not any user change channel in the duration of allocating USG bandwidth, the utility rate of bandwidth will decrease.

Hyunchul et al. [11] consider how to efficiently decrease video decoding delay by adding extra I-frames to normal video frames and the authors concurrently consider both broadcasting channel distribution state and video encoding structure as control variables to effectively guarantee channel zapping time.

The researchers above have the same pre-assumptions that all channels are sequentially arranged, and viewers can only do up/down channel-changing operations. However, these assumptions do not realistically model viewers' channel surfing behaviors. In general, viewers have preferences, such as hot-view channels and favorite channels, and the operations of channel changing are not only up/down operations, but also jump channels. In addition, the above research implements on core networks. Contrary to wireless networks, the bandwidth allocation must be put into consideration. So, the main idea of this article is to make preloads for hot-view channels and personal favorite channels and make suitable bandwidth allocation methods for increasing bandwidth utility rate.

System architecture
As shown in Figure 2, the system architecture consists of four roles: streaming server, WiMAX base station, proxy hard-disk, and mobile SS. The functions of these roles are described below.

1. *Streaming server*: Sends encoded media streaming to base-station or mobile SS.
2. *WiMAX base station*: Schedules bandwidth and allocates channels for channels which the user requests.
3. *Proxy hard-disk*: Stores all streaming media sent from streaming server. While it can offer video on demand and real-time IPTV service at the same time. Besides, the channels that we predicted as hot-view and personalized channels also preload on proxy
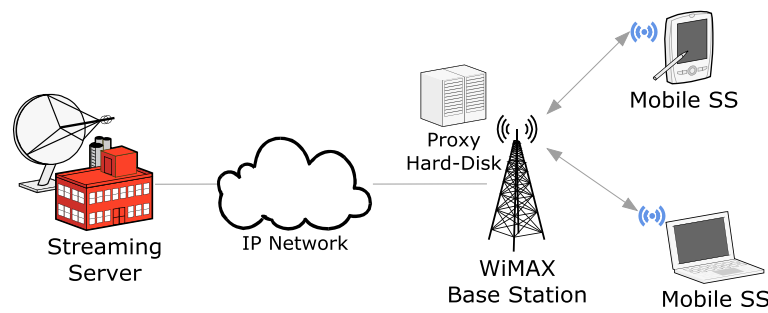
Figure 2 System architecture.

hard-disk. While user request for channel n, base station can get content from nearby proxy hard-disk to reduce IPTV channel zapping time.

4. *Mobile SS*: Sends request for channel n and switches multicast group between channels to fetch streaming.

Because of limited bandwidth, we must allocate bandwidth effectively. We divide the bandwidth into three parts, as shown in Figure 3. For a mobile IPTV program which is encoded with H.264 and 360 × 288 resolutions (CIF quality), video transferring bit rate is 384 Kbps, and audio is 48 Kbps. Including the 40 Kbps of overhead, the total required bandwidth is 500 Kbps.

Assume the base station offers 11 Mbps of total bandwidth. After calculating, we can get 11, 6, 4 channels to multicast at same time for regions I, II, and III, respectively.

Region I is reserved for top N1 viewers count. The base station periodically recalculates the ranking for viewers count. After re-ranked, the new top N1 viewers count channel will be allocated in region I.

Region II is reserved for single cast channels. While user request for channel n, but n is not belonged to N1, N2, and N3. At this moment, the channel n is treated as a newly requested channel. If region II still has bandwidth to be allocated, channel n will be set up; otherwise channel n will be en-queued until there is available bandwidth for serving.
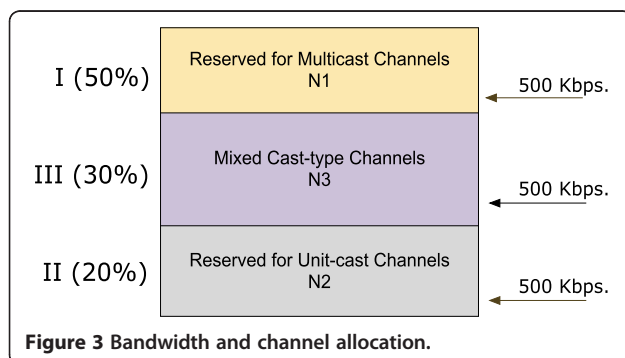
Region III is a mix of those channels associated with regions I and II. While re-calculating, as mentioned before in region I, some channels with decreasing viewers count will be kicked out of region I and be reallocated to region III (if there is available bandwidth in region III); in the same way, while more and more viewers join channels in region II, the viewers count of these channels will increase and perhaps these channels can be promoted to region III. These two cases mentioned above may happen at the same time and compete for bandwidth.
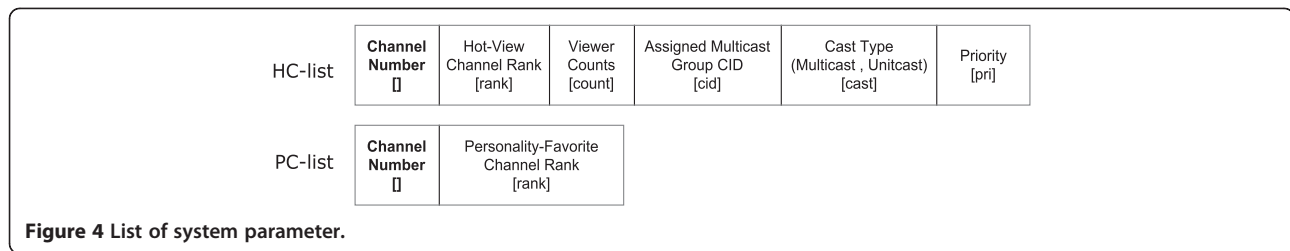
Figure 4 shows the primary parameters for the proposed system, hot-view channel list (HC-list), and personality-favorite channel list (PC-list). HC-list contains all the channels and is sorted by viewer counts. The viewer configures their own PC-list, and sends the list to the base station for preload (if bandwidth available) or scheduling.

HC-list is maintained by the base station, and the base station resorts the list by accumulating count of viewers. In addition, in every time unit (perhaps 30 or 60 min, usually a TV program playing duration), HC-list records rankings for every channel. According to the ranking on the HC-list, we can preload hot-view channels to proxy-hard in advance. Besides, viewer can send his PC-list while request channels to base station. The base station will reserve streaming in proxy hard-disk. So far, with the preloading in proxy hard-disk according to HC-list and PC-list, we can reduce IPTV channel zapping time and keep bandwidth allocation effectively. The other issue of how to allocate bandwidth will be discussed in the following section.

Bandwidth allocation algorithm
Algorithm 1—channel scheduler for region I
Algorithm 1 is routine operations. Base station periodically executes the function to decide which channels can be allocated to region I. In line 3, base station stores the old ranking list, and then resorts the list. After resorting, some channels will be replaced in region I and turn to region III, if there is bandwidth available in region III. If

Figure 3 Bandwidth and channel allocation.

HC-list	Channel Number []	Hot-View Channel Rank [rank]	Viewer Counts [count]	Assigned Multicast Group CID [cid]	Cast Type (Multicast , Unitcast) [cast]	Priority [pri]
PC-list	Channel Number []	Personality-Favorite Channel Rank [rank]				

Figure 4 List of system parameter.

bandwidth is unavailable, these channels turn to region II (line 12). The worst case is that there is still no available bandwidth in region II. If this is the case, then remove the single-cast channel in region II to service queue (line 15) or TT86d47313move the lowest viewer counts channel to service queue.

```
 1  function schedule_region_I()
 2  Begin
 3  A ← set of n in Region I
 4  sort HC-list order by HC-list[].count ASC
 5  B ← set of n in HC[] (1~N₁)
 6  Allocate bandwidth for channels in B in Region I
 7  C ← A − B
 8  for each n in C
 9  if (bandwidth_available(Region III))
10  allocate bandwidth for n to Region III
11  Else
12  if (bandwidth_available(Region II))
13  allocate bandwidth for n to Region II
14  Else
15  if (exist i in Region II and i is unicast)
16  en-queue i for service while bandwidth available
17  allocate bandwidth for n to Region II
18  Else
19  if (HC-list[n].count < j with min viewer in Region II)
20  en-queue n for service while bandwidth available
21  Else
22  en-queue j for service while bandwidth available
23  end
```

Algorithm 2—channel scheduler for region II

Algorithm 2 focuses on re-allocating bandwidth to region II. While there is available bandwidth and there are still some channels waiting service in service queue, these services will be de-queue and allocate bandwidth for service.

```
1 function schedule_region_II()
2 Begin
3 if (bandwidth_available(Region II) and queue is not
   empty)
4 de-queue n for service
5 allocate bandwidth for n in Region II
6 assign_channel_to_ss(n, ssid)
7 end
```

Algorithm 3—viewer makes a new request n

While a viewer make a new request for channel n, as described in Algorithm 3. First, check if channel n is in channel set N1, N2, or N3. If n belongs to N1, the viewer just joins that group for channel n; if n belongs to N2, channel n may transfer from region II to region I or III. If n belongs to N3, it means that channel n is on the air (similar to N1), the viewer just joins that group for channel n. If channel n does not belong to channel sets N1, N2, or N3 that means channel n is a new request, and base station checks if region II has sufficient bandwidth, if available, create a new group, else en-queue for waiting service.

```
 1  functionjoin_channel (n, ssid)
 2  Begin
 3  if (n ∈ N₁)
 4  assign_channel_to_ss(n, ssid)
 5  else if (n ∉ N₁ and n ∈ N₂)
 6  assign_channel_to_ss (n, ssid)
 7  if (bandwidth_available(Region III))
 8  assign n to N₃
 9  else if (n ∉ N₁ and n ∉ N₂ and n ∈ N₃)
10  assign_channel_to_ss(n, ssid)
11  else // n ∉ N₁ and n ∉ N₂ and n ∉ N₃
12  if (bandwidth_avialable(Region II))
13  allocate bandwidth to n on Region II
14  Else
15  en-queue n for service while bandwidth available
16  end
```

Algorithm 4—assign channel and CID to viewer

Algorithm 4 presents that all actions for a viewer request a channel n and join to multicast group. Line 3 means that accumulation for viewer counts. Higher viewer counts means the channel has a higher ranking and is more popular. In line 4, "CID" represents connection identifier. This is identified for WiMAX to multicast messages.

```
1 finction assign_channel_to_ss (n, ssid)
2 Begin
3 HC-list[n] ← HC-list[n].count + 1
4 ssid.cid ← HC-list[n].cid
5 if (HC-list [n].type = "unicast")
```

time \ channel	1 TVBS-N	2 CNN	3 SETN	4 Discovery	....	98 HBO	99 AXN	100 CINEMAX
08:00 ~ 09:00	23	25	1	4	......	44	22	6
09:00 ~ 10:00	21	2	23	71	......	24	1	29
10:00 ~ 11:00	19	1	4	9	......	19	29	33
~	~	~	~	~	~	~	~	~
21:00 ~ 22:00	22	33	14	29	......	4	1	39
22:00 ~ 23:00	3	4	29	20	......	55	42	87
23:00 ~ 24:00	3	4	99	23	......	55	66	24

Figure 5 Program guide and ranking in each time duration.

6 HC-list[n].type ← "multicast"
7 End

Algorithm 5—viewer make a leave request for channel n
Algorithm 5 shows what happens when the viewer wants to quit channel n. First, the viewer counts will decrease 1, and check if there is still any viewer in this multicast group. If the view count is greater than 2, there are no operations; if the view counts equals 1, this means that the channel has switched from a multicast channel to a unicast channel. Now let discuss three different cases, n belonging to N1, N2, or N3. If channel n belongs to N1, then check if there is bandwidth available in region II. If there is bandwidth available, move channel n to region II; otherwise en-queue channel n in service queue. When n belongs to N2, there is no operation. Lastly, when n belongs to N3, re-schedule for regions I and II.

Line 15 in Algorithm 5, while there is no viewers in the group, the group will be broken up. There are three cases to be discussed. If channel n belongs to N1, execute Algorithm 5 to do bandwidth rescheduling. If n belongs to N2, check service queue and de-queue requests to serve. If n belongs to N3, then execute Algorithms 1 and 2 to re-schedule bandwidth. The common operations after three cases checking region II for de-queue service to serve.

1 function leave_channel (n, ssid)
2 Begin
3 HC-list [n].count←HC-list [n],count − 1
4 if (HC-list [n].count=1)
5 if ($n \in N_1$)
6 if (bandwidth_available(Region II))
7 assign n to N_2
8 Else
9 en-queue n for service while bandwidth available
10 else if ($n \notin N_1$ and $n \in N_2$)
11 no operation
12 else // $n \notin N_1$ and $n \notin N_2$ and $n \in N_3$
13 schedule_region_I()
14 schedule_region_II()
15 else if (HC-list[n].count = 0)
16 HC-list[n].cast = ""
17 HC-list[n].cid = ""

18 release bandwidth for channel n
19 if ($n \in N_1$)
20 schedule_region_I()
21 else // $n \notin N_1$ and $n \in N_2$
1 22.NOP
22 else // $n \notin N_1$ and $n \notin N_2$ and $n \in N_3$
23 schedule_region_I()
24 schedule_region_II()
25 if (bandwidth_available(Region II))
26 de-queue n for service while bandwidth available
27 End

Programming guide with ranking
In IPTV, the design of the program guide is very important. Program guide can be in form of 1-day duration. If we set time duration as the rows, and channel lists as the columns, we get a full table as ranking list over 1 day, as shown in Figure 5. In this article, the ranking means that at specific time duration, the list ranks for all channels. For example, if the system time is 10:15 now, the ranking for channel 3 is 4th, and channel 4 is 9th.

As a result, the system ranking is changing through time.

Experiments
Simulation environment
We adopt the WiMAX module for NS-2 simulator to simulate the proposed solution [12]. We do several experiments on NS-2 simulator. The environment parameters we used are shown in Table 1. The simulation time is 60 min.

It is not possible to get the data from every user's watching habits. Therefore, we adopt IBM Synthetic

Table 1 Environment parameters

Parameters	Value
Bandwidth of the BS	11 Mbps
Number of IPTV channels	50 channels
Number of hot-view channel predict	10 channels
Number of personality-favorite channel predict	4 channels
N1	11
N2	5
N3	4
Simulation time	60 min

```
 1   #start(ms)  end(ms)  duration(ms)  channel
 2   1    15001    15000     12
 3   15002    18002    3000     13
 4   18003    154003   136000   21
 5   154004   156004   2000     24
 6   156005   257005   101000   25
 7   257006   347006   90000    28
 8   347007   364007   17000    34
 9   364008   468008   104000   39
10   468009   509009   41000    43
11   509010   511010   2000     48
12   511011   516011   5000     11
13   516012   521012   5000     12
14   521013   583013   62000    21
15   583014   659014   76000    25
16   659015   663015   4000     38
```

Figure 6 IBM synthetic data generator results.

Data Generator to generate user's watching habits data. The generated data are show in Figure 6.

Channel preload accurate rate

Accurate channel preloading rate has a direct impact on IPTV channel switch time. We will use our method to compare adjacent channel preload method [9].

Figure 7 shows that our channel preload method is better than the adjacent channel preload method. As number of viewers increases, our channel preload method has a more accurate trend. Because of more user activity can increase predict accurate rate.

Channel switch delay time

Channel switch delay time is one of the most important factors concerning QoE. Therefore, it is essential to keep the delay time as short as possible.

Channel switch time has a direct correlation to channel preload predict rate. Because of better channel preload predict rate will decrease network transmit time (channel switch delay time). From Figure 8, we can see our method has a shorter delay time.

Jitter

Jitter is another important factor concerning QoE. If networks have a bad condition, then users will experience a worse video fragment. Because of a lot of packets losses or heavy delay. From Figure 9, we can see our method has a better results.

Conclusion

In this article, we propose a bandwidth allocation algorithm to allocate bandwidth for wireless networks to solve the problem of insufficient bandwidth. In addition, multicasting by preloading hot-view and personality-favorite channel in proxy hard-disk can help reduce IPTV channel zapping time.
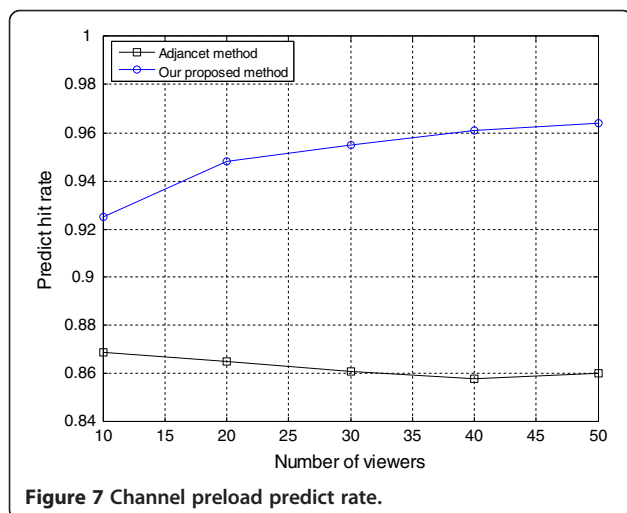

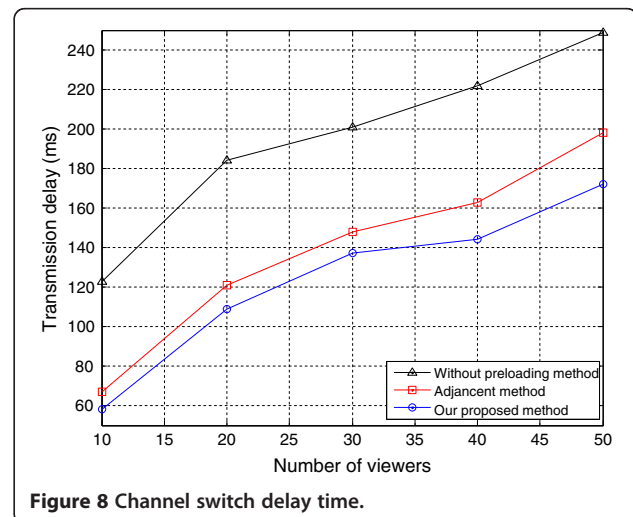
Figure 7 Channel preload predict rate.

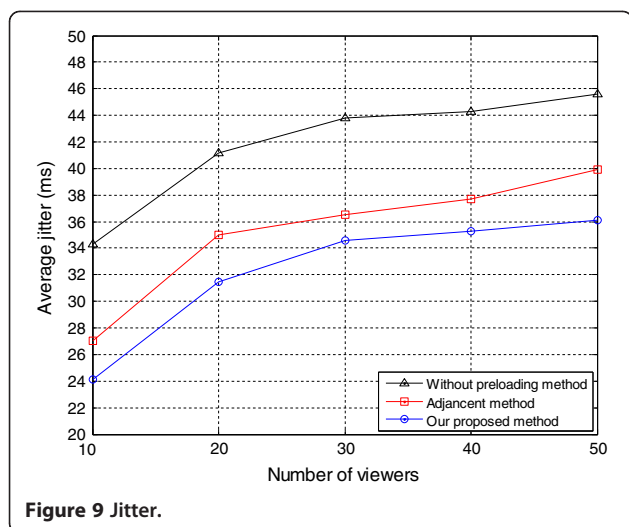**Figure 8 Channel switch delay time.**

Figure 9 Jitter.

In future work, we will try to offer different data rate streaming for three diverse service regions. In addition, we can support different QoS support for allocating bandwidth for VIP viewers and normal viewers.

Competing interests
The authors declare that they have no competing interests.

References
1. ITU-T (International Telecommunication Union Telecommunication Standardization Sector). http://www.itu.int/net/ITU-T/info/Default.aspx
2. IV Uilecan, Z Chi, GE Atkin, *Framework for delivering IPTV services over WiMAX wireless networks* (IEEE International Conference on Electro/Information Technology, Chicago IL, U.S.A, 2007), pp. 470–475
3. Open IPTV Forum, http://www.openiptvforum.org/.
4. Z Marcio Nieblas, B Graca, *A proposed approach for quality of experience assurance of IPTV* (First International Conference on Digital Society ICDS '07, Guadeloupe, French Caribbean, 2007), pp. 25–25
5. J Kishigami, *The role of QoE on IPTV services style* (Ninth IEEE International Symposium on Multimedia, Taichung, Taiwan, 2007), pp. 11–13. ISM
6. Air interface for fixed broadband wireless access systems. IEEE Standard 802.16 (2004)
7. IEEE Std 802.16e-2005, Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access System. (2006)
8. L Kyu Seol, R Sang Won, Y Hee Yong, A MAC layer multicasting approach for WiMAX access networks",2008 Sixth Annual IEEE International Conference on Pervasive Computing and Communications., 348–353.
9. C Chunglae, H Intak, J Yongil, L Hyeongho, Improvement of channel zapping time in IPTV services using the adjacent groups join-leave method. The 6th International Conference on Advanced Communication Technology **2**, 971–975 (2004)
10. L Dai-boong, J Hyunchul, S Hwangjun, An effective channel control algorithm for integrated IPTV services over DOCSIS CATV networks. IEEE Trans. Broadcast. **53**(4), 789–796 (2007)
11. J Hyunchul, S Hwangjun, L Dai-Boong, L Inkyu, An effective IPTV channel control algorithm considering channel zapping time and network utilization. IEEE Trans. Broadcast. **54**(2), 208–216 (2008)
12. NIST, IEEE 802.16 Module for NS-2. http://www.antd.nist.gov/seamlessandsecure/download.html