

RESEARCH

Open Access

The multicast system for seamless live multimedia in WLAN

Yoseop Woo and Iksoo Kim*

Abstract

This paper proposes the multicast system for seamless live multimedia in a wireless network. The multicast technique for live multimedia and a part of handover mechanism are performed on the buffered multicasting/switching agent (BMSA) in a wireless local area network (WLAN). The rest of the handover mechanism is performed on access points (APs). This paper uses a new Internet Protocol (IP) header that is composed of three AP numbers (AP_#s) and 1-bit overwrite bit (OB) to perform seamless service. The three AP_#s support a smooth handover for solving the cutoff service when mobile nodes (MNs) move among other APs in WLAN. The overhead for a new IP header (2 bytes long) is negligible in comparison with multimedia service, and the overhead for the multicast group address (MGA) packet that is used in the transmission of MGA is only 11 bits. The MGA packet is composed of two AP_#s, an overwriting group address (OGA) bit, and MGA. We confirmed that the proposed multicast system can decrease the number of channels required, the delay of service, and the possibility of packet duplication because multicast is not provided on the multimedia server but on the BMSA in WLAN.

Keywords: Multicast; Handover; Multimedia; VOD; WLAN

1. Introduction

The multimedia service on the Internet has many problems including insufficient network bandwidth in a wire/wireless network and an excessive load of the server. Especially, these problems are critical in a wireless network. Many technologies such as multicast, web caching, and content distributive network (CDN) have been studied to solve these problems, but they have their own problems too [1-3].

Nowadays, a smooth multimedia service is one of the most important services in a wire/wireless network. Providing multimedia service in a wireless local area network (WLAN) has many problems including cutoff service and reconnection to the server caused by the movement of mobile nodes (MNs) such as tablet PCs (iPad and Galaxy tab), laptop computers, and personal digital assistants (PDAs) [4,5]. Even though the MNs are moving among access points (APs) in WLAN, seamless multimedia services have to be achieved.

This paper presents a seamless multimedia system that can reduce the number of service channels to overcome

deficient wireless network bandwidth, alleviate the load of the server, and prevent reconnection to the server and cutoff service according to the MNs' movement. For solving these problems, this paper adopts the multicast technique to reduce the number of channels of a server and the service delay. It also adopts handover mechanism using AP identification numbers (AP_#s) as the Internet Protocol (IP) header to prevent cutoff service and reconnection caused by the MNs' movement [6,7].

This paper uses the buffered multicasting/switching agent (BMSA) to perform multicast in WLAN. The BMSA and AP perform handover mechanism using a 2-byte-long IP header that is made up of three AP_#s and overwrite bit. Because the multicast technique is not provided at the server on the Internet but on the BMSA in WLAN, the delay of multicasting service and the duplication of multicast packet can be reduced.

In the case of multicast, we confine our interest to live multimedia such as news, public TV programs, and sports events except on-demand services, but in the case of unicasting, we study all the multimedia programs whether they are live or not.

The rest of this paper is as follows: Section 2 describes the structure and operation of a seamless live multimedia

* Correspondence: iskim@incheon.ac.kr

Department of Information and Telecommunication Engineering, Incheon National University, 12-1 Songdo-dong, Yeonsu-ku, Incheon 406-772, South Korea

system and explains the proposed multicast and handover mechanism using BMSA applied for this system in WLAN. Sections 3 and 4 deal with the algorithm and the results of the simulation for the proposed system, respectively. Finally, we discuss our conclusions in Section 5.

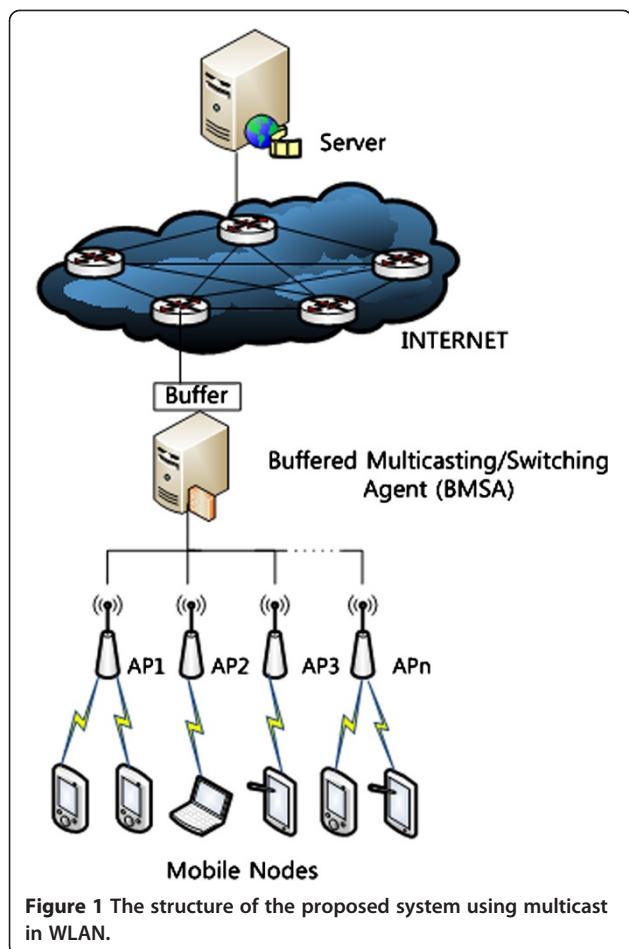
2. The structure and operation of seamless live multimedia system in WLAN

In this paper, the multicast system for seamless live multimedia service is made up of the multimedia server, a BMSA, a number of APs, and mobile devices (or mobile nodes) such as tablet PCs including iPad and Galaxy tab, PDAs, and laptop computers. Figure 1 shows the structure of this system [7,8]. The multicast technique for reducing the number of service channels and delay of service is performed by the BMSA in WLAN. Whenever a number of MNs request the same live multimedia item, the BMSA groups a single multicast group (MG) for them and then it generates a multicast group address (MGA; class D IP address) and sends the MGA packet to the MNs requesting the same item through the corresponding AP. Because the multicast technique is performed by the BMSA within WLAN and not by the multimedia server on the Internet, the multicast

service delay and the possibility of packet duplication can be decreased.

Furthermore, along with AP, the BMSA provides a smooth handover mechanism that prevents reconnection to the server caused by cutoff service when MN moves among APs. Therefore, the BMSA has to possess a MG table for multicast and an *AP:MNs* mapping table to indicate that MNs are on the coverage of a certain AP. The BMSA can switch the live multimedia streams transmitted from the server to the corresponding APs using the MG table and *AP:MNs* mapping table.

The AP assigns an IP address to MNs through DHCP generally. MNs are reassigned a new one whenever they travelling among APs. Thus, MNs have to reconnect to the server to sustain service when they are connected to other APs. In this paper, however, the AP assigns 2-byte-long *AP_#* fields as an IP header and IP address only if MN enters WLAN for the first time. After the AP assigns them to MN, the MN receives only an *AP_#* except IP address when the MN moves to other APs. Also, only 11 bits is added to the MGA packet in comparison to the traditional multicast. Thus, our proposed mechanism performs a seamless live multimedia service using the MGA packet and the *AP_#*s added as an IP header [9].



2.1. The structure of IP header

The proposed system uses three *AP_#* fields (2 bytes long) as a new IP header in addition to the IP address. Only the AP that was accessed for the first time when MNs enter into the WLAN can assign an IP address with an *AP_#* field. The other APs except the first accessed AP never assign an IP address to the MNs when they are moving among other APs. In this case, the APs generate their own *AP_#* only within *AP_#* fields. The IP address and *AP_#* assigned for the first time remain unchanged even though the MN travels among APs except the case that MN receives a MGA packet from the BMSA [5,6]. Thus, the two *AP_#*s of three *AP_#* fields are changed according to the MN's movement among APs. As mentioned earlier, we confine our interest for multicast to live multimedia programs and not on-demand ones. A study on using the proposed multicast system for all multimedia including on-demand services such as VOD, MOD, and NOD is in progress.

The first row of Figure 2a shows the format of the 2-byte-long IP header that is made up of three *AP_#* fields (initial state (15 bits): all 1's) and a 1-bit overwrite bit (OB). The second row of Figure 2a indicates the IP header with the IP address that is assigned by *AP_#12* accessed for the first time when the arbitrary MN (*MNi*) entered into WLAN.

The third row of Figure 2a shows that the *MNi* moved to *AP_#23* from *AP_#12*, and the first *AP_#12* and IP address (*IP_#32*) never changed according to the *MNi*'s

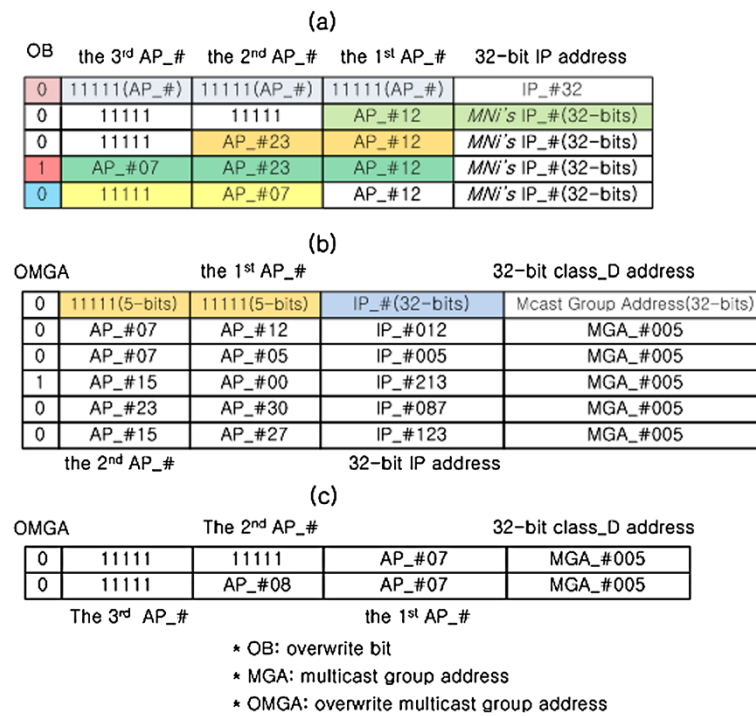


Figure 2 The structure of IP header packet and MGA packet of MSA. (a) A new IP header packet. (b) MGA packet. (c) IP header packet with MGA.

movement. Then the fourth row of Figure 2a shows that the MNi moved to AP_#23 and AP_#07 from AP_#12 sequentially. AP_#23 and AP_#07 assign only their own AP_# to the MNi, and they do not reassign an IP address even if the MNi enters within their coverage areas. At the moment, AP_#07 sends an IP header with OB = 1 to the BMSA and MNi. The BMSA changes AP:MNs' mapping table to indicate that MNi moved AP_#23 to AP_#07. Then the BMSA switches smoothly the transmission path for successive multimedia streams to AP_#07 for the MNi. As soon as the MNi receives an IP header from AP_#07, it overwrites the second field AP_#23 to AP_#07 and sets to 1 the third AP_# field. The MNi clears OB to 0 (the fifth row of Figure 2a) at the same time. Therefore, Figure 2a explains that multimedia streams requested can be switched smoothly from AP_#12 to AP_#23 and AP_#07 sequentially. In the fifth row of Figure 2a, the third AP_# field (11111) can be reused when the MNi moves AP_#07 to other APs. As shown in Figure 2a, even though the MNi accesses AP_#23 and AP_#07, AP_#12 and IP_#32 assigned by AP_#12 for the first time are never changed.

The 5-bit-long AP_# can manage 31 APs (AP_#00 to AP_#30) in WLAN, and it will be expanded if the number of APs is increased. The length of the three AP_# fields (2 bytes) is negligible in comparison with multimedia streams.

Figure 2b shows the MGA packet that transmits MGA to a number of MNs when they request the same live

multimedia. The MGA packet is composed of two AP_# fields, an IP_# field, multicast group address, and overwriting multicast group address (OMGA). The BMSA sends the MGA packet to the MNs through the corresponding APs using AP:MNs' mapping table.

Figure 2b indicates that five MNs are receiving the same MGA (MGA_#005) from the BMSA: two MNs (AP_#12:IP_#012, AP_#05:IP_#005) within the coverage area of AP_#07, another two MNs (AP_#00:IP_#213, AP_#27:IP_#123) within that of AP_#15, and a MN (AP_#30:IP_#087) within that of AP_#23. Thus, Figure 2b shows that five MNs within the range of three different APs (AP_#07, AP_#15, and AP_#23) are grouped with MGA_#005. The length of the MGA packet is 75 bits. Even though it has 75 bits, the added bits in this paper are only 11 bits (two AP_# fields and 1 bit for OMGA). The remaining parts of the MGA packet are the IP address and MGA, and those are the same as the traditional multicast [10]. The OMGA notifies that MNs received the MGA packet and overwrites MGA into the IP address for joining the multicast group.

The parts of Figure 2a, b are used as an AP:MNs mapping table and MG table on the BMSA, respectively. The former uses the third and the first AP_# and IP address of the IP header packet, and the latter uses the second AP_# and MGA fields of the MGA packet. The OMGA = 1 in Figure 2b notifies MNs to join with MGA_#005.

Figure 2c shows the IP header packet with MGA. This packet is used for joining a specific MG. The first row of

Figure 2c shows that a MN within the range of AP_#07 joins MGA_#005. Therefore, it can receive multicast live multimedia streams transmitted with MGA_#005. The second row of Figure 2c notifies the BMSA that more than one MNs serviced multicast within AP_#07 are moved to the coverage of AP_#08. The BMSA knows that it has to send the successive live multimedia streams to AP_#08 because a part of MG in AP_#07 already moved to AP_#08. Also, the BMSA sends multimedia streams (MGA_#005) to AP_#07 because other MNs that joined MGA_#005 may reside within AP_#07. However, AP_#07 has to notify the BMSA to stop sending successive multimedia streams with MGA_#005 when MNs do not join the group (MGA_#005 in AP_#07) anymore.

As mentioned above, the assigned AP_# and IP address by AP accessed for the first time are never changed. Also, other APs do not reassign a new IP address to MNs and assign only AP_# that indicates the MNs' movement. The BMSA can switch the successive multimedia streams from old AP to current AP because the BMSA manages the MNs' movement using its AP:MNs' mapping table. Therefore, the proposed mechanism supports a seamless live multimedia service without reconnection to the server and cutoff caused by the MNs' movement among APs.

The multicast technique is initiated on the BMSA when some MNs controlled by the same AP or different APs request the same live multimedia. The BMSA checks whether the live multimedia requested is already servicing. Then the BMSA generates MGA as soon as it receives the same request from other MNs when the requested live multimedia is servicing, and it sends MGA to the MNs requesting service. Next, the BMSA transmits live multimedia streams with MGA to the corresponding APs, and the APs transmit these streams to the multicast group. In traditional papers, the multimedia server makes multicast groups and is in charge of multicast transmission. However, the proposed mechanism performs multicast on the BMSA in WLAN. Thus, the necessary time for multicast transmission is shorter than that for the traditional one. Also, there is scarce data duplication, and the load of all routers on the Internet does not change because the multicast mechanism is performed in WLAN. The BMSA is the core of the proposed mechanism because multicast and a part of handover techniques are performed by the BMSA.

3. The algorithm for seamless live multimedia service in WLAN

As mentioned in Section 2, the proposed multicast and handover mechanism are simple and effective techniques for live multimedia service in WLAN. The proposed technique can reduce the service delay without cutoff and reconnection according to the MNs' movement among APs and the number of service channels. In

Section 3, to provide a seamless live multimedia service using multicast and smooth handover in WLAN, we present the algorithm written as a pseudo-code as follows:

Algorithm for seamless live multimedia service using multicast

```
class MN {
    SeamlessLiveMultimediaService() {
        ...
        if (FirstTimeAccess(MN#, AP#)) {
            addedIP = AP#.assignIP(MN#); // addedIP => AP#start:
            AP#start:IP#MN
            AP#.requestItem(addedIP, requestItem, MN#);
        } else { // moved AP while stream receiving
            addedIP = AP#.assignMovedIP(MN#); // addedIP =>
            AP#moved:AP#start:IP#MN
        }
    }
}

receiveStream(liveStream) {
    receive live stream through unicast or multicast IP
}

receiveMGA(liveStream) {
    receive new multicast group address
}
...
} // end of class MN

class BMSA {
    mappingTable mt; // (AP#moved, AP#start, IP#MN,
    Video#)
    requestLiveMultimediaItem(addedIP, requestItem,
    MN#) {
        ...
    }
}
```

```

recordMappingTable(addedIP, requestItem);
if (isServicingNow(requestItem)) {
    servicingStream = server.request(requestItem);
    multicast(addedIP, servicingStream);
} else {
    liveStream = server.request(requestItem);
    // server returns live multimedia stream
    sendStreamToAP(liveStream, AP#);
}
...
} // end of requestLiveMultimediaItem()

sendStreamToAP(liveStream, AP#) {
    currentAP = AP#;
    while (liveStream is not finish) {
        try { // handover(MN#) execution if AP accessed is changed
            during streams transmission
            currentAP.receiveStream(liveStream);
        } catch (moveMNNanotherAPEException(MN#)) {
            currentAP = handover(MN#); // current AP is changed
            through handover()
        }
    }
}

multicast(addedIP, servicingStream) {
    multicastGroupAddr
    = createMulticastGroupAndGenerateGroupAddress
    (addedIP, requestItem);
    AP.receiveMGA(multicastGroupAddr);
    sendStreamToAP(servicingStream, AP#);
}

handover(MN#) {
    return moved new AP# that find from mapping table
}

movedMN(addedIP) {
    recordMappingTable(addedIP, requestItem);
    // (AP#moved, AP#start, IP#MN, Video#)
    make and throw moveMNNanotherAPEException(MN)
}
...
} // end of class BMSA

class AP {
    receiveStream(liveStream) {
        MN.receiveStream(liveStream);
    }

    receiveMGA(multicastGroupAddr) {
        send multicast group address to all related MNs
    }

    assignIP(MN#) {
        return concatenated string "AP#start:AP#start:IP#MN"
    }
}

```



```

}

assignMovedIP(MN#) {

    addedIPMoved <= new concatenated string

    “AP#moved:AP#start:IP#MN”

    BMSA.movedMN(addedIPMoved);

    return addedIPMoved;

}

requestItem(addedIP, requestItem, MN#) {

    BMSA.requestLiveMultimediaItem(addedIP,

    requestItem, MN#);

}

...

} // end of class AP

```

4. The simulation results of the proposed system

This paper uses a BMSA, 10 APs, 100 MNs, and 100 multimedia contents as the environment of the simulation for the proposed system. We construct the simulation environment as shown in Table 1 to analyze the performance of the network model proposed in this paper. The nominal request rate (λ) is 40 per minute, and the duration of service for live multimedia is 10 min.

The service request follows a nonuniform Poisson distribution. The popularity of live multimedia in this system follows Zipf's distribution as in Equation 1. We let the popularity of the video contents be numbered 1 to 100 based on Zipf's law [11]. The numbered $i = 1$ is the most popular live multimedia item, and the numbered $i = 100$ is the least one.

$$P_N(i) = \Omega / i^\alpha, \quad (1)$$

where $\Omega = \left(\sum_{i=1}^N \frac{1}{i^\alpha} \right)^{-1}$. N is the total number of live multimedia serviced in the system. The popularity of live multimedia serviced is represented in i ($i = 1, 2, \dots, N$). The probability $P_N(i)$ is the conditional requesting probability for each live multimedia item. The α in Equation 1 is a skew factor that indicates the degree of concentration

Table 1 Simulation parameters

Parameter	Default	Range
Number of contents	100	50 ~ 300
Multimedia running time (min)	10	Not applicable
Request rate (request/min)	40	10 ~ 50
Number of BMSA	1	Not applicable
Number of AP	10	5 ~ 20

for the MNs' request for live multimedia. We set the skew factor α to 0.75 ($0 < \alpha < 1$) in order to simulate as closely as possible the real environment. The simulation was executed in Java programming.

Figure 3 compares the number of channels that used multicast and unicast for live multimedia service according to the service request rate ($\lambda = 5$ to 50). At the same time, the simulation uses the rates of multicast request which are 10%, 30%, and 50% of the service requests for 100 live multimedia contents. The number of necessary channels using the proposed mechanism according to the multicast request ratios 50%, 30%, and 10% when the service request rate (λ) is 40 requests/min is decreased to 44%, 28%, and 12%, respectively.

Figure 4 shows the total delay times of the proposed multicast service mechanism and the conventional one according to the service request rates ($\lambda = 5$ to 50). It shows that the total delay times of the proposed system decreases more when the service request rates increase. When the service request rate (λ) is 40 requests/min, the proposed mechanism decreased the service delay hops to 46%, 34.6%, and 13.5% according to the multicast request ratios 50%, 30%, and 10%, respectively. The total delay time using the conventional method is the same as that using unicast. The reason is that the conventional handover mechanism has to reconnect to the server on the Internet whenever the MNs move among APs. Also, the structure of the MGA packet and AP:MN mapping table is simpler than references [8,9].

5. Conclusions

This paper presents a combined mechanism that reduces the load of the multimedia server, uses efficiently insufficient network bandwidth, and prevents cutoff service and reconnection to the multimedia server caused by the MNs' movement among APs. The combined technology is made up of multicast for grouping the same live multimedia requests and smooth handover mechanism using AP_#s. The multicast and smooth handover are performed by the BMSA that is the core of the proposed system in WLAN. Therefore, this paper has no burden on the Internet for multicast. The smooth handover is performed using three AP_# fields (2 bytes long). The overhead for the combined mechanism is negligible

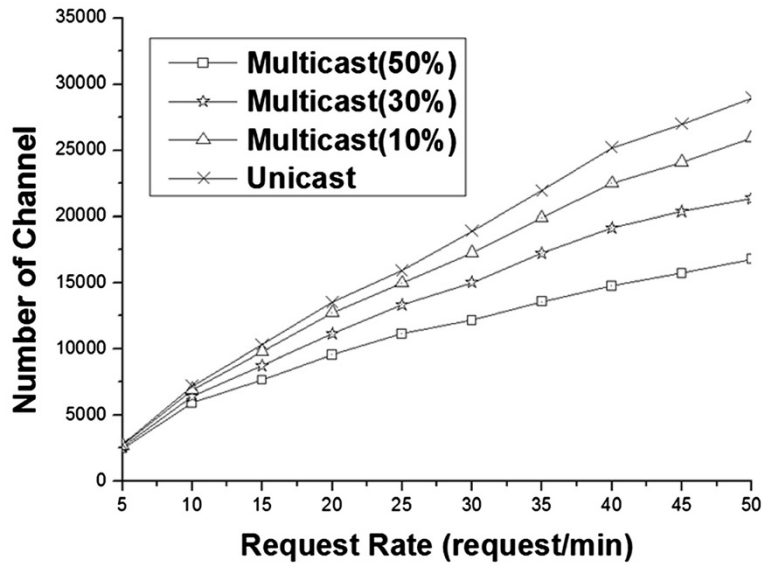


Figure 3 Comparison of the number of channels in unicast and multicast.

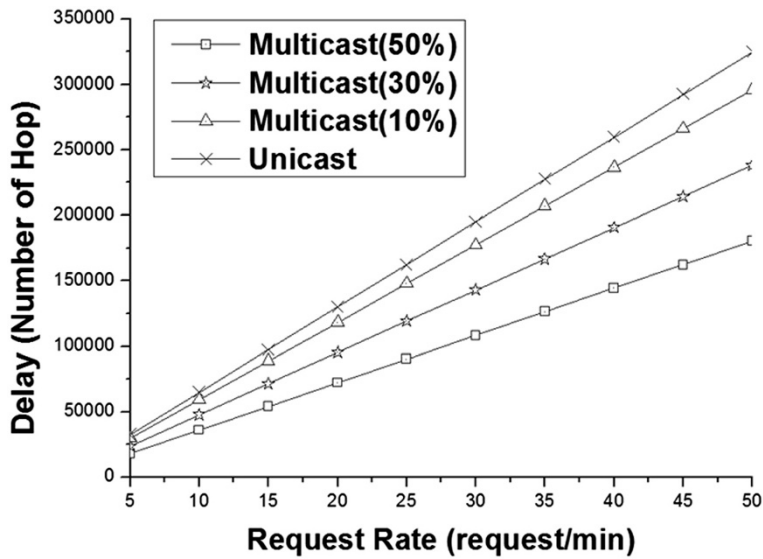


Figure 4 Comparison of service delay in the proposed system and traditional one.

in comparison with multimedia. The performance for the proposed mechanism enhanced about 10% to 46% for the number of channels required and about 7% to 47% for the delay according to the rates of multicast request and service request. This paper confirmed that the format of the MGA packet and AP:MN mapping table is simpler than references.

Competing interests

The authors declare that they have no competing interests.

Received: 22 March 2013 Accepted: 4 October 2013

Published: 17 October 2013

References

1. Y Cai, W Tavanapong, KA Hua, A double patching technique for efficient bandwidth sharing in video-on-demand systems. *J. Multi. Appl. T* **32**(1), 115–136 (2007)
2. D Katsaros, Y Manolopoulos, Web caching in broadcast mobile wireless environment. *IEEE. Internet. Comput* **8**(3), 37–44 (2004)
3. L Zhan-Sheng, L Da-Wei, B Hui-Juan, CRFP: a novel adaptive replacement policy combined the LRU and LFU policies, in *Proceedings of the IEEE 8th International Conference on Computer and Information Technology Workshops, 2008 (CIT Workshops 2008, Sydney, 8–11 July 2008)*, pp. 72–79
4. W Puangkor, P Pongpaibool, A survey of techniques for reducing handover latency and packet loss in mobile IPv6. (2006). IM2006306
5. H Gomaa, G Messier, B Davies, C Williamson, Media caching support for mobile transit users, in *Proceedings of the IEEE WiMob 2009 (Marrakech)*, pp. 79–84. 24–26 Sept
6. D Sarddar, P Mani, U Biswas, M Naskar, Fast handoff mechanism in wireless local area networks (WLAN) using neighbor graph algorithm. *Int. J. Comput. Appl. T* **25**(9), 36–40 (2011)
7. L Seoungyeol, K Iksoo, Multimedia service using equi-loaded cache in wireless network. *J. Korean Ins. Inf. Tech* **10**(3), 83–90 (2012)
8. L Seoungyeol, K Iksoo, W Yoseop, A seamless multimedia service in wireless network, in *Future Information Technology, Application, and Service. FutureTech 2012*, in *Proceedings of the 4th International Workshop on IT Service and Cloud Computing (ISCC 2012). Lecture Notes in Electrical Engineering*, vol. 164, ed. by JJ(JH) Park, VCM Leung, C-L Wang, T Shon, vol. 2 (Springer, Dordrecht, 2012), pp. 477–483. ISBN 2
9. S Lee, I Kim, Y Woo, Live-video service using multicast in wireless network. *Int. J. Multi. Ubiquitous. Eng* **7**(4), 17–22 (2012)
10. M Daniel, *IP Multicast with Applications to IPTV and Mobile DVB-H* (Wiley, New York, 2008)
11. L Breslau, P Cao, G Phillips, S Shenker, Web caching and Zipf-like distributions: evidence and implications, in *Proceedings of the IEEE INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 1 (IEEE, Piscataway, 1999), pp. 126–134

doi:10.1186/1687-1499-2013-246

Cite this article as: Woo and Kim: The multicast system for seamless live multimedia in WLAN. *EURASIP Journal on Wireless Communications and Networking* 2013 **2013**:246.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com