www.ietdl.org

Published in IET Communications
Received on 28th September 2012
Revised on 22nd May 2013
Accepted on 10th June 2013
doi: 10.1049/iet-com.2012.0591

ISSN 1751-8628

# Energy- and latency-efficient broadcasting mechanism supporting long-term evolution e-multimedia broadcast/multicast service transmission

*Haw-Yun Shin*

*Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung, People's Republic of China*
*E-mail: hawyun.shin@msa.hinet.net*

**Abstract:** To provide increasing multimedia broadcast/multicast services (MBMSs), the third generation partnership project modified long-term evolution release nine documents to further support evolved MBMSs (e-MBMS). This study designed a data-broadcasting method called evolved broadcast scheduling algorithm (eBSA). This eBSA provides new scheduling technology for e-MBMS and a new information searching method for user equipments (UEs) that simultaneously conserves UE energy and maintains favourable access latency of broadcast messages. The eBSA constructs index and message channels based on the message popularities, using the index channel to guide UEs to rapidly locate the position of the desired message in the message channel. If searching for unpopular messages, the index channel guides UEs directly to conduct a linear search. Consequently, a short mean access time and mean turning time can be attained. The results of numerical analysis verify that when access messages possess skewed-access characteristics, the proposed eBSA method performs exceptionally. Additionally, by adequately adjusting the parameter provided by the eBSA method, the optimal tradeoff between energy conservation and access latency performance can be attained.

## 1 Introduction

In wireless network environments used before the emergence of 4G wireless services, peer-to-peer methods were typically used to provide services. This hinders network operators from large-scale extension and promotion of multimedia broadcast/multicast services (MBMSs) that possess significant volumes of data. However with the development of mobile network technologies and diversification of client requirements, user demand to experience multimedia services through mobile networks has grown significantly. Consequently, the international standard organisation third generation partnership project (3GPP) approved the revisal of the long-term evaluation (LTE) release nine documents to support evolved MBMSs (e-MBMS) [1].

The structure of the BMS environment is shown in Fig. 1 [2]. The coverage area of service provided by operators is significant. However, required service content differs among regions. According to the varied requirements of different regions, operators distribute various service contents to multicast/broadcast over a single frequency network (MBSFN) areas. An MBSFN area comprises numerous eNodeB regions. All eNodeBs within a single MBSFN area share and distribute wireless resources through the multi-cell/multicast coordination entity (MCE) to complete within-MBSFN service transmission. In addition, eNodeBs located within a MBSFN area may concurrently belong to multiple MBSFNs. Different MBSFNs use different control parameters to prevent interactive interference, and user equipment (UE) can simultaneously receive services provided by multiple MBSFN areas.

The e-MBMS provided by the LTE is constructed by modifying existing 3G mobile network environment structures. In addition to the original network entity, e-MBMS adds two new sets of logical nodes, MCE and MBMS gateway (MBMS GW) to assist the BMS of the LTE system. As shown in Fig. 1, the BMS center (BMSC) is the current network entity of the 3G network structures. The BMSC is the distributing centre of BMSs in the LTE. After the BMSC collects service contents from service providers, it broadcasts session announcements to its designated area or region. The service types of the system possesses are conveyed to and notify all UEs. Subsequently, all types of BMS are scheduled and the service contents are transmitted to the MBMS GW based on and in order of the schedule results.

After the MBMS GW receives the service content packet sent from the BMSC, the service transmission is prepared (session start) to arrange wireless resources for the soon-to-be conducted BMS. Subsequently, service content data are transmitted to the corresponding MBSFN area. The MCE is similarly a set of logical nodes that can possibly be integrated into the eNodeB or other network entities. The MCE is responsible for allocating the time and frequency resources for multi-cell MBMS transmission to complete the BMS of the MBSFN area. The UE obtains the


© The Institution of Engineering and Technology 2013

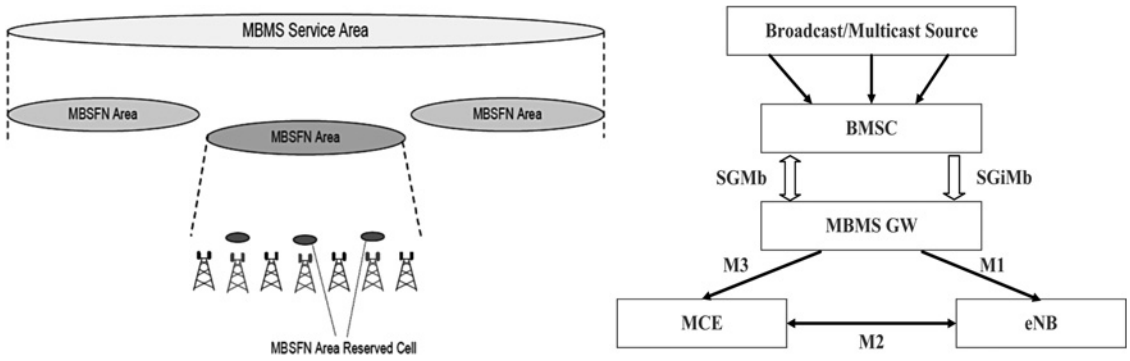


**Fig. 1** *MBSFA definition and e-MBMS service architecture*

modulation and coding scheme (MCS) and multicast channel (MCH) subframe allocation pattern (MSAP) of a broadcast service through the system information block. The MCS provides modulation and coding models for various channels to use and MSAP is used to define which subframes are used by certain MCHs during the scheduling intervals. After the UE determines the location of BMSs subframes through the MSAP, the dynamic schedule information is used to determine the initial position of each service to begin to receive service data [3].

Scheduling interval for BMSs can be customised from 40 to 640 m*s*, based on the service type and amount [4]. The MCE determines the MSAP of each MCH according to the quality of service (QoS) of each service within the scheduling interval. The distribution regulations of subframes for BMSs are as follows: (i) subframes 0, 4, 5 and 9 of each frame cannot be used in the MBSFN service transmission; (ii) subframes 1, 2 and 3 of each frame can be used in the MBSFN service transmission; and (iii) the first ten frames of the scheduling interval can be provided to MBSFN for service transmission. If the scheduling interval is greater than 100 ms, the MBSFN has 30 subframes in each scheduling interval that can be allocated for service transmission.

Recent academic and industrial studies on broadcast technologies have focused on mobile users' QoS and the goal of UE energy conservation. Therefore the BMS provided by e-MBMS must also consider these two crucial issues. To conserve the UE energy, 3GPP established the discontinuous reception (DRX) mechanism to allow UE to enter sleep mode when there is no data transmission [5]. However, the DRX mechanism is not applicable to BMSs. If some index information were added to service data that guides UE to switch between active and sleep modes appropriately and correctly receive required information, UE energy consumption can be reduced. Since 3GPP has not established any regulations or conventions regarding e-MBMS scheduling, the scheduling is designed and planned by operators.

This study develops an evolved broadcast scheduling algorithm (eBSA). This algorithm exhibits exceptional energy conservation and access latency performance, and attains a suitable tradeoff between these two performances. The eBSA method establishes the index channel and message channel based on the windmill concept. Popular messages are uniformly broadcasted at higher frequencies both in the index channel and message channel. The eBSA employs special indexing fields in each time slot to guide UEs to rapidly locate the position of the desired message in the message channel. If the search is for unpopular messages, the index channel guides UEs to enter the message channel directly to conduct a linear search. This design can achieve the minimum mean access time and mean tuning time and guarantee the maximum access time when UEs search unpopular messages. The result of numerical analysis indicates that when accessed messages possess skewed-access characteristics, the proposed eBSA offers superior performance compared with that of methods in already published research. The result also verifies that by adequately adjusting the parameter $P_e$ of the eBSA method, the optimal tradeoff that balances energy conservation and access latency performance can be achieved.

## 2 Related work

The metrics of the tuning time and access time are often used to evaluate the performance of broadcast systems [6]. Tuning time refers to the duration for which the UE remains active. It measures the energy that is consumed by the UE in the active mode. Conversely, access time refers to the speed with which a UE accesses the requested message. It represents the responsiveness of the system. Numerous studies [7–11] focused on broadcasting schedules to improve access time and/or tuning time. To cater for limited battery power, some air indexing techniques have been proposed to assist the mobile device in predicting the arrival time of requested data [6, 12, 13]. Chen *et al.* [12] proposed the organisation of an index tree in a broadcast channel, by using either Huffman coding with a fixed fanout or a greedy algorithm with variable fanout, to achieve a new optimal tuning time. Vaidya and Hameed [14] presented the lower bound for the UE mean access time in both uniform and non-uniform sizes. A scheduling algorithm was also defined, which achieved minimal mean access times.

Accurately obtaining information related to the access frequencies of broadcast data is a vital foundation for realising various broadcast schemes. Yu *et al.* [15] proposed a novel approach that exploited the available knowledge of broadcast misses and broadcast frequencies to refine the program to efficiently meet the requirements of the user population. They presented a statistical model based on the maximum-likelihood estimation for estimating the access frequencies. Stathatos *et al.* [16] demonstrated that frequently accessed data might be accurately obtained by monitoring the 'broadcast misses' that were observed through direct requests. The technique was to iteratively change the broadcasting patterns to broadcast the frequently accessed data.




© The Institution of Engineering and Technology 2013

A number of hybrid approaches that combine the broadcast and an on-demand service were also proposed [17–19]. Nicopolitidis *et al.* [17] estimated the demand for each item by the incoming requests for items that will not appear in the broadcast in the near future, and stated that, without requests for such broadcast missed items, the demand estimation is impossible. These studies focused on the manner in which UEs may receive the desired message quickly; however, all previous studies ignored the energy consumption for UEs because the broadcast system was regarded as an energy conservation system. Shin [20] exploited skewed-access to present an energy-efficient approach named WSA to organise messages in the broadcast channels for UEs to locate the desired message quickly and with a low energy requirement. However, when the broadcast message categories increase, using the WSA method can rapidly raise the mean access time.

In a recent study, Zhan *et al.* [21] combined current broadcast algorithms and network coding technology to increase system performance and bandwidth efficiency. In addition, [22] constructed an adaptive smart antenna-based wireless push system that integrates the locality of client demands, and proposed a BSA with favourable performance. Kim and Kang [23] classified broadcast messages into hot and cold types and broadcasted them using differing methods. They subsequently analysed the optimal classification method to obtain an enhanced broadcasting performance. Furthermore, Liaskos *et al.* [24] indicated that if QoS metrics improvement, such as the minimisation of the mean serving time or jitter, is overly emphasised, the broadcast system of a single channel reduces the client service ratio. Therefore Papadimitriou *et al.* [25] proposed a type of bandwidth assignment algorithm for service-ratio-oriented multiple broadcast channels to determine the optimal client service ratio. Finally, Liaskos *et al.* [26] compared the implementation costs of current broadcast system structures and algorithms and verified that the proposed method possessed minimum implementation costs.

## 3 Evolved broadcast scheduling algorithm

The UE uses FLUTE [27] to obtain service announcement, which contains the service list that informs users of the available service types. After subscribing to services, users receive a service key and user key to decipher the required service data [28]. Although UE can obtain the service list through service announcements, it must monitor the service notifications sent by the system at all times, resulting in the long-term active mode of UE and poor energy conservation. The eBSA concept divides each type of broadcast data into message and index portions. The message portion represents the broadcast content and the index portion stores a small volume of indexing information that guides UE to rapidly determine the required message. This method does not require constant monitoring of system notifications and appropriately guides UE to switch between active and sleep modes, thereby conserving energy and obtaining rapid message access. The eBSA establishes two services that represent the index and message portions in the BMSC. MBSFN subframes that distribute these two services are regarded as the index and message channels. After these broadcast index and message data are scheduled by the eBSA, they are transmitted separately and sequentially to the index channel and the message channel of the MBSFN for broadcast. By reading the index channel content, the UE can determine the correct time to return to the active mode after entering the sleep mode. With several reads of the index channel content, the position of the message searched in the message channel can be identified. Fig. 2 shows that the eBSA uses the BMS MSAP subframe to exemplify the index and message channels. According to the eBSA scheduling outcome, various messages use identical subframes to sequentially transmit various BMSs indices and messages at different times.

### 3.1 Structure of index channel and message channel

Fig. 2 shows the structure of the index channel and message channel. Each index in the index channel consists of ‘Key, Boundary Key, Extra Key, Next’ and ‘First’ fields. The function of each column is explained below.

- ‘Key’ field: record the key of to-be-broadcasted message.
- ‘Boundary Key’ field: if the key value of the searched message is smaller than the content of the ‘Boundary Key’ field, the message searched by the UE is a popular message and the search continues in the index channel. Otherwise, the searched message is an unpopular message and the UE conducts a linear search directly in the index channel.
- ‘Extra Key’ field: records the keys of those messages that follow in the next one, next two and next $E$ slot. $E$ is the number of extra keys.
- ‘First’ field: records the distance from the broadcasting index slot to the nearest index slot of the most popular message.
- ‘Next’ field: records the distance from the broadcasting index slot to the nearest index slot of the next popular message.

Assuming $K$ types of messages exist, their keys are $q_0, q_1, \ldots, q_{K-1}(q_0 < q_1 < \cdots < q_{K-1})$. When the UE begins to search for the desired message (considering the key is $q$), the boundary key is first extracted from the ‘Boundary Key’ field of the broadcasting index. If $q$ is larger than or equal to the boundary key, the desired message is an unpopular message. Thus, the UE is immediately to conduct a linear search in index channel to obtain the desired message. If $q$ is smaller than the boundary key, the desired message is a popular message. Subsequently, a further comparison of $q$ with the ‘Key’ field and ‘Extra Key’ field is conducted. The equivalence between $q$ and the ‘Key’ field or $q$ and ‘Extra Key’ field indicates that the desired message was located or was allocated in one of the next $E$ slots. If comparisons of $q$ with the ‘Key’ field and ‘Extra Key’ field are not equivalent, the UE enters the sleep mode. After passing the number of slots recorded in the ‘First’ field, the UE automatically returns to the active mode to re-read the index of the most popular messages. If UE further compared $q$ with the ‘Key’ and ‘Extra Key’ fields of this index and found that the information recorded by the index is not the desired message, the UE enters the sleep mode again. After passing the number of slots recorded in the ‘Next’ field, the UE automatically returns to the active mode to re-read the index of the next popular message. A further comparison was conducted to determine whether this index has recorded the information of the desired message. This method offers superior performance when dealing with messages with higher searching popularity. Fig. 3 shows the complete process of reading the desired message conducted by the UE.


© The Institution of Engineering and Technology 2013

doi: 10.1049/iet-com.2012.0591

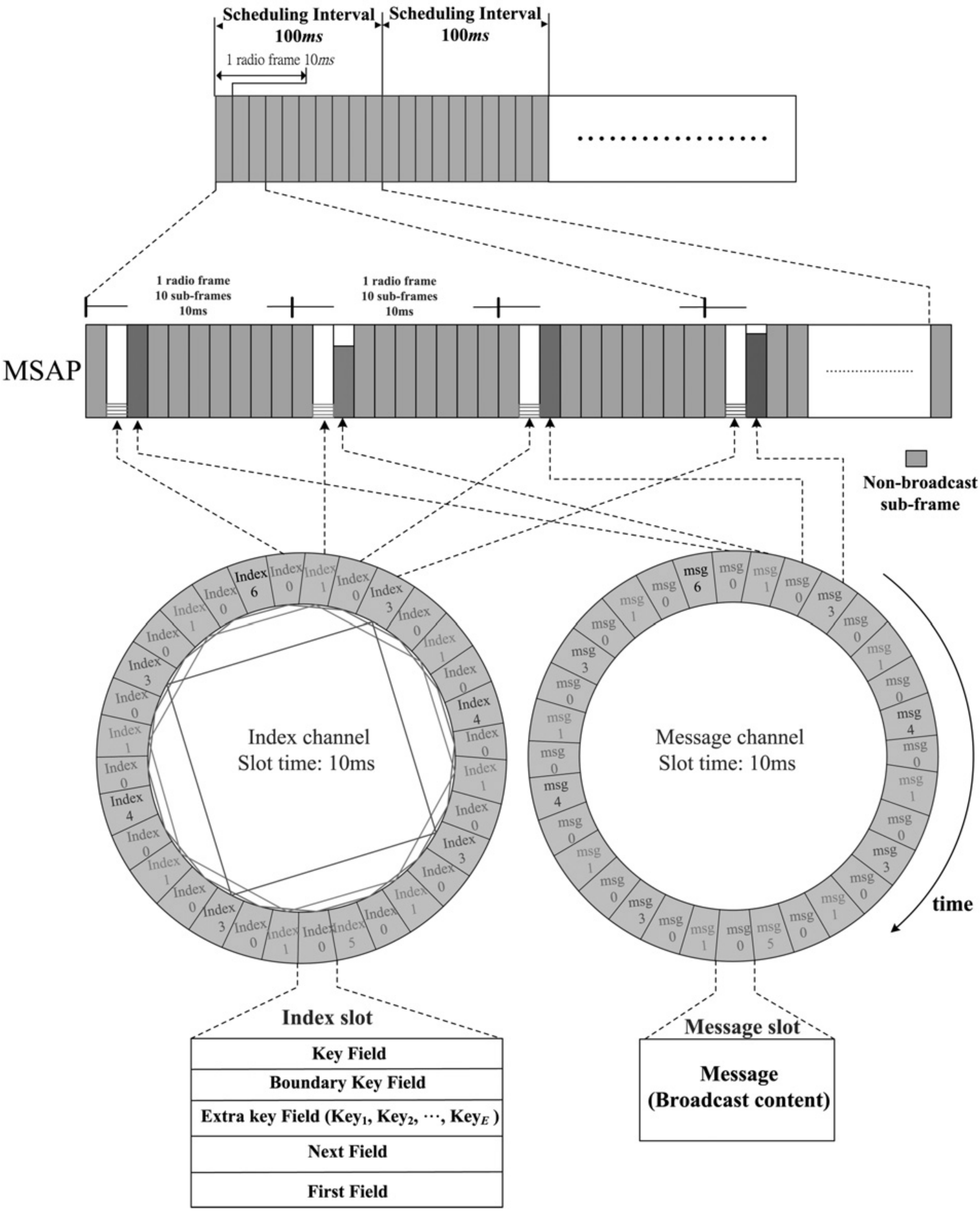


**Fig. 2** *Relationship between the scheduling interval, MSAP, index channel and message channel*

### *3.2 Slot allocation of index and message channels*

As shown in Fig. 2, the index channel and the message channel are considered as two slotted rings. The index information and message content of various message types can be uniformly distributed to the index and message channels and formed into various regular polygons based on popularity levels. Messages with higher popularity can form regular polygons with more sides. A UE may acquire a desired message at any time, and therefore the mean number of waiting slots is minimised if the broadcasted messages form a regular polygon in the ring.

Considering a periodical index channel with $C$ empty slots numbered 0, 1, 2, …, $C-1$, assume that the index information of message $q_i$ must be disseminated $C_i$ times $\left(C_i \leqq C\right)$. The slots number that is occupied by an index information of $q_i$ may be represented as a series $N_l^i (l = 0, \ldots, C_i - 1)$

$$N_l^i = \lfloor lC/C_i \rfloor \tag{1}$$



© The Institution of Engineering and Technology 2013

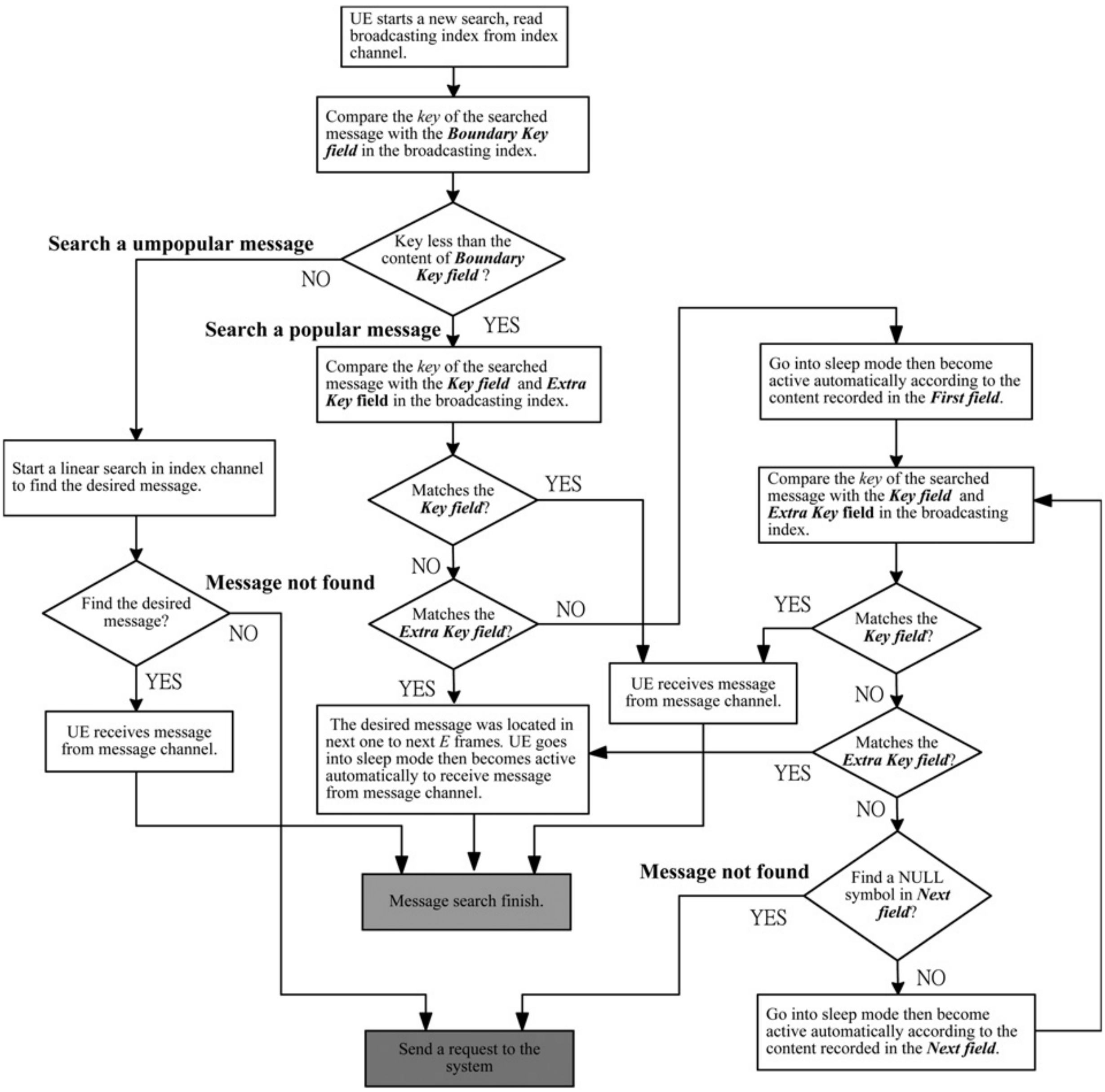


**Fig. 3** *Flowchart for a UE receiving a desired message*

When $C \gg C_i$, both the arranged messages and the remaining empty slots are almost symmetrical.

After index of one message type has been arranged completely, the remaining empty slots are renumbered and the same procedure is repeated for the one of next message type. To broadcast $K$ types of index ($C_0$, $C_1$, $C_2$, …, $C_{K-1}$) in $C$ time slots, the eBSA is used to form $K$ polygons that are as regular as possible. If $C$ is sufficiently large, each polygon is a regular polygon. Consequently, the index that are placed in the time slots are disseminated symmetrically, quantitatively and cyclically by the base station. Assume that the accessed frequencies of message $q_i (i = 0, \ldots, K-1)$ are sorted in descending order, the symmetry decreases gradually from message $q_0$ to message $q_{K-1}$.

When skewed access characteristics exist, most searches focus on popular messages. Although this method possesses exceptional performance in energy conservation and access latency, it sacrifices access latency when searching unpopular messages. To maintain the overall performance, the 'Boundary Key' field content in the index guides the UEs searching for unpopular messages directly to conduct linear search. Thus, a ratio value $P_e$ (the ratio of the number of popular message types and the total number of message types in the broadcast system, $P_e 1$) was defined to determine which of the keys from $q_0$ to $q_{K-1}$ can serve as a suitable 'Boundary Key'.

Assuming $K$ types of messages exist. During the statistical period, the number of access by UEs are $A_0, A_1, \ldots, A_{K-1}$ ($A_0 \geq A_1 \geq \cdots \geq A_{K-1}$). If $\sum_{i=0}^{B-1} A_i / \sum_{i=0}^{K-1} A_i \cong P_e$, messages $q_0$–$q_{B-1}$ are categorised as the popular message type. Conversely, $q_B$ to $q_{K-1}$ are categorised as the unpopular message type. $q_B$ is called the boundary key and recorded in the 'Boundary Key' field of each index. The UE searching popular messages rapidly determines the position of the desired message in the message channel from the index channel based on the eBSA searching algorithm. Conversely, the UE that searches unpopular messages directly conducts a linear search in the index channel under the guidance of the 'Boundary Key' field of index. When the number of message types increases, the time required for UE to access messages increases rapidly. Therefore the system can lower the $P_e$ value to increase the number of unpopular message types and reduce the time


© The Institution of Engineering and Technology 2013

doi: 10.1049/iet-com.2012.0591

required for UE to access messages. The closer the $P_e$ value is to one, the lower the number of unpopular message types.

The following steps explain the arrangement of message types into the index channel:

1. Calculate $C_i$ according to the accessed frequencies of message $q_i$, $i = K - 1, \ldots, 0$.
2. Allocate the index slots from message $q_0$ to message $q_{K-1}$ by using (1).
3. Fill the 'Key' field and the 'Boundary Key' field of each $q_i$index with $q_i$ and $q_B$, respectively, $i = 0, \ldots, K - 1$.
4. Fill in the 'First' field of each $q_i$ index with the distance between $q_i$ and the nearest $q_0$, $i = 1, \ldots, K - 1$.
5. Fill in the 'Next' field of each $q_i$ index with the distance from $q_i$ to the nearest $q_{i+1}$, $i = 0, \ldots, K - 2$.
6. Fill in the 'Next' field of each $q_{K-1}$ index with a 'NULL' symbol, which implies that the message $q_{K-1}$ is the last message type in the searching process.
7. Fill the $\text{key}_1$, $\text{key}_2$, …, $\text{key}_E$ in 'Extra key' field of each index. The $\text{'key'}_i$ is the key of the message that follows in the next $i$ slot.

Fig. 4 shows an example of the arrangement of various message types into index channel by using eBSA. Six message types are denoted as M1 to M6 in this example. The popularities from $M1$ to $M6$ are assumed to 16, 8, 4, 2, 1 and 1, respectively. A total of 32 time slots were used in this index channel and the boundary key is set to $q_5$ and number of extra keys is set to 2 ($E = 2$). As shown in Fig. 4, $M1$ is the most popular message type, which was broadcasted 16 times and formed a regular 16-sided polygon in the index channel. $M2$ was broadcasted eight times and formed a regular eight-sided polygon in the index channel. $M3$ and $M4$ were, respectively, broadcasted four times and two times in the index channel. $M5$ and $M6$ were categorised as unpopular messages.

## 4 Performance analysis

### 4.1 Energy conservation analysis

The 'mean tuning time' is the mean number of subframes in which the UE remains in the active mode. Let $T_{\text{MT}}$ denote the mean tuning time, $T_{\text{MT(eBSA)}}$ and $T_{\text{MT(LS)}}$ denote the mean tuning time that UEs access a popular message using eBSA and access an unpopular message using the linear search, respectively. Subsequently, $T_{\text{MT}} = T_{\text{MT(eBSA)}} + T_{\text{MT(LS)}}$ and

$$T_{\text{MT(eBSA)}} = \left\{ \sum_{i=0}^{B-1} \frac{A_i}{A_{\text{T}}} \left[ 1 + \sum_{j=0}^{i} \left( 1 - \frac{C_i}{C} \right)^{\text{j}E+1} \right] \right\} \text{T}_{\text{sub\_frame}} \tag{2}$$

$$T_{\text{MT(LS)}} = \left\{ \sum_{i=B}^{K-1} \left( \frac{A_i}{A_{\text{T}}} \frac{C}{2(E+1)C_i} \right) \right\} \text{T}_{\text{sub\_frame}} \tag{3}$$

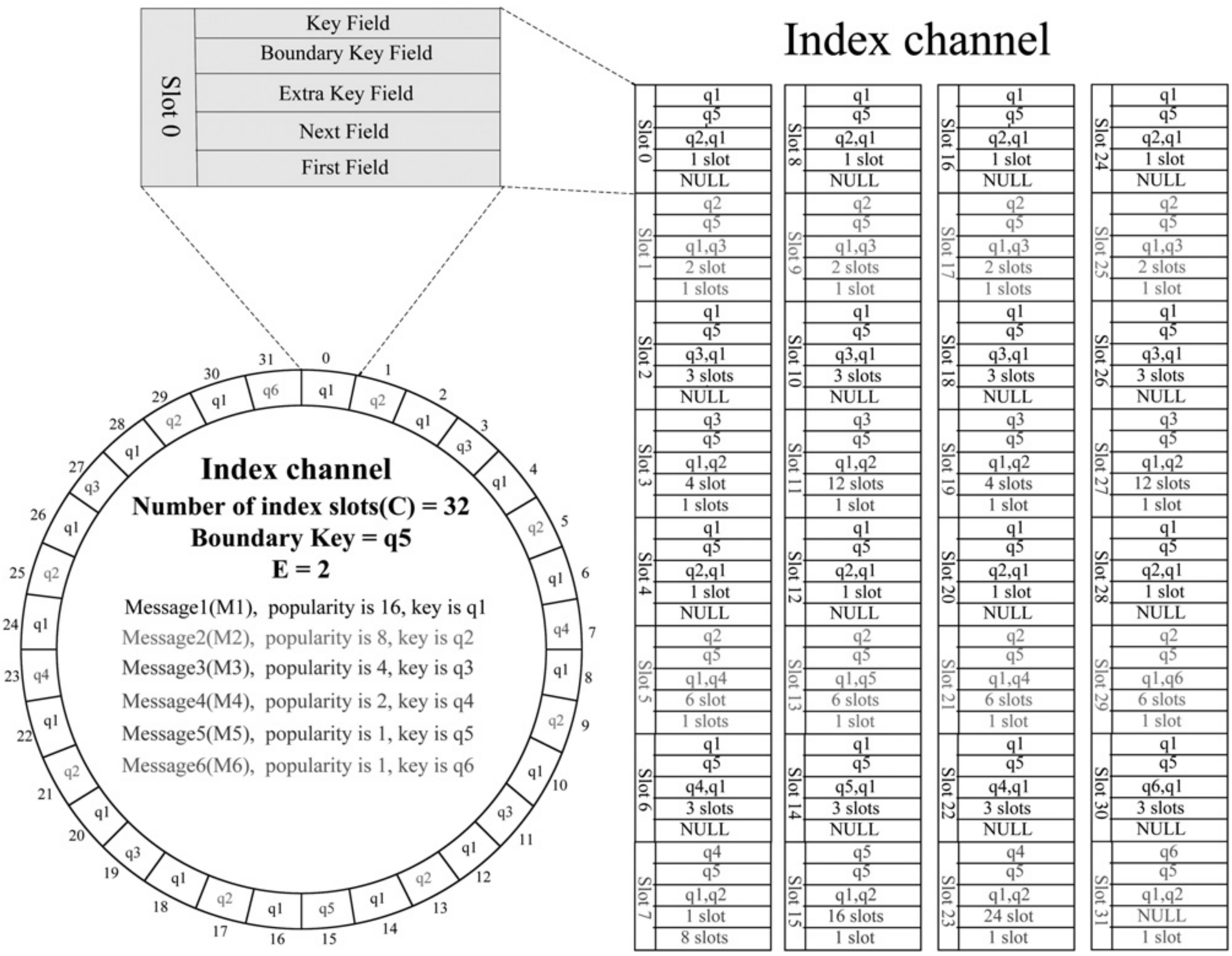


**Fig. 4** *Example explaining how to arrange different message types into index channel by using eBSA*




© The Institution of Engineering and Technology 2013

where $T_{\text{sub\_frame}}$ is 1 ms, $K$ is the number of message types, $B$ the boundary key number, $E$ the number of extra keys, $A_i$ the number of accesses of message $i$ ($i = 0, \ldots, K-1$), $A_T$ the total number of accesses in a statistical duration, $A_T = \sum_{i=0}^{K-1} A_i$, $C_i$ the number of index slots allocated to message $q_i$, where $C_i = C\lfloor A_i/A_T \rfloor$, $C_0 = C - \sum_{i=1}^{K-1} C_i$ and $C_i$ value is 1 or above.

The case where a UE requests a $q_0$ message is considered. The item of $A_0/A_T$ is the probability of message $q_0$ being accessed by a UE. The UE has a probability of $C_0/C$ consuming one unit of tuning energy and a probability of $(1 - C_0/C)$ consuming two units of tuning energy to acquire the desired message. The mean tuning time the UE acquires the message $q_0$ can be written as

$$\left\{\frac{A_0}{A_T}\left[\frac{C_0}{C} + 2\left(1 - \frac{C_0}{C}\right)\right]\right\}T_{\text{sub\_frame}}$$

Next, the case where a UE requests a $q_1$ message is considered. When a new search starts, the UE has a probability of $C_1/C$ to acquire the desired in the broadcasting message, in which case UE consumes one unit of tuning energy. If the broadcasting message is not desired, the UE has a probability of $[1 - (C_1/C)]\{1 - [1 - (C_1/C)]^E\}$ to find the desired in one of the next $E$ slots, in which case UE consumes two units of tuning energy. Otherwise, the UE has a probability of $[1 - (C_1/C) - [1 - (C_1/C)] - \{1 - [1 - (C_1/C)]\}^E)]$ to consume three units of tuning energy to acquire the desired one. Therefore the mean tuning time the UE acquires the message $q_1$ can be represented to

$$\left\{\frac{A_1}{A_T}\left[\frac{C_1}{C} + 2\left(1 - \frac{C_1}{C}\right)\left(1 - \left(1 - \frac{C_1}{C}\right)^B\right) + 3\left(1 - \frac{C_1}{C} - \left(1 - \frac{C_1}{C}\right)\left(1 - \left(1 - \frac{C_1}{C}\right)^E\right)\right)\right]\right\}T_{\text{sub\_frame}} \quad (4)$$

Similarly, the mean tuning time the UE acquires the message $q_i (i < B)$ can be generalised as

$$\left\{\frac{A_i}{A_T}\left[1 + \sum_{j=0}^{i}\left(1 - \frac{C_i}{C}\right)^{jE+1}\right]\right\}T_{\text{sub\_frame}} \quad (5)$$

### 4.2 Access latency analysis

'Mean access time' is the mean number of frames that a UE spend to complete message accessing. Let $T_{MA}$ denote the mean access time, $T_{MA(eBSA)}$ and $T_{MA(LS)}$ denote the mean access time that UEs access a popular message using eBSA and access an unpopular message using linear search, respectively. Subsequently, $T_{MA} = T_{MA(eBSA)} + T_{MA(LS)}$, and (see (6))

$$T_{MA(LS)} = \left\{\sum_{i=B}^{K-1}\left(\frac{A_i}{A_T}\frac{C}{2C_i}\right)\right\}T_{\text{frame}} \quad (7)$$

where $T_{\text{frame}}$ is 10 ms, $S_0$ is the mean number of index slots for which $q_i$ waits for the nearest index of $q_0$, $S(C_{j-1}, C_j, C)$ is the mean number of index slots for which $q_{j-1}$ waits for the nearest index of $q_j$, $j = 1, \ldots, B-1$.

Assume that index of $q_0$ is disseminated $C_0$ times in index channel, the distance between two adjacent index slots is $C/C_0$, and therefore $S_0 = C/2C_0$.

Consider if the index information of $q_i$ and $q_{i+1}$ are broadcasted $C_i$ and $C_{i+1}\left(C_i \geqq C_{i+1}\right)$ times in the index channel that has $C$ time slots, a regular $C_i$ polygon and a regular $C_{i+1}$ polygon on a circle are formed for indexes of $q_i$ and $q_{i+1}$, respectively. The relationships between $C_i$ and $C_{i+1}$ can be written as follows

$$C_i = \lceil C_i/C_{i+1} \rceil \left(C_i + C_{i+1} - \lceil C_i/C_{i+1} \rceil C_{i+1}\right) + \left(\lceil C_i/C_{i+1} \rceil - 1\right)\left(\lceil C_i/C_{i+1} \rceil C_{i+1} - C_i\right)$$

Let $\alpha = \lceil C_i + C_{i+1} \rceil$, $\beta = (C_i + C_{i+1} - \lceil C_i + C_{i+1} \rceil C_{i+1})$, $\gamma = \lceil C_i/C_{i+1} \rceil - 1$ and $\delta = (\lceil C_i/C_{i+1} \rceil C_{i+1} - C_i)$. The significance of $\alpha$, $\beta$, $\gamma$ and $\delta$ are that of a regular $C_{i+1}$-sided polygon with $C_{i+1}$ edges, the $\beta$ of these edges map to $\alpha$ vertices of a regular $C_i$-sided polygon and $\delta$ edges map to $\gamma$ vertices of a regular $C_i$-sided polygon. Consequently, the mean number of index slots for which $q_i$ waits for the nearest $q_{i+1}$ is

$$S\left(C_i, C_{i+1}, C\right) = \frac{1}{C_i}\left[\beta\sum_{k=0}^{\alpha-1}\left(kC/C_i + C/2C_i\right) + \delta\sum_{k=0}^{\gamma-1}\left(kC/C_i + C/2C_i\right)\right] \quad (8)$$

Additionally, the selection of the $P_e$ value significantly influences the performance. An appropriate $P_e$ value can determine the balance between energy conservation and access latency. Therefore after the $P_e$ value is set, the B value (boundary key number) of (2), (3), (6) and (7) can be determined using $\sum_{i=0}^{B-1} A_i/A_T \cong P_e$.

## 5 Numerical results and discussion

In this numerical analysis environment, assume that the scheduling interval does not exceed 100 ms. Thus, each frame possesses three subframes for transmitting broadcast service data. One subframe is used for the eBSA index channel and another subframe is used for the eBSA message channel. After various types of broadcast messages and relevant indices are scheduled through the BMSC, they are separately and sequentially transmitted to the message channel and index channel based on the scheduling results for broadcasting. In addition, this study assumed that all

$$T_{MA(eBSA)} = \left\{\begin{array}{l}\sum_{i=0}^{B-1}\frac{A_i}{A_T}\left\{\frac{C_i}{C}\left\{\sum_{j=0}^{E}(j+1)\left(1-\frac{C_i}{C}\right)^{j} + \sum_{j=0}^{i-1}\left\{\sum_{m=1}^{E}\left[S_0 + \sum_{n=1}^{j}S\left(C_{n-1}, C_n, C\right) + m\right]\left(1-\frac{C_i}{C}\right)^{(j+1)E+m}\right\}\right\}\right. \\ \left. + \left(S_0 + \sum_{j=1}^{i}S\left(C_{j-1}, C_j, C\right)\right)\left(1-\frac{C_i}{C}\right)^{(i+1)E+1}\right\}\end{array}\right\}T_{\text{frame}} \quad (6)$$


© The Institution of Engineering and Technology 2013

doi: 10.1049/iet-com.2012.0591

broadcast message types transmitted through the eBSA mechanism using identical MCS, QoS and block error rate (BLER) requirements.

To evaluate the performance of the proposed method regarding energy conservation and message access latency, numerical analysis was used to compare the following three methods: eBSA, eBSA_LS and the index tree with two fixed fanouts (index tree) [12]. The eBSA_LS method enables the UE to conduct a linear search in the index channel scheduled by the eBSA. The performance of this method is significantly greater than that of the normal linear search methods. Chen *et al.* [12] presented a favourable solution for minimising the tuning time. They proposed methods to build the index trees based on the popularity, and proved that an index tree with fixed fanout required minimal tuning time. To conduct a fair comparison of these three methods regarding the change of the mean tuning time and mean access time with increases in the number of message types, this study assumed that these three methods possessed the same broadcast cycle and number of broadcast channels.

The purpose of the parameter $P_e$ value was to establish the proportion of searches that should be completed in the index channel. eBSA allocates the number of time slots for each type of broadcast messages based on message popularity. Messages with higher popularity are allocated more time slots. Subsequently, eBSA guides UE to search required messages in the index channel in order of message popularity (from highest to lowest). Although this method shortens the search time for messages with higher popularity, the time required to search for lower-popularity messages increased substantially. When the number of message types increases, the time spent by UE searching for low-popularity messages significantly influences the system performance regarding access latency. An effective method entails removing abundant messages with lower popularity from the index channel, thereby enabling UE to directly access the required messages through a linear search in the data channel. Although this method increases power consumption, it significantly reduces the access latency of UE in searching for low-popularity messages. In the eBSA method, the messages that are broadcast through the index channel are categorised as popular messages, whereas messages that are broadcast through channels other than the index channel are categorised as unpopular messages. A $P_e$ of 1 represents that all messages are broadcast through the index channel (i.e. all broadcast messages are considered popular). A $P_e$ of 0 indicates that all messages are broadcast through channels other than the index channel (i.e. all broadcast messages are considered unpopular). When $P_e$ is 0.90, popular messages and unpopular messages account for 90 and 10% of all messages searched, respectively. Therefore the number of popular messages in the index channel can be changed by adjusting $P_e$. By adequately adjusting the $P_e$ value, the optimal balance between energy conservation and access latency can be attained.

The eBSA exploits the skewed-access characteristic in allocating time slots to various types of messages to improve energy consumption and access latency. Guided by the information in the index channel, UE switches to sleep mode during the majority of the UE search process for required messages and UE promptly returns to the active mode to read broadcast messages, thereby achieving energy conservation. When skewed-access characteristics exist, most of the message accesses by UE are concentrated on a small portion of popular messages. eBSA allocates the number of time slots based on the popularity of messages. This method shortens the search time for messages with higher popularity, thereby reducing the mean access latency for UE reading messages. A large degree of skewness (DS) implies that the access frequency markedly varies between various message types. The 'Zipf' formula was employed to deduce various DSs. Fig. 5 shows the access probability of various message types under various DS. When DS = 1, various message types possess the same access probability, or the system considers that all message types possess the same access probability. A DS of 2, 3 or 4 indicates the increasingly skewed access between message types. As shown in Fig. 5, message types with smaller numbers have greater access probability. The popularity of message types decreases sequentially from $q_0$ to $q_{99}$. In the real broadcast system, skewed access phenomena necessarily exist (i.e. DS ≠ 1). Thus, the subsequent performance analysis is conducted based on the access probability presented by DS = 2, DS = 3 and DS = 4.

Fig. 6 shows changes in the mean tuning time and mean access time with sequential increases in the number of message types when DS = 2. As shown in Fig. 6, although eBSA_LS possesses the minimum mean access time, it has the maximum mean tuning time. The index tree method, despite its optimal energy conservation performance, shows poor mean access time. This result contrasts with that of the eBSA_LS method. When $E$ = 0 and $P_e$ = 1, the eBSA method possesses inferior performance to that of the index tree method. After increasing the number of extra keys ($E$ = 3 and $E$ = 5), the mean tuning time of the eBSA method decreases significantly and the mean access time improves. When skewed-access characteristics exist, most of the message accesses are concentrated on a small portion of popular messages; thus, the amount of the unpopular messages is tremendous. When the number of broadcast messages increases, slightly reducing $P_e$ can decrease the number of popular messages in the index channel to shorten the time required for UE to search the required messages in the index channel; consequently, the mean access time is improved significantly. However, an increase in the number

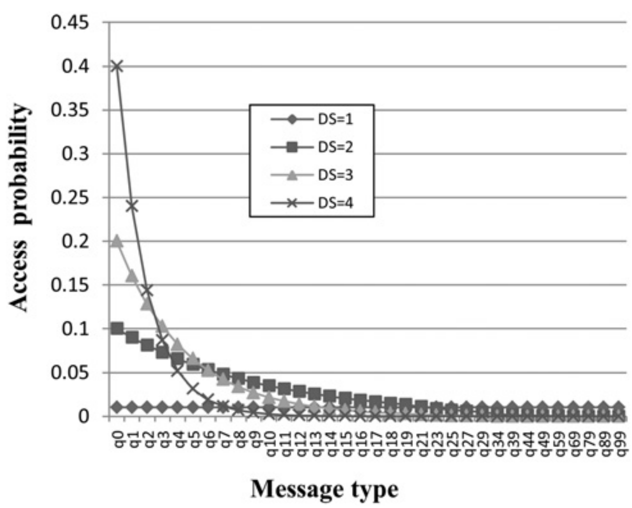


**Fig. 5** *Popularity (access probability) of message types for different DS. 'Zipf' formula is employed*
Popularity of $q_i = \left(\left(1/i^s\right)/\left(\sum_{j=0}^{k-1} 1/j^s\right)\right)$. $K$ is number of message types. $s$ = 0, 0.7, 1 and 1.5, respectively, represent DS = 1, 2, 3 and 4



© The Institution of Engineering and Technology 2013

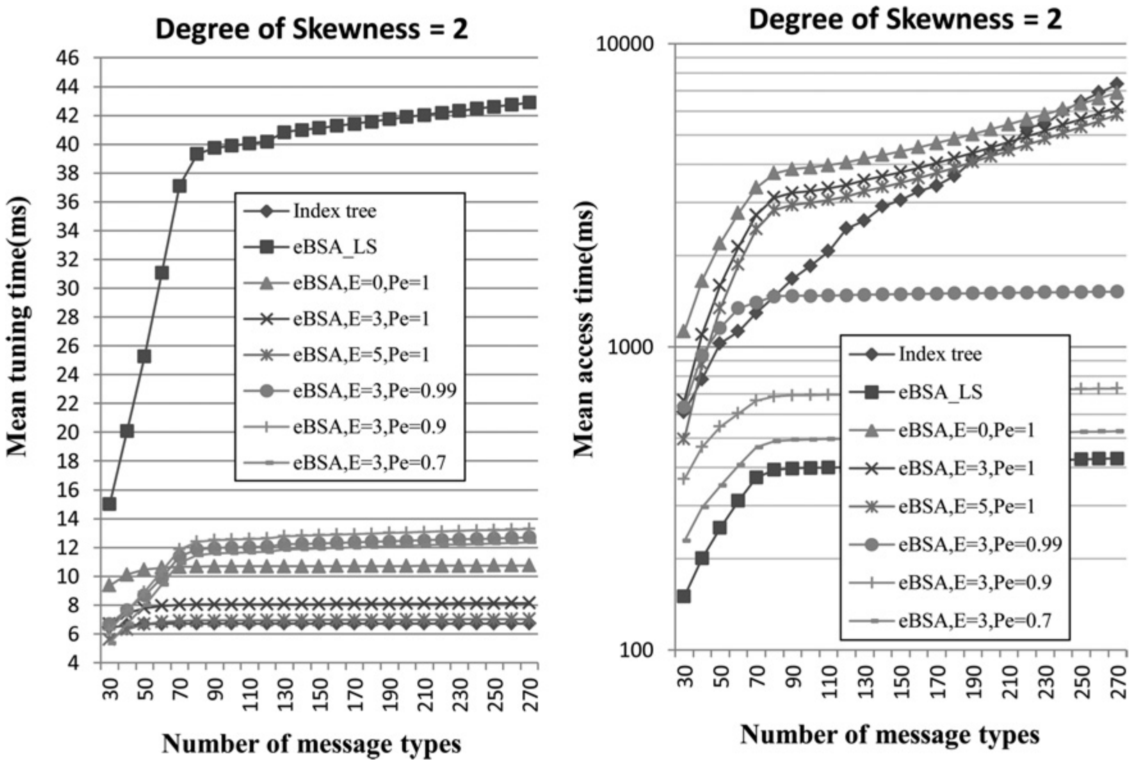


**Fig. 6** *Comparisons of the mean tuning time and mean access times among eBSA, eBSA_LS and index tree for DS = 2*

of unpopular messages increases power consumption for UE to search for required messages in the data channel using a linear search. Fig. 6 shows that if the $P_e$ is finely tuned to 0.9 or 0.7, the mean access time decreases substantially; however, the mean tuning time increases.

When DS = 4, the eBSA method achieves outstanding energy conservation performance. As shown in Fig. 7, when ($E = 3$ and $P_e = 1$) and ($E = 5$ and $P_e = 1$), the eBSA requires an extremely brief mean turning time to receive the required message. However, when DS = 4, the access

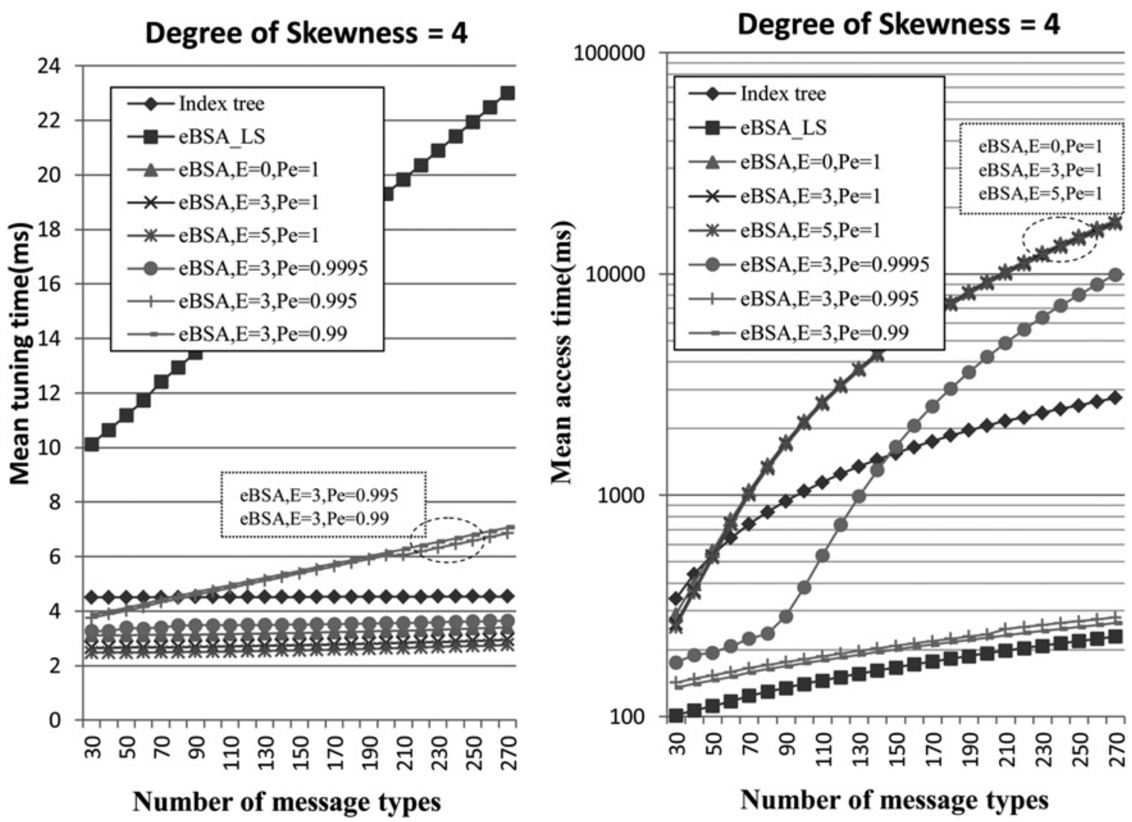


**Fig. 7** *Comparisons of the mean tuning time and mean access times among eBSA, eBSA_LS and index tree for DS = 4*


© The Institution of Engineering and Technology 2013
doi: 10.1049/iet-com.2012.0591

probability centres excessively on a few message types. When the number of message types increases, the eBSA ($P_e = 1$) method requires a relatively longer time to search the unpopular message types, thereby increasing the mean access time. Nevertheless, by adequately adjusting the parameter $P_e$ can avoid this phenomenon. As shown in Fig. 7, because $P_e$ is set at 0.995 or 0.99, the mean access time improves significantly and it performs in a superior manner to the index tree method. Thus, performance that is similar to that of eBSA_LS is attained. However, the mean tuning time is significantly shorter than that of the eBSA_LS method and superior to that of the index tree.

These discussion results indicate that when using the eBSA method, the mean tuning time and mean access times can be

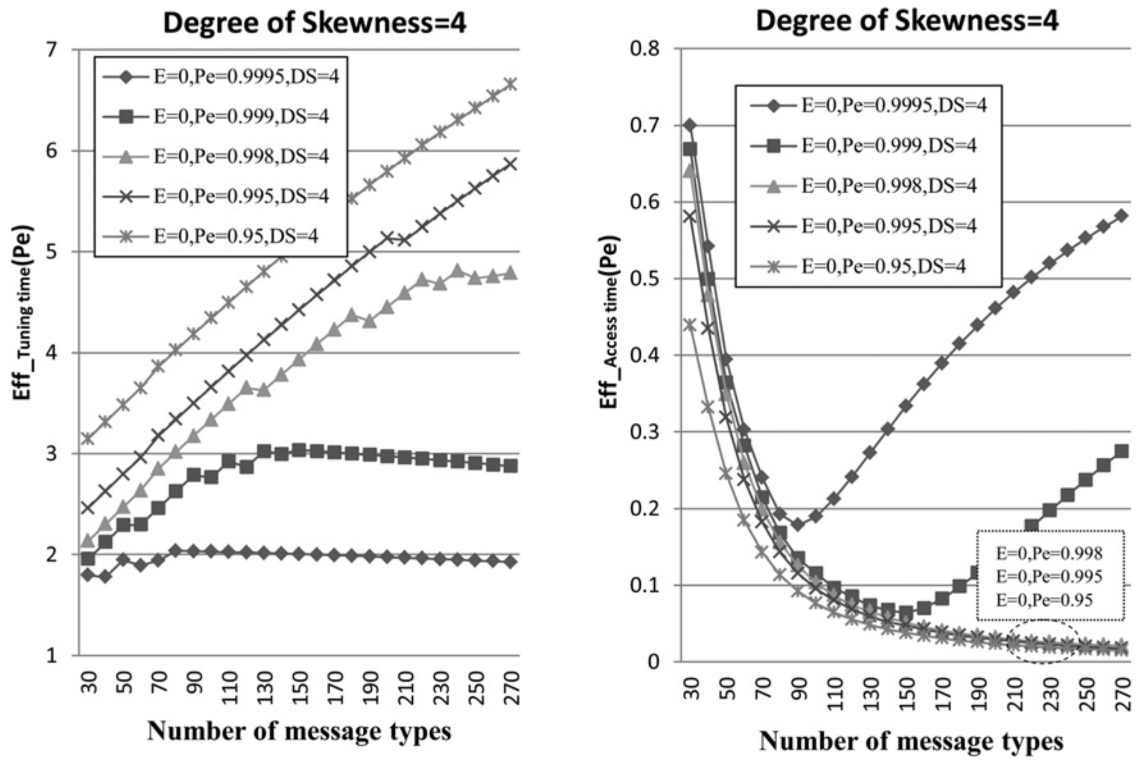


**Fig. 8** *Efficiency improvement of mean tuning time and mean access time for different value of $P_e$ when DS = 4 and E = 0*

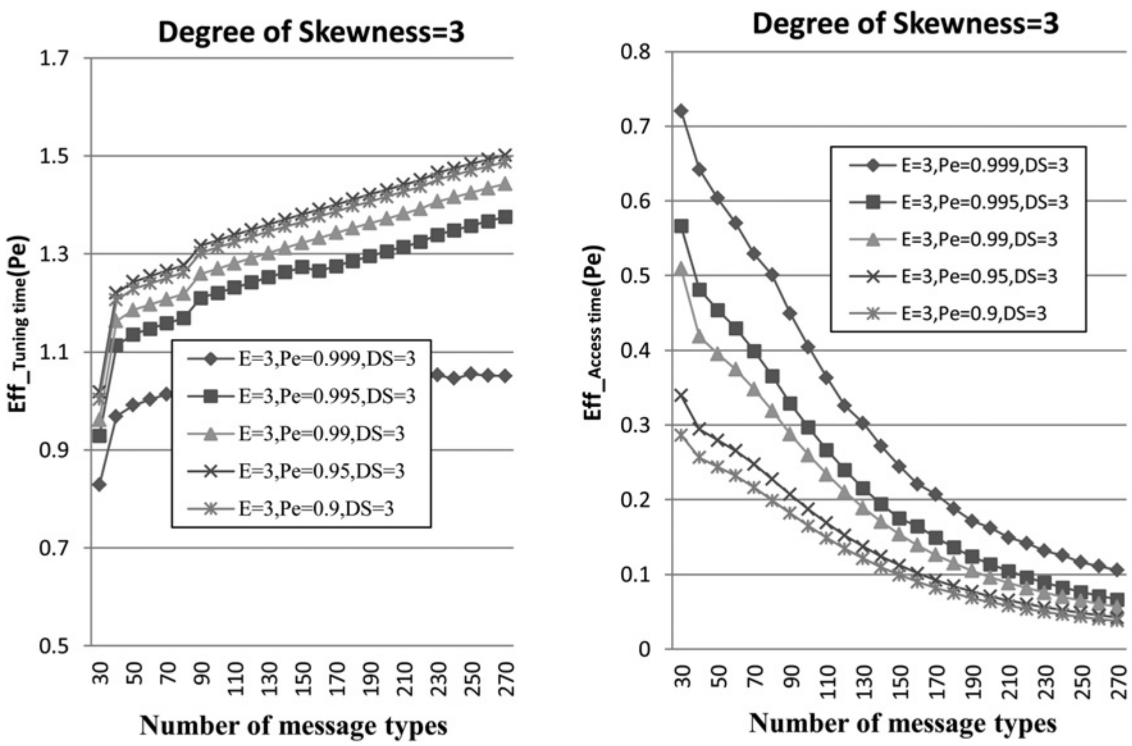


**Fig. 9** *Efficiency improvement of mean tuning time and mean access time using extra keys (E = 3) and different value of $P_e$ when DS = 3*


© The Institution of Engineering and Technology 2013

effectively improved by increasing the number of extra keys. If $P_e$ is adequately adjusted, a favourable tradeoff between energy conservation and access latency can be achieved. Eff_Tuning time ($P_e$) and Eff_Access time ($P_e$) were defined, respectively, as the performance improvement evaluations of the mean tuning time and the mean access time regarding different $P_e$ values, where

- Eff_Tuning time ($P_e$) = mean tuning time ($P_e < 1$ and $E = 0$)/mean tuning time ($P_e < 1$ and $E = 0$)
- Eff_Access time ($P_e$) = mean access time ($P_e < 1$ and $E = 0$)/mean access time ($P_e < 1$ and $E = 0$).

If the Eff_($P_e$) is less than 1, the performance is improved. The adjustment of the $P_e$ value can substantially improve the mean access time despite the slight increase in the mean tuning time. A $P_e$ value that provides a greater improvement in the mean access time and less influence on the mean tuning time was selected as the appropriate value of $P_e$.

Fig. 8 shows the variation of Eff_Tuning time ($P_e$) and Eff_Access time ($P_e$) regarding different $P_e$ values when DS = 4. When $P_e = 0.9995$ and $P_e = 0.999$, the Eff_Access time ($P_e$) curve reverses upwardly when the number of message types increases, indicating that an excessive number of popular messages influences the system performance. To improve the performance of the mean access time, more messages should be set as unpopular messages by reducing the value of $P_e$. As shown in Fig. 8, if the number of message types exceeds 150, a similar performance improvement of Eff_Access time ($P_e$) is attained at $P_e = 0.998$, $P_e = 0.995$ and $P_e = 0.95$. However, the increase of $Eff_{\text{\_Tuning time}}$ ($P_e$) at $P_e = 0.998$ is slight compared with other $P_e$ values. Thus, $P_e = 0.998$ is the better choice.

Fig. 9 shows the performance improved by the eBSA using extra keys and $P_e$ value adjustment concurrently. Using DS = 3 and $E = 3$ as examples, when $P_e = 0.999$, Eff_Tuning time ($P_e$) approaches 1 and Eff_Access time ($P_e$) decreases rapidly with the increasing number of message types. This shows that the eBSA significantly improves the mean access time without sacrificing the mean tuning time. Furthermore, the situation of $P_e = 0.9$ was observed. When the number of message types increases, although Eff_Tuning time ($P_e$) is slightly greater than 1, Eff_Access time ($P_e$) decreases rapidly to 0.05. This shows that the mean access time performance improved to nearly 20 times the original.

## 6 Conclusion

This study proposed an eBSA method to improve the LTE e-MBMS performance. The proposed eBSA provides e-MBMS with a novel scheduling technology and provides UEs with novel information search methods that can balance energy conservation and access latency performance of UE. By establishing the index channel based on the message popularities, the eBSA uses special indexing fields to guide UEs to rapidly determine the position of the desired message in the message channel. If the searched message is an unpopular message, the index channel guides UEs to conduct a linear search directly in the index channel. This design can attain a short mean access time and mean tuning time. This study compared the eBSA method with the eBSA_LS and index tree with two fixed fanouts (index tree) methods. The numerical analysis results indicate that when various types of accessed messages possess skewed-access characteristics, the proposed eBSA performance regarding access latency is superior to that of the index tree method and is similar to that of the eBSA_LS method. However, the energy conservation performance of the eBSA is prominent. Additionally, the eBSA method provides the parameter $P_e$ to distinguish the boundary between popular and unpopular broadcast messages. The numerical analysis results indicated that by adequately and appropriately adjusting the $P_e$ value, the optimal tradeoff balancing energy conservation and access latency performance can be attained.

## 7 Acknowledgment

This work was supported by the National Science Council, R.O.C., under grant NSC102-2221-E-019-031 and NSC101-2218-E-019-007.

## 8 References


1 3GPP RP-090350: 'WID for MBMS support in LTE', March 2009
2 3GPP TS 36.300 V9.2.0: 'E-UTRA and E_UTRAN, overall description, stage 2', July 2009
3 3GPP R2-093990: 'Details of eMBMS dynamic scheduling information', July 2009
4 3GPP R2-093778: 'Details of dynamic scheduling information', July 2009
5 3GPP TS 36.321 V9.0.0: 'Evolved Universal Terrestrial Radio Access (E-UTRA) Medium Access Control (MAC) protocol specification', September 2009
6 Xu, J., Lee, W.-C., Tang, X.: 'Exponential index: a parameterized distributed indexing scheme for data on air'. Proc. ACM/USENIX MobiSys, Boston, Massachusetts, 2004, pp. 153–164
7 Liaskos, C.K., Petridou, S.G., Papadimitriou, G.I.: 'Cost-aware wireless data broadcasting', *IEEE Trans. Broadcast.*, 2010, **56**, (1), pp. 66–76
8 Yao, Y., Tang, X., Lim, E.-P., Sun, A.: 'An energy-efficient and access latency optimized indexing scheme for wireless data broadcast', *IEEE Trans. Knowl. Data Eng.*, 2006, **18**, (8), pp. 1111–1124
9 Chung, Y.D., Yoo, S., Kim, M.H.: 'Energy- and latency-efficient processing of full-text searches on a wireless broadcast stream', *IEEE Trans. Knowl. Data Eng.*, 2010, **22**, (2), pp. 207–218
10 Yang, X., Bouguettaya, A.: 'Adaptive data access in broadcast-based wireless environments', *IEEE Trans. Knowl. Data Eng.*, 2005, **17**, (3), pp. 326–338
11 Sakata, T., Yu, J.X.: 'Statistical estimation of access frequencies problems, solutions and consistencies', *Wirel. Netw.*, 2003, **9**, (6), pp. 647–657
12 Chen, M.-S., Yu, P.S., Wu, K.-L.: 'Optimizing index allocation for sequential data broadcasting in wireless mobile computing', *IEEE Trans. Knowl. Data Eng.*, 2003, **15**, (1), pp. 161–173
13 Xu, J., Lee, W.-C., Tang, X., Gao, Q., Li, S.: 'An error-resilient and tunable distributed indexing scheme for wireless data broadcast', *IEEE Trans. Knowl. Data Eng.*, 2006, **18**, (3), pp. 92–404
14 Vaidya, N.H., Hameed, S.: 'Scheduling data broadcast in asymmetric communication envoronments', *Wirel. Netw.*, 1999, **5**, (3), pp. 171–182
15 Yu, J.X., Sakata, T., Tan, K.L.: 'Statistical estimation of access frequencies in data broadcast environments', *Wirel. Netw.*, 2000, **6**, (2), pp. 89–98
16 Stathatos, K., Roussopoulos, N., Baras, J.S.: 'Adaptive data broadcast in hybrid networks'. Proc. 23rd VLDB Conf., Athens, Greece, 1997, pp. 326–335
17 Nicopolitidis, P., Papadimitriou, G.I., Pomportsis, A.S.: 'Exploiting locality of demand to improve the performance of wireless data broadcasting', *IEEE Trans. Veh. Technol.*, 2006, **55**, (4), pp. 1347–1361
18 Hu, C.-L., Chen, M.-S.: 'Adaptive multichannel data dissemination: support of dynamic traffic awareness and push-pull time balance', *IEEE Trans. Veh. Technol.*, 2005, **54**, (2), pp. 673–686



© The Institution of Engineering and Technology 2013

doi: 10.1049/iet-com.2012.0591

19 Huang, J.-L.: 'AIDOA: an adaptive and energy-conserving indexing method for on-demand data broadcasting systems', *IEEE Trans. Syst. Man Cybern. A, Syst. Hum.*, 2008, **38**, (2), pp. 331–345
20 Shin, H.Y.: 'Exploiting skewed access and energy-efficient algorithm to improve the performance of wireless data broadcasting', *Comput. Netw.*, 2012, **56**, (4), pp. 1167–1182
21 Zhan, C., Lee, V.C.S., Wang, J., Xu, Y.: 'Coding-based data broadcast scheduling in on-demand broadcast', *IEEE Trans. Wirel. Commun.*, 2011, **10**, (11), pp. 3774–3783
22 Nicopolitidis, P., Kakali, V.L., Papadimitriou, G.I., Pomportsis, A.S.: 'On performance improvement of wireless push systems via smart antennas', *IEEE Trans. Commun.*, 2012, **60**, (2), pp. 312–316
23 Kim, S., Kang, S.H.: 'Scheduling data broadcast: an efficient cut-off point between periodic and on-demand data', *IEEE Commun. Lett.*, 2010, **14**, (12), pp. 1176–1178
24 Liaskos, C.K., Tsioliaridou, A.N., Papadimitriou, G.I.: 'More for less: getting more clients by broadcasting less data'. Proc. 10th Int. Conf. Wired/Wireless Internet Communications (WWIC 2012), Santorini, Greece, 2012, pp. 64–75
25 Papadimitriou, G., Nicopolitidis, P., Pomportsis, A.: 'Parallel data broadcasting for optimal client service ratio', *IEEE Commun. Lett.*, 2012, **16**, (11), pp. 1741–1743
26 Liaskos, C., Petridou, S., Papadimitriou, G.: 'Towards realizable, low-cost broadcast systems for dynamic environments', *IEEE/ACM IEEE Trans. Netw.*, 2011, **19**, (2), pp. 383–392
27 IETF RFC 3926: 'FLUTE – file delivery over unidirectional transport', October 2004
28 3GPP TS 26.346 V6.12.0: 'Technical specification group services and system aspects; Multimedia Broadcast/Multicast Service (MBMS); protocols and codecs', September 2008


doi: 10.1049/iet-com.2012.0591
© The Institution of Engineering and Technology 2013

Copyright of IET Communications is the property of Institution of Engineering & Technology and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.