Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**Connection Management and Relay Path Configuration**
Date Submitted	**2007-3-13**

Source(s)	Aik Chindapol Jimmy Chui Hui Zeng Siemens Corporate Research Princeton, NJ, 08540, USA	Voice: +1 609 734 3364 Fax: +1 609 734 6565 Email: aik.chindapol@siemens.com
	Teck Hu Siemens Networks Boca Raton, FL 33431, USA	
	Haihong Zheng, Yousuf Saifullah, Shashikant Maheshwari Nokia 6000 Connection Drive, Irving, TX, USA	Voice: +1 972 894 5000 Haihong.1.Zheng@nokia.com, Yousuf.Saifullah@nokia.com, Shashikant.Maheshwari@nokia.com
	Yuan-Ying Hsu Telcordia Applied Research Center Taiwan Co., Taipei, Taiwan	Voice: +886-2-37895177#4558 Fax: +886-2-26552078 yyhsu@tarc-tw.research.telcordia.com
	David Comstock, John Lee, Shang Zheng, Aimin Zhang Huawei Technologies No.98, Lane91, Eshan Road, Shanghai, P.R.C	Voice: +1 858 735 9382 dcomstock@huawei.com
	Jen-Shun Yang, Tzu-Ming Lin, Wern-Ho Sheen, Fang-Ching Ren, Chie Ming Chou, I-Kang Fu Industrial Technology Research Institute (ITRI)/ National Chiao Tung University	Voice: +886-3-5914616 Fax: +886-3-5820263 jsyang@itri.org.tw

(NCTU), Taiwan 195,Sec. 4, Chung Hsing Rd.
Chutung, Hsinchu, Taiwan 310, R.O.C.

Torsten Fahldieck
Alcatel-Lucent R&I
Holderaeckerstr.35, Stuttgart, Germany

Voice: +4971182132163
Fax: +4971182132453
torsten.fahldieck@alcatel-lucent.de

Erwu Liu, Dongyao Wang, Gang Shen, Kaibin Zhang, Jimin Liu, Shan Jin
Alcatel Lucent, R&I Shanghai, No.388, Ningqiao Road, Shanghai, P.R.C.

Voice: 86-21-50551240-8194
Fax: 96-21-50554554
{Erwu.liu, Dongyao.Wang, Gang.A.Shen, Kaibin.Zhang, Jimin.Liu, Shan.Jin} @alcatel-sbell.com.cn

Byung-Jae Kwak, Sungcheol Chang, Young-il Kim
ETRI
161, Gajeong-Dong, Yuseong-Gu, Daejeon, Korea 305-350

Voice: +82-42-860-6618
Fax: +82-42-861-1966
bjkwak@etri.re.kr

Changkyoon Kim, Kyu Ha Lee, Hyung Kee Kim
Samsung Thales Co., Ltd
San 12-1, Nongseo-Dong, Giheung-Gu, Yongin-City, Gyeonggi-Do, Korea 446-712

Voice: +82-31-280-9919
Fax: +82-31-280-1620
changkyoon.kim@samsung.com

Kenji Saito, Takashi Inoue
KDDI R&D Laboratories Inc.
Hikarino-oka 7-1, Yokosuka, Kanagawa 239-0847, Japan

Voice: +81 46 847 6347
Fax: +81 46 847 0947
saito@kddilabs.jp

Voice: +82 31 279 5248

Sungjin Lee,
Samsung Electronics

Fax: +82 31 279 5130
steve.lee@samsung.com

Re:	IEEE 80216j-07_007r2:" Call for Technical Comments and Contributions regarding IEEE Project 802.16j"
Abstract	This contribution proposes path and connection management procedures in multi-hop relay system.
Purpose	This contribution is provided as input for the IEEE 802.16j baseline document.
Notice	This document has been prepared to assist IEEE 802.16. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16.
Patent Policy and Procedures	The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures <http://ieee802.org/16/ipr/patents/policy.html>, including the statement "IEEE standards may include the known use of patent(s), including patent applications, provided the IEEE receives assurance from the patent holder or applicant with respect to patents essential for compliance with both mandatory and optional portions of the standard." Early disclosure to the Working Group of patent information that might be relevant to the standard is essential to reduce the possibility for delays in the development process and increase the likelihood that the draft publication will be approved for publication. Please notify the Chair <mailto:chair@wirelessman.org> as early as possible, in written or electronic form, if patented technology (or technology under patent application) might be incorporated into a draft standard being developed within the IEEE 802.16 Working Group. The Chair will disclose this notification via the IEEE 802.16 web site <http://ieee802.org/16/ipr/patents/notices>.

Connection Management and Relay Path Configuration

Siemens, Nokia, Telcordia, Huawei, ITRI, Alcatel-Lucent, ETRI, Samsung Thales, KDDI, Samsung

1 Introduction

In 802.16e [3], each connection (both management and data) is identified by a Connection ID (CID) [2]. There is no routing required; data is transmitted solely between the BS and the MS. In the Multihop Relay system, one or more RSs exist between an MR-BS and an MS. In order to forward traffic between MR-BS and MS, routing path needs to be established between them across the intermediate RSs. A path consists of a sequence of RS identifier, and is determined in a MR cell subject to a set of constraints such as availability of radio resource, radio quality of the link, QoS, load condition of a RS, etc..

This contribution proposes two simple path management schemes for multi-hop relay systems where the MR-BS makes centralized decision of a path. The MR-BS establishes the path by either informing all the RS along the path of relevant path information or embedding path information as part of connection management. In the first scheme, the MR-BS informs RS of the mapping between a connection as identified by a CID and an established path. The connection could be a regular transport connection established for a MS as defined in [3], basic and primary management CID allocated to RS/MS, or a tunnel connection as proposed in [8]. The RS builds up its routing table based on path and creates the binding relationship between CID and the path. In the latter scheme, each relay station is assigned a range of CIDs for which the relay is responsible. The parent node controls a superset of this CID range, and any child nodes (both RS and MS) are assigned disjoint subsets of the CID range. Because of this systematic structure, the relay path is established based on destination's CIDs and each relay station can recognize its packets and forward them to corresponding stations.

2 Data forwarding with explicit path information

2.1 Overview

In this section, we propose to use extended DSx (x represents Add, Change and Deletion) message to populate the routing path and path/CID binding information to the RSs on a specific path. Being different from legacy DSx messages defined for 802.16e, DSx signaling in multihop relay network is only processed by the RS along the selected path. To support constraint-based path establishment, Explicit-Route TLV and Path-ID TLV are included in the DSx message. To support path/CID binding operation, the DSx messages includes CIDs and service flow parameters. The CIDs could be regular MS transport CIDs, basic and primary CIDs, or tunnel CIDs. Furthermore, this extended DSx message also supports multiple path management operations in one signaling process.

The basic procedure of the path management proposed in this contribution is highlighted below.

- MR-BS creates routing paths, assigns an unique path id to the path, and populates the detailed path information to all the RS along the path
- MR-BS allocates CIDs to the RSs and MSs and creates a binding between CID and the path identified by path id. In the tunnelling case, the CID is the Tunnel CID (T-CID); while in the non-tunnelling case, the CID is the individual CID allocated to RS or MS.
- MR-BS populates the CID-path ID binding information to all the RSs along the path.
- Each RS should store the CID-path ID binding information into the routing table and derive the data forwarding table based on the detailed path information.
- When topology changes, due to events such as mobility, a new path may be created and/or the CID-path ID binding needs to be repopulated to every RS on the new path and removed from the old path.

2.2 Illustration of Topology Discovery and Path Management Procedures

Figure 1 illustrates the path establishment procedure during network entry for both MS and RS, as well as the binding procedure between the basic/primary management CID and selected paths. The network entry procedure is in line with [11].

- When RS1 attempts to conduct initial ranging, it sends regular RNG-REQ. After receiving a regular RNG-REQ, the MR-BS determines that RS1 directly attaches to it. MR-BS then sends the RNG-RSP to RS1. The other initial network entry procedures remain the same as MS. Such procedure may trigger the routing table update for RS1 in the MR-BS by including the basic and primary management CID of RS1. MR-BS also establish a path (P1: MR-BS, RS1) by sending a DSA*-REQ only to RS1 (not shown in Figure 1).
- When RS2 attempts to conduct initial ranging, it sends regular RNG-REQ. After receiving a regular initial RNG-REQ, RS1 replaces the Initial Ranging CID with its basic CID and sends it to the MR-BS. Upon receiving the RNG-REQ, MR-BS replaces RS1's basic CID with Initial Ranging CID and processes it. Then MR-BS determines that RS2 attaches to RS1 directly. It generates a RNG-RSP for RS2 and sends to RS1 using RS1's basic CID. Upon receiving the RNG-RSP, RS1 replaces its basic CID with Initial Ranging CID and sends it to RS2. The other initial network entry procedures remain the same as MS. MR-BS also establish a path (P2: MR-BS, RS1, RS2) by sending a DSA*-REQ, which is processed hop-by-hop by RS1 and RS2 (not shown in Figure 1). The binding between P1 and the basic and primary management CID of RS2 is included in the same message. MR-BS may also generate a new path id for the path between itself and RS1.
- When MS attempts to conduct initial network entry, it sends a regular RNG-REQ to RS2. RS2 replaces the Initial Ranging CID with its basic CID and sends it to the MR-BS. RS1 will just simply forward it to the MR-BS. Upon receiving the RNG-REQ, MR-BS determines that MS attaches to RS2 directly. It then calculates the relay path to be used toward MS (in this example, it's the relay path P2: MR-BS – RS1 – RS2), and then generates the basic and primary management CID for the MS. MR-BS sends RNG-RSP to RS2 using RS2's basic CID. Upon receiving the RNG-RSP, RS2 replaces its basic CID with Initial Ranging CID and sends it to MS.
- In order to inform all the RSs on the path of the routing information and optionally the service flow requirement for the basic and primary management CID of the MS, the MR-BS sends DSA*-REQ to all

the RSs on the path. The transmission mechanism of DSA-REQ message is hop-by-hop. Each RS receiving the request would process DSA*-REQ and store path/CID binding data in their routing table. This process is repeated until the DSA*-REQ reaches the last hop. The final RS replies with a DSA*-RSP. The further traffic sent over the basic and primary management CID will be routed by each RS through the identified path. MR-BS may generate a new path id for the path between itself and RS2 and log MS's basic/ primary management CID in the routing table.

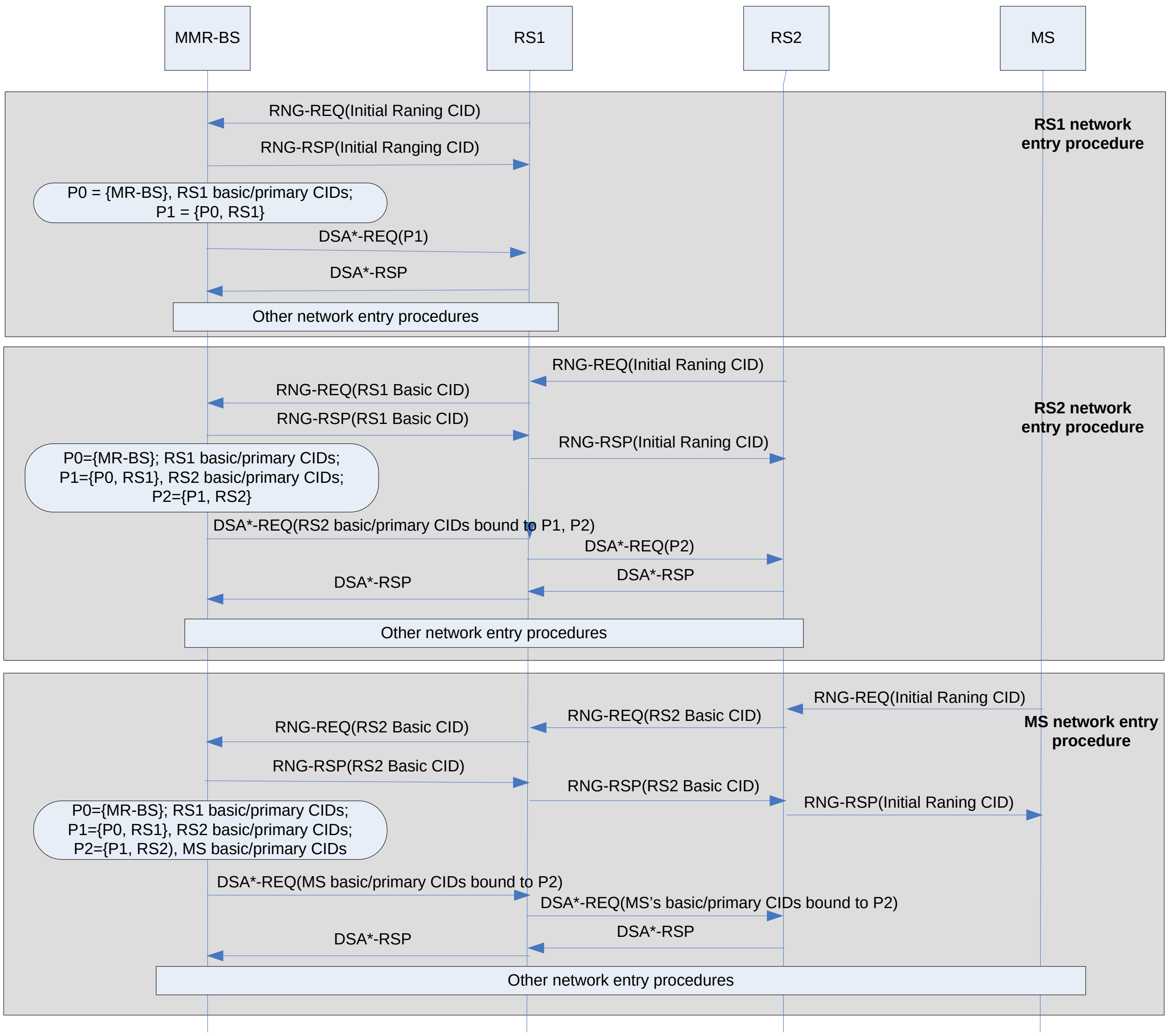


Figure 1: Illustration of Path Management Procedures During Network Entry

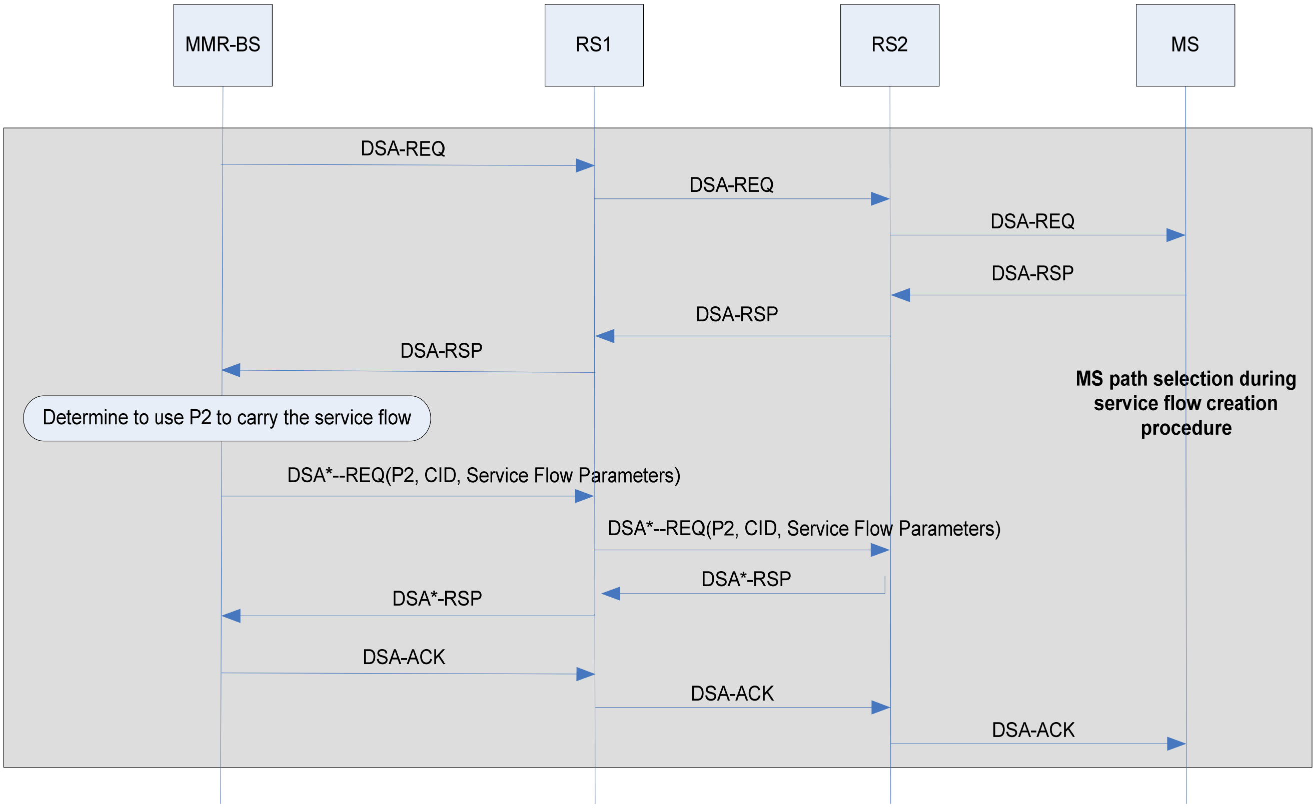


Figure 2: Illustration of Path Management Procedures During Service Flow Creation

As another example, Figure 2 shows the CID to path binding procedure in multi-hop relay system during the MR-BS initiated service flow creation procedure. Again, this example shows non-tunnel scenario.

- When MR-BS wishes to establish an uplink or downlink dynamic service flow, it sends DSA-REQ with MS CID. The DSA-REQ is forwarded by RS1 and RS2 to the MS. MS then responds with DSA-RSP, which is also forwarded by RS2 and RS1 to the MR-BS.
- Upon receiving a successful DSA-RSP, the MR-BS determines the path(s) to be used to carry the service flow. It then sends DSA*-REQ with RS1 CID. This message includes the selected Path-ID, the CID associated with the service flow and optionally the service flow parameter set to all the RSs on the path.
- Upon receiving the DSA*-REQ, RS1 obtains the mapping between the Path-ID and CID, which will be used to route the traffic for the specified service flow. The service flow parameters can be used for the RS to schedule the traffic for the specified service flow accordingly. RS1 derives the next hop (i.e., RS2) to further transmit the request based on the path information associated with the Path-ID, and

forwards the DSA*-REQ to RS2. RS2 processes the message in the same manner and responds with a DSA-RSP. RS1 updates the DSA-RSP and sends it to the MR-BS.

- The MR-BS completes the transaction by sending the acknowledgement message DSA-ACK to the MS.

3 Data forwarding with embedded path information

3.1 Overview

In this section, we propose to use systematic CID assignment to provide routing path information. By combining CIDs with the routing for each connection, the routing structure can be updated and maintained easily along with CIDs. To systematically assign CIDs to the MR-BS and RSs, the proposed CID allocation mechanism adopts the partitioning of the positive integers into subsets. The idea is to map these subsets to nodes in a network, which will assist in identifying the placement of the node in the tree. Each node of the tree represents a subset of **Z**, the set of all positive integers. The leaves of the tree are pairwise disjoint subsets of the integers. Each parent node is a superset of the union of its children. For example, in Figure 3, $B \supset (D \cup E)$ and $D \supset (H \cup J \cup K)$. The tree can grow; at a particular node, its children must satisfy two conditions. 1) the children must be subsets of the parent node; 2) the children are pairwise disjoint.

Due to this structure, any node (root, leaf, or intermediary) can determine whether a particular integer will exist in its subtree (with itself as the root). Intermediary nodes must distinguish between two types of integers; those that terminate at the node (terminal integers), and those that do not terminate at the node (non-terminal integers). We provide two examples of integer partitioning that assume only one terminal integer at each intermediary node, and briefly mention how multiple terminal integers (per intermediary node) can be attained.

In this section, we describe two methods to systematically assign CIDs. This can be accomplished by factorization into bit partition, or contiguous blocks.

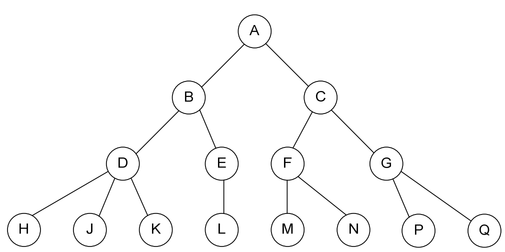


Figure 3: an example of a network tree (an abstract model)

3.1.1 Examples of integer partitioning: contiguous integer blocks

This is a simple implementation. The root node represents **Z**. Each of its children (1st tier nodes) are assigned a contiguous range of **Z** (and pairwise disjoint). For a particular 1st tier node (with range [p_1, p_2]), its children (2nd tier nodes) are each assigned a contiguous subset of [p_1, p_2] (and pairwise disjoint). This process continues for the entire tree. In Figure 4, we demonstrate how the tree in Figure 3 can partition the integers using contiguous integer block methods.

In Figure 4, the terminal integers for nodes B, C, D can be set to 1000, 2000, and 400 respectively. Allowing multiple terminal integers per intermediary node is trivial. We perform this CID assignment scheme *ignoring the MS in the topology*. This method is compatible with the notion of tunnel CIDs. Tunnel CIDs are the CIDs of the terminal access RS of the appropriate QoS service, but routing is considered a separate problem. With this systematic CID allocation scheme amongst the RSs, the tunnel CIDs may be distributed smartly so that routing is embedded within the CID structure with minimal signaling.

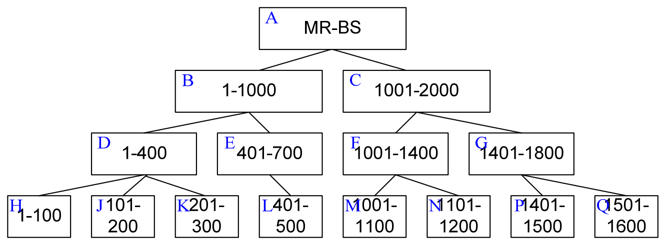


Figure 4: Systematic CID assignment using contiguous blocks. The choice of range length being multiples of 100 is arbitrary.

3.1.2 Examples of integer partitioning: bit partition

Each decimal number could also be converted into a binary number. Assume there are at most 2^k RSs could associate with one RS or BS directly, k bits would be used to identify each RS in the same level. The 1^{st} tier nodes that associate to the MR-BS directly would have CIDs with all possible number in lowest k bits. Their children (2^{nd} tier nodes) is identified by left shifting k bits of parent CID and set lowest k bits. This process continues for the entire tree. In this manner, the CID (without leading 0s) of any RS will be the prefix of CIDs of all its subordinate RSs.

To convert these values into subsets of **Z** (as discussed in Section 3.1) is simple; a n^{th} tier node will have a unique *nk*-bit sequence to identify itself, then the range this node could assign will be all numbers with this nk-bit sequence in the middle and begin with arbitrary number of "0"s as its prefix and with arbitrary combination of 0 and 1 as its suffix. The condition as set out in Section 3.1 is satisfied. We also demonstrate how the tree in Figure 3 can partition the integers using bit partition method in Figure 5.

We first define a parameter 2^k to identify the maximum number of subordinate RSs that the MR-BS or a RS could have. If k=0, each RS could only have one subordinate RS. For 1^{st}-tier RSs, which connect to the MR-BS directly, the MR-BS assigns IDs sequentially from 1 to 2^k as shown in Figure 5 by setting different values of the lowest k bits of the ID. We only show the lowest 8 bits of CIDs in Figure 5. For other n-tier RSs, the MR-BS left shifts k bits of its parent ID and sets the lowest k bits according to the arriving sequence of the RS. For example, RS_T and RS_U comes one after another to associate with RS_Q (ID: 00 01 00 11) after RS_R and RS_S in Figure 3. To assign an ID to RS_T, the MR-BS first perform left shift 2 bits of its parent ID and gets 01 00 11 00, and then it sets the lowest 2 bits as 10 since it is the third RS that attaches to RS_Q. Similarly, the MR-BS assigns 01 00 11 11 to RS_U after RS_T.

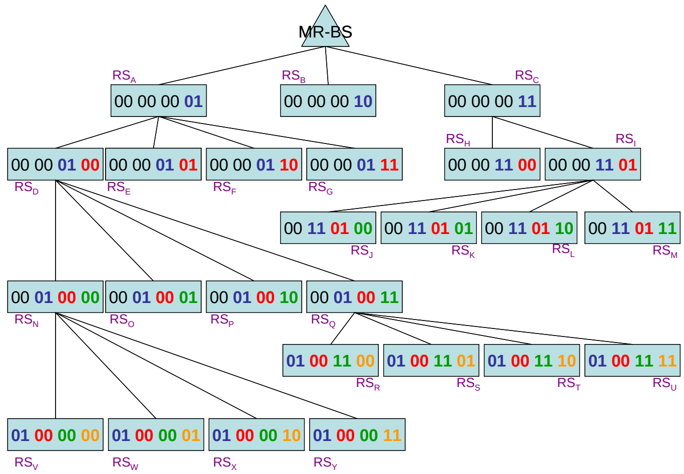


Figure 5: Systematic CID assignment using bit partition.

Note that a simple way of allowing multiple terminal values at intermediary nodes is to merge nodes. For example, the logical nodes H and J can represent the same physical node.

3.2 Examples of relay path configuration

In the following, we show examples how *contiguous and bit partitioning methods could be applied to relay path configuration.*

We take figure 6 for example. There are two MSs, which associate to RS L (CID: 00 01 01 00) and RS G (CID: 00 00 10 01), in the network. The MR-BS has records for these two MSs and knows their serving RSs. The whole relay path could be divided into two segments: from the source RS to the MR-BS and from the MR-BS to the destination RS. For upstream frames, each RS could easily know its parent CID by right shifting *k* bits of its own CID. For example, the CID of access RS L is 00 01 01 00, so its parent CID is 00 00 01 01 by right shifting 2 bits of its CID. For downstream frame received from its parent RS, the RS needs to determine if it should accept, forward, or discard the frame. When the tunneling [8] [10] is applied for relaying, the Tunnel CID could be set as the CID of destination RS. Each intermediate RS could compute if the destination RS belongs to its subordinate RSs by the algorithm in Figure 7. First of all, the RS compares if the destination CID is equal to its own CID and accepts the frame if these two CIDs are the same. If the match fails, it perform *k*-bit right shift of the destination CID and do the comparison with its own CID. If the shifted destination CID is the same as its own CID, it forwards the frame to its subordinate RS. Otherwise, it continues do the right shift and comparison for (maximal level-current level) times and discards the frame if all matches are failed. For example, RS C would know that RS G is its subordinate RS by right shifting the destination CID once.

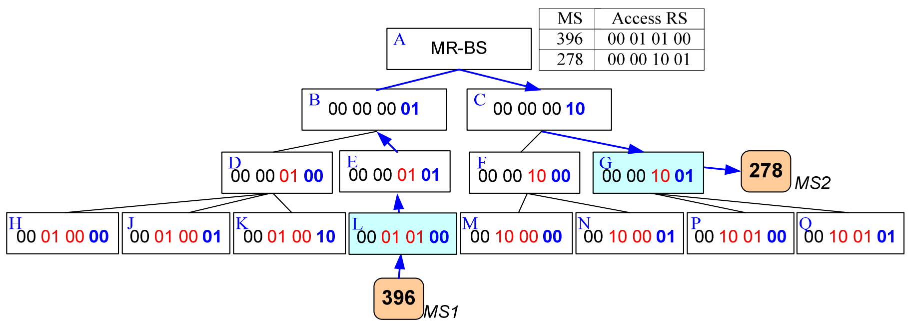


Figure 6: An example of relay path configuration using bit partition method.

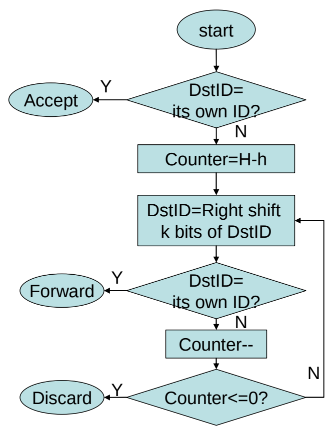


Figure 7: Subordinate RS differentiation algorithm

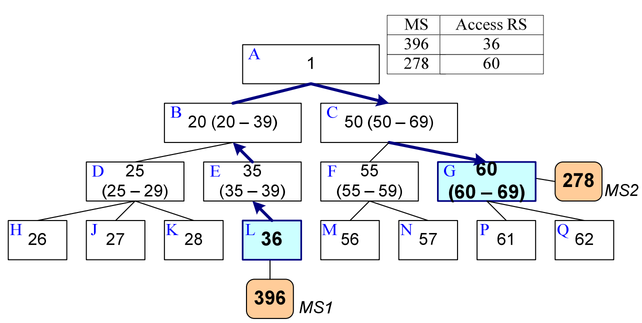


Figure 8: An example of relay path configuration using contiguous integer partitioning method. The number in parenthesis is the range of CIDs that the MR-BS could allocate to the subordinate RS.

Similarly with contiguous integer partitioning shown in Figure 8, the MR-BS keeps records of the access RS for each MS. For data directed towards MS2, the MR-BS sends the data to the access RS with CID 60. Since this CID belongs to the range of CID of the RS C, it forwards the data to the RS G. Meanwhile, the RS B ignores this data as the CID is not within its range. The similar procedure can be done on the uplink. Figure 9 illustrates procedures of path management with embedded information.

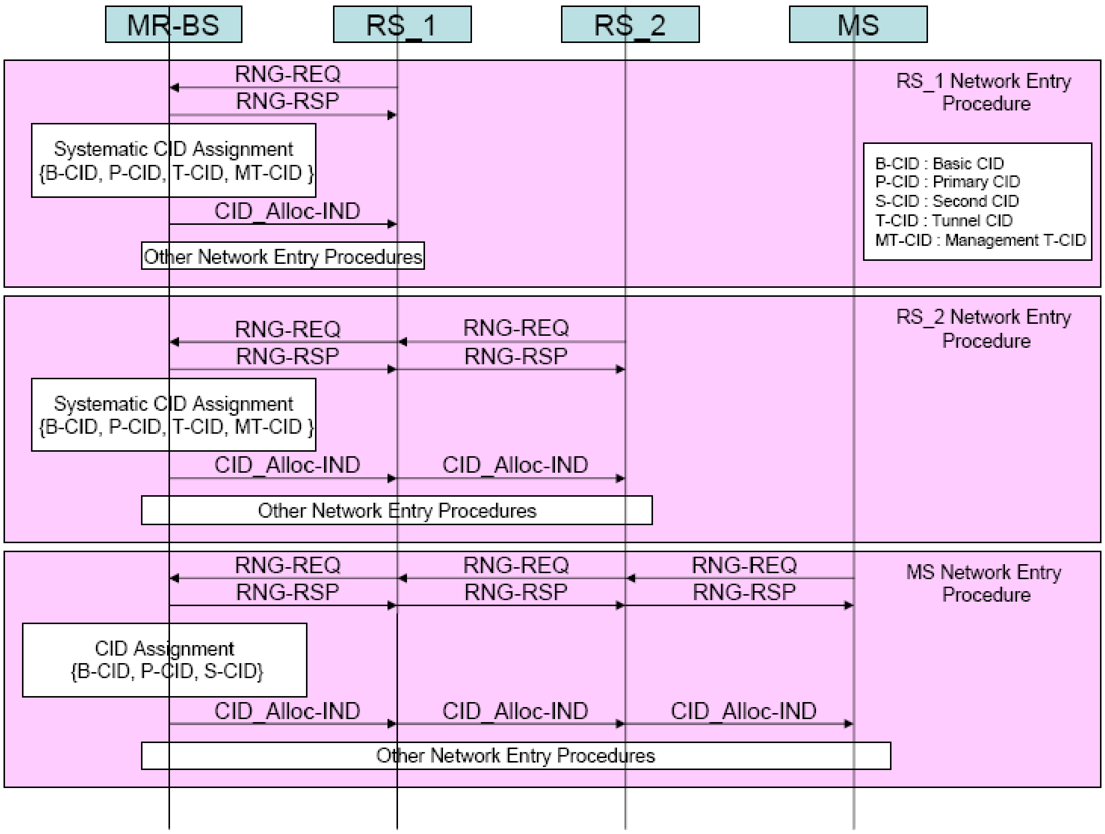


Figure 9: Illustration of embedded path management procedures.

To support dynamic topology such as MDHO and cooperative relaying, encapsulation of CIDs [6] or explicit path information can be used to perform path configuration, as described in the next or previous sections.

3.3 CID encapsulation

The MR-BS can send updates to reflect the changes in network topology from time to time. During transition stages, the length of time required to update the CID assignment is too lengthy. Furthermore, for cases such as MDHO, the CID assignment may be temporary. In this section, a solution to adapt to such changes in topology is presented.

The general idea of CID encapsulation is to have a dynamic method for temporarily changing CIDs. For example, it allows an intermediate node, who is assigned CID B, to relay a message with CID A, which the node is not directly responsible for. The following figure describes the structure of such MPDUs.

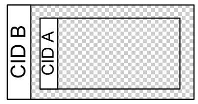


Figure 10: An inner MPDU with header CID A is encapsulated with an outer MPDU with header CID B.

The following two figures demonstrate how CID encapsulation can be used to perform changes in network topology. Node L has moved, and the BS knows to change L's parent from Node D to Node E. It is possible for this encapsulation to occur multiple times, depending on the severity of topological changes.

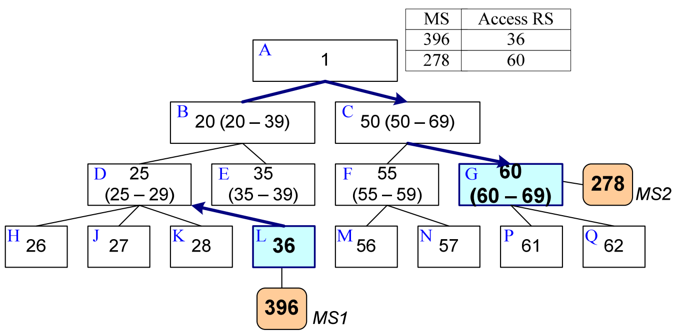
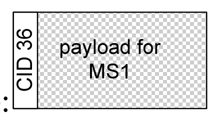


Figure 11: An example of a change in topology. The BS is aware that Node L has moved, and its parent should be changed from Node E to Node D.

Before: 

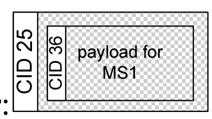
After: 

Figure 12: Before: Packet with CID 36 is routed to Node L. After: Packet with CID 25 is routed to Node E. Node E strips out inner MPDU, and retransmits a packet with CID 36.

4 Proposed Text

----------------------------Beginning of Text Changes---

[Add the following text into section 6.3.1.3]

<u>6.3.1.3.1 Addressing Scheme for Relaying</u>

<u>In the procedure of network entry and initialization for a new RS, the MR-BS may systematically assign CIDs, e.g. basic CIDs, MT-CIDs, and T-CIDs, for a RS. There are two CID assignment methods:</u>

<u>contiguous integer blocks as in Figure 6.3.1.3.X (a) and bit partition as in Figure 6.3.1.3.X (b). In the bit partition assignment, the MR-BS sets the lowest *k* bits in ascending order to RSs for RSs associated to the MR-BS directly where the maximum number of RSs the MR-BS or a RS could serve is 2^k. For other level-n RSs, which need *n* hops to reach the MR-BS, the MR-BS left shifts *k* bits of its parent CID and sets the lowest *k* bits according to the arriving sequence of the RS.</u>

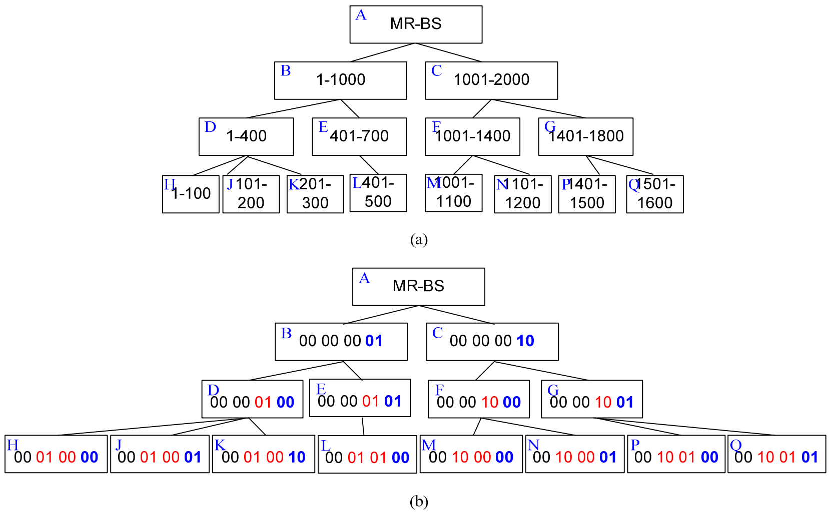


Figure 6.3.1.3.Y: CID range allocation example, (a) contiguous integer block, (b) bit partition method.

6.3.2.1 MAC header formats

[Insert the following at the end of 6.3.2.1:]

<u>The MAC header of the PDU from the MS to the MR-BS via the RS is encapsulated by the access RS, and the MAC header of the PDU from the MR-BS to the MS via the RS is decapsulated by the access RS.</u>

[Change the text in Table 4 as indicated:]

Table 4 – MAC header format

Syntax	Size	Notes
MAC Header() {		
HT	1 bit	0 = Generic MAC header 1 = Bandwidth request header
EC	1 bit	If HT = 1, EC = 0
if (HT == 0) {		

Type	6 bits	
Reserved	1 bit	Shall be set to zero
CI	1 bit	
EKS	2 bits	
CE	1 bit	0 – no CID encapsulation 1 – CID encapsulation is used
LEN	11 bits	
}		
Else {		
Type	3 bits	
BR	19 bits	
}		
CID	16 bits	
HCS	8 bits	
}		

6.3.2.3 MAC management messages

[Insert the following into table 14]

Table 14 – MAC Management messages

Type	Message name	Message description	connection
Xx	CID_ALLOC-IND	CID allocation message	Basic

Add the following text at the end of 6.3.2.3.10:

In multi-hop relay network, a DSA-REQ is also sent by MR-BS to populate the path information to every RS on the path and/or distribute the binding information between connections and a selected path. The MR-BS shall generate DSA-REQs in the form shown in Table T38. When a RS receives a DSA-REQ and it is not the last hop on the relay path, it shall also generate a DSA-REQ in the form shown in Table T38 and sends it to the next RS on the path.

The DSA-REQ message may contain the following TLVs:

Path Addition (see 11.21.1)
Specification of the path addition operations
Path CID Binding Update (see 11.21.2)
Specification of the path/cid binding operations including adding the binding between CIDs to the specific path.

The DSA-REQ shall contain the following TLVs:

HMAC/CMAC Tuple (see 11.1.2)
The HMAC/CMAC Tuple attribute contains a keyed message digest (to authenticate the sender). The HMAC Tuple attribute shall be the final attribute in the DSA message's attribute list.

Add the following text at the end of 6.3.2.3.11:

In multi-hop relay network, a DSA-RSP is also sent by a RS to confirm the path management operation requested in the correspondent DSA-REQ. The access RS on the last hop on a specific path should generate the DSA-RSP in the form shown in Table T39-1. When a RS receives a DSA-RSP, it shall update the confirmation code and generate a DSA-RSP in the form shown in Table T39-1 and sends it to the previous RS on the path.

Table 39-1 – DSA-RSP message format

Syntax	Size	Notes
DSA-RSP() {		
Management Message Type = 12	8 bits	
Transaction ID	16bits	
PM Confirmation Code	8 bits	
TLV Encoded Information	*Variable*	TLV specific
}		

Parameters shall be as follows:

Transaction ID
Transaction ID from corresponding DSA-REQ
PM Confirmation Code (see 11.21.8)
The appropriate Path Management Confirmation Code for the entire correspondent DSA-REQ.

The DSA-RSP shall contain the following TLVs:

HMAC/CMAC Tuple (see 11.1.2)

The HMAC/CMAC Tuple attribute contains a keyed message digest (to authenticate the sender). The HMAC Tuple attribute shall be the final attribute in the DSA message's attribute list.

Add the following text at the end of 6.3.2.3.13:

In multi-hop relay network, a DSC-REQ is also sent by MR-BS to update the binding between CIDs to a specified path, or to distribute the updated service flow parameter for a connection that is bound to the specified path. The MR-BS shall generate DSC-REQs in the form shown in Table T41. When a RS receives a DSC-REQ and it is not the last hop on the relay path, it shall also generate a DSC-REQ in the form shown in Table T38 and sends it to the next RS on the path.

The DSC-REQ message may contain the following TLVs:

Path CID Binding Update (see 11.21.2)
Specification of the path/cid binding operations including changing of service flow parameter of the CIDs bound to the specific path.

The DSC-REQ shall contain the following TLVs:

HMAC/CMAC Tuple (see 11.1.2)
The HMAC/CMAC Tuple attribute contains a keyed message digest (to authenticate the sender). The HMAC Tuple attribute shall be the final attribute in the DSC message's attribute list.

Add the following text at the end of 6.3.2.3.14:

In multi-hop relay network, a DSC-RSP is also sent by a RS to confirm the path management operation requested in the correspondent DSC-REQ. The access RS on the last hop on a specific path should generate the DSC-RSP in the form shown in Table T42-1. When a RS receives a DSC-RSP, it shall update the confirmation code and generate a DSC-RSP in the form shown in Table T42-1 and sends it to the previous RS on the path.

Table 42-1 – DSC-RSP message format

Syntax	Size	Notes
DSC-RSP() {		
Management Message Type = 12	8 bits	
Transaction ID	16bits	
PM Confirmation Code	8 bits	
TLV Encoded Information	*Variable*	TLV specific

}		

Parameters shall be as follows:

Transaction ID
Transaction ID from corresponding DSA-REQ
PM Confirmation Code (see 11.21.8)
The appropriate Path Management Confirmation Code for the entire correspondent DSA-REQ.

The DSC-RSP shall contain the following TLVs:

HMAC/CMAC Tuple (see 11.1.2)
The HMAC/CMAC Tuple attribute contains a keyed message digest (to authenticate the sender). The HMAC Tuple attribute shall be the final attribute in the DSA message's attribute list.

Add the following text at the end of 6.3.2.3.15:

In multi-hop relay network, a DSD-REQ is also sent by MR-BS to remove a path and/or remove the binding between connections and a selected path. The MR-BS shall generate DSD-REQs in the form shown in Table T44. When a RS receives a DSD-REQ and it is not the last hop on the relay path, it shall also generate a DSD-REQ in the form shown in Table T44 and sends it to the next RS on the path. The DSD-REQ message may contain the following TLVs:

Path ID (see section 11.21.4)
Specification of the path to be completely removed
Path CID Binding Removal (see 11.21.3)
Specification of the path/cid binding operations including removing the binding between CIDs to the specific path.

The DSD-REQ shall contain the following TLVs:

HMAC/CMAC Tuple (see 11.1.2)
The HMAC/CMAC Tuple attribute contains a keyed message digest (to authenticate the sender). The HMAC Tuple attribute shall be the final attribute in the DSD message's attribute list.

Add the following text at the end of 6.3.2.3.15:

In multi-hop relay network, a DSD-RSP is also sent by a RS to confirm the path management operation requested in the correspondent DSD-REQ. The access RS on the last hop on a specific path should generate the

DSD-RSP in the form shown in Table T44-1. When a RS receives a DSD-RSP, it shall update the confirmation code and generate a DSD-RSP in the form shown in Table T44-1 and sends it to the previous RS on the path.

Table 44-1 – DSD-RSP message format

Syntax	Size	Notes
DSD-RSP() {		
Management Message Type = 12	8 bits	
Transaction ID	16bits	
PM Confirmation Code	8 bits	
TLV Encoded Information	*Variable*	TLV specific
}		

Parameters shall be as follows:

Transaction ID
Transaction ID from corresponding DSA-REQ
PM Confirmation Code (see 11.21.8)
The appropriate Path Management Confirmation Code for the entire correspondent DSD-REQ.

The DSD-RSP shall contain the following TLVs:

HMAC/CMAC Tuple (see 11.1.2)
The HMAC/CMAC Tuple attribute contains a keyed message digest (to authenticate the sender). The HMAC Tuple attribute shall be the final attribute in the DSD message's attribute list.

[Insert the following subclause into section 6.3.2.3]

6.3.2.3.XX RS CID Allocation Indication (CID_ALLOC-IND) message

The CID_ALLOC-IND message shall be transmitted by the MR-BS to the RS during network entry/re-entry processes. When the network topology is changed or CID (re-)allocation is required, the MR-BS shall also transmit this message to related RSs to update CIDs. Upon receiving CID_ALLOC-IND, the RS shall (re-)configure CID allocation accordingly. The message format is shown in Table XX.

Table XX CID_ALLOC-IND message format

Syntax	Size	Note
CID_ALLOC-IND_Message_Format() {		
Management Message Type (TBD)	8 bits	
CID_Alloc_method	3 bits	0 : contiguous method

		1 : bit partition method 2-7 : reserved
CID_type	3 bits	0: basic CID 1: primary CID 2: T-CID 3: MT-CID 4-7: reserved
If (CID_Alloc_method = =0) {		
Start number of CID	16 bits	Starting point of the CID number
End number of CID	16 bits	End point of the CID number
}		
If (CID_Alloc_method = =1) {		
New CID for the RS	16 bits	
Hop count	8 bits	The new hop count of the RS to the MR-BS
K_Code	8 bits	The new maximum number of subordinate RSs that a RS could have
}		
}		

[Insert the followings in sections of 6.3.25]

6.3.25 Path Management for Relay

Based on the topology information obtained from topology discovery or update process, MR-BS makes centralized calculation for the path between MR-BS and an access RS for both uplink and downlink direction. The path creation is subject to the constraints such as the availability of radio resource, radio quality of the link, load condition of a RS, etc. The path calculation algorithm is out of scope of this specification.

Depending on the complexity of network topology, either embedded path management or explicit path management may be used.

6.3.25.1 Embedded Path Management for Relay

When the systematic CID allocation is used, the MR-BS shall update the CID range assigned to its subordinate RSs via the CID_ALLOC-IND message. The MR-BS shall be responsible for managing the entire

CID allocations within the MR-cell. By assigning systematic CIDs to RSs, the MR-BS already specifies the relay routing path of the connection.

When a relay station receives a MAC PDU with the CE field set in the MAC header, it shall remove the current MAC header and forward the payload as the new PDU. If CRC is used, the BS calculates the CRC for each packet.

6.3.25.2 Explicit Path Management for Relay

After MR-BS discovers the topology between a newly attached MS or RS and itself, or detects a topology update due to events such as mobility, MR-BS may remove an old path, establish a new path and inform the new path information to all the RSs on the path.

When connections are established or removed, MR-BS may distribute the mapping information between the connection and the path to all the RSs on the path. The connection could be a regular connection established for a MS (as defined in 802.16e) or a connection established for a RS (e.g., basic/primary management CID and tunnel connection). The path management procedures are specified below.

6.3.25.2.1 Path Establishment, Removal and Update

When a new path is discovered and calculated as specified in section 6.3.25.2, MR-BS sends a path establishment command to distribute the path information to all the RSs on that path by sending a DSA*-REQ message. The explicit path information and an uniquely assigned path id are included. The CIDs to be routed on this path and their associated service flow parameters are also included for path/CID binding operation. If DSA*-REQ is issued from an access RS, the explicit path path-ID and/or associated CIDs are included in the DSA-RSP message sent from the MR-BS.

If the MR-BS decides to remove an existing path (e.g. after an MRS handover), it sends DSD*-REQ message with the Path-ID. The RSs receiving the DSD*-REQ message should remove all the information related to the path, including the entry in the routing table, the binding between CIDs to the path, etc.

Upon receiving the DSA/DSD*-REQ, the RS performs the operation as requested in the message, and then sends the request to the next RS on the path. The next hop on the path is obtained from the explicit path information included in the DSA/DSD*-REQ message, or derived from the path information obtained from previous operation. Such process is repeated until the last RS on the path is reached. The last RS on the path then replies with an DSA/DSD*-RSP to the previous hop to report its operation status. The previous hop will update the response with its own operation status and forwards the DSA/DSD*-RSP to its previous hop on the path, until it reaches the MR-BS.

The MR-BS may aggregate multiple path management commands into one DSA*/DSD*-REQ message to save bandwidth. When the paths of different path management commands in the same message divaricates in an RS, the RS separates the path establishment or removal commands into different messages and transmits them to the appropriate next-hop RSs.

The MR-BS may establish the path in the following ways:

- Distributing the complete path information (including ids of all the RSs on the path) to the RSs on path
- Instructing the RSs how to generate the detailed path information based on the existing path. With this approach, each RS on the path forwards the instruction to the next hop RS on the path, as long as the next hop is aware of the existing path information; otherwise, the RS needs to generate the complete or remaining path information and send to the next hop RS. In the second case, when a RS receives a DSA*/DSD*-REQ message, if there are further hops on the path updated by the DSA*/DSD*-REQ message, the RS will regenerate a DSA*/DSD*-REQ message by deleting unused information in the old one, and send it to the next hop RS.

6.3.25.2.2 CID to Path Binding

A routing table that contains the mapping between a CID and one or more given paths needs to be updated when a new tunnel (identified by a Tunnel CID) is generated between the MR-BS and an access RS, or when a new connection (identified by a individual CID) is established for an RS or MS and the new connection is not put into a tunnel. The MR-BS selects one or more path to carry the traffic for the new connection, and informs all the RSs on the path of the binding between the path id and the supported CIDs by sending a DSA*-REQ message to all the RSs on the specified path. Such DSA*-REQ message contains the CIDs of the connections that will be routed through the specified path, the path-id and optionally the SFID and the service flow parameter for the connection. If the connection is a tunnel connection, the service flow is the aggregate service flow parameter for all the connections put into the tunnel.

When a RS on the path receives such DSA*-REQ message, it retrieves the CIDs and path id information and builds up the routing table, which will be used to route the traffic in the future for the specified CIDs. If the SFID and the QoS requirement are also present for certain connection, the RS saves them for scheduling the traffic for the specified CID. This process is repeated until the last RS along the path is reached. The last access RS then replies with the DSA-RSP.

If the MR-BS decides to cancel an existing binding between a path and one or more CID (e.g., after MS or MRS handover to another RS, or MS deregistration, or service flow deletion), it sends a DSD*-REQ message with the Path-Id and the affected CIDs to the associated RSs. The RSs receiving such DSD*-REQ should remove the record of the correspondent mapping in the routing table as well as the other context of the affected MS or MRS.

If the MR-BS decides to update the service flow parameter associated with a connection along a specific path, it sends a DSC*-REQ message with Path-ID together with the updated service flow parameter. As an example, as new transport connections are included into a tunnel, the MR-BS needs to recalculate the aggregate QoS for the tunnel and distribute the new service flow parameter to every RS on the path by sending a DSC*-REQ message.

Upon receiving a DSA*/DSC*/DSD*-REQ, the RS performs the operation as requested in the message, and then sends the request to the next RS on the path. The next hop on the path is obtained from the explicit path information included in message if available, or derived from the path information obtained from previous operation. Such process is repeated until the last RS on the path is reached. The last RS on the path then replies with an DSA*/DSC*/DSD*-RSP to the previous hop to report its operation status. The previous hop will update the response with its own operation status and forwards the DSA*/DSC*/DSD*-RSP to its previous hop on the path, until it reaches the MR-BS.

Multiple DSA*-REQ can be sent for the same CID to establish multiple paths to MS. This can be utilized for dynamic switching of traffic among multiple paths based on traffic condition or in case of macro diversity handoff.

The MR-BS may aggregate multiple CID to path binding commands in one DSx*-REQ message to save bandwidth. In addition, when a path is established for one or more connection, the CID to path binding/unbinding procedure can be conducted together with path establishment procedure by sending a single DSA*-REQ or DSD*-REQ to save bandwidth.

Insert new subclause 11.21

11.21 Path Management message encodings

The TLV encodings defined in this section are specific to the path management related MAC Management messages including DSA-REQ/RSP, DSC-REQ/RSP and DSD-REQ/RSP.

11.21.1 Path-Addition TLV

This field contains a compound attribute whose subattributes identifies Path ID, the direction of the path, the number of RSs on the path and an ordered list of RSs on the path as listed in Table S1.

Type	Length	Value	Scope
TBD	variable	Compound	DSA-REQ

Table S1 – Path-Addition Subattributes

Attribute	Content
Path ID	The ID of the path
Path Direction	The direction of the path
Existing Path ID	The ID of an existing path that is used to derive the information of the new path
Number of RS	The number of RSs in the ordered list of RSs
Ordered list of RSs	An ordered list of the basic CID of RSs that identifies the path in the

	case of non-presence of the Existing Path ID; or a ordered list of RSs that identifies the difference between the new path and the existing path in the case of presence of the Existing Path ID

11.21.2 Path-CID-Binding-Update TLV

This field contains a compound attribute whose subattributes identifies Path ID, the CIDs bound to the specified path, the service flow parameter associated with the CIDs as listed in Table S2.

Type	Length	Value	Scope
TBD	variable	Compound	DSA-REQ

Table S2 – Path-CID-Binding-Addition Subattributes

Attribute	Content
Path ID	The ID of the path
Number of CIDs	The number of CIDs bound to the path
List of CIDs	An list of CIDs that are bound to the path
List of service flow parameters	An list of service flow parameters associated with the CIDs bound to the path

11.21.3 Path-CID-Binding-Removal TLV

This field contains a compound attribute whose subattributes identifies Path ID, the CIDs bound to the specified path to be removed as listed in Table S3.

Type	Length	Value	Scope
TBD	variable	Compound	DSD-REQ

Table S3 – Path-CID-Binding-Removal Subattributes

Attribute	Content
Path ID	The ID of the path
Number of CIDs	The number of CIDs bound to the path to be removed
List of CIDs	An list of CIDs to be removed from the binding to the path

11.21.4 Path-ID TLV

This filed contains the ID of a path between MR-BS and a RS.

Type	Length	Value	Scope
TBD	TBD	ID of path	DSx-REQ, DSx-RSP, DSx-ACK

11.21.5 Path-Direction TLV

This field specifies the direction of the path, which could be uplink only, downlink only or both uplink and downlink.

Type	Length	Value	Scope
TBD	1	0 – uplink 1- downlink 2 – both uplink and downlink	DSA-REQ

11.21.6 Number-of-RS TLV

This field specifies the number of intermediate RSs on the path.

Type	Length	Value	Scope
TBD	1	Number of RSs on the path	DSA-REQ

11.21.7 Ordered-List-of-RS TLV

This field contains an ordered list of intermediate RSs on the path in the case of non-presence of the Existing Path ID; or a ordered list of RSs that identifies the difference between the new path and the existing path in the case of presence of the Existing Path ID. Note that if the Path Direction indicates for both uplink and downlink, then the ordered list of RS is for the downlink direction. The ordered list of RS for the uplink can be obtained by reverse the ordered list.

Type	Length	Value	Scope
TBD	Number of RS x 2bytes	An ordered list of basic CID of RSs on a path; if Path Direction == 2, then the ordered list of RS on the path is for the downlink direction	DSA-REQ

11.21.7 PM-Confirmation-Code TLV

TBD

11.21.8 Existing-Path-ID TLV

This filed contains the ID of a path between MR-BS and a RS.

Type	Length	Value	Scope
TBD	TBD	ID of an existing path	DSA-REQ

5 References

[1] IEEE C802.16j-06/004r1, “Recommendations on IEEE 802.16j”.

[2] IEEE 802.16-2004, “Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems”.

[3] IEEE 802.16e-2005, “Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands *and* Corrigendum 1”.

[4] IEEE C802.16j-06/014r1, “Harmonized definitions and terminology for 802.16j Mobile Multihop Relay"

[5] IEEE C802.16j-06/015, “Harmonized Contribution on 802.16j (Mobile Multihop Relay) Usage Models"

[6] IEEE C802.16j-07-126, “Routing with CID Encapsulation”

[8] IEEE C802.16j-06/274, “Proposal on addresses, identifiers and types of connections for 802.16j”.

[9] IEEE C802.16j-06/254, “Fast Connection Establishment and Maintenance with Relays”.

[10] IEEE C802.16j-06/170, “Connection Identification and Transmission for Relay Support”

[11] IEEE C802.16j-06_026r2.pdf, Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, Multihop Relay Specification

[12] C802.16j-07_032r1.pdf Topology Discovery in Multi-hop Relay System, Nokia, Huawei and Fujitsu

[13] C802.16j-06/274r3.pdf, Proposal on address, identifiers and types of connections for 802.16j, Intel et. al.

[14] C802.16j-07/093.pdf, DSx message extension for Constraint-Based routing and CID/path binding, Nortel