CHAPTER II

THEORIES AND LITERATURE REVIEWS

2.1 IEEE 802.16j Multi-hop Relay Network

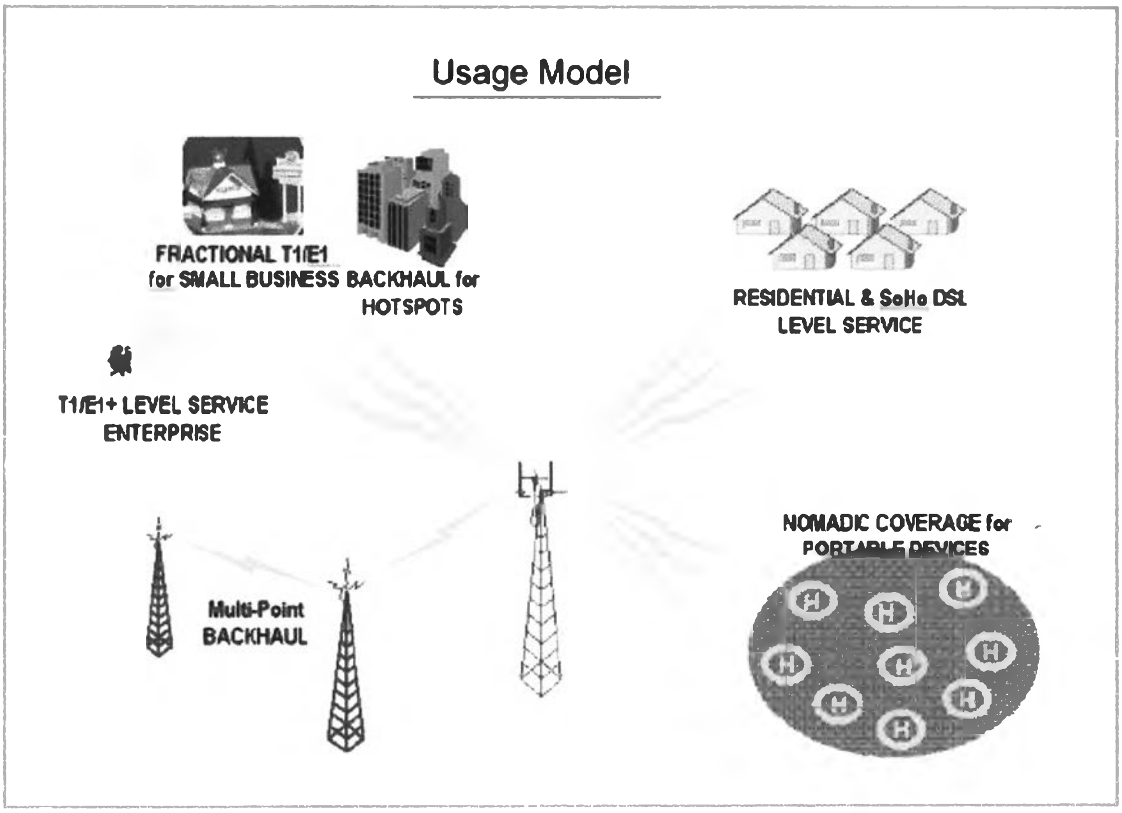


Figure 2.1: The usage model of IEEE 802.16j multi-hop network.

There are a number of different types of multi-hop wireless networks, notably ad hoc networks, sensor networks, and wireless mesh networks. Each of these network types has different characteristics (mobility rates, power constraints, scale, form factor, and so on) which result in different system designs (e.g., routing protocols, medium access control mechanisms). Additionally, another type of multi-hop wireless network is based on the relay architecture.

Relay-based systems typically comprise of small-form factor low-cost relays, which are associated with specific base stations (BSs). The relays can be used to extend the coverage area of a BS and/or increase the capacity of a wireless access system, called BS-RS system. Typically, it is envisaged that they could be used in the early stages of network rollout to provide coverage to a large area at lower cost than a BS-only solution. In addition, they are implemented to increase capacity of the high density area networks as well as coverage to coverage holes such as areas in the shadows of buildings.

Table 2.1: Comparison of 802.16j and 802.16e-2005 capabilities.

	IEEE 802.16e-2005	IEEE 802.16j
Topology	Point-to-Multi-Point only	Tree structure (Point-to-Multi-Point compatible, not ad hoc nor mesh)
Hops	Single-hop	Multi-hop
Traffic aggregation	No	Yes, over multi-hop path
System capacity	Lower	Higher within BS coverage area
Coverage	Lower	Higher
Cost	Higher	Lower
Mobility support	Yes	Yes
PHY support	OFDMA	OFDMA extension
Legacy 802.16e-2005 station	-	Backward compatibility

BS-RS system is the most applicable in network operator contexts, where the operator plans and deploys a wireless access network operating in licensed spectrum. Since the RS over 3 hops is compromise the throughput and delay, typically, they are characterized by tree-based routing in which end terminals connect to the BS over short routes (2–3 hops). However, even within this scope, many different designs are possible, and there are many works to be processed to understand the most appropriate use cases for different designs.

IEEE 802.16j is an amendment to the IEEE 802.16 standard that enables the functionalities of interoperable RSs and BSs. In this section, the key system features of the IEEE 802.16j MR network are summarized.

2.2 IEEE 802.16j Standard

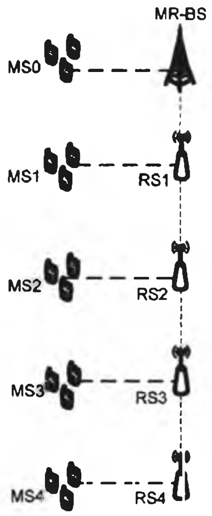


Figure 2.2: IEEE 802.16j multi-hop relay network.

The basic system architecture considered by IEEE 802.16j is shown in Figure 2.2, where two types of radio links are identified: an access link, and a relay link. BS that is capable of supporting the multi-hop relay is called the MR-BS. The access link is the radio link that originates or terminates at an MS, which is either a downlink (DL) or an uplink (UL), defined in IEEE 802.16-2004. The relay link is the radio link between an MR-BS and an RS, or between a pair of RSs; this link can be either UL or DL.

Each DL sub-frame and UL sub-frame are divided into an access zone and a relay zone. The DL/UL access zone is a portion of DL/UL sub-frame used for access-link transmission, and the DL/UL relay zone is a portion of a DL/UL sub-frame used for relay-link transmission. Note that each DL/UL sub-frame may have more than one relay zones.

Table 2.2: Comparison between transparent and non-transparent modes of operation.

	Transparent RS	Non-transparent RS
Coverage extension	No	Yes
Number of hops	2	2 or more
Inter RS cell interference	None	High
Performance	Within BS coverage: high Outer BS coverage: none	Within BS coverage: high Outer BS coverage: medium
RS cost	Low	High
Scheduling	Centralized scheduling only	Centralized or distributed scheduling

In order to enable RS operations with no change on the legacy MS specification, two types of RSs have been defined: non-transparent RS, and transparent RS. The non-transparent RS acts as a BS sector; therefore, the MR-BS has to assign a preamble index to each RS, and the RS transmits its own preamble, FCH (Frame Control Header) and MAP over the access zone.

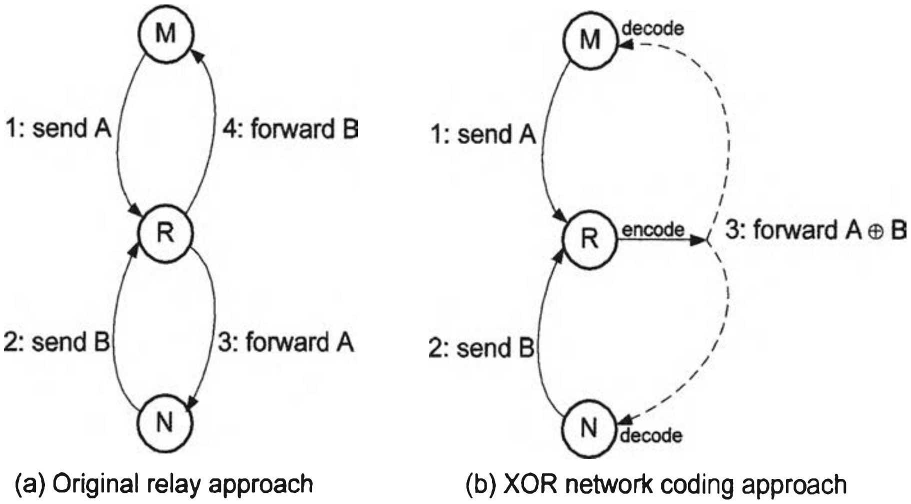
When an MS communicates with a non-transparent RS, it will receive a preamble, FCH, MAP and a data burst from the RS. On the other hand, if an MS communicates with a transparent RS, it will receive the data burst from the RS but receive the preamble, FCH and MAP from the MR-BS. Therefore, the transparent RS has to be centralized controlled by the MR-BS to transmit/receive the data burst over the designated sub-channels and symbol times. Note that the MR-BS and multiple RSs can serve a particular MS simultaneously so as to increase the received signal quality and to obtain cooperative diversity gain.

Consider the non-transparent RS, it can generate its own FCH and MAP without the instruction from the MR-BS. So the de-centralized control can be performed to reduce the messaging delay and the messaging overhead over relay links. Meanwhile, a group of RSs may transmit the same preamble, FCH, MAP and the data burst; these RSs will act as a single virtual station from the MS's point of view. In this situation, the MS will not initiate the handover procedure when moving between the grouped RSs. Moreover, the cooperative diversity gain will be obtained.

In this research, a Network Coding-Based Relay scheme, called NC-BR, and the corresponding frame structure are proposed. The proposed solution allows RS to combine two sets of data in the wireless backhaul using the XOR operation, and transmit it in a single transmission instead of two. Consequently, the throughput and the delay are improved. In addition, this approach reduces both of the number of transmissions from RSs, and the number of idle periods that caused by the limitation of signal interference which is considered as a waste. So, the throughput and the delay of the network can be significantly improved using the proposed method.

2.3 Benefit of relay station

RS aims to enhance the coverage and throughput of IEEE 802.16e per user. Compared with a base station (BS), RS needs no wire-line backhaul. Additionally, it requires much lower hardware complexity than BS. Thus, using RSs can notably reduce the deployment cost of the system. Unfortunately, there are some tradeoff in the case of the multi-hop relay network since Subscriber Stations (SSs) who connect to the RS with 2 or more hops away from the MR-BS, is suffered from the bottle-neck of the multi-hop, end-to-end throughput degradation and increasing of end-to-end delay. By applying proposed NC-BR, the network coding helps RSs transmit more data in the relay link. Consequently, it relieves both optimal offered load point and bottle-neck point. More details are described in the following section.



(a) Original relay approach (b) XOR network coding approach

Figure 2.3: Example of XOR network coding.

2.4 XOR Network Coding

The network coding is first introduced in [96]. It extends the traditional routing paradigm by allowing interior network nodes to perform coding operations rather than just routing. It has been shown in [97-149] that network coding can help a communication system to achieve a multicast capacity in lossless networks. Network Coding has recently been applied to wireless networks [150-184].

The application of network coding in wireless networks is facilitated by the broadcast nature of wireless medium. The best studied case is a three-node topology where two terminals wish to exchange mutual information through an intermediate relay. By applying network coding and utilizing broadcast capability at the relay, [97] demonstrated that the time it takes to exchange a message can be reduced from four time slots of native transmission to three. [100, 130-148] extend this idea to multiple terminals and implemented a practical protocol which used overhearing and opportunistic packet combining. Their works also showed remarkable throughput increase brought by network coding and physical-layer broadcast.

Figure 2.3 shows a simple example of the XOR network coding. Where in Figure 2.3 (a) represents the current relay approach, node M and N want to exchange a pair of packets via a router R. Node M sends its packet to the router R, which forwards it to node N, and node N sends its packet to the router R, which forwards it to node M. This process requires four transmissions. With a network coding relay approach, M and N send their respective packets to the router R, which XORs the two packets and broadcasts the XOR-ed version. M and N can obtain each other's packets by XOR-ing again with their own packets. This process takes three transmissions instead of four. Saved transmissions can be used to send new data, increasing the wireless throughput and also decrease the end-to-end delay.

2.5 Dijkstra algorithm

According to the congestion situation mentioned previously, finding the shortest and series of alternative paths from point-to-point in a large weighted graph is still required large computation time based on the complexity of the best existing algorithm. For this reason, various applications may not be served properly in the SPP. For example, applications that require high throughputs and low delay in a mobile infrastructure, such as the shortest travel time calculation in the communication network problem. Objects over the mobile environment need to transmit between locations over the communication channels. Thus, the transmission policy of an object must focus on the transmission time rather than the travelling distances. Therefore, the meaning of the SPP is referring to the smallest time count spending from the starting point to the required destination.

The communication network problem, particularly a road network problem, is a critical problem in many big cities, such as Bangkok and California. However, the development of a geo-positing system (GPS) helps transporters to find the best path from the point-to-point; information from mobile units comprised with GPS was gathered by a datacenter via a wireless environment. The proliferation of metropolitan wireless communication (802.16) and GPS, have created an environment that mobile unit can remotely exchange information with a datacenter that plays role in collecting and categorizing a data set from a mobile unit. Thus, the datacenter can solve a SPP problem by organizing and calculating submitted data. Transporters then have the high accuracy knowledge for SPP.

[2] is the classic of all label setting algorithms. The Dijkstra algorithm builds a shortest path tree with root S and stops when the shortest path to node T has been found. It visits all nodes which can be reached from S over a path with cost smaller than $d(T)$. In a typical road network of a city, the nodes are uniformly distributed in the plane and nearby nodes are connected by edges. In this case, the number of visiting nodes grows with the square of $d(T)$, and the number of visiting edges grows linearly with respect to the number of visiting nodes. Thus, the time complexity of this algorithm is $O(|V|^2+|E|)$. In fact, using a higher sophisticated implementation of the min-priority queue, an improvement to $O(V \log V + E)$ is possible.

2.6 A* algorithm

From the existing method, the weight of an edge connects between a pair of vertices in the graph from the digital map was an actual distance [1, 3-6]. The result of this method is the shortest-distance rather than the fastest time. Thus, a position-aware shortest-path algorithm can increase the efficiency. A resent method [4, 7-8] has made conversion from the travel-distance to the travel-time using speed limited of each transportation path, or actual travel-time of an individual mobile unit that submitted to the datacenter.

The A* algorithm [1, 3-8] takes the advantage of the fact that a lower bound for the traveling time from any nodes to the destination node can be calculated. This allows the directed search for the shortest path to heading for the direction of the destination node. The A* algorithm is very similar to the Dijkstra algorithm. Only the selection of the next node to be processed is based on the Euclidean distance value and distance among nodes. Thus, the time complexity of this algorithm is depended on the heuristic. In the worst case, the number of expanded nodes is exponential in the length of the solution (the shortest path), but it is polynomial when the heuristic function h meets the following condition: $|h(x) - h^*(x)| \leq O(\log h^*(x))$ where h^* is the optimal heuristic.

2.7 Distributed Shortest Path Algorithm for Hierarchically Clustered Data Networks

The Hierarchically Clustered Data Networks with *n* nodes can be classified into two categories: single origin shortest path problem (SOSP problem), and multiple origins shortest path problem (MOSP problem). Under the SOSP problem, the distributed version of the SOSP algorithm has the time complexity of $O(\log(n))$ [185-188], for parallel and distributed shortest path algorithm which is less than the general distributed shortest path algorithm with its time complexity is $O(\log^2(n))$ [189-222].

On the other hand, the MOSP algorithm minimizes the resources usages in each shortest path computations, including processors and communication links. So, the massive parallelization can be achieved. The parallel time complexity of the MOSP algorithm is $O(m\log(n))$, which is much less than the time complexity of $(M\log^2(n))$ of the non-parallel distributed shortest path algorithm. Here, *M* is the number of the shortest paths to be computed and *m* is a positive number related to the network complexity and *m* is much smaller than *M*. Moreover, the value of *m* is, almost, a constant when the network size increases [9-10, 185-222]. This method works well on the network assumption that all nodes or vertices have their own processors.

Table 2.3: Comparisons of related shortest-part finding algorithms.

Title	Description (Time complexity)	Drawbacks
Dijkstra algorithm.	The Dijkstra algorithm is the classic of all label setting algorithms. The Dijkstra algorithm builds a shortest path tree with root *S* and stops when the shortest path to node *T* has been found. It visits all nodes which can be reached from *S* over a path with cost smaller than *d(T)*. In a typical road network of a city, the nodes are uniformly distributed in the plane and nearby nodes are connected by	the number of visiting nodes grows with the square of *d*(*T*), and the number of visiting edges grows linearly with respect to the number of visiting

	edges. In this case the number of visiting nodes grows with the square of $d(T)$, and the number of visiting edges grows linearly with respect to the number of visiting nodes. Thus, the time complexity of this algorithm is $O(\|V\|^2+\|E\|)$. In fact, by using a more sophisticated implementation of the min-priority queue, an improvement to $O(V \log V + E)$ is possible.	nodes.
A* algorithm.	The A* algorithm takes advantage of the fact that a lower bound for the traveling time from any nodes to the destination node can be calculated. This allows the directed search for the shortest path in the direction of the destination node. The A* algorithm is very similar to the Dijkstra algorithm. Only the selection of the next node to be processed is based on the Euclidean distance value and distance among nodes. Thus, the time complexity of this algorithm is depends on the heuristic. In the worst case, the number of nodes expanded is exponential in the length of the solution (the shortest path), but it is polynomial when the heuristic function h meets the following condition: $\|h(x) - h^*(x)\| \leq O(\log h^*(x))$ where h^* is the optimal heuristic.	In the worst case, the number of nodes expanded is exponential in the length of the solution (the shortest path), but it is polynomial when the heuristic function h meets the following condition: $\|h(x) - h^*(x)\| \leq O(\log h^*(x))$ where h^* is the optimal heuristic.
Distributed Shortest Path Algorithm for	SOSP problem, the distributed version of the SOSP algorithm has the time complexity of $O(\log(n))$	This method worked well on network

Hierarchically Clustered Data Networks in SOSP problem.		assumption that every nodes or vertices have their own processor. But in case of only mobile unit have processor; more processing time and complexity are needed.
Parallel and distributed shortest path algorithm.	The MOSP algorithm minimizes the resources usages in each shortest path computations, including processors and communication links. So, the massive parallelization can be achieved. The parallel time complexity of the MOSP algorithm is $O(m\log(n))$,which is much less than the time complexity of $(M\log^2(n))$ of the non-parallel distributed shortest path algorithm. Here, M is the number of the shortest paths to be computed and m is a positive number related to the network situations and m is much smaller than M. Moreover, the value of m is, almost, a constant when the network size increases	This method worked well on network assumption that every nodes or vertices have their own processor. But in case of only mobile unit have processor; more processing time and complexity are needed.

The proposed method on the data transfer method and the path finding algorithm will be presented in the next chapter.