私立東海大學資訊工程與科學研究所

碩士論文

指導教授：呂芳懌

鏈狀拓撲之無線網狀網路的最大傳輸容量之研究

A Study on Maximum Transmission Capacity in Chain-Topology Wireless Mesh Networks

研究生：黃郁婷

中華民國 九十七年七月

摘要

本研究提出一個計算無線網狀網路中一條路徑最大容量的數學模型，包含訊號干擾、隱藏節點及 STDMA。亦發展一個稱作 Path_C 的演算法，用來決定一個節點在此拓撲下能用最大資料傳輸率、限制其速度或禁止其傳送。此模型跟演算法應用在鏈狀拓撲，其是一個普遍網路的基本結構。另提出 urgent path 的想法，主要為傳送緊急資料，如即時資料。當網路雍塞時，可透過此路徑傳送緊急通知的封包。實驗結果顯示，用 ns2 實驗之結果跟數學模型算出的結果錯誤率相差在 5%以下。

關鍵字：鏈狀拓撲、干擾、多跳躍、空間再利用、最大容量、無線網狀網路、隱藏節點

Abstract

Basically, a general-topology WMN consists of chain-topology paths (i.e., linear paths), and articulation nodes (i.e., nodes that each joins at least two linear paths). One should clearly realize characteristics of chain-topology links, before he/she can effectively explore behaviors of a mesh network. In the paper, we analyze details of signal interference, hidden-node problem and STDMA time slots among nodes of a chain-topology path WMN, and propose a mathematical model that can more accurately calculate maximum throughput of the path WMN. An algorithm, named Path_C, is also developed to determine whether a node in such a path WMN can transmit data with its maximum data rate or not, or should be prohibited to send data to next node. Simulation results show that error rates between simulated throughput by using ns2 and calculated by deploying the mathematical model are all less than 5%. Jitters, dropping rates and delay time of a chain-topology WMN are also evaluated.

Keywords: chain topology, interference, multi-hop, spatial reuse, maximum capacity, WMN, hidden nodes

誌謝

得以順利完成這兩年的研究與論文，我首先感謝我的爸媽與家人，你們的支持是我最大的動力，感謝你們多年來的栽培與支持，讓我順利的完成學業。所有學識上的研究成果榮耀，全都獻給你們。

誠摯的感謝指導教授呂芳懌博士，老師悉心的教導使我得以接觸無線網狀網路，在研究上極盡所能的指導，引領正確的研究方向，在為人處事之道上，時時苦語建言，對學生的研究生活與成長有莫大的影響。在此，向老師致上最深切的感謝。

此外，承蒙口試委員林宜隆教授、楊朝棟教授、林熙禎教授及蕭瑞祥教授對於本論文不吝指教，並給予學生我許多寶貴的建議，能獲得你們的肯定，是莫大的殊榮，在此致上最深的謝意。

最後，感謝在這二年來的日子裡，實驗室裡共同的生活點滴，學術上的討論，並感謝學長們小 B、小新、小賴、志揚、嘉琦及同學小葛跟文晉，和耀田、宏瑋、永倫、梗延等學弟們的幫忙，在我挫折與沮喪時聆聽，在面臨困頓的關卡時陪伴，並給予我諸多關懷與鼓勵。

最後，謹以此文獻給我摯愛的雙親。

Table of Contents

List of Tables

List of Figures

Chapter 1 Introduction

Recently, wireless mesh networks (WMNs) have emerged as a key technology and started an upsurge in the wireless research. The primary advantage of these networks is their rapid deployment and easy installation. Unlike traditional WiFi networks where each access point (AP) is connected to a wired network, WMNs is multi-hop communication. Basically, a general-topology WMN consists of chain-topology WMNs (i.e., linear paths), and articulation nodes (i.e., nodes that each joins at least two linear paths). One should clearly realize characteristics of chain-topology links, before he/she can effectively explore behaviors of a mesh network. In a WMN, a node or a path segment may be shared by two or more paths [1-5], or two nodes and/or two path segments may be located too close, resulting in signal interference and poor communication quality. Then, neither maximum throughput of the WMN can be achieved, nor wireless facilities are effectively utilized. The consequence is we have to reserve much more budget to equip many more facilities and upgrade network bandwidth if a higher throughput is required. Besides the chain-topology paths in a general-topology WMN, another example of chain-topology application is a wireless link between a remote sensor and its server due to no wired link between them. Of course, electricity should be available along the wireless link so that wireless routers can be powered. Other examples include a routing path between a triggered sensor and its base station in a wireless sensor network, and a routing path between two nodes in an wireless Ad hoc environment. They are all essential in data transmission.

However, what is the maximum throughput of a multi-hop wireless network. This subject has been one of hot research topics in recent years [6-8]. Aoun and Boutaba [6] pointed out that throughput degrades as offered load exceeds network capacity, and bounding the maximum offered load at a source node helps reducing packet losses (i.e. dropping) and prevents wasting bandwidth. For example, a node at the starting point of a chain injects more packets than the subsequent nodes could

forward. These packets are eventually dropped at later nodes. The time this node spends for sending those packets decreases the delivered throughput since it prevents effective transmissions at subsequent nodes. Therefore, if we desire each node of a multi-hop wireless network achieving its maximum throughput, effectively adjusting relevant measure (e.g., inhibit other nodes or paths to transmit data) to avoid packet loss is required. Several papers have also discussed this issue. Jain, et al. [7] modeled interference using a conflict graph, and presented methods for computing upper and lower bounds on optimal throughput for a given network and workload. Jun and Sichitiu [8] considered network interference and calculated collision domain. But, both the two papers dealt with no hidden node problem [9-11] and spatial reuse [9] in collision domains.

In this study, we analyze the possible phenomena of a chain-topology link, when several nodes on the link simultaneously transmit data to their common destination node which is the end node of the link and is either an articulation node or a node that connects to a wired link (i.e., a wired hot spot [12]). We also propose an algorithm, named Path_C, which can more accurately determine whether a node in such a topology can transmit data with its maximum data rate or not than other algorithms tested can do. Based on this algorithm, we derive a mathematical model that can calculate the maximum throughput and transmission capacity of a multi-hop chain-topology WMN. With this model, we can effectively utilize bandwidth and other resources of the WMN. In the following, we use chain-topology WMN, chain-topology path, chain path and linear path interchanging.

The rest of this paper is organized as follows. Chapter 2 introduces the background and related work. In chapter 3, we derive equations to compute maximum throughputs for different nodes in a chain-topology WMN, and describe how algorithm Path_C and the mathematic model work. Experimental results and discussion are presented in chapter 4. Chapter 5 concludes this paper and addresses our future research.

Chapter 2 Background and Related Work

2.1 Wireless Mesh Network

A WMN as shown in Figure 1 is composed of a set of mesh routers, namely Transient Access Points (TAPs), and one or many wired hot spots (WHSs) [12]. TAPs relay user data to Wired hot spots through which TAPs can connect to the Internet. Increasingly, WMNs are widely used to connect devices in the environments where wired network infrastructures do not exist, or are expensive or hard to deploy.

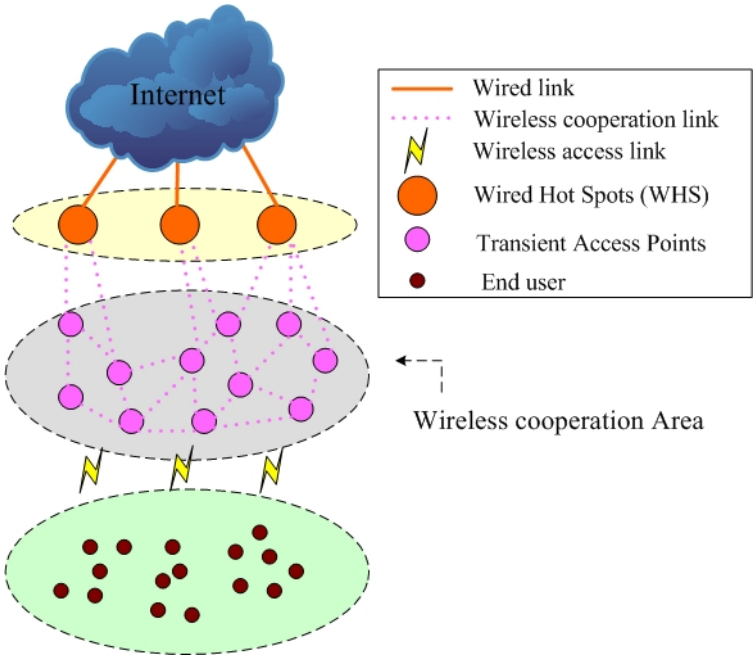

Figure 1. A wireless mesh network structure

2.2 Hidden Node Problem and STDMA

The hidden node problem [9-11] as shown in Figure 2 truly exists in multi-hop networks. Nodes A and D can not send data to B and E, respectively, at the same time because B (or D) is within the interference range of D (or B).

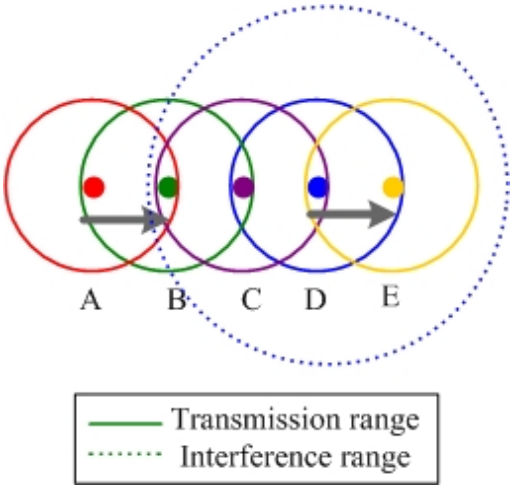

Figure 2. Hidden node problem (A is a hidden node of D)

Spatial reuse is to reuse a communication channel. The idea of spatial time division multiple access (spatial TDMA) raised in [9] is that when distance between two pairs of nodes, e.g., (A, B) and (E, F) shown in Figure 3, is far away enough, the communications between A and B and between E and F can be performed simultaneously, i.e., they use the same STDMA time slot. Such can effectively increase throughput of a WMN. The performance of spatial reuse is determined by various network characteristics, including type of radio (e.g., different carriers and different signal encoding/decoding approaches), network configuration [10], channel quality requirements (e.g., different signal/noise ratio), etc.

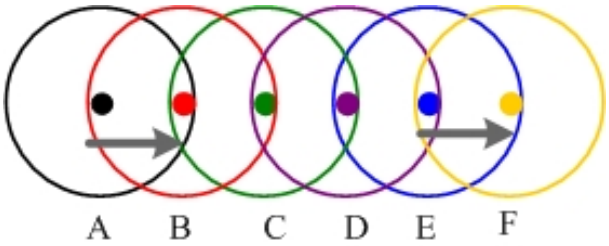

Figure 3. An example of STDMA. Nodes A and E can transmit data at the same time, i.e., they reuse the same communication channel. In other words, they share the same STDMA time slot.

2.3 Related work

Factors that affect maximum throughput of a WMN include signal interference

[6-8, 11], network bandwidth, hidden nodes [9-11], and transmission error rate [13-16]. Among them, signal interference is the key factor when bandwidth of the network is fixed. Hidden node problem resulted from collision in a wireless network is also a kind of signal interference. Maximum throughput and interference issues have been addressed by researchers. Gupta and Kumar [13] showed that under the following conditions when there are n identical but randomly located nodes in a wireless network, each node is capable of transmitting at W bits per second, and a fixed range of throughput is obtainable by each node for a randomly chosen destination, the maximum throughput is $\theta(\frac{W}{\sqrt{n \log n}})$ bits per second under a non-interference protocol. Aoun and Boutaba [6] claimed that a multi-hop wireless network can achieve its maximum throughput in particular offered load using progressive filling. The progressive filling approach starts offered rates at 0 and grows all rates together at the same pace. The procedure terminates when no flow can further increase its throughput. Jun and Sichitiu [8] also raised an interference related topic, but as stated above they considered no hidden nodes inside collision domains. Even Aoun and Boutaba [6] dealt with spatial reuse and hidden node problem, and Liu, Yan and Dai [17] further proposed concepts of short path and long path. They addressed throughput, however, in a simplified condition in STDMA time slot.

Guo, Wang and Lee [12] proposed a framework of Radio-Matching Protocol (RMP) shown in Figure 4. RMP utilizes two types of mesh routers, T-R and R-T. Each has two communication channels. The first channel of a T-R (R-T) router is for transmitting (receiving) packets, and the second for receiving (transmitting). Each router has a counter, counting 2, 1, and 0, iteratively. RMP has three states, ACTIVE, LISTEN and IDLE. It enters ACTIVE state when its counter = 2 or 1. In this state, packets can be transmitted and received without interference. RMP is in an LISTEN state when its counter = 0. Nodes in this state must keep silent for a period of time. An IDLE state is only for an initialization and error handling. When an error occurs during ACTIVE or LISTEN state, the node moves to IDLE state to restart. RMP works well and outperforms existing approaches. However, how to accurately

synchronize nodes is a crucial work.

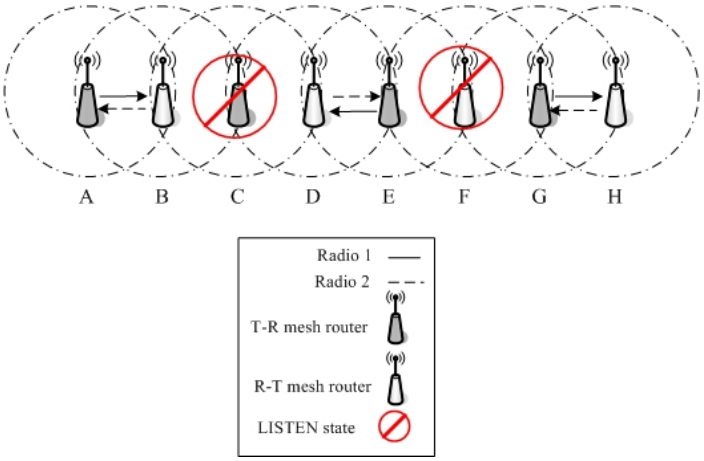

Figure 4. RMP framework

Chapter 3 The Proposed Scheme

The factors that influence transmission efficiency of a path (excluding external factors, like weather, wall, etc.) include:

(1). Number of nodes along the path [17]: the more nodes included, the longer transmission delay.

(2). Distance between two adjacent nodes: when the distance is fixed, number of nodes interfered by a communication segment or transmitting nodes can be determined. For example, if communication range (Radius) of a node is R, distance of two adjacent nodes is Q, Q≦R, and interference range of a node is IR, then the number of interfered nodes is $2*\lceil IR/Q \rceil$. Here, 2 represents two directions. In general, as shown in Figure 5, 2R<IR<3R.

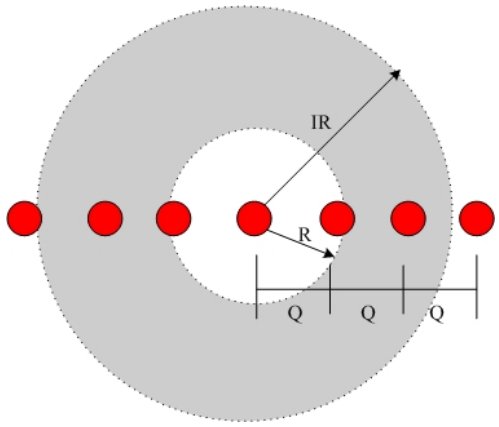

Figure 5. The relationship of Q (distance between two adjacent nodes), R (communication range (radius) of a node) and IR (interference range) of the node

(3). Remaining bandwidth of a path: which is calculated by B-B1, where B is the bandwidth of the path, B1 is bandwidth of all communication links/transmitting nodes interfering the path, and B≧B1.

(4). Hidden nodes [9-11]: which as described above is due to interference. In a chain path, when path length is longer than 3 (i.e., at least four nodes), hidden node problem may occur.

(5). Total bandwidth of the path: the wider the bandwidth, the higher data rate, and of course, the higher transmission efficiency.

(6). Packet size: When error rate is low, owing to less overhead, the larger the size of a packet, the more data transmitted and then the higher the throughput of the path.

(7). Data rate of a source node on the path: when the data rate does not saturate bandwidth of the given path, the higher the data rate, the higher its throughput.

In fact, maximum throughput is strongly related to transmission efficiency.

3.1 Efficiency of a Chain Path on Single Transmission Frequency

Attributes and variables used in this study are listed in Tables 1 and 2, respectively.

Table 1. Attributes used in this study

Attribute	Value	Meaning & Explanation
T_{SIFS}	10 μs	SIFS time
T_{slot}	20 μs	Time slot
T_{DIFS}	50 μs	DIFS time
T_P	144 μs	Time required to transmit preamble (Length of preamble is 144bits)
T_{PHY}	48 μs	Time required to transmit physical layer header (Length of physical layer header is 48 bits)
$CW_{\min}$	31	Size of a contention window
L_{P_DATA}	2000 bytes	Length of payload per packet
L_{H_DATA}	24 bytes	Length of MAC header
L_{Tail_DATA}	4 bytes	Length of CRC
L_{ACK}	14 bytes	Length of ACK (excluding packet header and tail)
L_{RTS}	44 bytes	Length of RTS
L_{CTS}	38 bytes	Length of CTS
$data\_rate$	11 Mbps	Speed of transmitting MAC packets
$basic\_rate$	1 Mbps	Speed of transmitting control frames

Table 2. Variables used in this study

T_{P_DATA}	Time required to transmit payload of a packet
T_{H_DATA}	Time required to transmit MAC header of a packet
T_{Tail_DATA}	Time required to transmit CRC of a packet
T_{P_ACK}	Time required to transmit ACK (excluding that of transmitting packet header and tail)
$\overline{CW}$	Time required for backoff
T_{RTS}	RTS time
T_{CTS}	CTS time
T_{pac}	Cycle time of transmitting a packet
T_{STDMA}^{avg}	Average packet delay for transmitting STDMA packets in a STDMA time slot
T_{hidden}^{avg}	Average packet delay for hidden node collision
N_{path}	Number of nodes along the path (excluding its destination node)
N_{in}	Maximum number of nodes in the interfered range (It should be 6, but we only consider upflow, so $N_{in}=3$)
N_{hid}	Number of nodes with hidden node problem
$Thr_{Path}^{\max}$	Maximum throughput of a path

Maximum throughput of a path can be calculated as follows.

$$Thr_{Path}^{\max}=\frac{n*L_{DATA}}{n*T_{pac}}=\frac{L_{DATA}}{T_{pac}} \tag{1}$$

where n is number of packets transmitted from source node to destination node, and T_{pac} (i.e., cycle time of transmitting a packet) is：

$$\begin{aligned} T_{pac} &= T_{DIFS}+\overline{CW}+T_{RTS}+T_{SIFS}+T_{CTS}+T_{SIFS}+T_{DATA}+T_{SIFS}+T_{ACK} \\ &= T_{DIFS}+\frac{CW_{\min}*T_{slot}}{2}+\frac{L_{RTS}*8}{basic\_rate}+T_{SIFS}+\frac{L_{CTS}*8}{basic\_rate}+T_{SIFS}+T_{DATA}+T_{SIFS}+T_{ACK} \end{aligned} \tag{2}$$

In (2), T_{DATA} (i.e., the time required to transmit DATA) is defined as:

$$T_{DATA}=T_P+T_{PHY}+T_{H\_DATA}+T_{P\_DATA}+T_{Tail\_DATA} \tag{3}$$

where $T_{H\_DATA}=\frac{L_{H\_DATA}*8}{data\_rate}$, $T_{P\_DATA}=\frac{L_{P\_DATA}*8}{data\_rate}$, and $T_{Tail\_DATA}=\frac{L_{Tail\_DATA}*8}{data\_rate}$

and T_{ACK} (i.e., the time required to transmit an ACK) is：

$$T_{ACK}=T_P+T_{PHY}+T_{P\_ACK} \tag{4}$$

where $T_{P\_ACK} = \frac{L_{ACK} * 8}{basic\_rate}$

(2) is derived under the assumption that there is no packet collision. This is the best case. $\overline{CW}$ is a random number between zero to one [18]. That is why $CW_{\min} * T_{slot}$ is divided by 2.

3.1.1 Previous Analysis

[17] analyzed throughput of a chain-topology WMN in two cases, short path ($N_{path} \le 3$) and long path ($N_{path} > 3$). Throughput of a short path is

$$Thr_{Path}^{\max} = \frac{L_{DATA}}{N_{Path} * T_{pac}} \quad [17] \tag{5}$$

A short path has neither hidden node problem nor STDMA time slot. Only one node can transmit data at any moment, otherwise interference will occur. Figure 6 shows an example. Hence, the throughput of $N_{path} = 1$ is two times that of $N_{path} = 2$, and three times that of $N_{path} = 3$. That is why denominator contains an N_{path}. A long path is a path of which $N_{path} > 3$. It has hidden node problem and some of its time slots can be spatially reused at the same time. Throughput of a long path is

$$Thr_{Path}^{\max} = \frac{L_{DATA}}{(N_{in} + 1) * T_{pac} + N_{hid} * T_{hidden}^{avg}} \quad [17] \tag{6}$$

where $N_{hid} \approx \min(N_{Path} - N_{in} - 1, N_{in})$. The first items $(N_{in} + 1) * T_{pac}$ is derived from $N_{path} * T_{pac} - N_{STDMA} * T_{pac}$ where N_{STDMA} is number of nodes that can reuse a spatial time slot. The authors combined N_{path} and N_{STDMA} together as $(N_{in} + 1)$. The second items ($N_{hid} * T_{hidden}^{avg}$) specifies conflicts generated by hidden nodes. The general expression of the throughput can be further derived by integrating (5) and (6).

$$Thr_{Path}^{\max} = \frac{L_{DATA}}{\{\min(N_{Path} - 1, N_{in}) + 1\} * T_{pac} + p_{hid} * \max(N_{Path} - N_{in},\ 0) * T_{hidden}^{avg}} \quad [17] \tag{7}$$

where $p_{hid} = N_{hid} / (N_{Path} - N_{in})$.

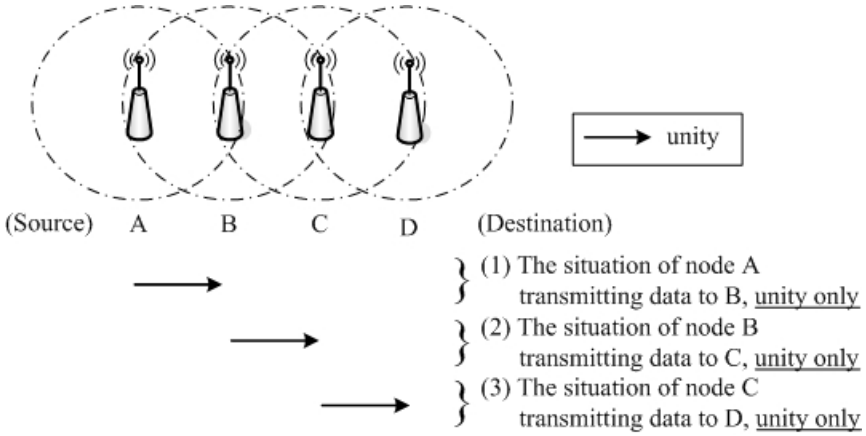

Figure 6. All situations when $N_{path} = 3$

3.1.2 Our Analyses and Derivation

In this study, we use three kinds of arrows, single-head, double-head and dotted arrows, to respectively represent, unity transmission, hidden node and STDMA transmission patterns, that may occur along a chain-topology path when a node on the path transmits data to its next node.

A chain-topology WMN can be analyzed in three ways, $N_{path} \le N_{in}$, $7 \ge N_{path} > N_{in}$ and $N_{path} > 7$. To simplify the following description, we use x→y to represent that node x transmits data to node y. Maximum throughput of $N_{path} \le N_{in}$ (here N_{in} = 3), denoted by $Thr_{pathi}^{\max}$, i=1, 2, 3, as shown in Figure 6 is the same as (5) since the three patterns are all unity patterns.

Maximum Throughputs of $7 \geqq N_{path} > N_{in}$

$N_{path} = 4$

In Figure 7, when node A transmits a packet to node B, two situations may occur. The first occurs when node A transmits packets, and its hidden node D keeps silent. Assume the probability is $p_{unity\ 4}$, and transmission time required is T_{pac}. The second

is that nodes A and D transmit packets simultaneously, resulting in a collision. Assume probability of the collision is $p_{hidden\ 4}$, and the time spent for transmitting a packet from node A to node B in this situation is T_{hidden}^{avg}, where $p_{unity\ 4} + p_{hidden\ 4} = 1$. Hence, the expected time that node A requires to send a packet to node B is $T_{hidden4} = p_{unity\ 4} * T_{pac} + p_{hidden\ 4} * T_{hidden}^{avg}$. We call A→B a hidden node pattern. D→E has the similar pattern. When B desires to transmit data to C, there is only one situation, i.e., unity pattern. C→D has the similar one. Therefore, we can derive maximum throughput of $N_{path} = 4$, denoted by $Thr_{path4}^{\max}$, $Thr_{Path4}^{\max} = \frac{L_{DATA}}{2 * T_{pac} + 2 * T_{hidden\ 4}}$, where $2 * T_{pac}$ represents B→C and C→D, and $2 * T_{hidden4}$ describes A→B and D→E.

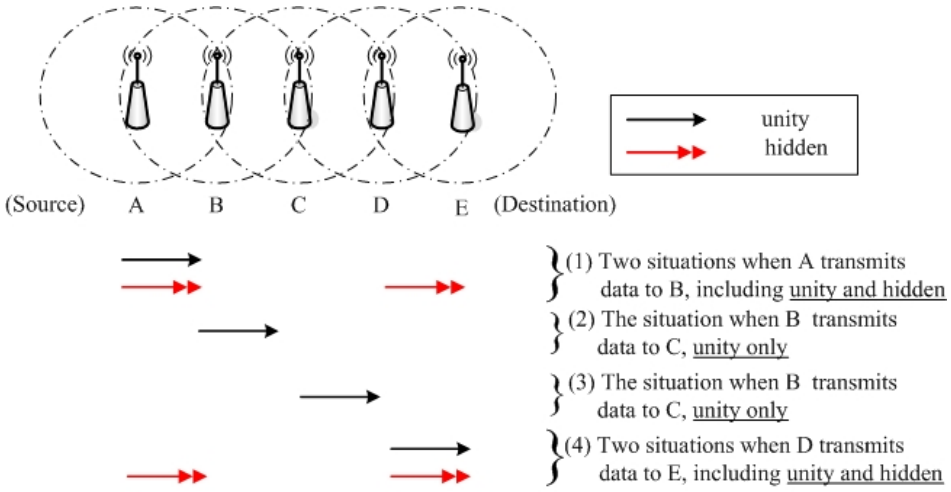

Figure 7. All situations when $N_{path} = 4$

$N_{path} = 5$

In Figure 8, B→C, like A→B in Figure 7, has a hidden node pattern. So, we can derive $T_{hidden5} = p_{unity\ 5} * T_{pac} + p_{hidden\ 5} * T_{hidden}^{avg}$ (in which $p_{unity\ 5} + p_{hidden\ 5} = 1$). The characteristic of C→D is similar to that of B→C in Figure 7. In addition when node A desires to transmit a packet to node B, three situations may occur. The first is that node A transmits a packet to node B, and its hidden node D keeps silent. Assume the

probability is $p_{unity\_5\_1}$ which is different from p_{hidde} for B→C. The second is that node A and D transmit packets at the same time, causing a collision. Assume the probability is p_{hidden_5}. The third is that node A and one of its STDMA node, e.g., E, transmit packets at the same time. We call A→B has a STDMA pattern. Assume that in this situation the time node A requires to transmit a packet to node B is T_{STDMA}^{avg}, and the probability is p_{STDMA}, where $p_{unity5\_1} + p_{hidde5\_1} + p_{STDMA5\_1} = 1$ and the expected time that node A requires to successfully transmit a packet to node B is $T_{STDMA} = p_{unity\_5\_1} * T_{pac} + p_{hidden\_5\_1} * T_{hidden}^{avg} + p_{STDMA\_5\_1} * T_{STDMA}^{avg}$. Therefore, maximum throughput of $N_{path} = 5$ $Thr_{Path5}^{\max} = \frac{L_{DATA}}{1*T_{pac} + 2*T_{hidden\_5} + 2*T_{STDMA\_5}}$, where $1*T_{pac}$ represents C→D (unity pattern), $2*T_{hidden\_5}$ describes B→C and D→E (i.e., two hidden node patterns), $2*T_{STDMA\_5}$ specifies A→B and E→F (i.e., two STDMA patterns).

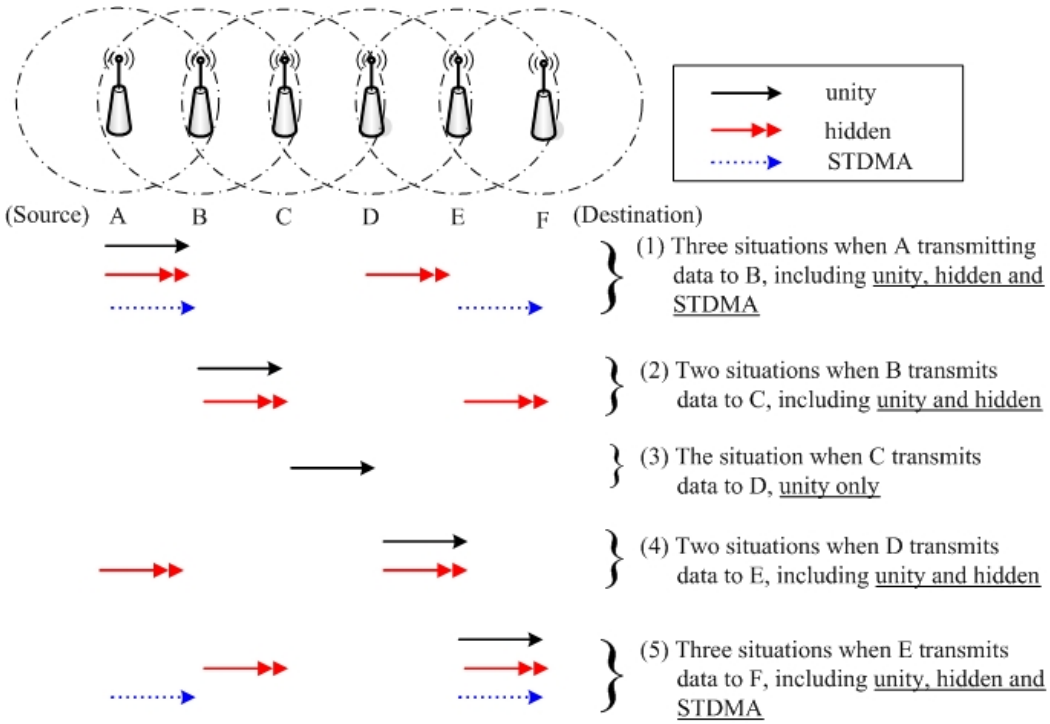

Figure 8. All situations when $N_{path} = 5$

$\underline{N_{path} = 6}$

Maximum throughput of $N_{path} = 6$, $Thr_{path6}^{\max}$, as shown in Figure 9, can be derived by the similar way. This time we have two STDMA patterns, denoted by $T_{STDMA\ 6\_1}$ and $T_{STDMA\ 6\_2}$, so

$Thr_{Path6}^{\max} = \frac{L_{DATA}}{0 * T_{pac} + 2 * T_{hidden\ 6} + 2 * T_{STDMA\ 6\_1} + 2 * T_{STDMA\ 6\_2}}$, where $2 * T_{hidden6}$ is to describe C→D and D→E, $2 * T_{STDMA\ 6\_1}$ is to specify A→B and F→G (for each of them, there are two possibilities of STDMA, e.g., A→B and E→F transmit data simultaneously, or A→B and F→G send data to next nodes at the same time), and $2 * T_{STDMA6\_2}$ is to represent B→C and E→F (for each of them, there is only one STDMA, e.g., B→C and F→G (or E→F and A→B) are performed simultaneously),

$T_{hidden\ 6} = p_{unity\ 6} * T_{pac} + p_{hidden\ 6} * T_{hidden}^{avg}$ (in which $p_{unity\ 6} + p_{hidden\ 6} = 1$),

$T_{STDMA6\_i} = p_{unity\ 6\_i} * T_{pac} + p_{hidden\ 6\_i} * T_{hidden}^{avg} + p_{STDMA\ 6\_i} * T_{STDMA}^{avg}$ (in which $p_{unity\ 6\_i} + p_{hidden\ 6\_i} + p_{STDMA\ 6\_i} = 1$), i=1, 2 and $p_{STDMA\ 6\_1} > p_{STDMA\ 6\_2} > 0$, because $p_{STDMA\ 6\_1}$ has two STDMAs, e.g., (A→B and E→F) and (A→B and F→G), but $p_{STDMA\ 6\_2}$ has only one STDMA, e.g., B→C and F→G.

For simplicity's sake, combinations of $N_{path} = 4$ to $N_{path} = 10$ are summarized in Table 3, in which x, y, z of (x, y, z) represent numbers of unity, hidden and STDMA nodes, respectively, and from which we can further derive maximum throughputs for $N_{path} = 7$ to $N_{path} = 10$, where (1, 0, 0) represents it is a unity pattern; (1, i, 0) represent. That it is a hidden node pattern including one unity, and i hidden nodes; and (1, i, j) denotes that it is a STDMA pattern including one unity, i hidden nodes and j STDMA

nodes. For example, when $N_{path}=6$, $p_{STDMA\ 6\_1}=\frac{2}{1+1+2}$ (see Table 3, A→B or F→G), $p_{STDMA\ 6\_2}=\frac{1}{1+1+1}$ (see Table 3, B→C or E→F).

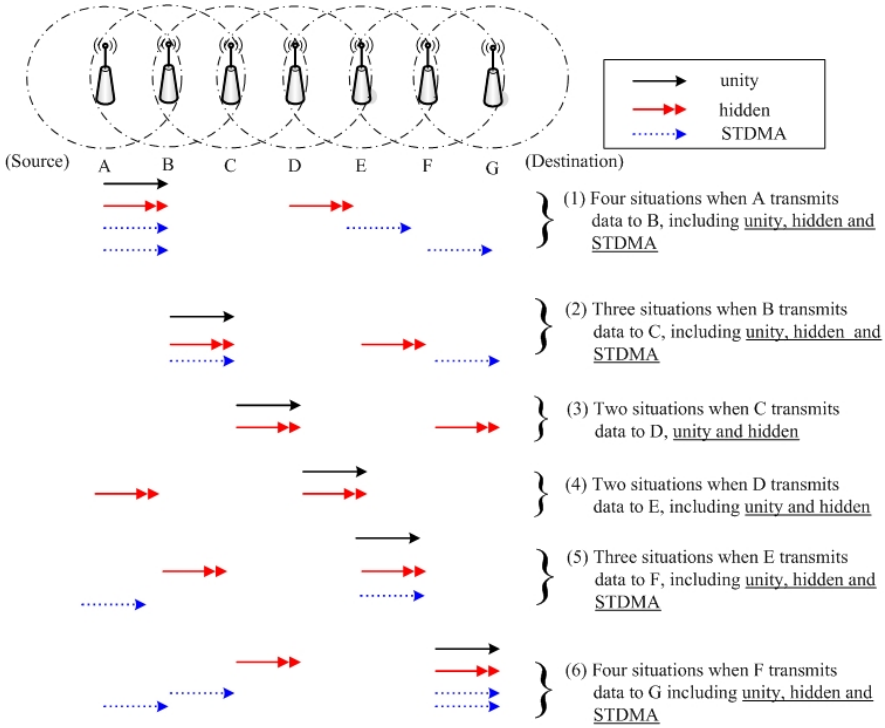

Figure 9. All situations when $N_{path}=6$

Table 3. All situations occur from when $N_{path}=4$ to $N_{path}=10$, where x, y and z of (x, y, z) represent unity, hidden and STDMA nodes, respectively. (1, 0, 0) represents it is a unity pattern, (1, i, 0) represents that it is a hidden node pattern including one unity and i hidden nodes, and (1, i, j) describes that it is a STDMA pattern including one unity, i hidden nodes and j STDMA nodes.

Source node / N_{path}	A→B	B→C	C→D	D→E	E→F	F→G	G→H	H→I	I→J	J→K	Refer to
4 (Dest. is E)	(1,1,0)	(1,0,0)	(1,0,0)	(1,1,0)							Figure 7
5 (Dest. is F)	(1,1,1)	(1,1,0)	(1,0,0)	(1,1,0)	(1,1,1)						Figure 8
6 (Dest. is G)	(1,1,2)	(1,1,1)	(1,1,0)	(1,1,0)	(1,1,1)	(1,1,2)					Figure 9
7 (Dest. is H)	(1,1,3)	(1,1,2)	(1,1,1)	(1,3,0)	(1,1,1)	(1,1,2)	(1,1,3)				
8 (Dest. is I)	(1,1,4)	(1,1,3)	(1,1,2)	(1,3,1)	(1,3,1)	(1,1,2)	(1,1,3)	(1,1,4)			

9 (Dest. is J)	(1,1,6)	(1,1,4)	(1,1,3)	(1,3,2)	(1,3,3)	(1,3,2)	(1,1,3)	(1,1,4)	(1,1,6)		
10 (Dest. is K)	(1,1,9)	(1,1,6)	(1,1,4)	(1,3,3)	(1,3,5)	(1,3,5)	(1,3,3)	(1,1,4)	(1,1,6)	(1,1,9)	
…	…	…	…	…	…	…	…	…	…	…	

$\underline{N_{path} = 7}$

When $N_{path} = 7$, (see Table 3), $Thr_{Path7}^{\max} = \frac{L_{DATA}}{0*T_{pac} + 1*T_{hidden\_7} + 2*T_{STDMA\_7\_1} + 2*T_{STDMA\_7\_2} + 2*T_{STDMA\_7\_3}}$, where $T_{hidden\_7} = p_{unity\_7} * T_{pac} + p_{hidden\_7} * T_{hidden}^{avg}$ (in which $p_{unity\_7} + p_{hidden\_7} = 1$), the corresponding item in Table 3 is (1, 3, 0). $T_{STDMA\_7\_i} = p_{unity\_7\_i} * T_{pac} + p_{hidden\_7\_i} * T_{hidden}^{avg} + p_{STDMA\_7\_i} * T_{STDMA}^{avg}$ (in which $p_{unity\_7\_i} + p_{hidden\_7\_i} + p_{STDMA\_7\_i} = 1$), i =1, 2, 3, (the corresponding items in Table 3 are (1, 1, 3), (1, 1, 2) and (1, 1, 1), respectively) and $p_{STDMA\_7\_1} > p_{STDMA\_7\_2} > p_{STDMA\_7\_3} > 0$, since each of A→B and G→H has more STDMA chances than those of B→C and F→G, and each of B→C and F→G in turn has more STDMA opportunities than those of C→D and E→F.

Also, i=1 for $T_{STDMA\_7\_1}$ represents A→B and G→H. The representation in the following is further simplified to i=1: (A→B and G→H). Similarily, i=2 and i=3 can be simplified to i=2: (B→C and F→G); and i=3: (C→D and E→F), respectively.

Now, we can summary the maximum throughput for $7 \geq N_{path} > N_{in}$, it is

$$Thr_{Path}^{\max} = \frac{L_{DATA}}{\max(N_{in} - N_{path} + 3,\ 0) * T_{pac} + \min(N_{in} - N_{path} + 5,\ 2) * T_{hidden} + 2 * T_{STDMA}} \quad (8)$$

where $T_{hidden\ n} = p_{unity\ n} * T_{pac} + p_{hidden\ n} * T_{hidden}^{avg}$ (in which $p_{unity\ n} + p_{hidden\ n} = 1$), $T_{STDMA} = (N_{path} - 4) * T_{STDMA\ n\_1} + \max(N_{path} - 5,\ 0) * T_{STDMA\ n\_2} + \max(N_{path} - 6,\ 0) * T_{STDMA\ n\_3}$, here $T_{STDMA\ n\_i} = p_{unity\ n\_i} * T_{pac} + p_{hidden\ n\_i} * T_{hidden}^{avg} + p_{STDMA\ n\_i} * T_{STDMA}^{avg}$ (in which $p_{unity\ n\_i} + p_{hidden\ n\_i} + p_{STDMA\ n\_i} = 1$), i = 1 to $N_{path} - 4$ (i.e., $N_{path} = 4$ has no STDMA).

2) Maximum throughput on N_{path} >7

$\underline{N_{path} = 8}$

When $N_{path} = 8$ (see Table 3),

$$Thr_{Path8}^{\max} = \frac{L_{DATA}}{0*T_{pac} + 0*T_{hidden\_8} + 2*T_{STDMA\_8\_1} + 2*T_{STDMA\_8\_2} + 2*T_{STDMA\_8\_3} + 2*T_{STDMA\_8\_4}},$$

where $T_{hidden8} = p_{unity\_8} * T_{pac} + p_{hidden\_8} * T_{hidden}^{avg}$ (in which $p_{unity\_8} + p_{hidden\_8} = 1$ and $T_{STDMA\_8\_i} = p_{unity\_8\_i} * T_{pac} + p_{hidden\_8\_i} * T_{hidden}^{avg} + p_{STDMA\_8\_i} * T_{STDMA}^{avg}$, i=1, 2, 3, 4 (respectively for (1, 1, 4), (1, 1, 3), (1, 1, 2) and (1, 3, 1)). Here, i=1: (A→B and H→I); i=2: (B→C and G→H); i=3: (C→D and F→G); i=4: (D→E and E→F), $p_{unity\_8\_i} + p_{unity\_8\_i} + p_{hidden\_8\_i} = 1$, and $p_{STDMA\_8\_4} > p_{STDMA\_8\_3} > p_{STDMA\_8\_2} > p_{STDMA\_8\_1} > 0$.

$\underline{N_{path} = 9}$

When $N_{path} = 9$ (see Table 3),

$$Thr_{Path9}^{\max} = \frac{L_{DATA}}{0*T_{pac} + 0*T_{hidden\_9} + 2*T_{STDMA\_9\_1} + 2*T_{STDMA\_9\_2} + 2*T_{STDMA\_9\_3} + 3*T_{STDMA\_9\_4}},$$

where $T_{hidden\_9} = p_{unity\_9} * T_{pac} + p_{hidden\_9} * T_{hidden}^{avg}$ (in which $p_{unity\_9} + p_{hidden\_9} = 1$, and $T_{STDMA\_9\_i} = p_{unity\_9\_i} * T_{pac} + p_{hidden\_9\_i} * T_{hidden}^{avg} + p_{STDMA\_9\_i} * T_{STDMA}^{avg}$, i=1, 2, 3, 4. Here, i=1: (A→B and I→J); i=2: (B→C and H→I); i=3: (C→D and G→H); i=4: (D→E, E→F and F→G), and $p_{STDMA\_9\_4} > p_{STDMA\_9\_3} > p_{STDMA\_9\_2} > p_{STDMA\_9\_1} > 0$.

When $N_{Path} > 7$, there is no individual unity and hidden node. They all co-exist with a STDMA. Now, we can derive the maximum throughput for a chain-topology WMN when $N_{path} > 7$ as follows.

$$Thr_{Pathn}^{\max} = \frac{L_{DATA}}{2*(T_{STDMA\ n\_1} + T_{STDMA\ n\_2} + T_{STDMA\ n\_3}) + \max(N_{path} - 6,\ 0)*T_{STDMA\ n\_4}} \quad (9)$$

where $T_{STDMA\ n\_i} = p_{unity\ n\_i} * T_{pac} + p_{hidden\ n\_i} * T_{hidden}^{avg} + p_{STDMA\ n\_i} * T_{STDMA}^{avg}$ (in which

$p_{unity\ n\_i} + p_{hidden\ n\_i} + p_{STDMA\ n\_i} = 1$), $n = N_{path}$ and i= 1, 2, 3, 4 see Table 3.

In the following, we will introduce how to deploy this model to determine whether a segment of a path can transmit data with its maximum data rate or not.

3.2. Data Rate of a Node

In this study, when a source S attempts to send data, it uses AODV routing protocol to discover a path [19] under the situation that no routing information is available. S initiates a route discovery process by broadcasting RREQs to the entire network. A RREQ packet [19] consists of RREQ sequence number, RREQ originator, hop-count to the source, immediate RREQ sender, etc. An intermediate node later on receiving duplicated RREQs from different senders will record these senders' information in its RREQ table, and discard the packet. It only rebroadcasts the first received RREQ. The activities together are called route discovery phase since their purpose is to discover a routing path. After that, packets can be delivered to their destination through the path.

In a chain topology, a node intending to send data to the destination node issues (rather than broadcasts) a RREQ packet [19] which will flow through subsequent nodes to the destination node. We create a table, named C_table, for the destination node in route discovery phase. When the destination node receives a RREQ from a node x, it creates a tuple in C_table for x. The tuple has four fields, hop_count, source_ip, destination_ip and starting_order. The first field records number of nodes that a RREQ packet has traveled through, which is also length of the corresponding routing path, i.e., N_{Path}. The second and the third respectively copy originator and destination IP addresses conveyed on the RREQ packet. The fourth field records the order that nodes start transmitting data packets, rather than its RREQ packet, to the destination node. Before a tuple is inserted into C_table, the initial value of the field starting_order is null. We also create a variable sequence_count to keep track the order that nodes start sending their data packets, and its initial value is zero. After the route discovery phase, when the destination node receives a data packet p it

respectively compares the source_ip and destination_ip fields in its C_table with p's source and destination addresses. If there is a matched record R, and R's starting_order = null, implying that p is the first packet that its source node sends to the destination, then sequence_count is increased by one, and let R's starting_order = sequence_count, indicating that the source node is the starting_orderth node that sends its first packet to the destination node. If there is a matched record but its starting_order ≠ null, i.e., it is not the first data packet sent by this source node, or there is no matched record, the packet is discarded. The latter means that the source node is an un-registered node.

Assume that there is a path P of chain topology linearly connecting k nodes n_1, n_2, n_3 …n_k together. Among these nodes, q nodes (e.g., n'_1, n'_2 …n'_q, q≦k-1) transmit data to their common target n_k, where n_k is the only node that connects to the Internet through a wired path. When traffic of a path segment M on P is saturated, a node r that intends to transmit packets through the path segment is not allowed to send right now, otherwise interference will occur, where r∈{n'_1, n'_2, …n'_q}. We determine whether r is allowed to send its packets or not by the following algorithm. If yes, the algorithm further checks to see what data rate r can send packets.

Path_C Algorithm /* Algorithm for determining the data rate a node can send its data */

Input: A path P of chain topology that linearly connects k nodes n_1, n_2, n_3…n_{k-1}, n_k together; q nodes of P, n'_1, n'_2…n'_q, that have sent RREQ packets to n_k during route discovery phase, q≦k-1; n'_ks C_table which has q records keeping corresponding information of the q nodes.

Output: Whether a node r intending to transmit data to n_k can transmit data right now or not? If yes, what data rate r can transmit its data?

{

1. P_{rem} = 0; /*initialization*/
2. If (sequence_count ≧ 1) { /*at least one node has transmitted data to n_k*/
   For(starting_order =1 to sequence_count){ /*starting_order: a field of a C_table, recording order that nodes sent their first data packets to n_k; here, we

assume that priorities of n'_1, n'_2…n'_q are the same; if they are different, this step should follow their transmission priorities.*/

2.1 Calculate maximum throughput for node $n'_{starting\_order}$, i.e., $Thr^{\max}_{Path\ n'\ starting\_order}$ by deploying (5), (8) or (9) according to N_{path} value.

2.2 Percentage of throughput used is

$$p_{rem} = p_{rem} + \frac{curret\_data\_rate}{Thr^{\max}_{Path\ n'\ starting\_order}} \quad (10)$$

/* current_data_rate is the data rate the node $n'_{starting\_order}$ currently transmits its data*/

2.3 If ($p_{rem} > p_{threshold}$) goto step 4; /*theoretical value of $p_{threshold}$ is 1*/}}

3. If (a node r intends to transmit data to n_k with a specific data rate, e.g., $data\_rate(r)$, where r∈{n'_1, n'_2, …n'_q}, and r has a corresponding record in n'_ks C_table, e.g., record R)

{ If R's starting_order = =null) { /* the packet received is the first data packet sent by r */

3.1 Calculate maximum throughput for node r, $Thr^{\max}_{Path}(r)$;

3.2 Throughput that r can gain from remaining bandwidth of P is $Thr_{rem}(r)$, which is calculated by deploying (11).

$$Thr_{rem}(r) = Thr^{\max}_{Path\ r} * (1 - p_{rem}) \quad (11)$$

3.3 If (data_rate(r) > $Thr_{rem}(r)$) $data\_rate(r) = Thr_{rem}(r)$; /*r's data rate can not exceed its current maximum allowable data rate */

Else {allow r to transmit data to n_k with $data\_rate(r)$; stop;.}}}

4. Inhibit r to transmit data to n_k;} /*other nodes use up the bandwidth of P*/

In this algorithm, $p_{threshold} = 1$ is an ideal case. In fact, the maximum throughputs of calculated and simulated results are not the same. Therefore, we reduce the threshold ($p_{threshold}$) to a specific value, 0.95, to meet the real situation. However, one may figure out that the denominators of (10), $Thr^{\max}_{Path\ n'\ starting\_order\ s}$, are

different. How fractions of different paths can be summed up. The reason is each $Thr_{Path}^{\max}$ represents. The maximum capacity of that path segment. If j% of $Thr_{Path}^{\max}$ is used. Other path segment Y with $Thr_{Path\ Y}^{\max}$ should not inject $Thr_{Path\ Y}^{\max}$ * (100-j)% of traffic, otherwise interference will occur.

3.3. Efficiency of a path on RMP framework

The RMP framework uses a counter to solve the binary exponential backoff problem [12]. At the beginning of its chain operation, each node is in IDLE state. A node x (except the first node) is activated by an RTS (request to send) frame generated by its precedent node. After that, x triggers its counter starting with the value of 2, by invoking no RTS and CTS. Therefore, $T_{pac} = T_{DATA} + T_{ACK}$, omitting T_{DIFS}, $\overline{CW}$, T_{RTS}, T_{CTS}, and T_{SIFS} from (2).

The RMP framework also solves the hidden terminal and exposed terminal problems. Hence, it can use the same STDMA time slot all the same time. But, Guo, Wang and Lee [11] did not analyze RMP's maximum throughput in detail. We analyze it in the following. Throughput analysis of a path based on the RMP framework can be also divided into two cases, $N_{path} \le N_{in}$ and $N_{path} > N_{in}$.

If $N_{path} \le N_{in}$, $Thr_{Path}^{\max}$ is the same as (5)

Throughput of $N_{path} > N_{in}$ is:

$$Thr_{Path}^{\max} = \frac{L_{DATA}}{T_{pac} * N_{in}} \tag{12}$$

General expression of the throughput can be derived by integrating (5) and (12).

$$Thr_{Path}^{\max} = \frac{L_{DATA}}{T_{pac} * \min(N_{in}, N_{path})} \tag{13}$$

In RMP framework, Path_C algorithm can be invoked so that its throughput can be further improved.

3.4. Probabilities of hidden nodes and STDMA

In the following, we analyze probabilities of hidden nodes and STDMAs given a chain-topology path which has n+1 nodes (including the front n nodes and their common destination node).

3.4.1 Probability of hidden nodes

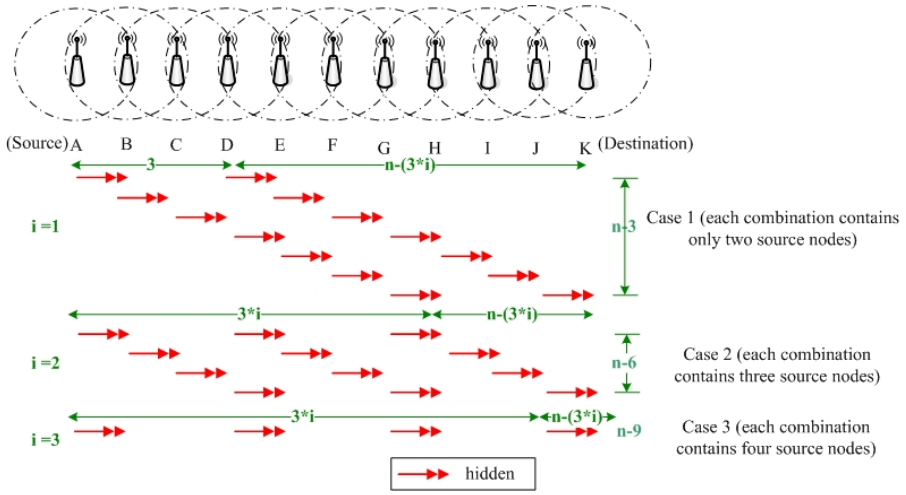

Figure 10. When n = 10, at most 4 nodes are involved as hidden nodes, and number of possible cases that cause hidden node problem is 12

The number of possible cases that hidden node problem may occur is

$$n_{hidden} = \sum_{i=1}^{\left\lfloor \frac{n}{3} \right\rfloor} n-(3*i) \ \ where\ n \geq 3 \tag{14}$$

Where $\left\lfloor \frac{n}{3} \right\rfloor$ is maximum number of nodes causing hidden node problem. For example, when n=10 as shown in Figure 10,

$$n_{hidden} = \sum_{i=1}^{\left\lfloor \frac{n}{3} \right\rfloor} n-(3*i) \ = \sum_{i=1}^{3} 10-(3*i) = [10-(3*1)]+[10-(3*2)]+[10-(3*3)] = 7+4+1 = 12$$

3.4.2 Probability of STDMA nodes

We divide the analysis into three cases, n<5, 5≦n≦8 and 9≦n.

Case 1: When n<5, $n_{STDMA} = 0$

Case 2: When 5≦n≦8

$$n_{STDMA} = \sum_{i=0}^{n-5}(n-i-4) \tag{15}$$

For example, when n = 8 as shown in Figure 11 should be

$$n_{STDMA} = \sum_{i=0}^{n-5}(n-i-4) = \sum_{i=0}^{8-5}(8-i-4) = 4+3+2+1 = 10$$

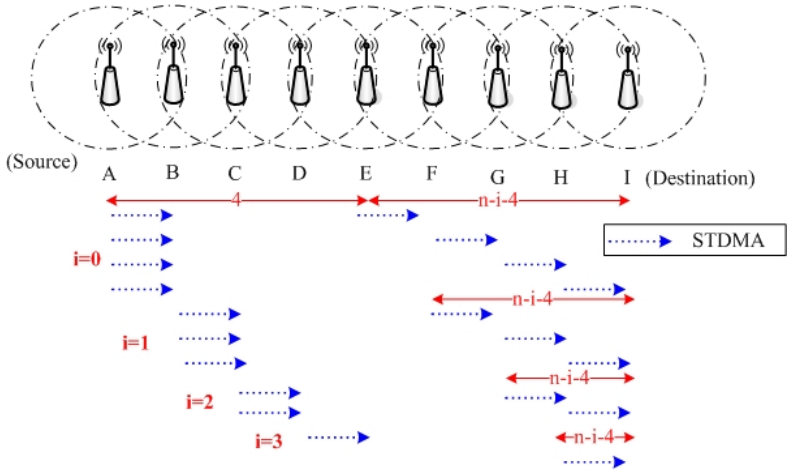

Figure 11. When n = 8, only two nodes can transmit data simultaneously in a STDMA time slot, and number of possible cases that allow two nodes to transmit data simultaneously in a STDMA time slot is 10.

Case 3: When 9≦n

The number of possible cases that allow only two nodes to transmit data simultaneously in a STDMA time slot is

$$\sum_{i=0}^{n-5}(n-i-(2-1)*4) \tag{16}$$

The number of possible cases that allow exactly three nodes to transmit data simultaneously in a STDMA time slot is

$$\sum_{j=0}^{n-9}\sum_{i=j}^{n-9}(n-i-(3-1)*4) \tag{17}$$

The number of possible cases that allow exactly four nodes to transmit data simultaneously in a STDMA time slot is

$$\sum_{k=0}^{n-13}\sum_{j=k}^{n-13}\sum_{i=j}^{n-13}(n-i-(4-1)*4) \tag{18}$$

Why there are three $\sum_s$ in (17)? The reason is that we need three variables to identify positions of three source nodes. The possible positions of the 4th source node are described by (n-i-(4-1)*4) since it has (n-i-(4-1)*4) different positions. Here, we can conclude that if we allow exact k nodes to transmit their data simultaneously, k-1 variables, of course k-1 $\sum_s$, are required. The maximum $k = \left\lfloor \frac{n}{4} \right\rfloor$.

Therefore, number of STDMA n_{STDMA} can be derived by integrating (16), (17), (18) and what mention above.

$$n_{STDMA} = \underbrace{\sum_{i=0}^{n-5}(n-i-(2-1)*4)}_{k=2} + \underbrace{\sum_{j=0}^{n-9}\sum_{i=j}^{n-9}(n-i-(3-1)*4)}_{k=3} + \underbrace{\sum_{s=0}^{n-13}\sum_{j=s}^{n-13}\sum_{i=j}^{n-13}(n-i-(4-1)*4)}_{k=4} + \ldots +$$

$$+ \underbrace{\sum_{l=0}^{n-(4*q+1)}\sum_{j=l}^{n-(4*q+1)} \cdots \sum_{t=s}^{n-(4*q+1)}\sum_{i=t}^{n-(4*q+1)}(n-i-(4-1)*4)}_{k=\left\lfloor \frac{n}{4} \right\rfloor}$$

For example, n = 13 as shown in Figure 12, number of STDMA is

$$\sum_{i=0}^{n-5}(n-i-(2-1)*4) + \sum_{j=0}^{n-9}\sum_{i=j}^{n-9}(n-i-(3-1)*4) + \sum_{s=0}^{n-13}\sum_{j=s}^{n-13}\sum_{i=j}^{n-13}(n-i-(4-1)*4)$$
$$= (9+8+7+6+5+4+3+2+1) + [(5+4+3+2+1)+(4+3+2+1)+(3+2+1)+(2+1)+1] + 1 = 81$$

Now, from a overall view point, when a node transmits a packet in a time slot t, we can calculate the probability that the transmission is in a unity, a hidden node or a STDMA transmission mode (rather than pattern) given a chain-topology path P of length L. Let n_{unity} be number of unity of P, where $n_{unity} = L$, and

n *be* $n_{unity} + n_{hidden} + n_{STDMA}$. Then, the probabilities that the transmission is in a unity mode, a hidden-node mode and a STDMA mode is

$p(unity) = n_{unity} / n$, $p(hidden) = n_{hidden} / n$ and $p(STDMA) = n_{STDMA} / n$, respectively.

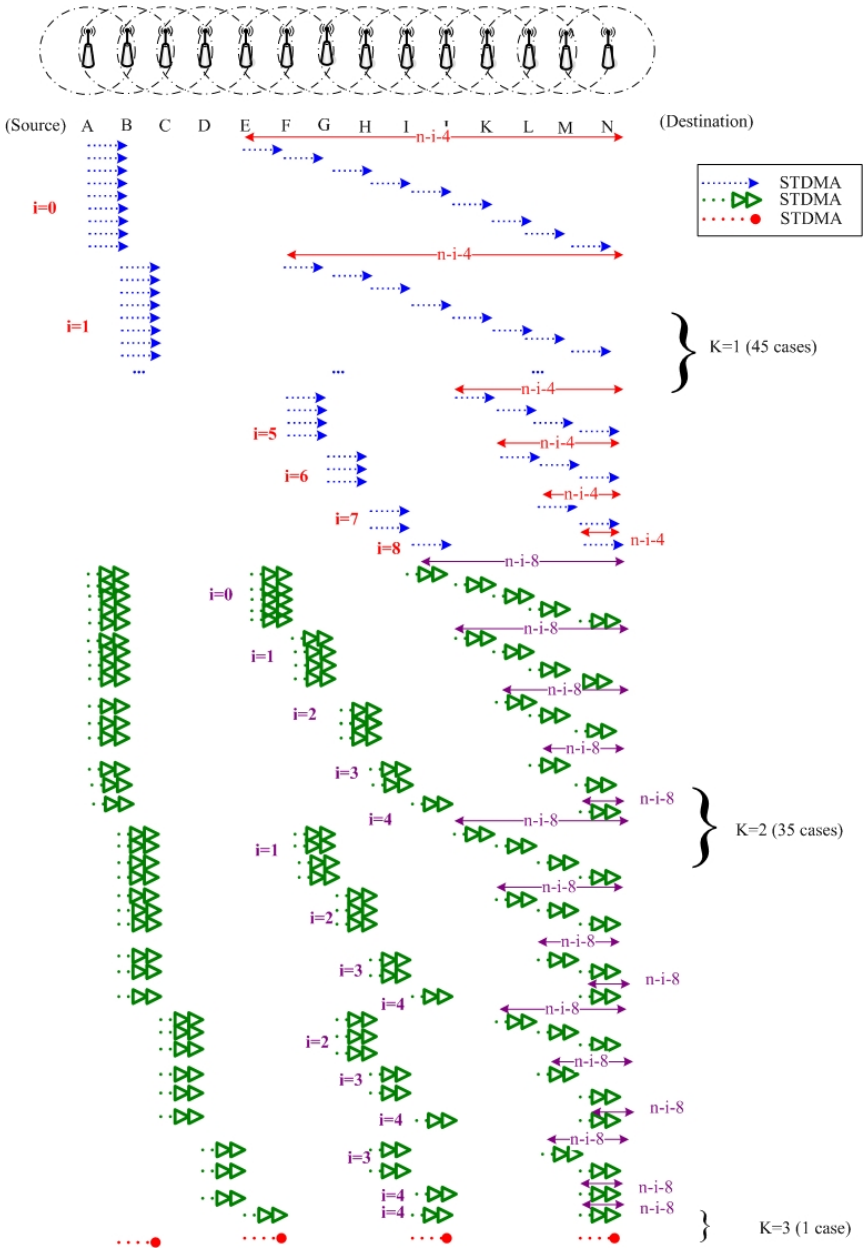

Figure 12. When n = 13, number of all possible combinations of STDMA is 81 (= 45+35+1)

Chapter 4 Experimental Environments and Results

In this study, we used *ns*-2 [20] as our simulation tool. Four cases listed in Table 4 are used in the following experiments.

Table 4 Four cases are used in the following experiments

Case	$N_{Path\ 1} = 3$ (Kbps)	$N_{Path\ 2} = 5$ (Kbps)	$N_{Path\ 3} = 9$ (Kbps)	$p_{rem} > p_{threshold}$
1	512	400	300	No
2	512	400	310	Yes
3	512	512	1024	Yes
4	1024	768	300	Yes

Among the four cases, $p_{rem}s$ of cases 2 to 4 are all larger than $p_{threshold}$. Each simulation was performed fifteen times all on a 500 * 500 m^2 wireless field. Each time lasted 200 seconds. CBR (Constant Bit Rate) data was transmitted to the same destination. Channel bandwidth is 11 Mbps. Attributes and parameters used are listed in Attributes and variables used in this study are listed in Tables 1 and 2, respectively.

and 2, respectively. Communication range and interference range of a node is 40 m and 90 m, respectively. We employed progressive filling [6] to adjust data rates for nodes to achieve their maximum throughput.

The first experiment evaluates throughputs of several related algorithms and error rates of throughputs between their individual simulation and calculation results. The second experiment compares throughputs of a chain-topology path when cases 1 to case 4 are involved. The third and the fourth study packet loss rates of case 1 to 4 and jitters of cases 1 and 2. Transmission delays of cases 1 to 4 are explored in the fifth experiment. The sixth compares system performance of a chain-topology path, when one, two and four channels (transmission frequencies) are used.

4.1 Error rates between calculated and simulation results

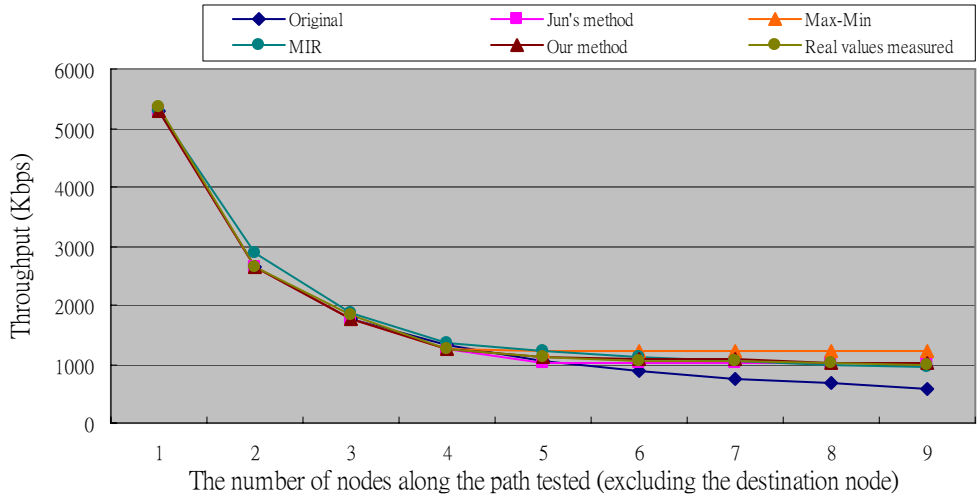

Figure 13 Calculated throughputs of the algorithms tested

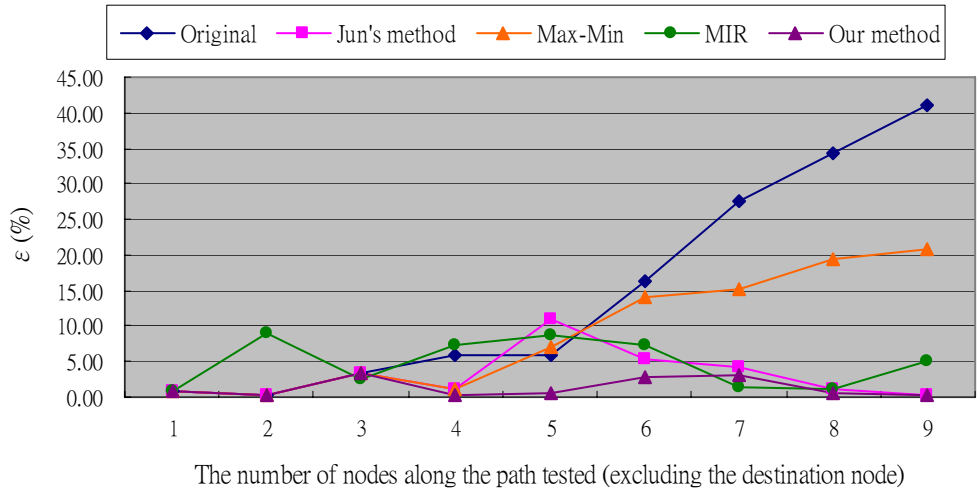

Figure 14 Error rates between simulated throughput (i.e., "Real values measured" in Figure 13) and calculated throughputs of the algorithms tested

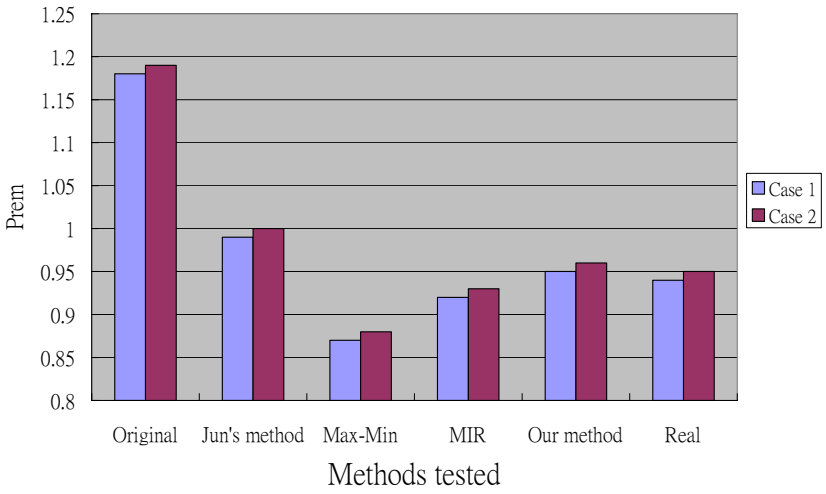

Figure 15 Relation between p_{rem} and $p_{threshold}$ given cases 1 and 2

In the first experiment, we evaluate several existing algorithms, including Jun's method [8], Max-Min [6], MIR [10], original and our method, and their derived mathematical equations for calculating maximum throughput of a chain-topology path. The algorithm, that can more accurately approximate the real measured values, will be the one that can accurately predict the maximum throughput of a transmission path. It can then effectively control the performance of the path. The "original" means this algorithm does not consider hidden nodes and STDMA time slots, only dealing with signal interference. Jun and Sichitiu [8] considered network interference and calculated collision domain. But as stated above they (i.e., original and Jun's methods) dealt with no hidden node problem and spatial reuse in collision domains. The Max-Min [6] claimed that its throughput reaches a steady state around 1205.75 Kbits/s when number of nodes is beyond 4. The throughput MIR derived is $C_p = \frac{1}{\sqrt{N_i}} \times S_p \times P_c$ where C_p is path capacity, S_p per-path spatial reuse number, P_c is power consumption gain and N_i is the number of nodes along a path. "Real values measured" represents its throughputs are really measured on-line. According to (2), throughput of $N_{path} = 1$ is as follows.

$$T_{pac} = 50 + 310 + 352 + 10 + 304 + 10 + 1666.909 + 10 + 304 = 3016.909 \mu s$$

According to (5), when $N_{path}=1$,

$Thr_{Path}^{\max}=\frac{8*2000}{1*3016.909}=5.303Mbps$, and when $N_{path}=2$,

$Thr_{Path}^{\max}=\frac{8*2000}{2*3016.909}=2.6515Mbps$ and so on. The calculated results real values measured gives really measured throughputs are illustrated in

Figure 13 which shows that our method is the one most close to the real measured throughputs.

Error rates ε_s between calculated and simulated results for algorithms tested, defined as $\left|\frac{V_{simulation}-V_{calculation}}{V_{simulation}}\right|$, are shown in

Figure 14 in which ε_s of our method on different numbers of nodes involved are all less than 5%. In a short path, ε is quite stable, whereas, in a long path, due to interference, hidden-node problem and STDMA, ε is a little unstable. But, error rates of the original and Max-Min when $N_{path\ 3}=9$ are high up to 40% and 20%, respectively. Because the original deals with no hidden nodes and STDMA time slots, and Max-Min optimistically considers that throughput will theoretically reach a steady state when number of hops is beyond 4. In fact, when number of nodes exceeds 4 as stated in Max-Min [6] the measured throughputs decrease slightly rather than keep steady.

Figure 15 illustrates the relation between p_{rem} and $p_{threshold}$ given cases 1 and 2. It is clear that original and Jun's methods under-estimate $p_{rem\ s}$, since they omit T_{STDMA} effect. Max-Min and MIR over-estimate $p_{rem\ s}$. The reason is mentioned above. That is why the former (latter) two's $p_{rem\ s}$ are higher (lower) than real values measured.

4.2 Throughputs

According to Path_C algorithm, we inhibit a node to transmit data when accumulated $p_{rem} > p_{threshold}$. Without losing its generality, in the second experimentwe use the chain-topology WMN shown in Figure 16 as our experimental network platform in which data are transmitted through three path segments to their common destination node J. In case 1, as stated above the three paths are $N_{path\ 1} = 3$ (starts transmission at time = 0, data rate 1 = 512Kbps and source node=G), $N_{path\ 2} = 5$ (starts transmission at time = 50 sec, data rate 2 = 400Kbps and source node=E), and $N_{path\ 3} = 9$ (starts transmission at time =100 sec, data rate 3 = 300Kbps and source node=A). Figure 17 plots their individual throughputs, and total throughputs of the three paths. During simulation, the third path is allowed to transmit data, because $p_{rem} = \sum_{i=1}^{3} \frac{current\_data\_rate}{Thr_{path\ i}^{\max}(start\_sequence)} = \frac{0.512}{1.768} + \frac{0.4}{1.120} + \frac{0.3}{1.002} = 0.29 + 0.36 + 0.3 = 0.95$ is smaller than or equal to $p_{threshold}$, where $Thr_{Path}^{\max}s$ for paths 1, 2 and 3 are $\frac{8*2000}{3*3016.909}(=1.768Mbps)$, $\frac{8*2000}{(3016.909+5618.791+5650)}(=1.120Mbps)$ and $\frac{8*2000}{(3016.909+6751.155+6200)}(=1.002Mbps)$ according to (5), (8) and (9), respectively, and the $p_{threshold}$ given is 0.95. The test environment of case 2 is the same as that of case 1. The only difference is that data rate of the third path is higher than 300Kbps (e.g., 310Kbps). Hence, $p_{rem} = 0.29 + 0.36 + 0.31 = 0.96$ is now higher than the given $p_{threshold}$, resulting in severe interference (see Figure 18). Hence, node A in case 2 has to reduce its data rate to $Thr_{rem}(A)$ (=300Kbps); otherwise its total throughputs are lower than those shown in Figure 17. The test environment of case 3 is the same as that of case 1. The differences are that data rate of the second path is 512Kbps, and the third path is 1024Kbps. When the third path starts to transmit data, the traffic load exceeds the bandwidth, $p_{rem} = 0.29 + 0.46 + 1.02 = 1.77$. As shown in Figure 19, $N_{Path\ 3}$ contributes small throughputs to total throughputs. The test environment of case 4 is the same as that of case 1. The differences are that data rate of the first path is 1Mbps, and the second path is 768Kbps). When the second path starts to transmit data, the traffic load exceeds the bandwidth,

$p_{rem} = 0.57 + 0.69 + 0.3 = 1.56$. As shown in Figure 20, throughputs of $N_{Path\ 3}$ are almost zero and $N_{Path\ 2}$ contributes small throughputs.

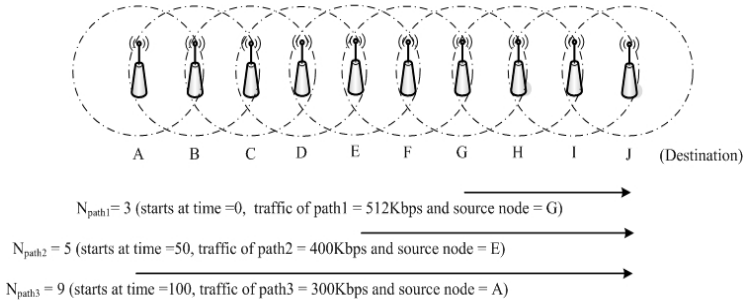

Figure 16. Three paths of a chain topology send data to their common destination node J at different times and with different data rates (source nodes of paths 1, 2 and 3 are nodes G, E and A, respectively)

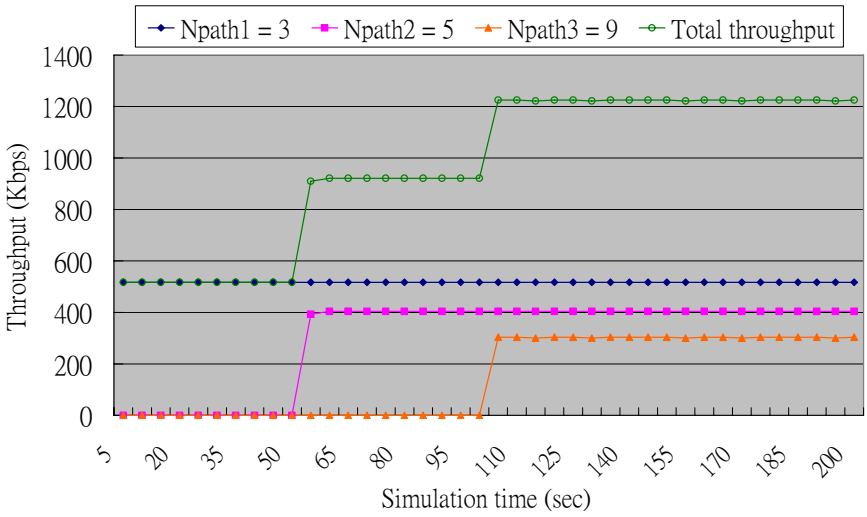

Figure 17. Case 1: when $N_{path\ 3} = 9$ starts to transmit data and $p_{rem} \leqq p_{threshold}$, where $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ individually start at 0th, 50th, and 100th sec with 512Kbps, 400Kbps and 300Kbps data rates, respectively.

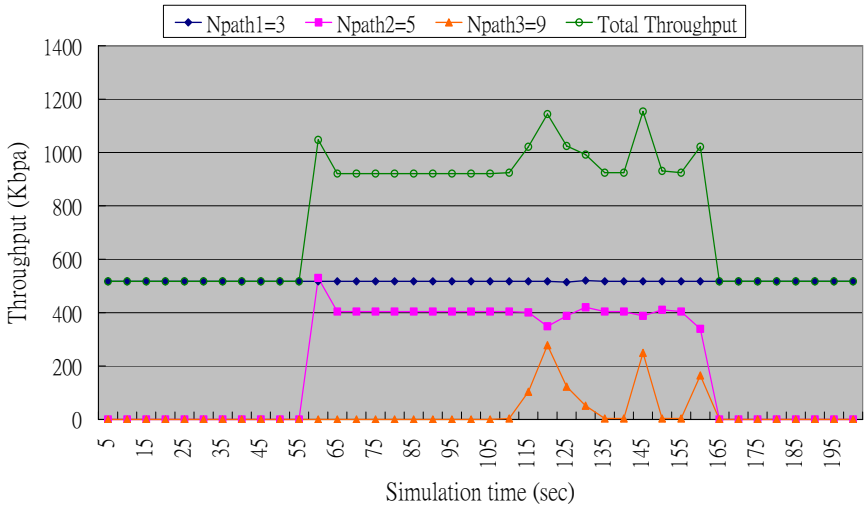

Figure 18. Case 2: when $N_{path\ 3} = 9$ starts to transmit data and $p_{rem} > p_{threshold}$, where $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ individually start at 0^{th}, 50^{th}, and 100^{th} sec with 512Kbps, 400Kbps and 310Kbps data rates, respectively.

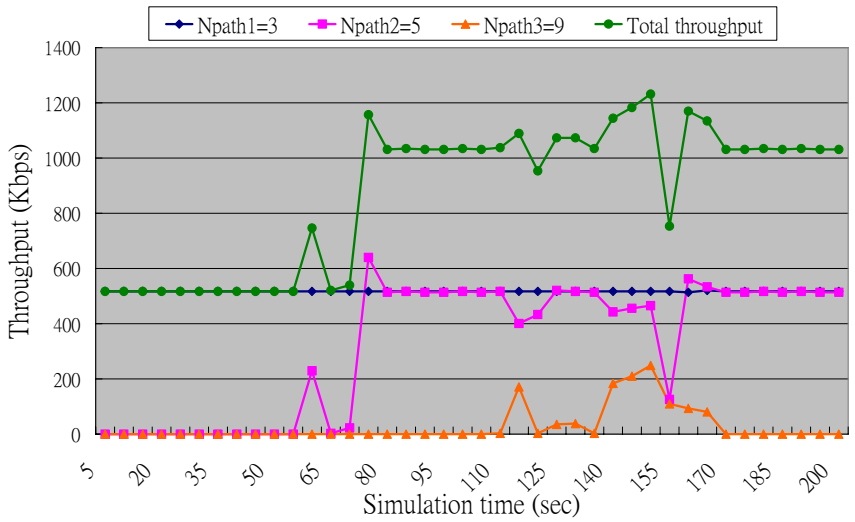

Figure 19 Case 3: when $N_{path\ 3} = 9$ starts to transmit data and $p_{rem} > p_{threshold}$, where $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ individually start at 0^{th}, 50^{th}, and 100^{th} sec with 512Kbps, 512Kbps and 1024Kbps data rates, respectively.

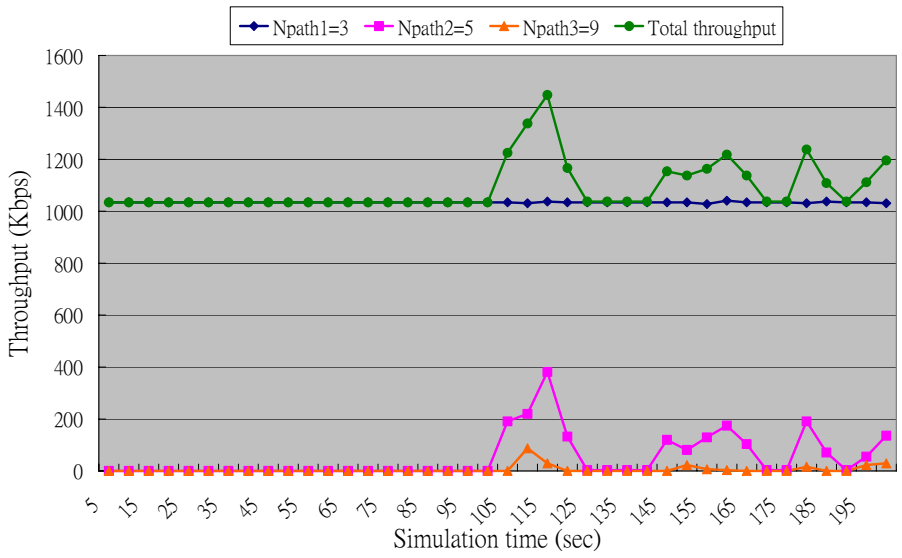

Figure 20 Case 4: when $N_{path\ 3}=9$ starts to transmit data and $p_{rem} > p_{threshold}$, where $N_{path\ 1}=3$, $N_{path\ 2}=5$ and $N_{path\ 3}=9$ individually start at 0th, 50th, and 100th sec with 1024Kbps, 768Kbps and 300Kbps data rates, respectively.

Figure 18 show that severe interference at 165 seconds even if the data rate of $N_{path\ 3}$ is only 310kbps.

Table 5 $N_{path\ 3}$, $N_{path\ 2}$ and $N_{path\ 1}$ start transmitting their packets to their common destination node J at 0th, 50th and 100th second from system starts up

path		Initial data rate (Mbps)	Data rates (Mbps)			
			0-50 sec	50-100 sec	100-150 sec	150-200 sec
1	N_{path3}	1.5	0.748	0.277	0.019	0.070
	N_{path2}	0.4	0	0.4	0.397	0.369
	N_{path1}	0.512	0	0	0.512	0.512
2	N_{path3}	1.5	0.753	0.021	0	0
	N_{path2}	1.5	0	0.937	0.0009	0.004
	N_{path1}	1.5	0	0	1.5	1.5
3	N_{path3}	1	0.746	0.253	0.065	0.026
	N_{path2}	0.4	0	0.4	0.397	0.4
	N_{path1}	0.512	0	0	0.512	0.512

In the second portion of the second experiment, topology shown in Figure 16 is also deployed. But we reverse start times of the three path segments. The start times of $N_{path\ 3}$, $N_{path\ 2}$ and $N_{path\ 1}$ are at 0th, 50th and 100th sec, respectively.

Table 5 shows the experimental results. We can see that even though $N_{path\ 3}$ is the first node that transmits data to the destination node, when the path is saturated, its throughputs sharly decrease. Its bandwidth is preempted by nodes close to the destination node. We call this phenomena neighbor-contention effect, which is resulted from the fact that $N_{Path\ 3}'s$ ACK delivery is longer than $N_{Path\ 1}'s$ and $N_{Path\ 2}'s$.

Throughputs on Faulty Environments

In the third portion, experiments are conducted on a faulty environment in which error rates of a path are increased from 0.02 to 0.2. Figure 21 shows the accumulated error rates along a path when error rate of a node is 0.9. We recover the three path segments' transmission sequence, i.e., $N_{path\ 1}$ goes the first, and $N_{path\ 3}$ the last.

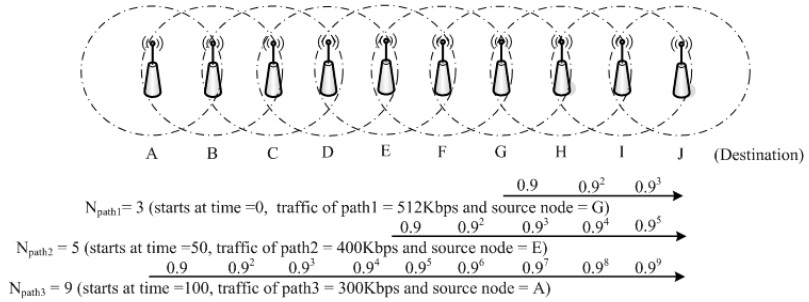

Figure 21. When error rate is 0.1, packet delivery rate between nodes A and B is 0.9, and that between nodes B and C is 0.9*0.9 and so on.

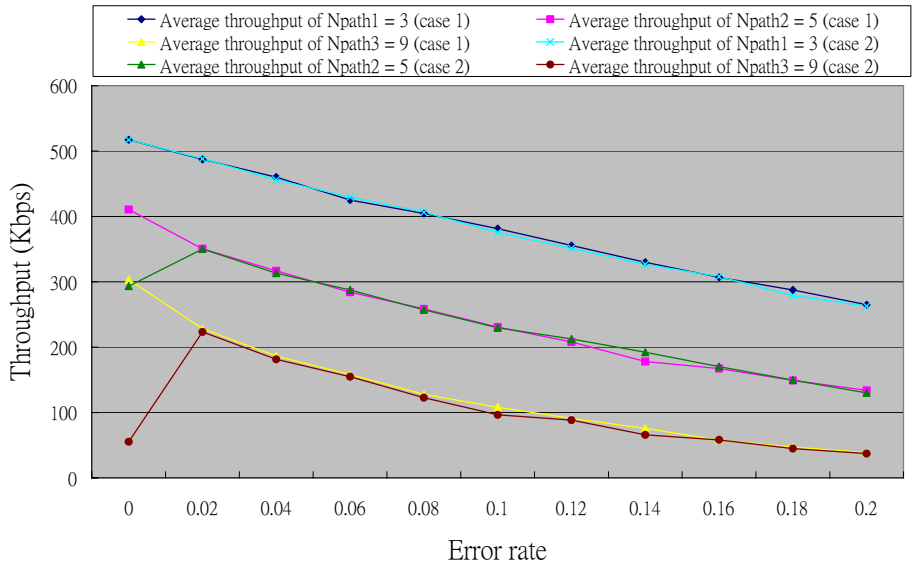

Figure 22.Average throughputs of $N_{path\ 1}=3$, $N_{path\ 2}=5$ and $N_{path\ 3}=9$ in cases 1 and 2, when error rates increase from 0 to 0.2

Figure 22 show that average throughputs of $N_{path\ 1}=3$, $N_{path\ 2}=5$ and $N_{path\ 3}=9$ in cases 1 and 2, when error rates range between 0 to 0.2. We can see that from error rate = 0.02 to error rate =0.2, throughputs of case 1 and case 2 are almost the same, no matter originally $p_{rem} \le p_{threshold}$ or $p_{rem} > p_{threshold}$ on error rate =0. This because error rate = 0.02 is sufficient to reduce traffic to a specific level to make the path unsaturated.

4.3 Loss rate

Table 6. Packet loss rates of case 1 (error rate =0)

path \ Loss rate	Packets sent (packets)	Packets received (packets)	Loss rate (%)
$N_{path\ 1}$	$6400\,((6400/200)*(2000*8)\simeq 512kbps)$	6400	0
$N_{path\ 2}$	$3751\,((3751/150)*(2000*8)\simeq 400kbps)$	3749	0.053319115
$N_{path\ 3}$	$1875\,((1875/100)*(2000*8)\simeq 300kbps)$	1875	0

Table 7. Packet loss rates of case 2 (error rate =0)

path \ Loss rate	Packets sent (packets)	Packets received (packets)	Loss rate (%)
$N_{path\ 1}$	$6400\,((6400/200)*(2000*8)\simeq 512kbps)$	6400	0

$N_{path\ 2}$	3751 ($(3751/150)*(2000*8) \simeq 400kbps$)	3646	2.79925
$N_{path\ 3}$	1938 ($(1938/100)*(2000*8) \simeq 310kbps$)	907	53.19917

Table 8. Packet loss rates of case 1 (error rate = 0.1)

Loss rate / path	No. of packets sent (packets)	No. of packets received theoretically V_{theo} (packets)	No. of packets received really V_{real} (packets)	Theoretical loss rate (%)	Real loss rate (%)	Error rate = $\left\|\frac{V_{theo}-V_{real}}{V_{theo}}\right\|$
$N_{path\ 1}$	6400	4665.6 ($=0.9^3*6400$)	4718	27.1	26.28	1.12%
$N_{path\ 2}$	3751	2214.93 ($=0.9^5*3751$)	2140	40.95	42.95	3.38%
$N_{path\ 3}$	1875	726.41 ($=0.9^9*1875$)	665	61.26	64.53	8.45%

Table 9. Packet loss rates of case 2 (error rate = 0.1)

Loss rate / path	No. of packets sent (packets)	No. of packets received theoretically V_{theo} (packets)	No. of packets received V_{real} (packets)	Theoretical loss rate (%)	Real loss rate (%)	Error rate = $\left\|\frac{V_{theo}-V_{real}}{V_{theo}}\right\|$
$N_{path\ 1}$	6400	4665.6 ($=0.9^3*6400$)	4645	27.1	26.28	0.44%
$N_{path\ 2}$	3752	2214.93 ($=0.9^5*3752$)	2130	40.95	42.95	3.83%
$N_{path\ 3}$	1938	750.82 ($=0.9^9*1938$)	596	61.26	64.53	20.62%

Tables 6 and 7 show results of the third experiment, i.e., packet loss rates of cases 1 and 2. It is evident that packet loss rates of case 1 are not better than those of case 2 on error rate =0. The reason is that packets issued by $N_{path\ 3}$ severely interfere path 2. In the following, we evaluate of a chain-topology WMN's loss rate on a faulty environment. In case 1, when error rate = 0.1, theoretical delivery rates of paths 1, 2 and 3 as shown in Table 8 are $\frac{6400-4665.6}{6400}=27.1\%$, $\frac{3751-2214.93}{3751}=40.95\%$ and $\frac{1875-726.41}{1875}=61.26\%$, respectively. Their error rates between theoretical number of packets received V_{theo} and real number of packets received V_{real}, defined as $\left|\frac{V_{theo}-V_{real}}{V_{theo}}\right|$, are 1.12%, 3.38% and 8.45%, respectively. They are all less than 10%.

Table 9 lists the results of the second portion of the third experiment on case 2 when error rate = 0.1. It is evident that error rate between the theoretical and real

numbers of packets received for $N_{Path\ 3}$ is high up to 20.62% (> 8.45% in Table 8) due to interference.

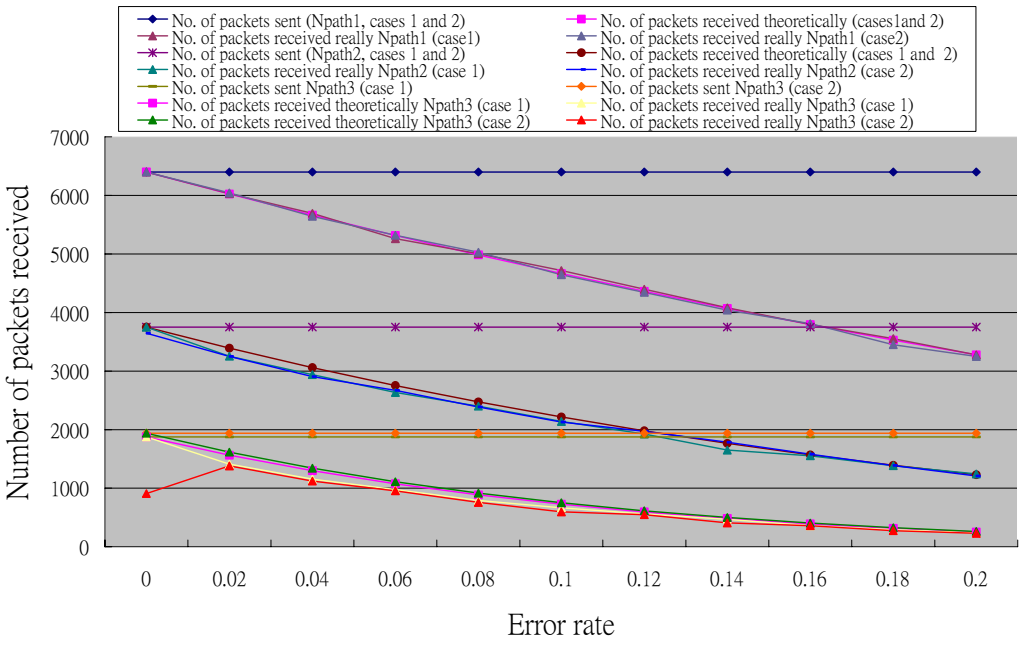

Figure 23. Number of packets received theoretically and really when $N_{path\ 1} = 3$ in cases 1 and 2

Figures 23 illustrates experimental results on different error rates. We can see that the simulation results are close to their theoretical values for $N_{Path\ 1}$ and $N_{Path\ 2}$ both in cases 1 and 2. Also, in case 2 number of packets received on $N_{path\ 3} = 9$, when error rate = 0, is lower than that when error rate = 0.02, because packets are dropped on error rate = 0.02. When packets sent by node A arrive at node E, data rate is 1787.55 $(=1938*0.98^4)\ packets/\text{sec}$ which is lower than the threshold 1869.2 $(\simeq 1938*0.991^4)$ so that no interference on node E occurs, i.e., error rate = 0.991 will make $p_{rem} = 0.95$.

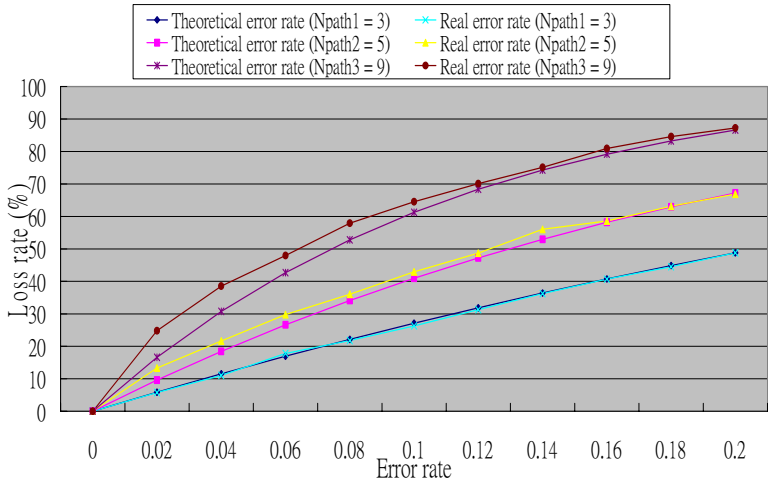

Figure 24. The error rates of theoretical and real values in case 1

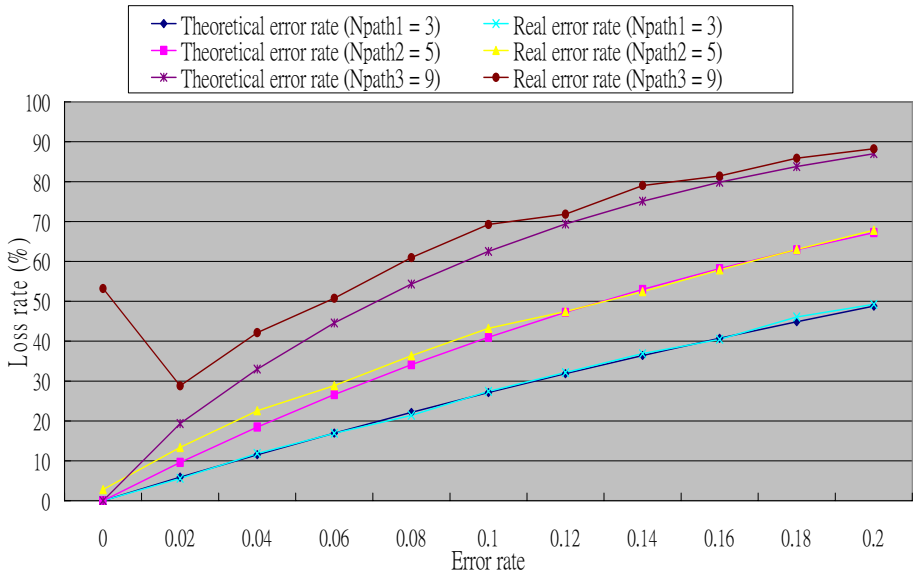

Figure 25. The error rates of theoretical and real values in case 2

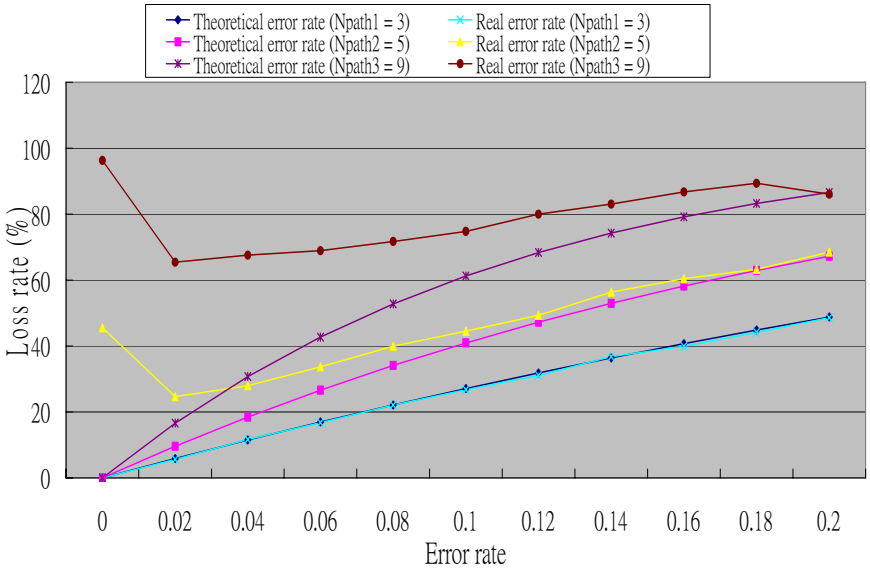

Figure 26. The error rates of theoretical and real values in case 3 (512Kbps、512Kbps、1024Kbps)

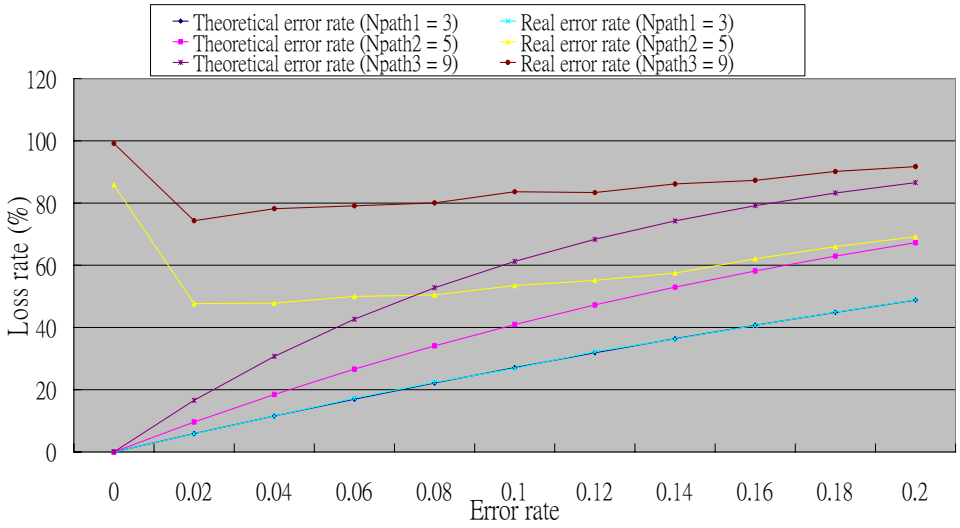

Figure 27 The error rates of theoretical and real values in case 4 (1Mbps、768Kbps、300Kbps)

Figures 24 and 25 show that when error rate = 0, $N_{path\ 3} = 9$'s packet loss rates in case 2 are far higher than those in case 1, since the path is saturated. Some packets are then dropped. Figure 25 shows that when $0.02 \leqq$ error rate $\leqq 0.06$, $N_{path\ 3} = 9$'s packet loss rates are smaller than those on error rate = 0. Figure 26 shows that in case

3 $N_{path\ 2}=5$'s packet loss rate on error rate = 0 is higher than those on 0.02≦error rate≦0.1. $N_{path\ 3}=9$'s packet loss rate on error rate = 0 is almost 100% and is higher than those when error rates ranging from 0.02 to 0.2. Figure 27 shows that when error rate = 0, $N_{path\ 2}=5$ and $N_{path\ 3}=9$'s packet loss rates in case 4 are respectively higher than those when error rate = 0.2. Now, we can conclude that packet loss rates are caused by tow factors, error rates we set and signal interference/collision, and different levels of interference will result in different levels of packet loss rates when the path length is the same, e.g., for $N_{path\ 2}=5$ and $N_{path\ 3}=9$, real packet loss rates shown in Figure 27 due to more severe interference/collision are higher than those in Figure 26 . Further, a node far away from the destination node has higher packet loss rates also because of higher probability of interference/collision.

4.4 Jitters

Table 10. Jitters of the two cases tested

Jitters / path	Average jitter of case 1	Average jitter of case 2
$N_{path\ 1}$	-1.8E-05	-3.8e-06
$N_{path\ 2}$	-2.5E-05	-0.00137
$N_{path\ 3}$	-0.000102344	-0.02771

From Table 10, we can realize that the three paths' jitters in case 2 are higher than those in case 1.

Table 11. Jitters of the two cases tested (error rate = 0.1)

Jitters / path	Average jitter of case 1	Average jitter of case 2
$N_{path\ 1}$	-1.3E-05	2.33E-05
$N_{path\ 2}$	1.15E-05	-6E-05
$N_{path\ 3}$	2.65E-05	0.000162

In the fourth experiment, we used multicast loss model [20], in which lost packets will not be retransmitted. Tables 10 and 11 list average jitters on error rate = 0

and error rate = 0.1, respectively, for both cases 1 and 2. Average jitters shown in Table 10 are higher than those listed in Table 11 also due to higher probability of interference/collision. Figures 28 and 29 show average jitters of cases 1 and 2, respectively. Figure 28 illustrates that jitters of $N_{Path\ 3}$ are higher than those of $N_{Path\ 1}$ and $N_{Path\ 2}$. From the two figures, we can realize that when error rate = 0, $N_{path\ 2}$'s and $N_{path\ 3}$'s jitters in case 2 are respectively higher than those in case 1, also due to path saturation which causes more interference/collision. Jitters on error rate = x both in cases 1 and 2 are very close, where x = 0.02, 0.04 …0.1 …0.2. Figures 30, 31 and 32 respectively illustrate maximum jitters, minimum jitters and standard deviations of both cases 1 and 2. Here, we can conclude that jitters and standard deviations of a node far away from the destination node are higher than those of a node near the destination node. The reason is that longer transmission delay and high probability of interference/collision cause higher jitters.

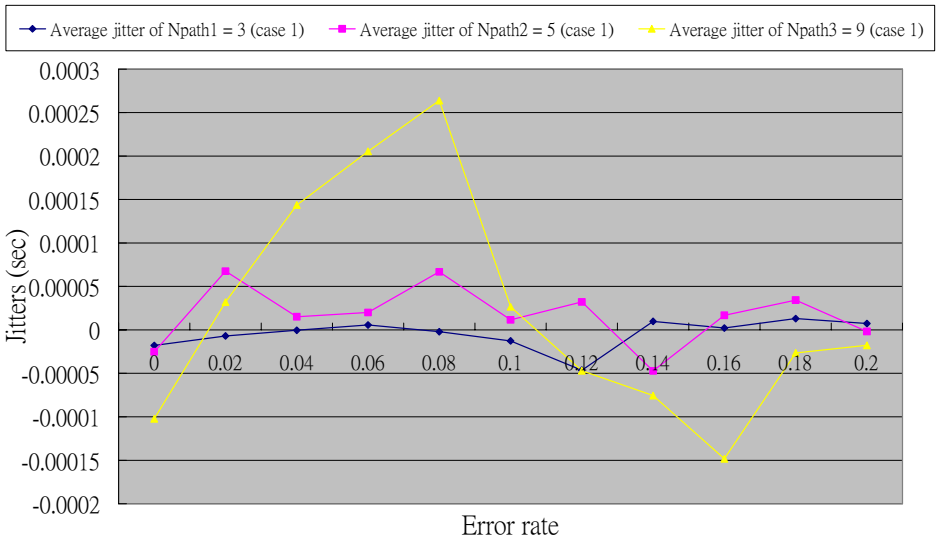

Figure 28. Average jitters of $N_{path1} = 3$, $N_{path2} = 5$ and $N_{path3} = 9$ in cases 1

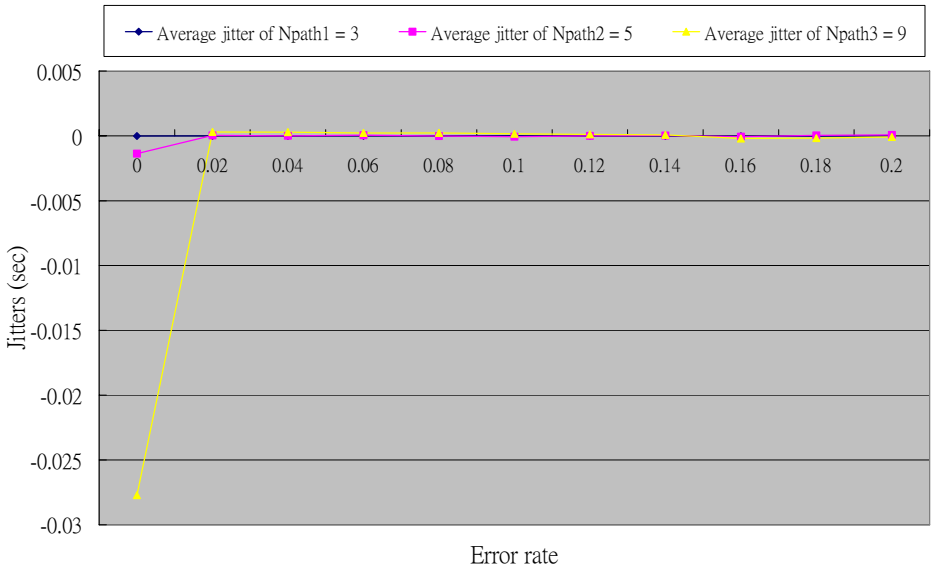

Figure 29. Average jitters of $N_{path1} = 3$, $N_{path2} = 5$ and $N_{path3} = 9$ in cases 2

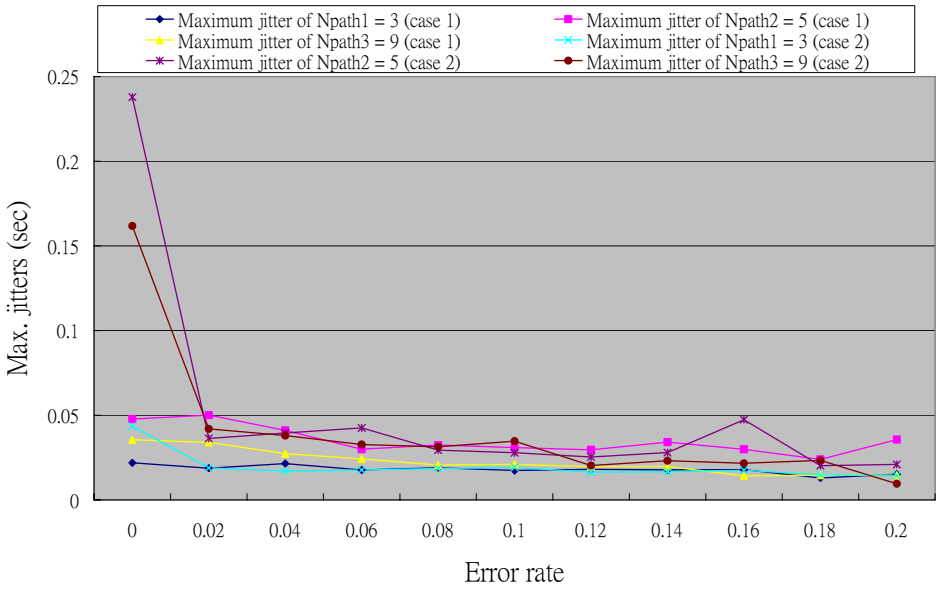

Figure 30. Maximum jitters of $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ in cases 1 and 2

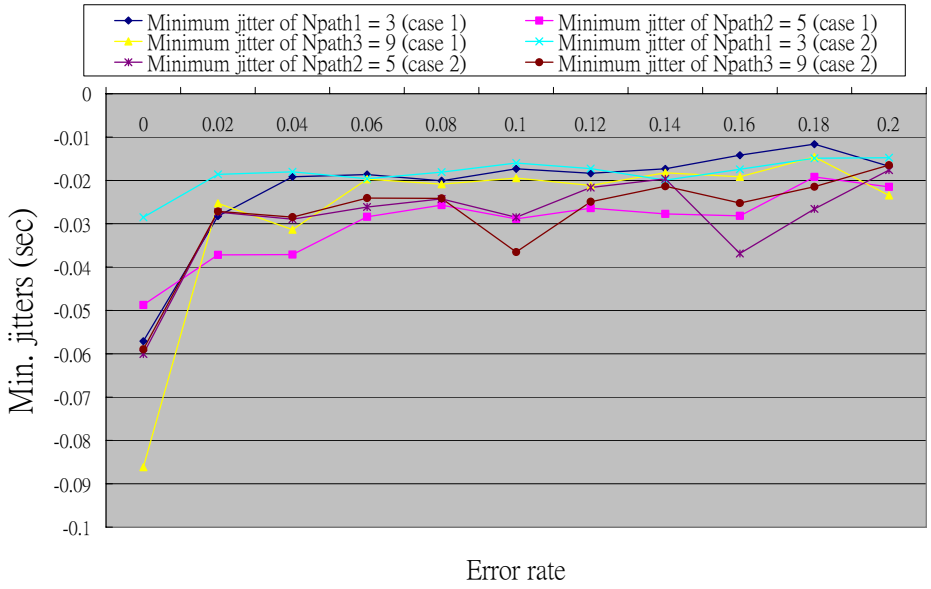

Figure 31. Minimum jitters of $N_{path\ 1}=3$, $N_{path\ 2}=5$ and $N_{path\ 3}=9$ in cases 1 and 2

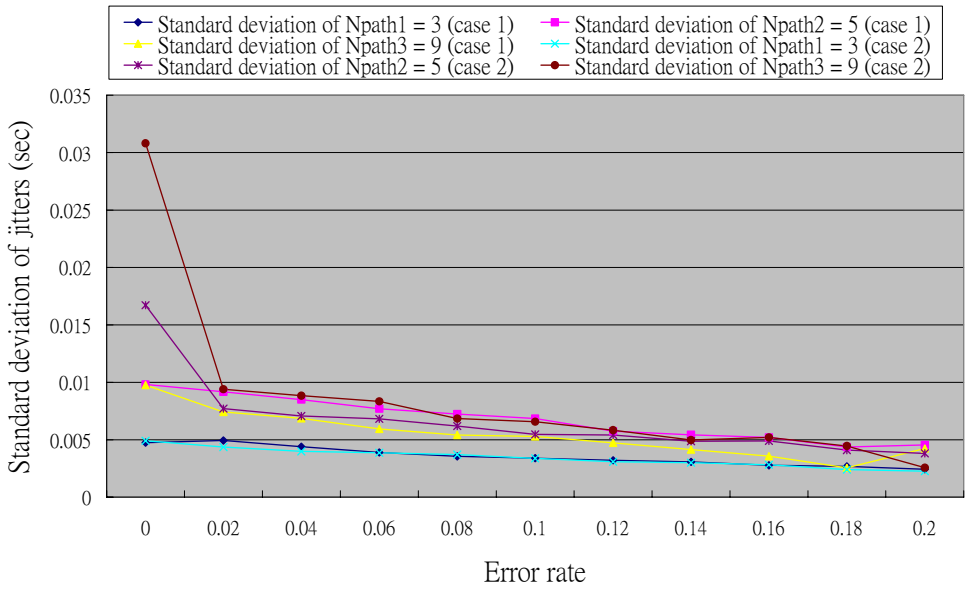

Figure 32. Standard deviation of jitters when $N_{path\ 1}=3$, $N_{path\ 2}=5$ and $N_{path\ 3}=9$ in cases 1 and 2

4.5 Transmission Delay

The fifth experiment deals with transmission delays. Figure 33 shows that the transmission delays of $N_{path\ 2}=5$ and $N_{path\ 3}=9$ are respectively about 5/3 times and

3 times that of $N_{path\ 1} = 3$. From Figures 33 to 35, we can see that a higher total data rate results in more severe interference/collision which will cause longer transmission delay. The delays approach to almost a constant when no network segment is saturated, i.e., a higher data rate will result in a higher error rate value where delay time starts approaching to a constant. For example, transmission delays of the three path segments as shown in Figure 34, 35 and 36 (respectively cases 2, 3 and 4) restore to the delays of the three path segments in case 1 on 0.02, 0.08 and 0.14, respectively. Also, when error rate =0, transmission delays of the three path segments in case 2 are respectively longer than those in case 1. And the delays in cases 1 and 2 on the same error rate x are almost the same where x=0.02, 0.04 …0.2. Figures 37 to 39 illustrate maximum and minimum delays and their standard deviations when cases 1 and 2 are involved.

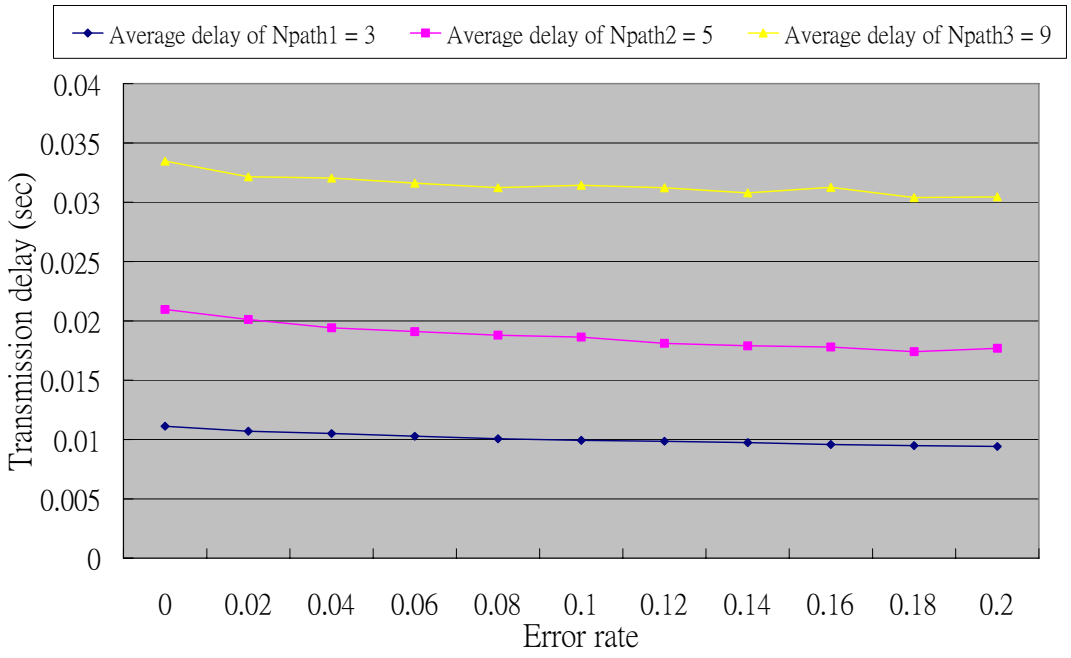

Figure 33. Average transmission delay of $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ in cases 1

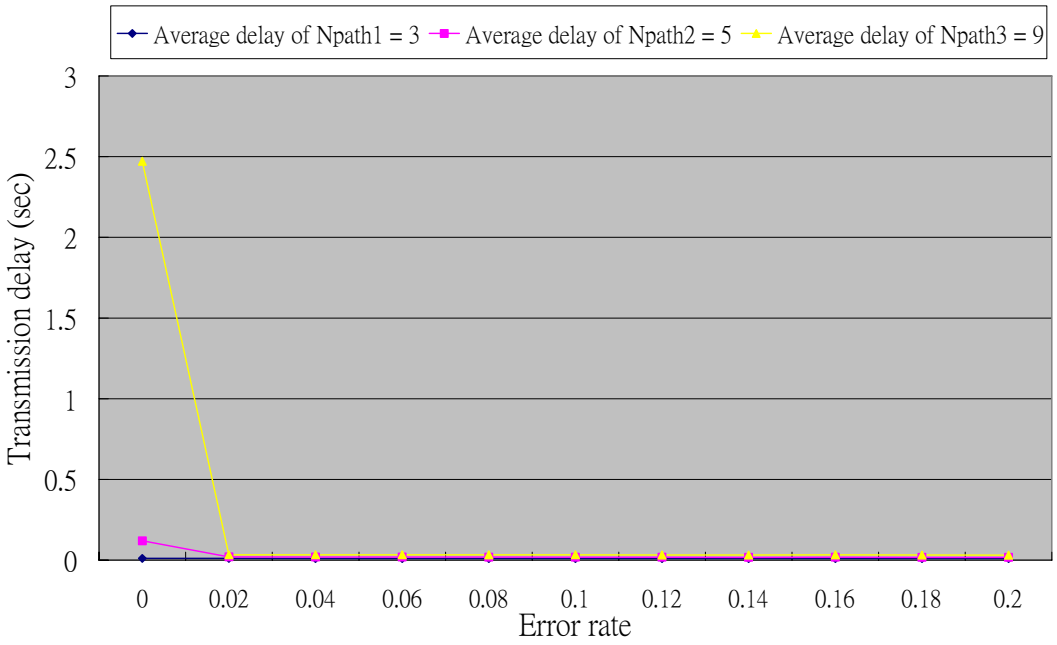

Figure 34. Average transmission delay of $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ in cases 2

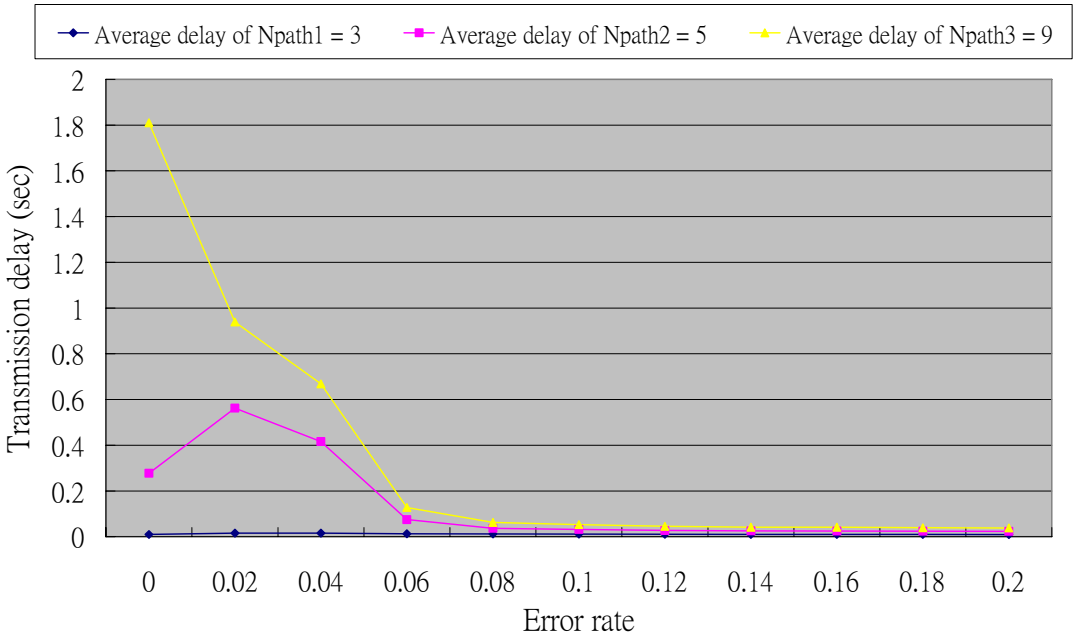

Figure 35. Average transmission delay of $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ in cases 3 (512Kbps、512Kbps、1024Kbps)

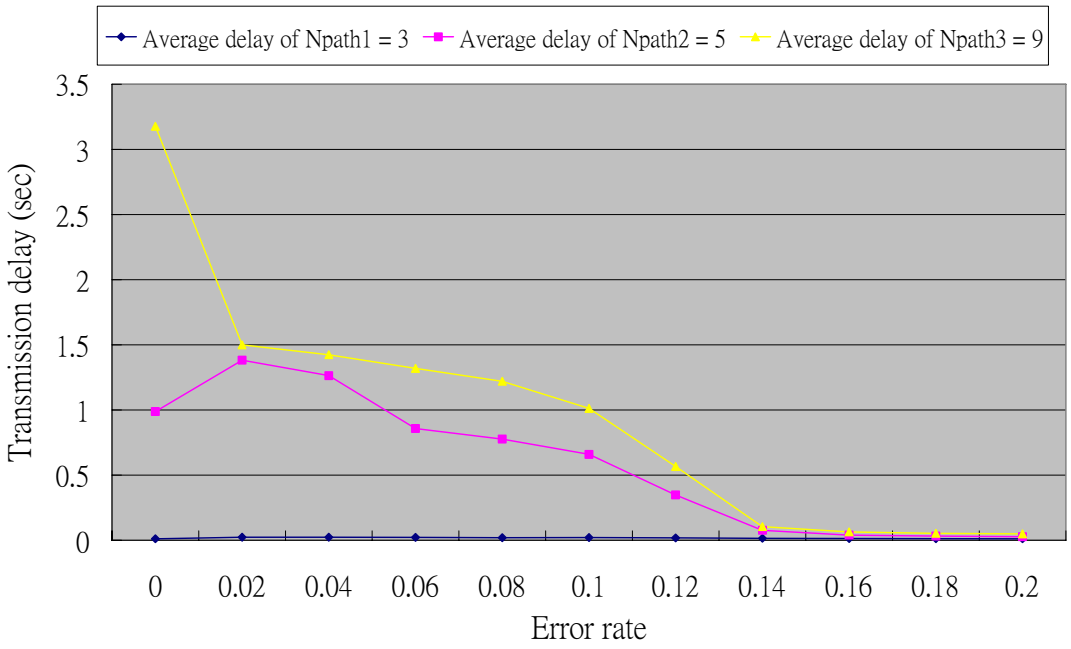

Figure 36. Average transmission delay of $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ in cases 4 (1M、768Kbps、300Kbps)

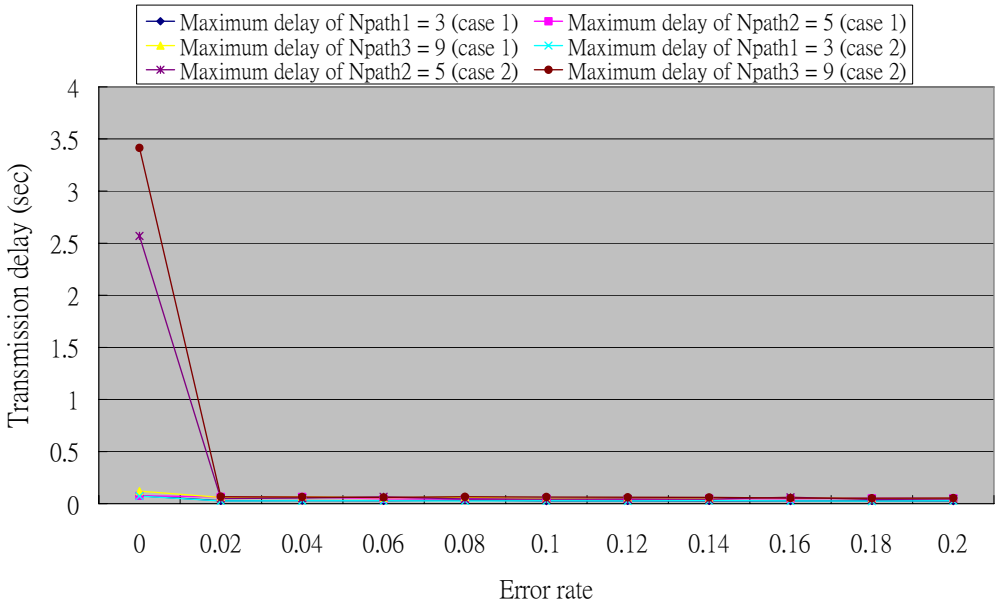

Figure 37. Maximum transmission delay of $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ in cases 1 and 2

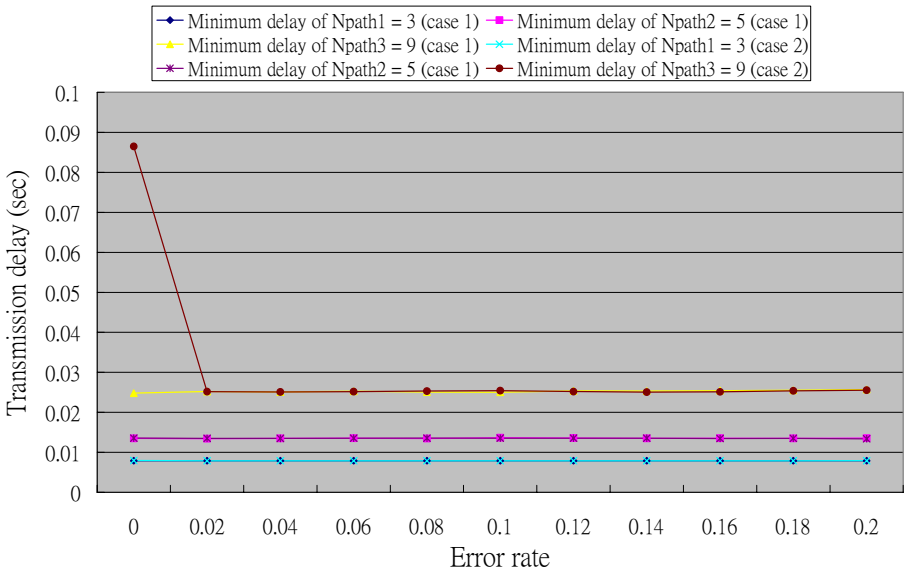

Figure 38. Minimum transmission delay of $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ in cases 1 and 2

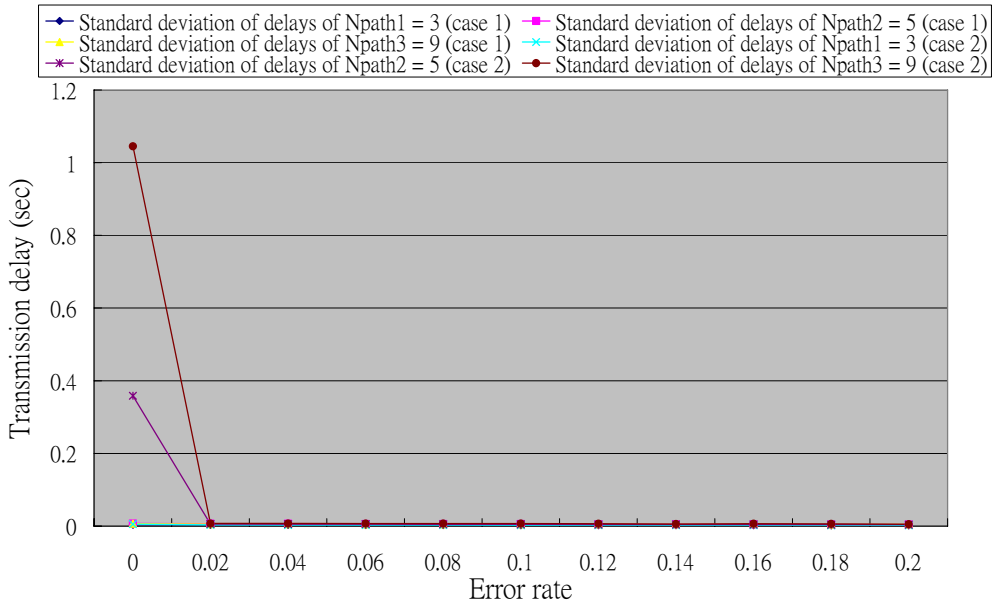

Figure 39. Standard deviation of transmission delays when $N_{path\ 1} = 3$, $N_{path\ 2} = 5$ and $N_{path\ 3} = 9$ in cases 1 and 2

4.6 Throughput comparison of multi-frequency approaches

Figure 40 shows that a path of $N_{path} = 4$ starts to transmit data in a four-channel

path which deploys four types of mesh routers as a set. Each router, A, B, C, or D, transmits data with a channel which is different from the other three routers' channels. Channels do not interfere one another. Its throughput is $Thr_{Path}^{\max} = \frac{L_{DATA}}{T_{pac}}$. In general, $N_{in} = 3$. But by adding a hidden node, number of nodes influenced by a node is four. Hence, four channels will result in an interference-free communication environment. The disadvantage is that when due to some reason users want to insert an extra node to the path, channels of all nodes along one of the two sides should be readjusted. Of course, a set of four nodes with different channels can be inserted directly, but their distance should be lengthened to avoid any interference with existing nodes, and cost is another consideration.

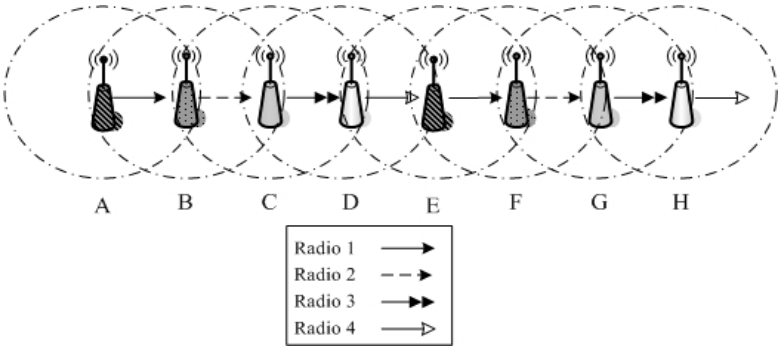

Figure 40. A four-channel framework

shows that when $N_{path} = 4$ throughput of a four-channel path is four times that of single channel. RMP's throughput is two times that of single channel, because two channels can transmit data at the same time with different radios. Table 12 lists single-channel's, RMP's and four-channel's transmission delays and jitters. We can see that when error rate is larger, transmission delay will be shorter, regardless of whether single-channel, RMP or Four-channel is in use. This is because less contention occurs along the routing paths. When transmission delay is also longer, jitter is also larger. The reason is stated above. Further, their throughputs approximate to $(error\ rate)^4 * Tho\ i$, where . $Tho\ i$ is throughput on error rate =0 and i may be single-channel, RMP or Four-channel. We can also see that four-channel performs the

best. Here, we can conclude that bottleneck of a chain-topology WMN is severe among four consecutive nodes when single-channel is used.

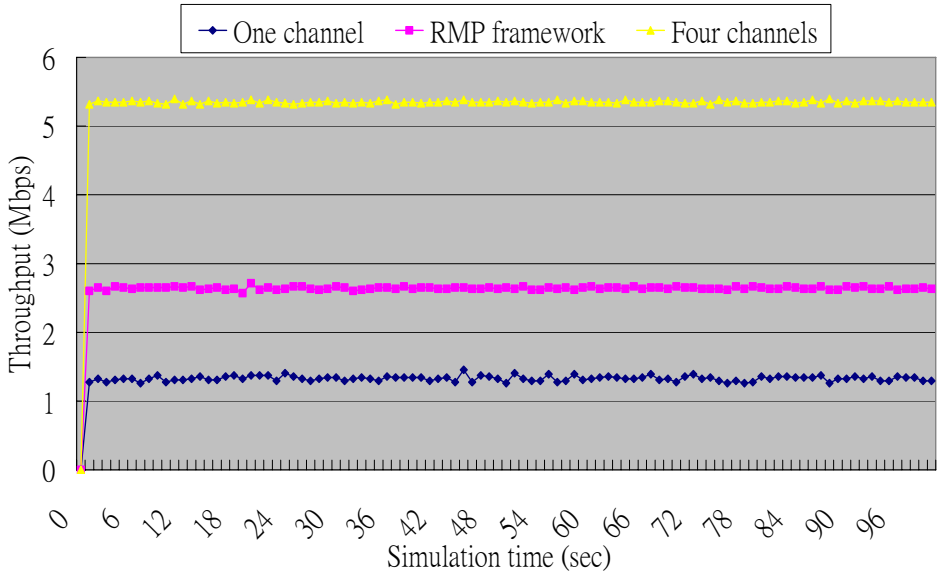

Figure 41. Throughputs of three different frameworks including single-channel, RMP and four-channel frameworks when $N_{path} = 4$.

Table 12 Average transmission delays and jitters of three different frameworks tested

framework / QoS	Error rate	Single-channel	RMP	Four-channel
Transmission delay	0	0.65904	0.451	0.205
	0.04	0.65046	0.423	0.201
	0.08	0.64965	0.396	0.198
	0.12	0.64913	0.376	0.196
	0.16	0.6452	0.342	0.191
	0.2	0.6431	0.316	0.189
Jitters	0	0.002027	0.001407	0.000641756
	0.04	0.002004	0.001396	0.000641701
	0.08	0.001996	0.001371	0.000641305
	0.12	0.001981	0.001341	0.000641137
	0.16	0.001943	0.001305	0.000640923
	0.2	0.001907	0.001296	0.000640218
Throughput Mbps	0	1.32819	2.641998	5.350576
	0.04	1.152	2.2507	4.5779

	0.08	0.9676	1.927	3.804
	0.12	0.8279	1.613	3.2023
	0.16	0.7072	1.339	2.6835
	0.2	0.6068	1.1018	2.2077

Chapter 5 Conclusion and Future Research

In this paper, we propose a capacity-computation model that calculates the maximum throughput of a chain-topology WMN. We also propose Path_C algorithm to determine whether a path segment can transmit data with its maximum data rate or not so that transmission on the path segment will not interfere data transmission on other segments, and be interfered by other segments. We inhibit a node to start transmitting its data when accumulated $p_{rem} > p_{threshold}$. Such can avoid signal interference among nodes and help to dramatically increase throughput of a chain-topology WMN. Theoretically, $p_{threshold}$ in the Path_C algorithm is 1, but its real value measured is 0.95.

Simulation results reveal that error rates of throughputs between simulated by using ns2 and calculated by deploying (5), (8) or (9) are all less than 5%. Loss rates, jitters and delays of a chain-topology WMN are also evaluated in this study. Finally, a single-channel, RMP and four-channel WMNs are also studied to show that signal interference on a path of single channel is severe.

In the future, we would like to develop the Path_C algorithm for a general topology WMN. This time characteristics of an articulation node should be carefully studied. Deriving the mathematical models for behaviors and reliability of such a system so that users can predict and realize its behaviors and it reliability, respectively, with the models is also one of our future works.

References

[1] N.S. Nandiraju, D.S. Nandiraju and D.P. Agrawal, “Multipath Routing in Wireless Mesh Networks,” IEEE International Conference on Mobile Ad hoc and Sensor Systems, 2006, pp. 741 – 746.

[2] Y.H. Yuan, H.M. Chen and M. Jia, “An Optimized Ad-hoc On-demaind Multipath Distance Vector (AOMDV) Routing Protocol”, Asia-Pacific Conference on Communications, 2005, pp. 569–573

[3] X. Li and L. Cuthbert, “On-demand Node-Disjoint Multipath Routing in Wireless Ad hoc Networks,” IEEE International Conference on Local Computer Networks, 2004, pp. 419 – 420.

[4] X. Hu and M.J. Lee, “An Efficient Multipath Structure for Concurrent Data Transport in Wireless Mesh Networks,” Computer Communications, vol. 30, issue 17, Nov. 2007, pp. 3358-3367.

[5] S. Rai, O. Deshpande, O. Canhui, C.U. Martel and B. Mukherjee, “Reliable Multipath Provisioning for High-Capacity Backbone Mesh Networks,” IEEE/ACM Transactions on Networking, vol. 15, issue 4, Aug. 2007, pp. 803 – 812.

[6] B. Aoun and R. Boutaba, “Max-Min Fair Capacity of Wireless Mesh Networks,” IEEE International Conference on Mobile Adhoc and Sensor Systems, 2006, pp. 21 – 30.

[7] K. Jain, J. Padhye, V. Padmanabhan and L. Qiu, “Impact of Interference on Multi-Hop Wireless Network Performance,” In ACM Annual International Conference on Mobile Computing and Networking, 2003, pp. 66–80.

[8] J. Jun and M.L. Sichitiu, “The Nominal Capacity of Wireless Mesh Networks,” IEEE Wireless Communications, vol. 10, issue 5, Oct. 2003, pp. 8 – 14.

[9] T. Scherer and T. Engel, “Bandwidth Consumption for Providing Fair Internet Access in Wireless Mesh Networks,” IEEE Communications Society on Sensor and Ad Hoc Communications and Networks, vol. 3, Sept. 2006, pp. 746 – 750.

[10] Y. M. Lu, D. Grace and P.D. Mitchell, “Capacity Evaluation of a Multi-hop Wireless Ad hoc Network Using Minimum Impact Routing,” IEEE International Conference on Wireless Communications, Networking and Mobile Computing, 2006, pp. 1 – 4.

[11] D.R. Guo, K. Wang and L.S. Lee, “Efficient Spatial Reuse in Multi-Radio,Multi-Hop Wireless Mesh Networks,” Vehicular Technology Conference, 2007, pp. 1076 – 1080.

[12] N.B. Salem and J.P. Hubaux, “Securing Wireless Mesh Networks,” IEEE Wireless Communications, vol. 13, issue 2, Apr. 2006, pp. 50 – 55.

[13] P. Gupta and P.R. Kumar, “The Capacity of Wireless Networks” IEEE Transactions on Information Theory, vol. 46, no. 2, Mar. 2000, pp.388 – 404.

[14] M. Grossglauser and D. Tse, “Mobility Increases the Capacity of Ad Hoc Wireless Networks,” IEEE/ACM Transactions on Networking, vol. 10, issue 4, Aug. 2002, pp. 477 – 486.

[15] W. Song and X. Fang, “Routing with Congestion Control and Load Balancing in Wireless Mesh Networks,” International Conference on ITS Telecommunications, 2006, pp. 719 – 724.

[16] R. Draves, J. Padhye and B. Zill, “Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks,” International Conference on Mobile Computing and Networking, 2004, pp. 114 – 128.

[17] Y.Q. Liu, W. Yan and Y.F. Dai, “A Path Capacity Analytical Model for Wireless Networks,” Journal of Software, vol.17, no.4, Apr. 2006, pp. 854-859.(in chinese)

[18] “IEEE 802.11 Wireless LAN,” http://totoro.cs.nthu.edu.tw/~com53100/WirelessLAN.ppt

[19] “RFC 3561 - Ad hoc On-Demand Distance Vector (AODV) Routing ,” July 2003, http://www.faqs.org/rfcs/rfc3561.html

[20] USC/ISI, “The Network Simulator - ns-2,” [Online]Available: http://www.isi.edu/nsnam/ns/