

ระบบทดสอบเครือข่ายแอตสอกไร้สายสำหรับประเมินการควบคุมแบบเวลาจริงในการส่งข้อมูลสื่อผสม



นางสาวพนิตา วีระวุฒิพล

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2548

ISBN: 974-17-4334-3

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

MANET TESTBEDS FOR EVALUATION OF REAL-TIME CONTROLS IN
MULTIMEDIA TRANSMISSION



Miss Panida Veeravuttiphol

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Electrical Engineering

Department of Electrical Engineering

Faculty of Engineering

Academic Year 2005

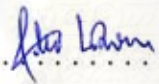
ISBN: 974-17-4334-3


หัวข้อวิทยานิพนธ์ ระบบทดสอบเครือข่ายแอตซอกไร้สายสำหรับประเมินการควบคุมแบบเวลาจริง
ในการส่งข้อมูลสื่อผสม
โดย นางสาวพนิดา วีระวุฒิพล
สาขาวิชา วิศวกรรมไฟฟ้า
อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.เชาวน์ดิศ อัสวกุล

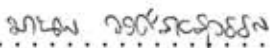
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้เป็น
ส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต



..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร. จิเรก ลาวัณย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์


..... ประธานกรรมการ
(อาจารย์ สุวิทย์ นาคพิระยุธ)


..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.เชาวน์ดิศ อัสวกุล)


..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.มานพ วงศ์สายสุวรรณ)


..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.ชัยเชษฐ์ สายวิจิตร)

พนิตา วีระวุฒิพล : ระบบทดสอบเครือข่ายแอตชอกไร้สายสำหรับประเมินการควบคุมแบบเวลาจริงในการส่งข้อมูลสื่อผสม (MANET TESTBEDS FOR EVALUATION OF REAL-TIME CONTROLS IN MULTIMEDIA TRANSMISSION) อ.ที่ปรึกษา : ผศ. ดร. เซวาน์ดิศ อัครกุล, 77 หน้า. ISBN: 974-17-4334-3.

วิทยานิพนธ์ฉบับนี้นำเสนอกลไกควบคุมแบบเวลาจริงสำหรับการส่งข้อมูลสื่อผสมในเครือข่ายแอตชอกเพื่อควบคุมให้จัตเตอร์ที่โหนดปลายทางมีค่าน้อยในขณะที่ค่าประวิงเวลาอยู่ในช่วงที่ยอมรับได้ โดยอาศัยการจัดการควบคุมที่โหนดระหว่างทาง และจะทดสอบกลไกที่นำเสนอในในระบบทดสอบ 2 รูปแบบ คือแบบไร้สายในกรณีเครือข่ายมีขนาดเล็ก ส่วนแบบมีสายจะใช้ได้ในสถานการณ์ที่เครือข่ายมีขนาดใหญ่ขึ้น รวมถึงพิจารณาการเคลื่อนที่ของแต่ละโหนดด้วย เนื่องจากอาศัยการเปลี่ยนแปลงของสัญญาณไร้สายผ่านทางตัวควบคุมที่เรียกว่า เกตเวย์เลียนแบบแทน โดยที่ในระบบทดสอบทั้งสองแบบจะใช้ข้อมูลต้นฉบับในการส่งเป็นข้อมูลวิดิทัศน์ที่ถูกบีบอัดตามมาตรฐาน MPEG-4 และส่งในรูปแบบของกราฟฟิกทั้งแบบอัตราบิตคงที่ และอัตราบิตแปรผันได้

กลไกการควบคุมแบบเวลาจริงที่นำเสนอในวิทยานิพนธ์ฉบับนี้ประกอบไปด้วย 3 วิธี คือ การควบคุมแบบเวลาจริงแบบกระจายโดยการควบคุมขีดเริ่มเปลี่ยนของบัฟเฟอร์เพื่อการปรับให้เป็นปกติ (Distributed Real-time Control with Regulating Buffer Threshold : DRC-RBT), การควบคุมแบบเวลาจริงแบบกระจายโดยใช้ป้ายระบุค่าประวิงเวลา (Distributed Real-time Control with Delay Time Tag : DRC-DTT) และ การควบคุมแบบเวลาจริงแบบกระจายที่ปรับตัวได้ (Adaptive Distributed Real-time Control : ADRC) โดยในวิธี DRC-RBT ใช้หลักการเก็บกลุ่มข้อมูลจำนวนหนึ่งไว้ในบัฟเฟอร์ก่อนจนเมื่อถึงเวลาที่เหมาะสมแล้วจึงค่อยทยอยส่งออกไป ซึ่งเวลานี้ก็ขึ้นอยู่กับค่าขีดเริ่มเปลี่ยนที่กำหนด ในขณะที่ DRC-DTT จะอาศัยการหน่วงเวลาเพิ่มให้กับกลุ่มข้อมูลที่มาเร็วเกินไปเมื่อเทียบกับค่าประวิงเวลาของกลุ่มข้อมูลก่อนหน้า ส่วน ADRC จะอาศัยการหน่วงเวลาเพิ่มสำหรับกลุ่มข้อมูลที่มาเร็วเกินไปเช่นกัน แต่จะคำนวณค่าประวิงเวลาที่ต้องหน่วงเพิ่มจากค่าประวิงเวลาเฉลี่ยของแต่ละกลุ่มข้อมูลแทน และสำหรับกลุ่มข้อมูลที่มาช้าเกินไปก็จะถูกตัดทิ้ง

ผลการทดสอบที่ได้แสดงให้เห็นว่า วิธี DRC-RBT จะทำงานได้ดีในเครือข่ายที่โหนดไม่เคลื่อนที่ และสนใจที่จัตเตอร์เป็นหลัก โดยรูปแบบของกราฟฟิกที่ส่งจะต้องเป็นแบบอัตราบิตคงที่ด้วย ในขณะที่วิธี DRC-DTT จะทำงานได้ดีในเครือข่ายที่มีการรบกวนเนื่องจากกลุ่มข้อมูลมีค่าประวิงเวลาสูงผิดปกติไม่มาก และมีอัตราการสูญเสียกลุ่มข้อมูลไม่มากด้วย ส่วนวิธี ADRC เนื่องจากใช้ค่าประวิงเวลาเฉลี่ยในการตัดสินใจ จึงทำให้สามารถทำงานได้ดีแม้เครือข่ายจะมีการรบกวนเกิดขึ้นก็ตาม โดยที่ประสิทธิภาพการทำงานจะขึ้นอยู่กับการใช้ค่าดังกล่าวหน้าให้เหมาะสมกับรูปแบบของแต่ละสถานการณ์ด้วย

ภาควิชา วิศวกรรมไฟฟ้า ลายมือชื่อนิสิต พนิตา วีระวุฒิพล
 สาขาวิชา วิศวกรรมไฟฟ้า ลายมือชื่ออาจารย์ที่ปรึกษา
 ปีการศึกษา 2548

467 07014 21 : MAJOR ELECTRICAL ENGINEERING

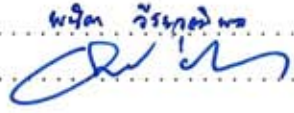

KEY WORD: MANET/ TESTBED/ REAL-TIME CONTROL/ JITTER.

PANIDA VEERAVUTTIPHOL : MANET TESTBEDS FOR EVALUATION OF REAL-TIME CONTROLS IN MULTIMEDIA TRANSMISSION. THESIS ADVISOR: ASST. PROF. CHAODIT ASWAKUL, Ph.D., 77 pp. ISBN: 974-17-4334-3.

In this thesis, real-time control mechanisms for multimedia transmission in an ad-hoc network have been proposed. The objective of these mechanisms is to minimize the packet jitter while maintaining the resultant packet delay in an acceptable range. Unlike existing algorithms, the proposed control algorithms are intended to operate at intermediate nodes. These algorithms have been tested in two platforms: wireless and wired testbeds. The implemented wireless testbed has been suitably applied in case of small networks. In contrast, by using an emulator gateway to model ad-hoc environments, the wired testbed has been applied in case of larger networks possibly with consideration of node mobility.

Three approaches of real-time control mechanisms, Distributed Real-time Control with Regulating Buffer Threshold (DRC-RBT), Distributed Real-time Control with Delay Time Tag (DRC-DTT) and Adaptive Distributed Real-time Control (ADRC), have been presented in this thesis. For the DRC-RBT mechanism, packets are kept in buffers and sequentially forwarded according to their appropriate time which is defined by a buffer threshold value. For the DRC-DTT mechanism, each incoming packet that arrives too early, compared with the delay of the previous packet, will be delayed to maintain minimum packet jitter. The same technique of packet delay compensation is employed in the ADRC mechanism but, in addition to DRC-DTT, the added delay is calculated according to the moving average of the delays of the previous packets. Packets with excessive delay will be discarded.

Experimental results reveal that each of the proposed methods performs well in different situations. In a static network, DRC-RBT is a preferable method; however, this method can serve only the incoming traffic of constant bit rate nature. On the other hand, DRC-DTT gives a good performance under lossless situations. And finally, by appropriately adjusting the assigned weight value of average delay in ADRC, the obtained numerical tests suggest that ADRC can operate efficiently in both lossless and lossy network scenarios.

Department	Electrical Engineering	Student's signature 
Field of study	Electrical Engineering	Advisor's signature 
Academic year	2005		

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สามารถสำเร็จลุล่วงไปได้เนื่องด้วยความช่วยเหลืออย่างดียิ่งของผศ.ดร.เชาวนดิศ อัครกุล อาจารย์ที่ปรึกษา ที่คอยให้ทั้งแนวทาง คำแนะนำ คำปรึกษา และช่วยเหลือเป็นอย่างดีทุกครั้งเมื่อประสบปัญหา พร้อมทั้งยังให้ความสนับสนุนในด้านอุปกรณ์ทุกอย่างที่จำเป็นต้องใช้ในวิทยานิพนธ์นี้ รวมถึงผศ.ดร.ชัยเชษฐ์ สายวิจิตร และ อ.ภัทรชาติ โกมลภิติ ที่คอยให้คำแนะนำที่เป็นประโยชน์กับงานอย่างมาก ผู้วิจัยต้องขอกราบขอบพระคุณเป็นอย่างยิ่ง

นอกจากนั้น ยังมีเพื่อน ๆ ทุกคนที่คอยให้ความช่วยเหลือและเป็นกำลังใจให้กันเสมอมา ขอขอบคุณโอดิ์ ต้นนก อี๊ด ขวัญ เต้ แก๊ป ก้อง อาร์ต น้องส้ม ชาย เต๊ะ มิก โธ่ ตา ก้อย เบิ้ล มิ่ง พีคิต พีพีต ที่มาช่วยในการทดลองที่ต้องขั้บรด โดยเฉพาะโอดิ์ที่มีส่วนร่วมกับการทดลองภาคสนามของวิทยานิพนธ์นี้ ขอขอบคุณหน้าที่ให้ความรู้ และคำแนะนำที่มีประโยชน์ในด้านการบีบอัดวิดีโอ ขอขอบคุณตา ก้อย โธ่ ที่ร่วมทุกข์ร่วมสุขกันมาตลอดระยะเวลาหลายเดือน รวมถึงเพื่อน ๆ คนอื่นที่อาจจะไม่ได้เอ่ยชื่อมา ณ ที่นี้ ขอขอบคุณทุกคนมาก ๆ ที่ทำให้ช่วงเวลานี้เป็นเวลาที่

มีค่ายิ่ง
สุดท้ายนี้ต้องขอขอบคุณครอบครัวที่อบอุ่น ทั้งคุณพ่อ คุณแม่ และพี่ชายที่ให้ความรัก ความห่วงใย และคอยให้ความสนับสนุนทั้งกำลังใจและกำลังทรัพย์เป็นอย่างดีเสมอมา

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ง
บทคัดย่อภาษาอังกฤษ	จ
กิตติกรรมประกาศ	ฉ
สารบัญ	ช
สารบัญตาราง	ญ
สารบัญภาพ	ฎ
บทที่	
1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 แนวทางของวิทยานิพนธ์	3
1.3 วัตถุประสงค์ของงานวิทยานิพนธ์	4
1.4 ขอบเขตวิทยานิพนธ์	4
1.5 ขั้นตอนการดำเนินงาน	5
1.6 ประโยชน์ที่คาดว่าจะได้รับ	5
1.7 ประมวลวิทยานิพนธ์	6
2 หลักการและทฤษฎี	7
2.1 ลักษณะเฉพาะของเครือข่ายแอดฮอค	7
2.2 มาตรฐาน IEEE 802.11	8
2.2.1 ชั้นกายภาพตามมาตรฐาน IEEE 802.11b	9
2.2.2 การควบคุมการเข้าถึงตัวกลาง	9
2.3 การคำนวณจิตเตอร์	10
2.4 การส่งข้อมูลสื่อผสมแบบต่อเนื่อง	11
2.4.1 การส่งตามคำขอ	12
2.4.2 การส่งสัญญาณสด	12
2.5 มาตรฐานในการบีบอัดข้อมูล	13
2.5.1 มาตรฐานของ ITU-T	14
2.5.2 มาตรฐานของ ISO/IEC	14
2.6 แบบจำลองการเคลื่อนที่	16

บทที่	หน้า
2.6.1	16
2.6.2	17
2.6.3	17
2.7	18
2.7.1	19
2.7.2	20
2.7.3	21
2.7.3.1	23
2.7.3.2	23
2.7.4	24
3	26
3.1	26
3.1.1	26
3.1.2	26
3.1.2.1	28
3.1.2.2	29
3.1.2.3	34
3.2	35
3.2.1	35
3.2.2	37
3.2.3	38
4	41
4.1	42
4.2	43
4.3	44
4.4	50
4.5	54

บทที่	หน้า
4.5.1 ผลกระทบจากการเปลี่ยนเส้นทางการส่งข้อมูล	54
4.5.2 ผลกระทบจากสภาพแวดล้อมที่เปลี่ยนแปลง	57
4.5.2.1 สภาพแวดล้อมสำหรับการทดลองที่โนดไม่เคลื่อนที่	57
4.5.2.2 สภาพแวดล้อมสำหรับการทดลองที่โนดมีการเคลื่อนที่	58
4.5.2.3 ผลการทดลองและการวิเคราะห์	63
5 บทสรุปและข้อเสนอแนะ	65
5.1 บทสรุป	65
5.2 ข้อเสนอแนะ	66
รายการอ้างอิง	68
ภาคผนวก	71
บทความทางวิชาการจากวิทยานิพนธ์	72
ประวัติผู้เขียนวิทยานิพนธ์	77



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ตารางที่		หน้า
ตารางที่ 2.1	คุณลักษณะของชั้นกายภาพตามมาตรฐาน IEEE 802.11b	9
ตารางที่ 2.2	พารามิเตอร์ต่าง ๆ ตามมาตรฐาน IEEE 802.11	11
ตารางที่ 5.1	สรุปเปรียบเทียบบิตเตอร์และค่าประวิงเวลาของกลไกควบคุมทั้ง 3 วิธีในแต่ละสถานการณ์ . .	66



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

รูปที่	หน้า	
รูปที่ 1.1	จิตเตอร์ที่เกิดขึ้นในกรณีทำการควบคุมที่โหนดปลายทางเทียบกับที่โหนดระหว่างทาง	4
รูปที่ 2.1	ลักษณะของเครือข่ายแอดฮอก	7
รูปที่ 2.2	ระดับชั้นโพรโทคอลของ IEEE 802.11	8
รูปที่ 2.3	ขั้นตอนการเข้าถึงช่องสัญญาณตามมาตรฐาน 802.11	10
รูปที่ 2.4	แผนภาพแสดงการหาค่าจิตเตอร์ของกลุ่มข้อมูลที่ i และ j	11
รูปที่ 2.5	โครงสร้างของการส่งแบบต่อเนื่องตามคำขอ	12
รูปที่ 2.6	โครงสร้างของการส่งสัญญาณสดแบบต่อเนื่อง	13
รูปที่ 2.7	วิวัฒนาการของมาตรฐานการบีบอัดวิดีโอ	13
รูปที่ 2.8	ตัวอย่างเพิ่มข้อมูลวิดีโอต้นฉบับที่ใช้ในการทดลอง	15
รูปที่ 2.9	อัตราบิตที่ใช้ส่งของกราฟฟิกแบบอัตราบิตแปรผันได้	15
รูปที่ 2.10	รูปแบบการเคลื่อนที่ของโหนดตามแบบจำลองการเคลื่อนที่การเดินทางแบบสุ่ม	16
รูปที่ 2.11	รูปแบบการเคลื่อนที่ของโหนดตามแบบจำลองการเคลื่อนที่โดยสุ่มปลายทาง	17
รูปที่ 2.12	รูปแบบการเคลื่อนที่ของแบบจำลองการเคลื่อนที่ตามกันของรถ	18
รูปที่ 2.13	ระบบทดสอบของ D. A. Maltz และ J. Broch	19
รูปที่ 2.14	ระบบทดสอบ EWANT	20
รูปที่ 2.15	ระยะการส่งและรูปแบบการเคลื่อนที่ของโหนดในระบบทดสอบ EWANT	21
รูปที่ 2.16	โครงสร้างการเชื่อมต่อของ MobiEmu	22
รูปที่ 2.17	โครงสร้างของระบบ MobiEmu	22
รูปที่ 3.1	โครงสร้างของการส่งข้อมูลที่ใช้ในระบบทดสอบ	27
รูปที่ 3.2	รูปแบบการทดลองของระบบทดสอบแบบไร้สาย	27
รูปที่ 3.3	โครงสร้างของระบบทดสอบแบบมีสาย	28
รูปที่ 3.4	ลักษณะค่าประวิงเวลาของการส่งกลุ่มข้อมูลผ่านเครือข่าย	29
รูปที่ 3.5	สแตจที่เกิดเวรย์เลียนแบบใช้ในการจำลองการส่งกลุ่มข้อมูล	30
รูปที่ 3.6	รูปแบบการทดลองวัดค่าเวลาประวิงการประมวลผล การแพร่กระจาย และการส่ง	30
รูปที่ 3.7	ภาพขณะทดลองวัดค่าเวลาประวิงการประมวลผล การแพร่กระจาย และการส่ง	31
รูปที่ 3.8	ค่าประวิงเวลาเฉลี่ยของการส่งกลุ่มข้อมูลในแต่ละขนาดระหว่างคู่โหนดที่ระยะห่างต่าง ๆ	32
รูปที่ 3.9	ความเร็วเฉลี่ยของการส่งกลุ่มข้อมูลในแต่ละขนาดระหว่างคู่โหนดที่ระยะห่างต่าง ๆ	32
รูปที่ 3.10	ความเร็วสูงสุดของการส่งกลุ่มข้อมูลในแต่ละขนาดระหว่างคู่โหนดที่ระยะห่างต่าง ๆ	33

รูปที่	หน้า
รูปที่ 3.11 อัตราการสูญเสียของกลุ่มข้อมูลในแต่ละขนาดระหว่างคูโนดที่ระยะห่างต่าง ๆ	33
รูปที่ 3.12 อันตรภาคชั้นของค่าประวิงเวลาในช่วง 0-70 เมตรเทียบกับการแจจแจงลือกปรกติ	34
รูปที่ 3.13 อันตรภาคชั้นของค่าประวิงเวลาในช่วง 70-100 เมตรเทียบกับการแจจแจงลือกปรกติ	35
รูปที่ 3.14 แผนภาพเวลาของการส่งที่ไม่มีการควบคุมใด ๆ	36
รูปที่ 3.15 แผนภาพเวลาของการส่งที่ควบคุมโดย DRC-RBT	37
รูปที่ 3.16 กระบวนการตัดสินใจของ DRC-DTT	38
รูปที่ 3.17 แผนภาพเวลาของการส่งที่ควบคุมโดย DRC-DTT	39
รูปที่ 3.18 กระบวนการตัดสินใจของวิธี ADRC	40
รูปที่ 4.1 จิตเตอร์เฉลี่ยของทราฟฟิคที่มีอัตราบิตคงที่ในวิธี ADRC เมื่อปรับเปลี่ยนค่าถ่วงน้ำหนัก	42
รูปที่ 4.2 ค่าประวิงเวลาของทราฟฟิคที่มีอัตราบิตคงที่ในวิธี ADRC เมื่อปรับเปลี่ยนค่าถ่วงน้ำหนัก	43
รูปที่ 4.3 จิตเตอร์เฉลี่ยของทราฟฟิคที่มีอัตราบิตแปรผันได้ในวิธี ADRC เมื่อปรับเปลี่ยนค่าถ่วงน้ำหนัก	43
รูปที่ 4.4 ค่าประวิงเวลาเฉลี่ยของทราฟฟิคที่มีอัตราบิตแปรผันได้ในวิธี ADRC เมื่อปรับเปลี่ยนค่าถ่วงน้ำหนัก	44
รูปที่ 4.5 จิตเตอร์เฉลี่ยเมื่อปรับเปลี่ยนค่าขีดเริ่มเปลี่ยน	45
รูปที่ 4.6 ค่าประวิงเวลาเฉลี่ยเมื่อปรับเปลี่ยนค่าขีดเริ่มเปลี่ยน	45
รูปที่ 4.7 จิตเตอร์เฉลี่ยของทราฟฟิคที่มีอัตราบิตคงที่เมื่อปรับเปลี่ยนจำนวนช่วงเชื่อมต่อ	46
รูปที่ 4.8 ค่าประวิงเวลาเฉลี่ยของทราฟฟิคที่มีอัตราบิตคงที่เมื่อปรับเปลี่ยนจำนวนช่วงเชื่อมต่อ	46
รูปที่ 4.9 จิตเตอร์เฉลี่ยของทราฟฟิคที่มีอัตราบิตแปรผันได้เมื่อปรับเปลี่ยนจำนวนช่วงเชื่อมต่อ	47
รูปที่ 4.10 ค่าประวิงเวลาเฉลี่ยของทราฟฟิคที่มีอัตราบิตแปรผันได้เมื่อปรับเปลี่ยนจำนวนช่วงเชื่อมต่อ	47
รูปที่ 4.11 จิตเตอร์เฉลี่ยของทราฟฟิคที่มีอัตราบิตคงที่เพื่อเปรียบเทียบการควบคุมแบบเวลาจริงในแต่ละวิธี	48
รูปที่ 4.12 ค่าประวิงเวลาเฉลี่ยของทราฟฟิคที่มีอัตราบิตคงที่เพื่อเปรียบเทียบการควบคุมแบบเวลาจริงในแต่ละวิธี	49
รูปที่ 4.13 จิตเตอร์เฉลี่ยของทราฟฟิคที่มีอัตราบิตแปรผันได้เพื่อเปรียบเทียบการควบคุมแบบเวลาจริงในแต่ละวิธี	49
รูปที่ 4.14 ค่าประวิงเวลาเฉลี่ยของทราฟฟิคที่มีอัตราบิตแปรผันได้เพื่อเปรียบเทียบการควบคุมแบบเวลาจริงในแต่ละวิธี	50
รูปที่ 4.15 จิตเตอร์เฉลี่ยของทราฟฟิคที่มีอัตราบิตคงที่เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูล	51
รูปที่ 4.16 ค่าประวิงเวลาเฉลี่ยของทราฟฟิคที่มีอัตราบิตคงที่เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูล	51
รูปที่ 4.17 จิตเตอร์เฉลี่ยของทราฟฟิคที่มีอัตราบิตแปรผันได้เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูล	52
รูปที่ 4.18 ค่าประวิงเวลาเฉลี่ยของทราฟฟิคที่มีอัตราบิตแปรผันได้เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูล	52

รูปที่	หน้า
รูปที่ 4.19 อัตราการสูญเสียกลุ่มข้อมูลตลอดเส้นทางของกราฟฟิกที่มีอัตราบิดคองที่เมื่อปรับเปลี่ยนอัตรา การสูญเสียกลุ่มข้อมูลต่อหนึ่งช่วงเชื่อมต่อ	53
รูปที่ 4.20 อัตราการสูญเสียกลุ่มข้อมูลตลอดเส้นทางของกราฟฟิกที่มีอัตราบิดแปรผันได้เมื่อปรับเปลี่ยน อัตราการสูญเสียกลุ่มข้อมูลต่อหนึ่งช่วงเชื่อมต่อ	53
รูปที่ 4.21 จำนวนช่วงเชื่อมต่อที่เวลาต่าง ๆ ของรูปแบบการเคลื่อนที่ที่พิจารณา	55
รูปที่ 4.22 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดคองที่ในระบบทดสอบแบบมีสายในกรณีที่โนดเคลื่อนที่	55
รูปที่ 4.23 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิดคองที่ในระบบทดสอบแบบมีสายในกรณีที่โนดเคลื่อนที่	56
รูปที่ 4.24 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้ในระบบทดสอบแบบมีสายในกรณีที่โนด เคลื่อนที่	56
รูปที่ 4.25 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิดแปรผันได้ในระบบทดสอบแบบมีสายในกรณีที่โนด เคลื่อนที่	57
รูปที่ 4.26 ทอพอโลยีที่พิจารณาในระบบทดสอบแบบไร้สาย	58
รูปที่ 4.27 ภาพขณะทดลองในระบบทดสอบแบบไร้สายกรณีที่โนดไม่เคลื่อนที่	58
รูปที่ 4.28 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดคองที่ในระบบทดสอบแบบไร้สายที่ไม่มีการเคลื่อนที่	59
รูปที่ 4.29 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิดคองที่ในระบบทดสอบแบบไร้สายที่ไม่มีการเคลื่อนที่	59
รูปที่ 4.30 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้ในระบบทดสอบแบบไร้สายที่ไม่มีการเคลื่อนที่	60
รูปที่ 4.31 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิดแปรผันได้ในระบบทดสอบแบบไร้สายที่ไม่มีการเคลื่อนที่	60
รูปที่ 4.32 ภาพขณะทดลองในระบบทดสอบแบบไร้สายกรณีที่โนดเคลื่อนที่โดยใช้รถ 4 คันขับตามกัน	61
รูปที่ 4.33 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดคองที่ในระบบทดสอบแบบไร้สายที่โนดเคลื่อนที่	61
รูปที่ 4.34 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิดคองที่ในระบบทดสอบแบบไร้สายที่โนดเคลื่อนที่	62
รูปที่ 4.35 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้ในระบบทดสอบแบบไร้สายที่โนดเคลื่อนที่	62
รูปที่ 4.36 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิดแปรผันได้ในระบบทดสอบแบบไร้สายที่โนดเคลื่อนที่	63

บทที่ 1

บทนำ

เทคโนโลยีการติดต่อสื่อสารในปัจจุบันได้พัฒนาก้าวหน้าขึ้นอย่างรวดเร็วและต่อเนื่อง ดังนั้นจึงได้มีงานวิจัยมากมายมุ่งเน้นถึงการพัฒนาให้การติดต่อสื่อสารมีประสิทธิภาพดียิ่งขึ้น ซึ่งวิทยานิพนธ์ฉบับนี้ก็เป็นส่วนหนึ่งที่ได้นำเสนอวิธีการที่มีส่วนช่วยพัฒนาเทคโนโลยีการสื่อสารให้มีคุณภาพดีขึ้น โดยเนื้อหาในบทนี้ได้กล่าวถึงความเป็นมาและความสำคัญของปัญหาที่นำมาศึกษา จากนั้นได้เสนอแนวทางของวิทยานิพนธ์ วัตถุประสงค์ของวิทยานิพนธ์ ขอบเขตของวิทยานิพนธ์ รวมถึงขั้นตอนดำเนินงาน และประโยชน์ที่คาดว่าจะได้รับจากวิทยานิพนธ์ฉบับนี้

1.1 ความเป็นมาและความสำคัญของปัญหา

การติดต่อสื่อสารถือเป็นปัจจัยสำคัญในการดำรงชีวิตของมนุษย์ โดยที่ผ่านมามีการพัฒนาเทคโนโลยีในด้านการสื่อสารต่าง ๆ มากมายเพื่อตอบสนองความต้องการของมนุษย์ ซึ่งในปัจจุบันเทคโนโลยีหนึ่งทีถือว่ามีพัฒนาไปอย่างรวดเร็วและเข้ามามีบทบาทกับชีวิตมนุษย์มากก็คือ การสื่อสารไร้สาย ไม่ว่าจะเป็นโทรศัพท์เคลื่อนที่ การสื่อสารผ่านดาวเทียม เครือข่ายแลนไร้สาย รวมถึงเครือข่ายแอดฮอค (ad hoc network) ที่นักวิจัยจำนวนมากกำลังให้ความสนใจ [1] และคาดว่าในอนาคตจะมีการนำไปประยุกต์ใช้งานกันอย่างแพร่หลายทั้งในด้านการทหาร และในเชิงธุรกิจ ทั้งนี้รวมถึงการนำไปใช้งานร่วมกับระบบขนส่งอัจฉริยะ (Intelligent Transportation System: ITS) เพื่อเพิ่มความปลอดภัยให้กับผู้ขับขี่ [2], [3] ซึ่งการใช้เครือข่ายแอดฮอคนั้นมีข้อดีคือไม่ต้องอาศัยโครงสร้างพื้นฐานของโครงข่ายสื่อสารในพื้นที่ และไม่จำเป็นต้องมีศูนย์กลางการควบคุมการแลกเปลี่ยนข้อมูล เนื่องจากแต่ละคันภายในระยะการส่งสัญญาณสามารถรับส่งข้อมูลระหว่างกันเองได้ นอกจากนี้รถแต่ละคันยังสามารถส่งต่อข้อมูลไปให้รถคันอื่น ๆ ในระยะการส่งถัดไปได้อีกด้วย ทำให้การลงทุนในการติดตั้งระบบไม่สูงมากนัก และสามารถนำไปประยุกต์ใช้กับตัวรับรู้ (sensor) ได้ เช่น หากตัวรับรู้ในรถคันหนึ่งตรวจพบบริเวณที่มีแอ่งน้ำขังอยู่ตรงทางโค้ง ซึ่งอาจก่อให้เกิดอุบัติเหตุได้ ก็ทำการส่งสัญญาณไปเตือนรถคันอื่น ๆ ที่กำลังจะผ่านบริเวณนี้ให้ระมัดระวังเป็นพิเศษ หรือหากทำการลงทุนเพิ่มโดยการติดตั้งเกตเวย์ไว้ตามจุดสำคัญ ๆ ริมถนนเพื่อใช้งานร่วมกับเครือข่ายแอดฮอคของรถ ก็ยังช่วยให้ผู้ที่นั่งอยู่ในรถสามารถเข้าถึงอินเทอร์เน็ตเพื่อใช้งานในรูปแบบต่าง ๆ ได้ ทั้งนี้ก็รวมถึงการใช้บริการสื่อผสมด้วย โดยขณะนี้ก็ได้มีงานวิจัยเกี่ยวกับเครือข่ายแอดฮอคที่อยู่ระหว่างการกำหนดมาตรฐานและการวิจัยที่สำคัญมากมาย เช่น โพรโทคอลจัดเส้นทาง [4], [32] การเข้าถึงช่องสัญญาณ [6] การควบคุมคุณภาพการให้บริการ [7] เพื่อให้สามารถนำไปใช้กับเครือข่ายจริงได้อย่างมีประสิทธิภาพ แต่ปัญหาที่สำคัญคือ จะทำการทดสอบได้อย่างไรว่าวิธีการนั้น ๆ มีประสิทธิภาพและเหมาะสมกับการใช้งานในระบบเครือข่ายจริง

การทดสอบประสิทธิภาพของวิธีการหรือโพรโทคอลต่าง ๆ ที่นักวิจัยนำเสนอ นั้นสามารถแบ่งออกได้เป็น 3 วิธีหลัก ๆ คือ การจำลองโดยใช้โปรแกรมคอมพิวเตอร์ (ที่นิยมใช้กันมากเช่น ns-2 [8] และ glomosim [9]), การวิเคราะห์ทางคณิตศาสตร์ [10] และการสร้างระบบทดสอบ (testbed) [11]-[16] ซึ่งสองวิธีแรกเป็นวิธีที่งาน

วิจัยส่วนใหญ่นิยมใช้กัน เนื่องจากวิธีแรกเป็นวิธีที่ทำได้ง่าย สะดวก และสามารถทำการจัดการระบบให้เป็นอย่างที่ต้องการได้ ไม่ว่าจะเป็นการย่อหรือขยายขนาดของระบบที่ทำการทดสอบ สร้างสถานการณ์ที่ต้องการให้เกิดขึ้นในขณะใดก็ได้ ซึ่งความสามารถนี้ได้รวมถึงการกำหนดหรือตัดรายละเอียดบางอย่างออกเพื่อทำให้ระบบจัดการได้ง่ายขึ้น อย่างไรก็ตามการจำลองโดยใช้โปรแกรมคอมพิวเตอร์ยังไม่สามารถทำการจำลองสภาพแวดล้อมทุกอย่างได้เหมือนในสถานการณ์จริง เช่น ปรากฏการณ์ต่าง ๆ ของคลื่นแม่เหล็กไฟฟ้า การส่งข้อมูลผ่านชั้นการสื่อสารข้อมูล (data communication layer) ในระบบปฏิบัติการจริง ๆ และระบบการทำงานของโปรแกรมคอมพิวเตอร์ยังเป็นการประมวลผลข้อมูลแบบรวมศูนย์ (centralized data processing) ซึ่งไม่เหมือนกับในระบบจริงที่เหตุการณ์ต่าง ๆ สามารถเกิดขึ้นได้ตลอดเวลาและไม่เป็นลำดับ ส่วนในวิธีที่สองนั้นก็ต้องใช้การตั้งสมมติฐานในการแก้ปัญหาเหมือนกัน ด้วยเหตุนี้จึงทำให้สองวิธีนี้อาจจะให้ผลที่คลาดเคลื่อนจากระบบจริงได้ ดังนั้นในงานวิจัยนี้จึงได้เลือกใช้วิธีการสร้างระบบทดสอบ เพราะถึงแม้ว่าจะมีความซับซ้อนมากกว่าและต้องใช้ต้นทุนสูงกว่า แต่ก็สามารถจำลองสภาพแวดล้อมต่าง ๆ ได้ใกล้เคียงกับระบบจริงมากกว่าเช่นกัน

ระบบทดสอบเครือข่ายแอดฮอกในงานวิจัยที่ผ่านมาสามารถแบ่งได้เป็น 2 รูปแบบ คือแบบไร้สาย และแบบมีสาย โดยที่แบบไร้สายจะเป็นระบบทดสอบที่สร้างให้เหมือนกับการใช้งานในระบบจริงทุกอย่าง มีข้อดีคือเป็นการรับส่งด้วยคลื่นแม่เหล็กไฟฟ้า ในสภาพแวดล้อมที่มีการรบกวนของสัญญาณเหมือนกับในระบบจริง ซึ่งจะทำให้ได้ผลการทดสอบที่เชื่อถือได้ เหมาะสำหรับการใช้ทดสอบประสิทธิภาพของโพรโทคอลในรูปแบบการทำงานจริง ๆ หลังจากผ่านการทดสอบในขั้นต้นด้วยวิธีการอื่น ๆ ที่ง่ายกว่ามาแล้ว แต่ข้อเสียคือ ในสภาพแวดล้อมเช่นนี้ทำให้ยากต่อการตรวจหาข้อผิดพลาดของโพรโทคอลที่ทดสอบ เนื่องจากพารามิเตอร์ต่าง ๆ ที่มีผลกระทบกับการทดลองมีค่าเปลี่ยนแปลงไปเรื่อย ๆ ขึ้นอยู่กับเวลาที่ทำการทดลอง รวมถึงการเคลื่อนที่ของแต่ละโหนดที่ต้องใช้การเคลื่อนที่จริง ดังนั้นจึงไม่สามารถทำการทดลองในสถานการณ์เดิมซ้ำใหม่ได้ และยังไม่สามารถสร้างบางสถานการณ์ที่ต้องการให้เกิดขึ้นได้ เช่น ปัญหาสถานีซ่อนเร้น (hidden station problem) หรือปัญหาสถานีรับสัญญาณ (exposed station problem) อีกทั้งการสร้างระบบทดสอบยังต้องใช้งบประมาณสูงมากในการทดลอง เนื่องจากต้องใช้เครื่องคอมพิวเตอร์หนึ่งเครื่องแทนหนึ่งโหนด ทำให้ยากต่อการขยายขนาดของระบบที่ทดลอง งานวิจัยที่ใช้ระบบทดสอบประเภทนี้ ได้แก่ [11], [12] ส่วนงานใน [13] ก็เป็นแบบไร้สายเช่นกัน แต่อาศัยการจำลองการเคลื่อนที่แทนการเคลื่อนที่จริง

จากปัญหาและข้อจำกัดที่เกิดขึ้นในระบบทดสอบแบบไร้สายดังที่กล่าวมานั้น จึงมีนักวิจัยส่วนหนึ่งเลือกใช้ระบบทดสอบแบบมีสายแทน [14], [16] เพราะถึงแม้ว่าสภาพแวดล้อมในการทดสอบจะไม่เหมือนกับระบบจริงเท่ากับในระบบทดสอบแบบไร้สาย แต่ก็ได้มีการเลียนแบบให้ใกล้เคียงที่สุด โดยอาศัยการหน่วงเวลาเพิ่มเพื่อเลียนแบบความเร็วในการรับส่งข้อมูล ส่วนการเคลื่อนที่ของแต่ละโหนดจะอาศัยการกำหนดโหนดข้างเคียงให้กับโหนดนั้น ๆ ที่เวลาต่าง ๆ และเนื่องจากในระบบทดสอบประเภทนี้ผู้ทดสอบสามารถจำลองสถานการณ์ใด ๆ ได้ตามต้องการ และพารามิเตอร์ต่าง ๆ ก็ทำการควบคุมได้ ทำให้มีความเหมาะสมมากกว่าสำหรับนำไปใช้ทดสอบการทำงานของโพรโทคอลในขั้นต้นที่ยังต้องมีการตรวจแก้จุดบกพร่อง รวมทั้งยังสามารถขยายขนาดของเครือข่ายที่ทดสอบได้ตามต้องการ ในขณะที่งบประมาณก็ไม่สูงมากนัก ซึ่งจากที่กล่าวมาจะเห็นว่าระบบทดสอบที่ควรเลือกใช้ขึ้นอยู่กับสถานการณ์ที่ต้องการทดสอบ ดังนั้นวิทยานิพนธ์นี้จะเลือกทำทั้ง 2 แบบ โดยระบบทดสอบแบบไร้สายจะใช้สำหรับทดลองในเครือข่ายที่มี

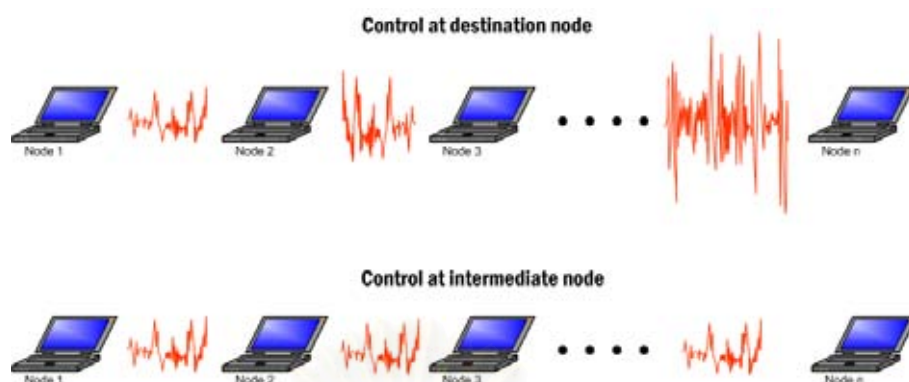
ขนาดเล็ก ส่วนแบบมีสายจะใช้ในสถานการณ์ที่เครือข่ายมีขนาดใหญ่ขึ้น และมีการเปลี่ยนแปลงทอพอโลยี รวมถึงพิจารณาการเคลื่อนที่ของแต่ละโนดด้วย

จากการที่เครือข่ายแอตฮอกมีการพัฒนาเติบโตขึ้นอย่างรวดเร็วนี้ การให้บริการสื่อผสม (multimedia service) คงจะเข้ามามีบทบาทในไม่ช้า ซึ่งในปัจจุบันการให้บริการสื่อผสมนี้ถือว่าเป็นที่นิยมอย่างมากในเครือข่ายอินเทอร์เน็ต ไม่ว่าจะเป็นการเรียกดูข้อมูลภาพเคลื่อนไหว หรือการประชุมผ่านทางเครือข่าย โดยคุณภาพของการให้บริการ (Quality of Service: QoS) ก็ขึ้นอยู่กับหลายปัจจัย ทั้งการสูญหายของกลุ่มข้อมูล (packet loss) ค่าประวิงเวลา (delay) และค่าแปรผันของการประวิงเวลาหรือจิตเตอร์ (jitter) ซึ่งโดยส่วนของจิตเตอร์นั้นถือว่ามีความสำคัญมากสำหรับการส่งข้อมูลสื่อผสมแบบต่อเนื่อง (multimedia streaming) เนื่องจากหากจิตเตอร์มีค่ามาก จะทำให้การแสดงผลข้อมูลภาพหรือเสียงนั้น ๆ ไม่ต่อเนื่อง ส่งผลให้ผู้ใช้บริการเกิดความไม่พอใจได้ นอกจากนี้จิตเตอร์ที่มีค่ามากยังส่งผลต่อขนาดของบัฟเฟอร์ที่โนดปลายทางด้วย [17] ดังนั้นในงานวิจัยนี้จึงพิจารณาการลดค่าจิตเตอร์เป็นหลัก ในขณะที่ยังพยายามควบคุมให้ค่าประวิงเวลาและการสูญหายของกลุ่มข้อมูลอยู่ในช่วงที่สามารถยอมรับได้

บริการสื่อผสมที่ใช้กันอยู่ทุกวันนี้ใช้วิธีการควบคุมจิตเตอร์ที่โนดปลายทาง โดยการเก็บกลุ่มข้อมูลจำนวนหนึ่งไว้ในบัฟเฟอร์ก่อนแล้วจึงค่อยนำออกแสดงผล [18] ซึ่งวิธีการนี้ทำได้ง่าย เนื่องจากต้องมีการจัดการที่โนดปลายทางเท่านั้น แต่การจัดการบัฟเฟอร์ในลักษณะนี้อาจไม่เหมาะกับเครือข่ายแอตฮอกเนื่องจากเป็นเครือข่ายไร้สาย ค่าประวิงเวลามีความไม่แน่นอนสูง ทำให้ต้องใช้บัฟเฟอร์ที่มีขนาดใหญ่ ซึ่งส่วนนี้อาจจะเป็นปัญหาได้ เนื่องจากในเครือข่ายแอตฮอกอุปกรณ์ที่ใช้ไม่ได้มีเพียงเครื่องคอมพิวเตอร์เท่านั้น อาจจะมีอุปกรณ์ขนาดเล็กอื่น ๆ ที่มีขนาดบัฟเฟอร์ที่จำกัดอยู่ด้วย ดังนั้นวิธีการควบคุมจิตเตอร์ที่มีประสิทธิภาพจึงมีความสำคัญและจำเป็นต่อเครือข่ายแอตฮอกมาก

1.2 แนวทางของวิทยานิพนธ์

จากปัญหาที่กล่าวมาข้างต้น ในงานวิจัยนี้จึงได้นำเสนอกลไกการควบคุมแบบเวลาจริงแบบกระจาย (distributed real time control) โดยให้โนดระหว่างทาง (intermediate node) สามารถช่วยควบคุมจังหวะการรับส่งข้อมูลได้ ซึ่งทำให้โนดปลายทางไม่จำเป็นต้องรับภาระหนักจนเกินไป และยังช่วยลดขนาดของบัฟเฟอร์ที่โนดปลายทางได้อีกด้วย ดังแสดงในรูปที่ 1.1 จากรูปจะเห็นว่าในวิธีที่มีการควบคุมจิตเตอร์ที่โนดปลายทางเพียงโนดเดียว ที่โนดปลายทางนี้จะต้องรองรับจิตเตอร์ที่มีค่ามากเนื่องมาจากการแปรผันของค่าประวิงเวลาที่เกิดขึ้นหลายช่วงเชื่อมต่อ (hop) แต่ในวิธีควบคุมแบบกระจายที่ให้โนดระหว่างทางมีส่วนช่วยจัดการด้วยนี้ จะทำให้การแปรผันของค่าประวิงเวลาลดลงเหลือเพียงในระยะหนึ่งช่วงเชื่อมต่อเท่านั้น ซึ่งมีผลทำให้จิตเตอร์ที่โนดปลายทางมีค่าน้อยลงด้วย และในวิทยานิพนธ์นี้ยังได้สร้างระบบทดสอบสำหรับประเมินผลวิธีการควบคุมที่นำเสนอ เพื่อให้การทดสอบมีสภาพแวดล้อมใกล้เคียงกับระบบจริงมากที่สุด ระบบทดสอบที่สร้างนี้จะเป็นทั้งแบบไร้สาย และมีสายแล้วแต่ความเหมาะสมของสถานการณ์ที่ทดสอบ ทั้งนี้ขึ้นอยู่กับขนาดของเครือข่ายและรูปแบบการเคลื่อนที่ (mobility model) ที่พิจารณา



รูปที่ 1.1 จิตเตอร์ที่เกิดขึ้นในกรณีทำการควบคุมที่โหนดปลายทางเทียบกับที่โหนดระหว่างทาง

1.3 วัตถุประสงค์ของงานวิทยานิพนธ์

ออกแบบและพัฒนากระบวนการรับส่งข้อมูลแบบสื่อผสมในเครือข่ายแอตสอกเพื่อให้คุณภาพการให้บริการดีขึ้น โดยพิจารณาจากการลดค่าจิตเตอร์ที่โหนดปลายทางเป็นหลัก ในขณะที่ค่าประวิงเวลาและการสูญหายของกลุ่มข้อมูลยังคงอยู่ในช่วงที่สามารถยอมรับได้ พร้อมทั้งสร้างระบบทดสอบสำหรับประเมินผลวิธีการควบคุมที่นำเสนอ เพื่อให้การทดสอบมีสภาพแวดล้อมใกล้เคียงกับระบบจริงมากที่สุด โดยระบบทดสอบที่สร้างนี้จะเป็นทั้งแบบไร้สาย และมีสายแล้วแต่ความเหมาะสมของสถานการณ์ที่ทดสอบ โดยขึ้นอยู่กับขนาดของเครือข่ายและรูปแบบการเคลื่อนที่ที่พิจารณา

1.4 ขอบเขตวิทยานิพนธ์

ในงานวิจัยนี้นำเสนอกลไกควบคุมแบบเวลาจริงสำหรับการรับส่งข้อมูลสื่อผสมเพื่อทำให้จิตเตอร์ที่โหนดปลายทางมีค่าน้อยลง ขณะที่ค่าประวิงเวลาและการสูญหายของกลุ่มข้อมูลอยู่ในช่วงที่สามารถยอมรับได้ และจะทำการทดสอบกลไกที่นำเสนอนี้ในระบบทดสอบ 2 รูปแบบ คือแบบไร้สายในกรณีที่เครือข่ายมีขนาดเล็ก ส่วนแบบมีสายจะใช้ในสถานการณ์ที่เครือข่ายมีขนาดใหญ่ขึ้น และมีการเปลี่ยนแปลงทอพอโลยี รวมถึงพิจารณาการเคลื่อนที่ของแต่ละโหนดด้วย โดยแบบจำลองการเคลื่อนที่ในวิทยานิพนธ์นี้จะใช้แบบจำลองการเคลื่อนที่ตามกันของรถ (Car-Following Models) [19] เพื่อให้สอดคล้องกับตัวอย่างการใช้งานเทคโนโลยีสื่อสารแบบแอตสอกในสถานการณ์ของระบบขนส่งแบบอัจฉริยะ ซึ่งในการทดลองจะส่งแฟ้มข้อมูลวิดีโอที่มีการบีบอัดตามมาตรฐาน MPEG-4 [25] เนื่องจากเป็นมาตรฐานที่ออกแบบมาเพื่อให้เหมาะสำหรับการส่งข้อมูลสื่อผสมแบบต่อเนื่องและยังสามารถใช้กับการส่งด้วยอัตราบิตต่ำได้ ทำให้เหมาะสำหรับเครือข่ายไร้สาย ส่วนรูปแบบของทราฟฟิกที่ทดลองจะใช้ 2 แบบ คือ แบบอัตราบิตคงที่ (constant bit rate: CBR) และแบบอัตราบิตแปรผันได้ (variable bit rate: VBR) เนื่องจากการส่งข้อมูลสื่อผสมแบบต่อเนื่องโดยปกติจะมีอยู่ 2 ลักษณะ คือ การส่งตามคำขอ (on-demand streaming) และ การส่งสัญญาณสด (live streaming) ซึ่ง

ในแบบส่งตามคำขอจะเป็นการส่งแฟ้มข้อมูลที่มีการบีบอัดไว้แล้วและเก็บอยู่ในอุปกรณ์หน่วยเก็บ (storage device) เมื่อมีการร้องขอจากผู้ให้บริการแล้วจึงค่อยนำข้อมูลในแฟ้มข้อมูลนั้นส่งออกไปด้วยอัตราบิตคงที่ ส่วนแบบส่งสัญญาณสดจะเป็นการจับภาพจากต้นฉบับ (source) ในเวลาจริง จากนั้นจึงทำการบีบอัดแล้วส่งออกไป ซึ่งข้อมูลที่ถูกส่งออกไปนี้จะมีอัตราไม่คงที่ เนื่องจากขนาดของข้อมูลในแต่ละเวลาขึ้นอยู่กับเนื้อหาของภาพในขณะนั้น ดังนั้นกราฟฟิกที่ใช้ในการส่งข้อมูลสื่อผสมแบบต่อเนื่องจึงมีทั้งแบบอัตราบิตคงที่ และอัตราบิตแปรผันได้ รวมทั้งในวิทยานิพนธ์นี้จะพิจารณาการควบคุมสายงาน (flow control) เป็นหลัก และจะไม่คำนึงถึงโพรโทคอลหาเส้นทาง (routing protocol) โดยมีสมมติฐานว่าทุกโหนดทราบเส้นทางไปยังโหนดอื่น ๆ ในเครือข่าย และไม่ต้องมีกระบวนการร้องขอเส้นทางเมื่อทอพอโลยีมีการเปลี่ยนแปลง แต่สามารถสลับไปใช้เส้นทางใหม่ได้ทันที

1.5 ขั้นตอนการดำเนินงาน

1. ศึกษาความรู้พื้นฐานเกี่ยวกับเครือข่ายแอดฮอก และระบบทดสอบในงานวิจัยที่ผ่านมา
2. ศึกษาการใช้งานและการเขียนโปรแกรมบนระบบปฏิบัติการลินุกซ์ (linux operating system)
3. ศึกษาการเขียนโปรแกรมติดต่อสื่อสารผ่านทางซ็อกเก็ต (socket programming)
4. ทำการทดลองเพื่อหารูปแบบของค่าประวิงเวลาและการสูญหายของกลุ่มข้อมูลที่ต้องใช้เลียนแบบในระบบทดสอบแบบมีสาย
5. ออกแบบวิธีควบคุมแบบเวลาจริงเพื่อช่วยลดจitterที่โหนดปลายทาง
6. สร้างระบบทดสอบแบบมีสายพร้อมทั้งเขียนโปรแกรมเพื่อทดสอบวิธีควบคุมที่นำเสนอ
7. สร้างระบบทดสอบไร้สายพร้อมทั้งปรับปรุงโปรแกรมบางส่วนให้เหมาะสมกับสถานการณ์จริง
8. ทดลองและวัดผลจากวิธีควบคุมที่นำเสนอในระบบทดสอบทั้ง 2 แบบ
9. วิเคราะห์และสรุปผลการทดลองที่ได้ พร้อมทั้งจัดทำรายงานฉบับสมบูรณ์

1.6 ประโยชน์ที่คาดว่าจะได้รับ

สามารถนำวิธีควบคุมที่ได้นำเสนอนี้ไปใช้ในการรับส่งข้อมูลสื่อผสมในเครือข่ายแอดฮอก เพื่อให้มีคุณภาพการให้บริการดีขึ้น รวมทั้งยังสามารถนำระบบทดสอบที่สร้างขึ้นนี้ไปใช้ในการทดลองประเมินผลวิธีการอื่น ๆ ได้อีกในอนาคต

1.7 ประมวลวิทยานิพนธ์

บทที่ 1 บทนำ: กล่าวถึงปัญหาและความสำคัญของการควบคุมจิตเตอร์สำหรับการส่งข้อมูลสื่อผสมแบบต่อเนื่อง เหตุผลในการเลือกใช้ระบบทดสอบเพื่อประเมินผลวิธีการที่นำเสนอ รวมถึงประเภทของระบบทดสอบในงานวิจัยที่ผ่านมา พร้อมทั้งวัตถุประสงค์และแนวทางของวิทยานิพนธ์นี้

บทที่ 2 หลักการและทฤษฎี: กล่าวถึงหลักการพื้นฐานที่จำเป็นประกอบไปด้วยลักษณะของเครือข่ายแอดฮอก มาตรฐาน IEEE 802.11 หลักการคำนวณจิตเตอร์ วิธีการส่งข้อมูลสื่อผสมแบบต่อเนื่อง ตลอดจนมาตรฐานที่ใช้ในการบีบอัดข้อมูล พร้อมทั้งกล่าวถึงแบบจำลองการเคลื่อนที่ รวมถึงระบบทดสอบในงานวิจัยที่ผ่านมา

บทที่ 3 โครงสร้างของระบบทดสอบและกลไกการควบคุมแบบเวลาจริง: กล่าวถึงโครงสร้างของระบบทดสอบทั้งแบบมีสายและไร้สายที่ใช้ในวิทยานิพนธ์นี้ รวมถึงการหารูปแบบของค่าประวิงเวลาที่นำมาใช้ในระบบทดสอบแบบมีสาย และยังได้กล่าวถึงหลักการทำงานของกลไกการควบคุมแบบเวลาจริงที่นำเสนอทั้ง 3 วิธี

บทที่ 4 ผลการทดสอบ: กล่าวถึงผลจากการประเมินกลไกแบบเวลาจริงที่นำเสนอทั้ง 3 วิธีในระบบทดสอบทั้งสองแบบ แบ่งออกเป็น 5 การทดลองหลัก โดย 2 การทดลองแรกจะเป็นการหาค่าด่างน้ำหนักของวิธี ADRC และค่าขีดเริ่มเปลี่ยนของวิธี DRC-RBT ที่เหมาะสม ส่วนอีก 3 การทดลองจะเป็นการศึกษาผลกระทบจากขนาดของเครือข่าย อัตราการสูญเสียกลุ่มข้อมูล และการเคลื่อนที่ของโหนด

บทที่ 5 บทสรุปและข้อเสนอแนะ: สรุปงานวิจัยทั้งหมดในวิทยานิพนธ์ฉบับนี้และเสนอแนวทางในการพัฒนา งานวิจัยต่อไป

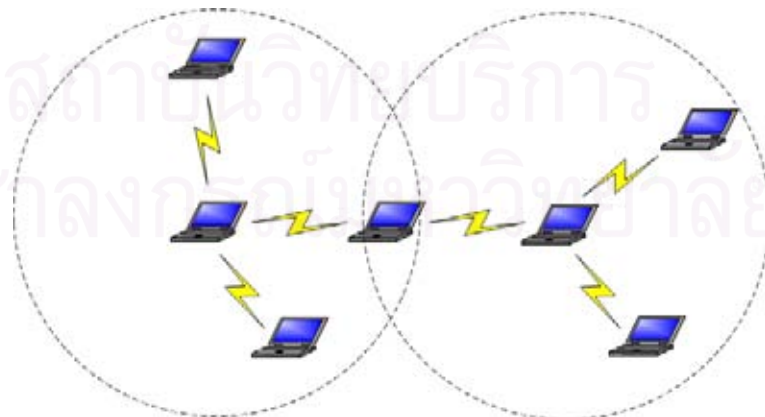
บทที่ 2

หลักการและทฤษฎี

เนื้อหาในบทนี้จะกล่าวถึงหลักการและทฤษฎีที่เป็นพื้นฐานในการทำวิทยานิพนธ์นี้ ในส่วนแรกจะกล่าวถึงลักษณะของเครือข่ายแอดฮอก มาตรฐาน IEEE 802.11 ซึ่งเป็นตัวกำหนดคุณลักษณะของการส่งทางกายภาพ และการเข้าถึงตัวกลางของเครือข่ายไร้สาย ส่วนถัดมาจะกล่าวถึงหลักการคำนวณจิตเตอร์ วิธีการส่งข้อมูลสี่ผสมแบบต่อเนื่อง ตลอดจนมาตรฐานที่ใช้ในการบีบอัดข้อมูล (compression) และส่วนสุดท้ายจะกล่าวถึงแบบจำลองการเคลื่อนที่ รวมถึงระบบทดสอบในงานวิจัยที่ผ่านมา

2.1 ลักษณะเฉพาะของเครือข่ายแอดฮอก

เครือข่ายแอดฮอกเป็นเครือข่ายการสื่อสารแบบไร้สายที่ไม่มีรูปแบบการเชื่อมต่อที่แน่นอน ทอพอโลยี (topology) มีการเปลี่ยนแปลงตลอดเวลา เนื่องจากไม่มีการเชื่อมต่อเข้ากับสถานีฐาน (base station) เหมือนในกรณีของระบบเซลลูลาร์ หรือแอคเซสพอยต์ (access point) ในเครือข่ายเฉพาะที่ไร้สาย (wireless local area network) แบบมีโครงสร้างพื้นฐาน (infrastructure mode) ทุกโหนดในเครือข่ายแอดฮอกนี้สามารถเคลื่อนที่ได้อย่างอิสระ และจะสามารถติดต่อได้โดยตรงเฉพาะกับโหนดข้างเคียง (neighbour node) ที่อยู่ในระยะการส่ง (transmission range) ของโหนดนั้น ๆ เท่านั้น หากโหนดใดต้องการจะติดต่อกับโหนดอื่นที่อยู่นอกเหนือระยะการส่ง จะต้องส่งผ่านโหนดระหว่างกลาง (intermediate node) ซึ่งจะส่งข้อมูลนั้นต่อไปจนกระทั่งถึงโหนดปลายทาง ดังนั้นในเครือข่ายแอดฮอกนี้ ทุกโหนดต้องสามารถปฏิบัติตัวเป็นอุปกรณ์จัดเส้นทาง (router) ได้ ดังรูปที่ 2.1

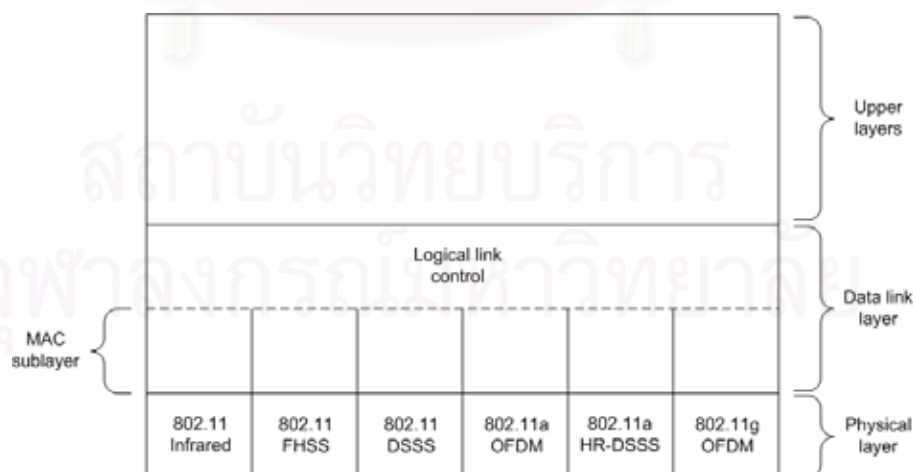


รูปที่ 2.1 ลักษณะของเครือข่ายแอดฮอก

2.2 มาตรฐาน IEEE 802.11

เป็นมาตรฐานที่ได้รับการตีพิมพ์ครั้งแรกในปี 1997 [27] เพื่อใช้กำหนดคุณลักษณะของตัวกลางทางกายภาพ (physical medium) และการเข้าถึงตัวกลาง (medium access) สำหรับเครือข่ายไร้สาย ซึ่งในตอนแรกได้มีการกำหนดให้ชั้นกายภาพ (physical layer) ทำการส่งได้ 3 รูปแบบ โดยแบบแรกจะส่งผ่านอินฟราเรด (infrared) ส่วนอีกสองแบบจะส่งผ่านคลื่นวิทยุความถี่ 2.4 GHz ที่มีมอดูเลชันแบบ FHSS (Frequency Hopping Spread Spectrum) และ DSSS (Direct Sequence Spread Spectrum) โดยความถี่ที่ใช้นี้อยู่ในย่าน ISM (Industrial Scientific and Medical) ซึ่งเป็นย่านที่ถูกจัดสรรให้ใช้งานด้านวิทยาศาสตร์ อุตสาหกรรม และการแพทย์ได้อย่างสาธารณะ หลังจากนั้นก็ได้มีการกำหนดมาตรฐานขึ้นมาเพิ่มอีกเพื่อให้มีแบนด์วิดท์ในการส่งสูงขึ้น ด้วยการใช้อมอดูเลชันสองรูปแบบใหม่ คือ OFDM (Orthogonal Frequency Division Multiplexing) และ HR-DSSS (High Rate Direct Sequence Spread Spectrum) ซึ่งมาตรฐานที่เป็นที่รู้จักดีเช่น 802.11a, 802.11b และ 802.11g โดยระดับชั้นโพรโทคอล (protocol stack) ของมาตรฐาน IEEE 802.11 แสดงได้ดังรูปที่ 2.2

จากรูปจะเห็นว่าชั้นเชื่อมโยงข้อมูล (data link layer) สามารถแบ่งเป็นชั้นย่อยได้อีก 2 ชั้น คือ LLC (logical link control) และ MAC (medium access control) โดยชั้นย่อย LLC จะทำหน้าที่ควบคุมความผิดพลาด (error control) และควบคุมอัตราการส่ง (flow control) ข้อมูล พร้อมทั้งจัดการให้รูปแบบของกลุ่มข้อมูลที่ใช้ติดต่อกับชั้นเครือข่าย (network layer) เป็นรูปแบบเดียวกันถึงแม้จะถูกส่งด้วยมาตรฐาน IEEE 802 ที่ต่างกันก็ตาม ส่วนชั้นย่อย MAC จะทำหน้าที่ควบคุมการเข้าถึงตัวกลางในการส่งข้อมูล เนื่องจากระบบทดสอบในวิทยานิพนธ์นี้ใช้อุปกรณ์ตามมาตรฐาน IEEE 802.11b ในการทำงาน ดังนั้นจะขอลำดับถึงรายละเอียดของ IEEE 802.11b เท่านั้น



รูปที่ 2.2 ระดับชั้นโพรโทคอลของ IEEE 802.11

2.2.1 ชั้นกายภาพตามมาตรฐาน IEEE 802.11b

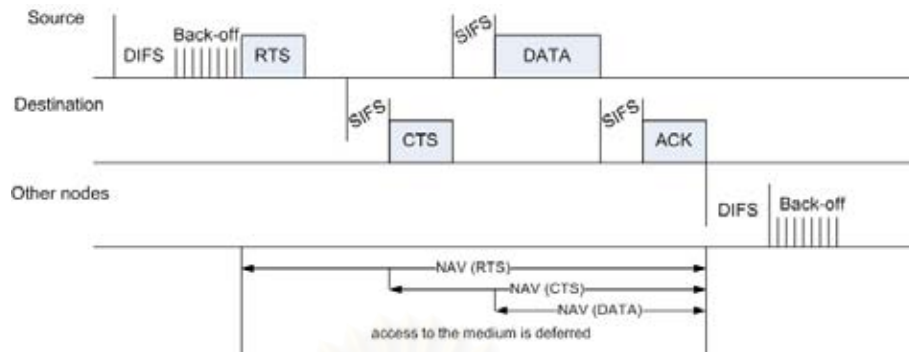
มาตรฐานนี้จะสามารถรองรับอัตราเร็วในการส่งได้ 4 ระดับ คือ 11, 5.5, 2, 1 Mbps โดยจะมีลักษณะมอดูเลชันที่ต่างกัันดังแสดงในตารางที่ 2.1 ซึ่งความสามารถในการส่งแต่ละระดับจะขึ้นอยู่กับอัตราส่วนความแรงของคลื่นพาห์ที่ได้รับต่อสัญญาณรบกวน (carrier to noise ratio)

ตารางที่ 2.1 คุณลักษณะของชั้นกายภาพตามมาตรฐาน IEEE 802.11b

Data Rate	Chipping Code Length	Modulation	Symbol Rate	Bits/Symbol
1 Mbps	11 (Barker sequence)	DBPSK	1 Msps	1
2 Mbps	11 (Barker sequence)	DQPSK	1 Msps	2
5.5 Mbps	8 (CCK)	DQPSK	1.375 Msps	4
11 Mbps	8 (CCK)	DQPSK	1.375 Msps	8

2.2.2 การควบคุมการเข้าถึงตัวกลาง

ตามมาตรฐาน IEEE 802.11 กำหนดให้มีการเข้าถึงตัวกลางได้ 2 รูปแบบ คือ แบบ PCF (Point Coordinate Function) และแบบ DCF (Distributed Coordinate Function) โดย PCF เป็นการเข้าถึงตัวกลางแบบใช้ศูนย์กลางในการควบคุม ซึ่งจะใช้กับเครือข่ายที่มีแอสซิงโครนัส ส่วนแบบ DCF จะเป็นการเข้าถึงตัวกลางแบบไม่ใช้ศูนย์กลางในการควบคุม ซึ่งเป็นรูปแบบที่ใช้ในเครือข่ายแอสซิงโครนัส ดังนั้นในส่วนนี้จะขอกล่าวถึงเฉพาะ DCF เท่านั้น โดยในรูปแบบ DCF นี้ได้กำหนดให้ใช้โพรโทคอลเข้าถึงตัวกลาง CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) ร่วมกับการใช้ binary exponential back-off โดยเมื่อโหนดใด ๆ ต้องการส่งข้อมูล จะทำการตรวจสอบก่อนว่าในเวลานั้นช่องสัญญาณว่างหรือไม่ หากตรวจสอบได้ว่าช่องสัญญาณว่างอยู่เป็นเวลาเท่ากับ DIFS (DCF Inter-Frame Spacing) ก็ทำการเปลี่ยนไปอยู่ในสถานะ back-off เพื่อรอที่จะส่ง RTS (Request-To-Send) ออกไป โดยการส่ง RTS นี้จะเป็นการบอกให้โหนดอื่น ๆ ในระยะการส่งทราบว่าช่องสัญญาณในขณะนี้ไม่ว่างแล้ว เนื่องจากกำลังจะมีการส่งข้อมูลเกิดขึ้น หากต้องการจะส่งข้อมูลจะต้องรอก่อนว่าโหนดต้นทางดังกล่าวส่งเรียบร้อยแล้ว โดยระยะเวลาที่ใช้ในการส่งข้อมูลจะถูกนับมาเทียบกับ RTS ด้วย ซึ่งค่านี้จะนำมาใช้ในการกำหนด NAV (Network Allocation Vector) เพื่อแทนสถานะว่าขณะนี้ไม่สามารถส่งข้อมูลใด ๆ ได้ หลังจากโหนดปลายทางได้รับ RTS แล้วจะรอเป็นระยะเวลาเท่ากับ SIFS (Short Inter-Frame Spacing) ก่อน จากนั้นจะส่ง CTS (Clear-To-Send) กลับมาให้โหนดต้นทางเพื่อบอกให้ทราบว่าโหนดปลายทางอยู่ในสถานะพร้อมรับข้อมูลแล้ว และยังเป็นการแจ้งให้โหนดอื่น ๆ ที่อยู่ในระยะการส่งของโหนดปลายทางทราบเช่นกันว่าช่องสัญญาณกำลังจะถูกใช้งาน หากต้องการจะส่งข้อมูลให้รอก่อน เมื่อโหนดต้นทางได้รับ CTS ตอบกลับมา ก็จะรอเป็นเวลา SIFS แล้วจึงค่อยส่งข้อมูลออกไป และเมื่อโหนดปลายทางได้รับข้อมูลแล้วก็รอเป็นเวลา SIFS แล้วส่ง ACK (Acknowledge) ตอบกลับมา โดยขั้นตอนการส่งข้อมูลที่กล่าวมานี้แสดงดังรูปที่ 2.3



รูปที่ 2.3 ขั้นตอนการเข้าถึงช่องสัญญาณตามมาตรฐาน 802.11

เวลาที่ต้องรอกอยู่ในสถานะ back-off จะถูกสุ่มขึ้นตามสมการ (2.1) โดย CW หรือ contention window จะมีค่าเริ่มต้นที่ CW_{min} ในการพยายามส่งครั้งแรก แต่หากการส่งล้มเหลวค่านี้ก็เพิ่มขึ้นตามสมการ (2.2) และจะเพิ่มขึ้นไปเรื่อย ๆ ทุกครั้งที่การส่งล้มเหลวจนกว่าจะถึง CW_{max} จึงจะไม่เพิ่มขึ้นอีก ซึ่งค่า CW_{min} , CW_{max} และ t_{slot} ที่มาตรฐาน IEEE 802.11 กำหนดไว้แสดงดังตารางที่ 2.2

$$\text{backoff window} = \text{rand}(0, CW) * t_{slot} \tag{2.1}$$

$$CW_{new} = 2 * CW_{current} - 1 \tag{2.2}$$

ตารางที่ 2.2 พารามิเตอร์ต่าง ๆ ตามมาตรฐาน IEEE 802.11

Parameter	802.11 (FHSS)	802.11 (DSSS)	802.11 (IR)	802.11b
t_{slot}	50 μsec	20 μsec	8 μsec	20 μsec
SIFS	28 μsec	10 μsec	10 μsec	10 μsec
PIFS	SIFS + t_{slot}			
DIFS	SIFS + (2 * t_{slot})			
CW_{min}	15	31	63	31
CW_{max}	1023	1023	1023	1023

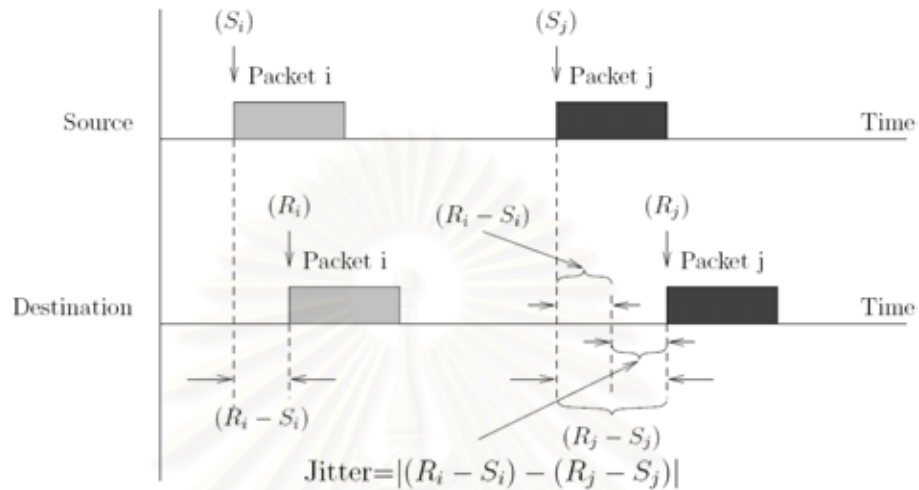
2.3 การคำนวณจิตเตอร์

จิตเตอร์หรือการแปรผันของค่าประวิงเวลาของแต่ละกลุ่มข้อมูล สามารถหาได้จากสมการ (2.3) โดยพารามิเตอร์ต่าง ๆ แสดงดังรูปที่ 2.4

$$\text{Jitter} = |(R_j - S_j) - (R_i - S_i)| = |(R_j - R_i) - (S_j - S_i)| \tag{2.3}$$

โดยที่ S_i คือ เวลาที่โนดต้นทางส่งกลุ่มข้อมูลที่ i ออกไป

R_i คือ เวลาที่โนดปลายทางได้รับกลุ่มข้อมูลที่ i



รูปที่ 2.4 แผนภาพแสดงการหาค่าจitterของข้อมูลกลุ่มที่ i และ j

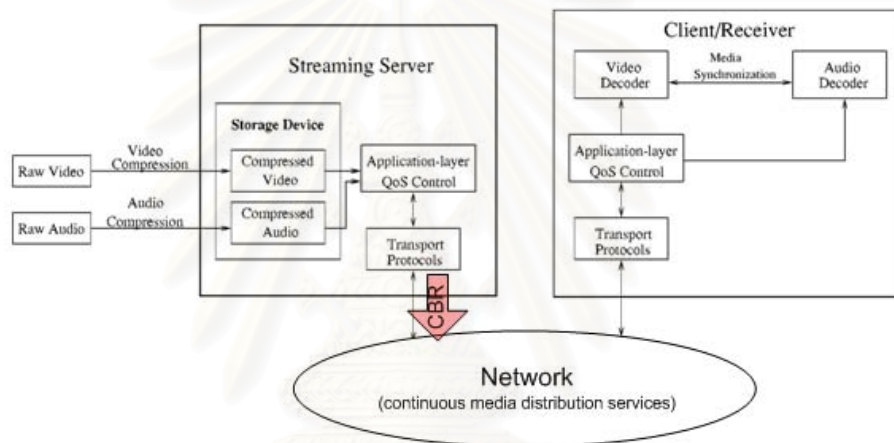
2.4 การส่งข้อมูลสื่อผสมแบบต่อเนื่อง

ปัจจุบันการให้บริการสื่อผสมผ่านเครือข่ายอินเทอร์เน็ตได้มีการเติบโตขึ้นเรื่อย ๆ เนื่องจากเทคโนโลยีการสื่อสารที่ก้าวหน้าขึ้นอย่างรวดเร็วทำให้สามารถรองรับการใช้งานในรูปแบบต่าง ๆ ได้มากขึ้น โดยที่ปกติแล้วในการเรียกดูข้อมูลสื่อผสมผ่านทางเครือข่ายนั้น สามารถทำได้ 2 วิธี คือ ทำการดาวน์โหลดแฟ้มข้อมูล (file) นั้น ๆ มาเก็บไว้ในเครื่องของผู้ใช้บริการให้ครบทั้งแฟ้มข้อมูลก่อน จากนั้นจึงค่อยนำออกแสดงผล ซึ่งวิธีนี้มีข้อดีคือ การแสดงภาพจะมีความต่อเนื่องตลอด และยังสามารถเรียกดูส่วนอื่น ๆ ของแฟ้มข้อมูลนั้นได้ทันทีด้วย เนื่องจากมีข้อมูลของทั้งแฟ้มข้อมูลเก็บอยู่ในเครื่องของผู้ใช้บริการแล้ว แต่ข้อเสียที่สำคัญของวิธีนี้คือต้องใช้เวลาในการดาวน์โหลดข้อมูลมาทั้งแฟ้มข้อมูลก่อน ซึ่งหากแฟ้มข้อมูลมีขนาดใหญ่มากก็จะเสียเวลารอนานด้วยเช่นกัน ส่วนอีกวิธีคือการส่งแบบต่อเนื่อง โดยวิธีนี้ไม่จำเป็นต้องรอให้ดาวน์โหลดข้อมูลครบทั้งแฟ้มข้อมูลก่อน แต่จะเป็นการแสดงผลหลังจากดาวน์โหลดแฟ้มข้อมูลมาไว้ในบัฟเฟอร์ได้บางส่วน ในขณะที่ทำการดาวน์โหลดเพิ่มไปพร้อม ๆ กันด้วย ดังนั้นในวิธีนี้ผู้ใช้บริการจะสามารถเปิดแฟ้มข้อมูลได้เกือบจะทันทีที่เริ่มดาวน์โหลด แต่ก็มีข้อเสียคือ หากจitter มีค่ามากจะทำให้ในการแสดงผลภาพและเสียงไม่ต่อเนื่อง ส่งผลให้ผู้ใช้บริการไม่พอใจได้ แต่อย่างไรก็ตาม วิธีการส่งแบบต่อเนื่องนี้ก็ยังเป็นที่ยอมรับอย่างมาก เพราะนอกจากจะไม่ต้องเสียเวลารอดาวน์โหลดข้อมูลทั้งแฟ้มข้อมูลแล้ว ยังมีข้อดีในเชิงธุรกิจด้วย เนื่องจากวิธีนี้ผู้ใช้บริการไม่สามารถเก็บแฟ้มข้อมูลที่ดาวน์โหลดมาไว้เพื่อเผยแพร่ต่อได้ ทำให้เพียงแค่เรียกดูจากเครื่องแม่ข่าย (server) เท่านั้น อีกทั้งการส่งแบบต่อเนื่องยังสามารถให้บริการแพร่สัญญาณสด (live broadcasting) ได้อีกด้วย ดังนั้นหากสามารถควบคุมจitter ให้มีค่าน้อยได้ ก็จะเป็นการกำจัดข้อเสียในด้านนี้ของการส่ง

แบบต่อเนื่องออกไป ซึ่งไม่เพียงแต่จะทำให้คุณภาพการให้บริการดีขึ้นเท่านั้น ยังสามารถช่วยลดขนาดของบัฟเฟอร์ที่ โหนดปลายทางได้อีกด้วย โดยการให้บริการข้อมูลสื่อผสมแบบต่อเนื่องที่ใช้กันทั่วไปมีอยู่ 2 ลักษณะคือ

2.4.1 การส่งตามคำขอ

จะเป็นลักษณะของข้อมูลที่มีการบันทึกและบีบอัดไว้แล้ว โดยจัดเก็บเป็นแพ้มข้อมูลอยู่ในอุปกรณ์หน่วยเก็บ เมื่อมีผู้ใช้ร้องขอจึงค่อยทำการส่งออกไปด้วยอัตราบิตคงที่ที่เหมาะสมกับความสามารถของช่องสัญญาณของผู้ใช้ บริการ ซึ่งลักษณะนี้ผู้ใช้บริการจะสามารถเรียกดูข้อมูลได้ตลอดเวลา โดยที่การแสดงผลจะเริ่มต้นที่ส่วนแรกของ แพ้มข้อมูลเสมอ หลังจากนั้นหากผู้ใช้บริการต้องการเรียกดูส่วนอื่นของแพ้มข้อมูลก็สามารถร้องขอมาได้ โครงสร้างของการส่งในลักษณะนี้แสดงดังรูปที่ 2.5

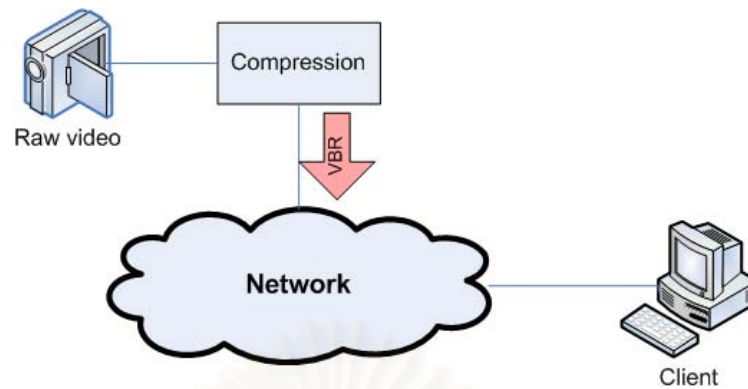


รูปที่ 2.5 โครงสร้างของการส่งแบบต่อเนื่องตามคำขอ

2.4.2 การส่งสัญญาณสด

ในวิธีนี้ข้อมูลที่ส่งจะเป็นการจับภาพจากต้นฉบับ (source) ในเวลาจริง จากนั้นจึงทำการบีบอัดแล้วส่งออกไป ซึ่งข้อมูลที่ถูกส่งออกไปนี้จะมีอัตราบิตไม่คงที่ เนื่องจากขนาดของข้อมูลในแต่ละเวลาขึ้นอยู่กับเนื้อหาของภาพในขณะนั้น ดังแสดงในรูปที่ 2.6 และเนื่องจากวิธีนี้เป็นการจับภาพในเวลาจริงทำให้ผู้ใช้บริการสามารถเรียกดูได้เฉพาะตอนที่มีการส่งสัญญาณมาเท่านั้น ซึ่งข้อมูลที่ได้รับจะเป็นสัญญาณภาพในขณะนั้น ๆ ไม่สามารถเลือกดูได้เหมือนกรณีการส่งตามคำขอ โดยที่วิธีการส่งแบบต่อเนื่องที่เป็นสัญญาณสดนี้สามารถนำไปใช้ในการแพร่สัญญาณเพื่อส่งออกไปหาผู้ใช้บริการหลายคนพร้อม ๆ กันได้ด้วย

การควบคุมจิตเตอร์ที่ใช้อยู่ในปัจจุบันนั้นจะเป็นการควบคุมที่โหนดปลายทางเพียงโหนดเดียว โดยในตอนเริ่มต้นจะเก็บข้อมูลส่วนแรกของวิดิทัศน์ไว้ในบัฟเฟอร์ให้ได้จำนวนหนึ่งก่อน แล้วจึงค่อยนำข้อมูลในบัฟเฟอร์นั้นออกแสดงผล ในขณะที่เดียวกันก็ทำการรับข้อมูลส่วนต่อมาเพิ่มเข้าไปในบัฟเฟอร์ด้วย ทำให้สามารถแสดงผลต่อเนื่องได้เรื่อย ๆ โดยค่าประวิงเวลาในตอนเริ่มต้นมีค่าน้อยมาก ซึ่งบัฟเฟอร์ที่ใช้เก็บข้อมูลนี้จะต้องมีขนาดใหญ่พอที่จะรองรับความ

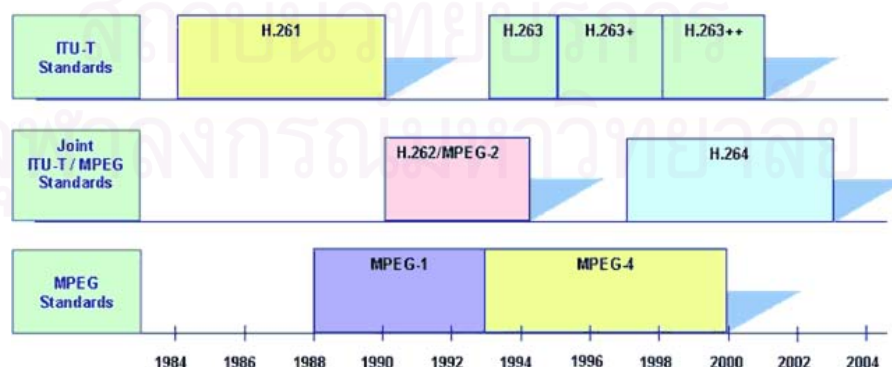


รูปที่ 2.6 โครงสร้างของการส่งสัญญาณสดแบบต่อเนื่อง

แปรผันของค่าประวิงเวลาในแต่ละกลุ่มข้อมูลได้ โดยเฉพาะอย่างยิ่งในเครือข่ายไร้สายที่ค่าประวิงเวลาที่มีความไม่แน่นอนสูง จึงทำให้บัพเฟอร์ในส่วนนี้ต้องมีขนาดใหญ่มากขึ้นไปอีก แต่เนื่องจากข้อจำกัดทางด้านขนาดของอุปกรณ์ที่ใช้ในเครือข่ายแอตฮอกอาจจะทำให้ไม่สามารถเพิ่มขนาดของบัพเฟอร์ได้ตามต้องการ ดังนั้นหากต้องการส่งข้อมูลแบบต่อเนื่องในเครือข่ายแอตฮอก สิ่งที่ต้องทำคือควบคุมจิตเตอร์ให้อยู่ในระดับที่โนดปลายทางสามารถจะรองรับได้

2.5 มาตรฐานในการบีบอัดข้อมูล

โดยปกติแล้วข้อมูลวีดิทัศน์ที่ได้จากต้นฉบับจะเป็นข้อมูลดิบที่มีขนาดใหญ่มาก ไม่สะดวกต่อการบันทึกจัดเก็บ อีกทั้งยังไม่สามารถส่งผ่านเครือข่ายได้ ดังนั้นจึงจำเป็นต้องใช้การบีบอัดเข้าช่วยเพื่อลดขนาดของข้อมูลลง ซึ่งในปัจจุบันได้มีการพัฒนาประสิทธิภาพของกระบวนการบีบอัดขึ้นเรื่อย ๆ เพื่อให้ได้อัตราการบีบอัดสูง ในขณะที่คุณภาพของภาพและเสียงยังดีอยู่ รวมทั้งเพื่อให้เหมาะกับการใช้งานในแต่ละประเภท โดยมี 2 องค์กรหลัก คือ ITU-T (International Telecommunication Union) และ ISO/IEC (International Standard Organization) เป็นผู้พัฒนาและกำหนดขึ้นเป็นมาตรฐาน [28] ดังรูปที่ 2.7 โดยหลัก ๆ จะแบ่งได้ดังนี้



รูปที่ 2.7 วิวัฒนาการของมาตรฐานการบีบอัดวีดิทัศน์

2.5.1 มาตรฐานของ ITU-T

- H.261 [20] เป็นมาตรฐานเริ่มแรกที่ออกแบบมาเพื่อใช้กับระบบการประชุมสัมมนาทางวิดีโอผ่านโครงข่าย ISDN (ISDN videoconference network)
- H.263 [21] พัฒนาขึ้นมาจาก H.261 เพื่อให้วิดีโอมีคุณภาพดีขึ้นและสามารถใช้งานในอัตราบิตต่ำได้
- H.264 [22] เป็นมาตรฐานใหม่ล่าสุดที่ทั้งสององค์กรร่วมกันพัฒนาเพื่อให้การบีบอัดมีประสิทธิภาพดียิ่งขึ้นสามารถใช้ได้กับงานทางวิดีโอดิจิทัลได้อย่างเต็มรูปแบบ แต่ในปัจจุบันยังไม่มีการนำมาใช้งานจริง

2.5.2 มาตรฐานของ ISO/IEC

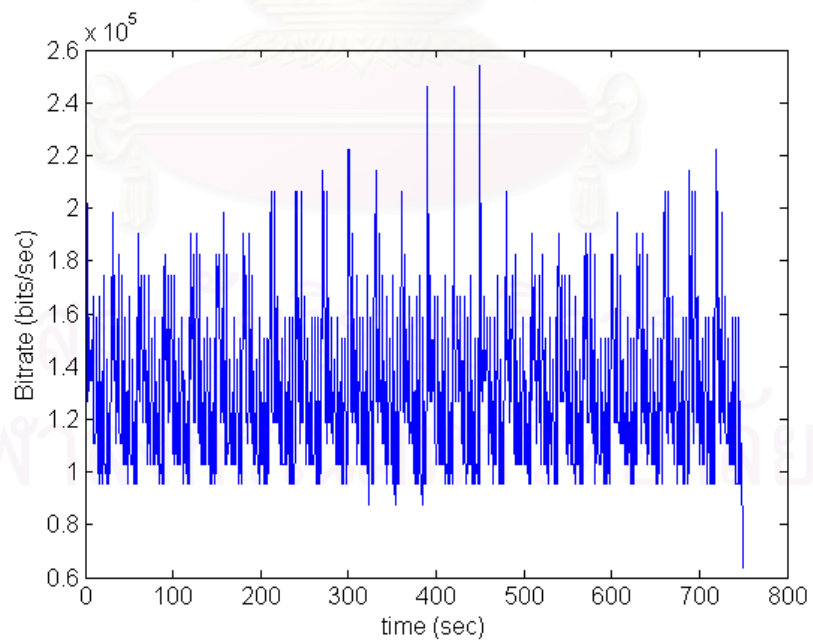
- MPEG-1 [23] เป็นมาตรฐานที่ออกแบบมาสำหรับการเก็บวิดีโอดิจิทัล (digital video) ลงในสื่อเก็บข้อมูลประเภทซีดี (CD) เหมาะกับการใช้งานที่อัตราบิตสูง ๆ อยู่ในช่วงประมาณ 1.5 เมกะบิตต่อวินาที
- MPEG-2 [24] เป็นมาตรฐานที่ทั้งสององค์กรร่วมกันพัฒนาขึ้นจาก MPEG-1 เพื่อใช้สำหรับโทรทัศน์ดิจิทัลความละเอียดสูง (high definition television :HDTV) โทรทัศน์ผ่านดาวเทียมและเคเบิล รวมถึงพัฒนาการเก็บข้อมูลลงสื่อเก็บข้อมูลประเภทดีวีดี (DVD) เพื่อให้มีประสิทธิภาพมากยิ่งขึ้น โดยมีอัตราบิตอยู่ในช่วงตั้งแต่ 2 เมกะบิตต่อวินาที จนถึง 8 เมกะบิตต่อวินาที
- MPEG-4 [25] เป็นมาตรฐานที่พัฒนาขึ้นเพื่อให้สามารถใช้งานได้ตั้งแต่อัตราบิตต่ำไปจนถึงอัตราบิตสูง ๆ ในขณะที่คุณภาพของภาพและเสียงยังดีอยู่ โดยมีจุดประสงค์เพื่อใช้สำหรับการส่งข้อมูลสื่อผสมผ่านเครือข่าย และเนื่องจากสามารถใช้ในอัตราบิตต่ำมากได้ จึงเหมาะสำหรับใช้ในเครือข่ายไร้สายด้วย โดยมีอัตราบิตต่ำได้ถึง 10 กิโลบิตต่อวินาที จนถึง 10 เมกะบิตต่อวินาที

ซึ่งในส่วนของวิทยานิพนธ์นี้จะเลือกใช้การบีบอัดตามมาตรฐาน MPEG-4 เนื่องจากเป็นมาตรฐานที่นิยมใช้ในการส่งข้อมูลสื่อผสมแบบต่อเนื่อง ประกอบกับสามารถใช้กับการส่งด้วยอัตราบิตต่ำได้ จึงเหมาะกับการใช้งานในเครือข่ายแอดฮอกที่เป็นเครือข่ายไร้สายด้วย

สำหรับการทดลองในวิทยานิพนธ์นี้ จะใช้แฟ้มข้อมูลวิดีโอต้นฉบับ (video source file) ที่มีชื่อว่า salesman.qcif ขนาด 176x144 จุดภาพ (pixel) ดังแสดงตัวอย่างในรูปที่ 2.8 และนำไปบีบอัดด้วยมาตรฐาน MPEG-4 ซึ่งในการทดลองจะส่งออกไปด้วยอัตราเร็ว 15 เฟรมต่อวินาที และกำหนดให้ขนาดของกลุ่มข้อมูลเท่ากับ 1024 ไบต์ โดยรูปแบบของกราฟฟิกจะแบ่งเป็น 2 ชนิด คือแบบอัตราบิตคงที่ และอัตราบิตแปรผันได้ ในรูปที่ 2.9 แสดงอัตราบิตที่เวลาใด ๆ ของกราฟฟิกแบบอัตราบิตแปรผันได้ ซึ่งค่าที่ได้ขึ้นอยู่กับเนื้อหาของภาพในแต่ละเฟรม ส่วนกราฟฟิกแบบอัตราบิตคงที่ในการทดลองจะกำหนดให้มีอัตราบิตเท่ากับอัตราบิตเฉลี่ยของกราฟฟิกแบบอัตราบิตแปรผันได้นั้นคือ 128.15 กิโลบิตต่อวินาที



รูปที่ 2.8 ตัวอย่างเพิ่มข้อมูลวิดีโอที่สนับต้นฉบับที่ใช้ในการทดลอง



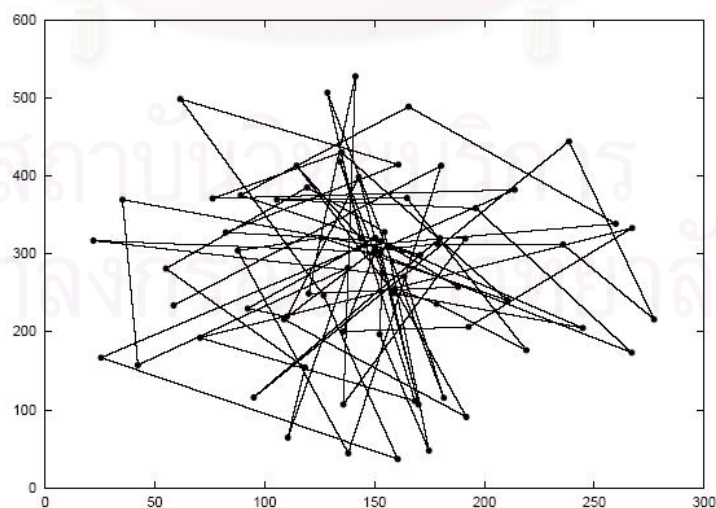
รูปที่ 2.9 อัตราบิตที่ใช้ส่งของกราฟฟิกแบบอัตราบิตแปรผันได้

2.6 แบบจำลองการเคลื่อนที่

แบบจำลองการเคลื่อนที่ที่ใช้ในการทดลองถือว่ามีความสำคัญต่อการประเมินผลโพรโทคอลหรือวิธีการที่นำเสนอมาก โดยสามารถแบ่งได้เป็น 2 ประเภทหลัก ๆ คือ แบบจำลองที่การเคลื่อนที่ของแต่ละโหนดไม่ขึ้นแก่กัน (entity mobility model) และแบบจำลองที่โหนดเคลื่อนที่ไปเป็นกลุ่ม (group mobility model) [29] ซึ่งแบบจำลองที่การเคลื่อนที่ของแต่ละโหนดไม่ขึ้นแก่กันยังแบ่งออกได้อีกหลายแบบ แต่มีอยู่ 2 แบบที่นักวิจัยส่วนใหญ่เลือกใช้ คือ

2.6.1 แบบจำลองการเคลื่อนที่การเดินแบบสุ่ม

ในแบบจำลองการเคลื่อนที่การเดินแบบสุ่ม (Random Walk Mobility Model) [30] นี้ โหนดจะเคลื่อนที่จากตำแหน่งปัจจุบันไปตำแหน่งใหม่โดยการสุ่มทิศทางและความเร็วในการเคลื่อนที่ขึ้น ซึ่งจะทำให้การสุ่มจากช่วง $[\text{speedmin}, \text{speedmax}]$ และ $[0, 2\pi]$ ที่มีการกำหนดไว้ก่อนแล้ว โดยการเคลื่อนที่แต่ละครั้งนั้นจะมีช่วงเวลาที่ไม่เคลื่อนที่คงที่ หรือระยะทางที่เคลื่อนไปคงที่ อย่างใดอย่างหนึ่ง และหากโหนดมีการเคลื่อนที่ไปชนขอบของพื้นที่ในการจำลอง โหนดนั้นจะกระเด็นกลับมาตามกฎการสะท้อน จากนั้นก็จะเคลื่อนที่ต่อไปในทิศทางใหม่ รูปที่ 2.10 แสดงตัวอย่างรูปแบบการเคลื่อนที่ของโหนดเมื่อใช้แบบจำลองการเคลื่อนที่การเดินแบบสุ่ม เมื่อกำหนดให้พื้นที่ในการจำลองเท่ากับ 300×600 เมตร ความเร็วในการเคลื่อนที่ของโหนดอยู่ระหว่าง 0 ถึง 10 เมตรต่อวินาที ระยะเวลาที่โหนดเคลื่อนที่แต่ละครั้งเท่ากับ 60 วินาที และให้จุดเริ่มต้นของโหนดคือจุดกึ่งกลางของพื้นที่นั้น ซึ่งจะเห็นว่าการเคลื่อนที่ในแบบจำลองนี้ไม่สามารถแสดงรูปแบบการเคลื่อนที่ในสถานการณ์จริงได้อย่างถูกต้อง เนื่องจากทำให้เกิดกรณีที่เป็นไปไม่ได้ในการเคลื่อนที่จริง เช่น การเคลื่อนที่ไปด้วยความเร็วแล้วหยุดกะทันหันและเปลี่ยนทิศทาง การเคลื่อนที่ที่พื้นที่

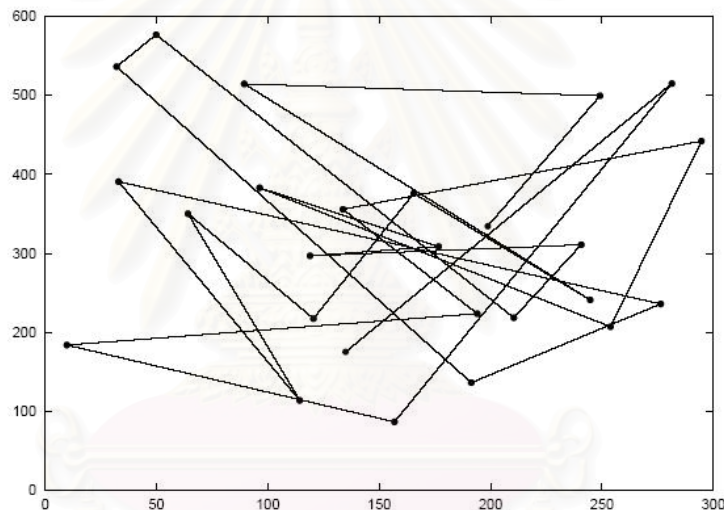


รูปที่ 2.10 รูปแบบการเคลื่อนที่ของโหนดตามแบบจำลองการเคลื่อนที่การเดินแบบสุ่ม

2.6.2 แบบจำลองการเคลื่อนที่โดยสุ่มปลายทาง

ในแบบจำลองการเคลื่อนที่แบบสุ่มปลายทาง (Random Waypoint Mobility Model) [31] โหนดจะมีการหยุดชั่ววาระหนึ่ง (pause time) ก่อนที่จะเปลี่ยนทิศทางหรือความเร็วในการเคลื่อนที่ โดยเวลาที่โหนดหยุดนิ่งนี้จะมีค่าคงที่ตามที่ได้อำหนดไว้ล่วงหน้าแล้ว ในตอนเริ่มต้นนั้นโหนดจะหยุดที่ตำแหน่งเริ่มต้นเป็นเวลา pause time จากนั้นจะสุ่มตำแหน่งปลายทางที่จะเคลื่อนที่ไป รวมทั้งความเร็วในการเคลื่อนที่ด้วย ซึ่งสุ่มได้จากการแจกแจงเอกรูป (uniform distribution) ในช่วง [minspeed, maxspeed] และเมื่อถึงตำแหน่งปลายทางแล้วก็ต้องหยุดเป็นเวลา pause time อีกครั้งก่อนที่จะสุ่มหาตำแหน่งใหม่ต่อไป รูปที่ 2.11 แสดงตัวอย่างรูปแบบการเคลื่อนที่ของโหนดเมื่อใช้แบบจำลองการเคลื่อนที่โดยสุ่มปลายทาง เมื่อกำหนดให้ความเร็วในการเคลื่อนที่ของโหนดอยู่ระหว่าง 0 ถึง 10 เมตรต่อวินาที และจุดเริ่มต้นที่สุ่มได้ในตัวอย่างนี้อยู่ที่ตำแหน่ง (133,180)

ซึ่งจากทั้งสองแบบจำลองจะเห็นว่า ค่อนข้างจะเป็นไปได้ยากในสถานการณ์จริง เนื่องจากการเคลื่อนที่



รูปที่ 2.11 รูปแบบการเคลื่อนที่ของโหนดตามแบบจำลองการเคลื่อนที่โดยสุ่มปลายทาง

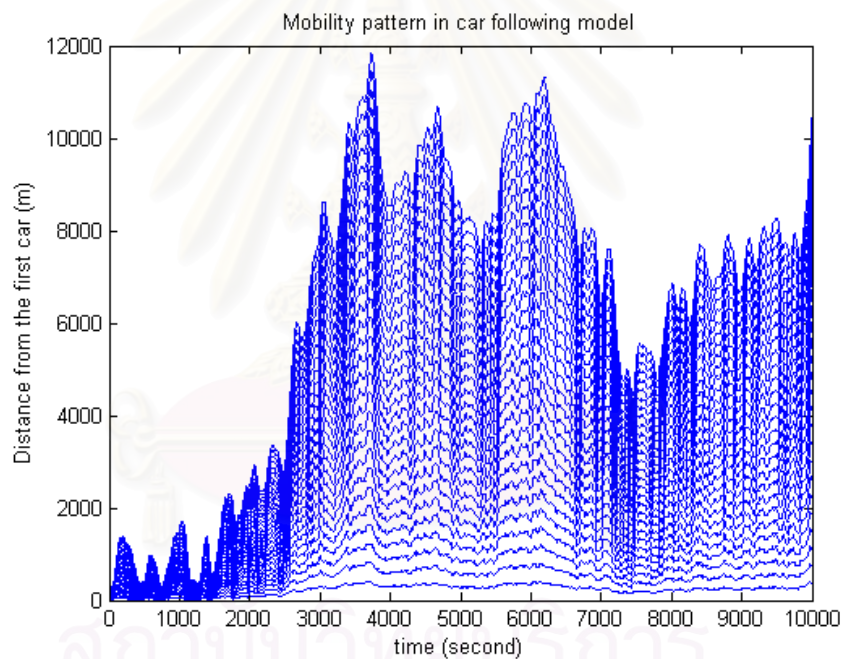
ไปมาแบบไม่มีทิศทางแน่นอนอยู่ในพื้นที่ที่จำกัด โดยพื้นที่ดังกล่าวก็เป็นพื้นที่โล่งที่ไม่มีสิ่งกีดขวางใด ๆ เลย ดังนั้นเพื่อให้การเคลื่อนที่มีลักษณะใกล้เคียงกับสถานการณ์จริงมากขึ้น ระบบทดสอบในวิทยานิพนธ์นี้จึงเลือกใช้แบบจำลองการเคลื่อนที่ตามกันของรถ [19] จำลองสถานการณ์การใช้งานเครือข่ายแอตสอกของรถบนถนนที่มีช่องเดินรถเดียว (single lane)

2.6.3 แบบจำลองการเคลื่อนที่ตามกันของรถ

แบบจำลองการเคลื่อนที่ตามกันของรถ (Car Following Model) นี้เป็นลักษณะการเคลื่อนที่ของรถแล่นตามกันบนถนนที่มีช่องเดินรถเดียว ความเร่งของรถคันที่ตามหลังจะขึ้นอยู่กับความเร็วของรถคันข้างหน้า เนื่องจากไม่

สามารถแข่งขันไปได้ ซึ่งความเร่งของรถคันหลังหาได้จากสมการ (2.4) โดยที่ $\dot{x}_f(t)$ และ $\ddot{x}_f(t)$ เป็นความเร็วและความเร่งของรถคันที่แล่นตามหลัง ณ เวลา t ใด ๆ ตามลำดับ ส่วน $\dot{x}_l(t)$ เป็นความเร็วของรถคันที่แล่นอยู่ข้างหน้า ณ เวลา t ใด ๆ λ เป็นค่าสัมประสิทธิ์การตอบสนองต่อสิ่งกระตุ้น และ T เป็นเวลาตอบสนอง (response time) ของคนขับ รูปที่ 2.12 แสดงตัวอย่างของรูปแบบการเคลื่อนที่ตามกันของรถ 30 คัน โดยตำแหน่งของรถทั้งหมดตอนเริ่มต้นได้จากการสุ่มแบบแจกแจงเอกรูป (uniform distribution) ภายในพื้นที่ 200 เมตร และความเร่งของรถคันแรกในแต่ละช่วงเวลา (time slot) จะกำหนดจากการสุ่มแบบแจกแจงเอกรูปเช่นกันโดยมีค่าอยู่ระหว่าง -2 ถึง 2 เมตรต่อวินาที² λ เท่ากับ 0.368 วินาที⁻¹ และ T เท่ากับ 0.9 วินาที ซึ่งค่า λ และ T ที่ใช้นี้อ้างอิงตาม [19] จากรูปจะเห็นว่าในตอนแรกรถจะแล่นอยู่รวมกันเป็นกลุ่ม และจะกระจายออกเรื่อย ๆ ในเวลาต่อมา

$$\ddot{x}_f(t + T) = \lambda[\dot{x}_l(t) - \dot{x}_f(t)] \quad (2.4)$$



รูปที่ 2.12 รูปแบบการเคลื่อนที่ของแบบจำลองการเคลื่อนที่ตามกันของรถ

2.7 ระบบทดสอบในงานวิจัยที่ผ่านมา

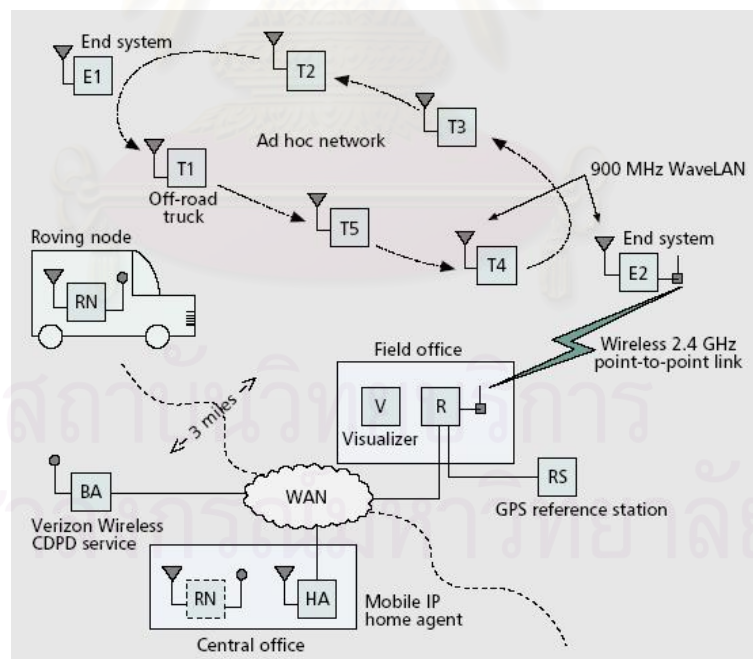
ที่ผ่านมางานวิจัยที่เกี่ยวข้องกับการสร้างระบบทดสอบการทำงานของเครือข่ายแอต สอกยังถือว่าไม่มากนัก เนื่องจากส่วนใหญ่ใช้วิธีการสร้างแบบจำลองโปรแกรมคอมพิวเตอร์ กลุ่มนักวิจัยที่ใช้วิธีการสร้างระบบทดสอบที่ผ่านมามักจะสร้างเพื่อพัฒนาและทดสอบประสิทธิภาพการทำงานของโพรโทคอลจัดเส้นทาง [11]-[13] ในหัวข้อนี้จะขอยกตัวอย่างงานวิจัยของระบบทดสอบที่มีโครงสร้างชัดเจนมา 3 บทความ โดยในบทความแรกจะเป็นระบบทดสอบ

แบบไร้สาย มีการเคลื่อนที่ของโนดจริง ส่วนในบทความที่สองก็เป็นระบบทดสอบแบบไร้สายเช่นกัน แต่ใช้การจำลองการเคลื่อนที่ของโนดแทนการเคลื่อนที่จริง และบทความสุดท้ายจะเป็นระบบทดสอบแบบมีสายที่อาศัยการจำลองการเชื่อมต่อของแต่ละคูโนดในเครือข่ายแอดฮอค

2.7.1 ระบบทดสอบของ D. A. Maltz และ J. Broch

ระบบทดสอบนี้เป็นระบบทดสอบขนาดใหญ่ของเครือข่ายแอดฮอคหลายช่วงเชื่อมต่อ (multihop) [11] ใช้ทดสอบประสิทธิภาพการทำงานของโพรโทคอลจัดเส้นทาง DSR [32] ในสถานการณ์จริง โดยระบบทดสอบประกอบด้วยส่วนเครือข่ายแอดฮอคที่มีพื้นที่ประมาณ 700x300 เมตร และมีเกตเวย์เชื่อมต่อกับอินเทอร์เน็ตด้วย ซึ่งการวัดผลทำได้จากการส่งโพลด์ที่เป็นเสียง (voice) ข้อมูลขนาดใหญ่ (bulk data) และข้อมูลที่ส่ง ณ เวลาจริง (real-time data) ในระหว่างที่โนดมีการเคลื่อนที่ อุปกรณ์ที่ใช้ในระบบทดสอบนี้ประกอบด้วยแล็ปท็อปคอมพิวเตอร์ที่ติดตั้งไว้กับรถยนต์เพื่อให้สามารถเคลื่อนที่ได้ และแผ่นวงจรประสานเครือข่ายไร้สาย (wireless network interface card) ที่ใช้คือ 900 MHz WaveLAN-I radios ซึ่งไม่มีโพรโทคอลการเข้าถึงช่องสัญญาณ (MAC protocol) ตามมาตรฐาน IEEE 802.11 รวมอยู่ด้วย เนื่องจากในเวลาที่ทำการสร้างระบบทดสอบนั้นยังไม่มีอุปกรณ์ที่รองรับ

รูปที่ 2.13 แสดงภาพรวมเชิงตรรกะ (logical overview) ของระบบทดสอบนี้ ซึ่งประกอบไปด้วยโนด 8 โหนด



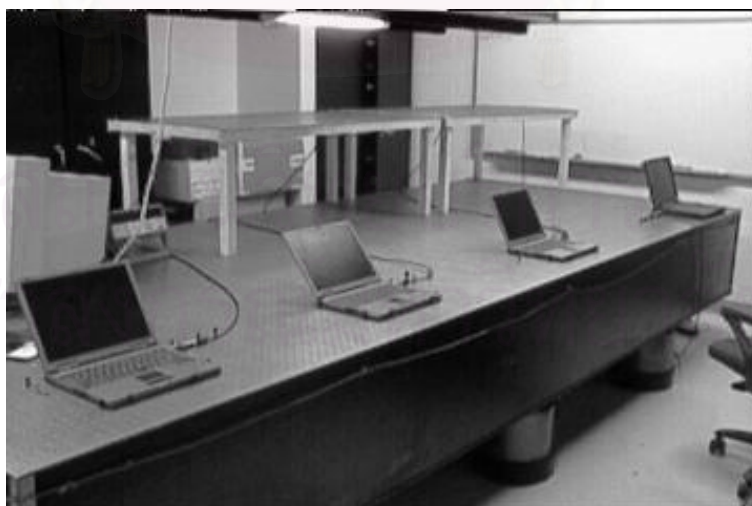
รูปที่ 2.13 ระบบทดสอบของ D. A. Maltz และ J. Broch

โดยเป็นโนดที่มีการเคลื่อนที่จริง ๆ ภายในเครือข่ายแอดฮอค 5 โหนด คือ T1-T5 และโนดที่ไม่เคลื่อนที่ (stationary node) อีก 2 โหนด คือ E1 และ E2 ส่วนอีก 1 โหนดจะเป็น roving node (RN) เคลื่อนที่จากสำนักงานกลาง

(central office) ไปยังส่วนเครือข่ายแอตฮอก โดยระหว่างที่โนด RN เคลื่อนที่ออกจากส่วนเครือข่ายเฉพาะที่ไร้สายบริเวณสำนักงานกลางแล้ว แต่ยังไม่ถึงพื้นที่ของส่วนเครือข่ายแอตฮอก โหนด RN จะยังคงเชื่อมต่อกับอินเทอร์เน็ตได้อยู่โดยผ่านทาง CDPD (cellular digital packet data) ดังนั้นโนด RN นี้จะต้องสามารถเชื่อมต่อได้กับ 3 เครือข่ายคือ ระบบเครือข่ายเฉพาะที่ไร้สาย ระบบ CDPD และ เครือข่ายแอตฮอก ซึ่งจะใช้ Mobile IP [33] เข้ามาช่วยในการติดต่อ ทำให้ยังสามารถติดต่อได้ถึงแม้ว่าจะเคลื่อนที่ออกไปจากเครือข่ายเดิมแล้ว ส่วนของเครือข่ายแอตฮอกจะมีการเชื่อมต่อกับส่วนของอินเทอร์เน็ตผ่านทางโนด E2 ซึ่งจะทำหน้าที่เป็นเกตเวย์เชื่อมต่อไปยังสำนักงานสนาม (field office) โดยใช้ความถี่ 2.4 GHz ซึ่งจะไม่รบกวนกับส่วนความถี่ที่ใช้ภายในเครือข่ายแอตฮอก และในการทดลองนี้จะมีจีพีเอส (global positioning system, GPS) คอยตรวจสอบตำแหน่งของโนดและส่งค่าให้แสดงผลทางเครื่องแสดงภาพ (visualizer) เพื่อทำการเฝ้าสังเกตได้ในระหว่างที่ทำการทดลอง

2.7.2 ระบบทดสอบ EWANT

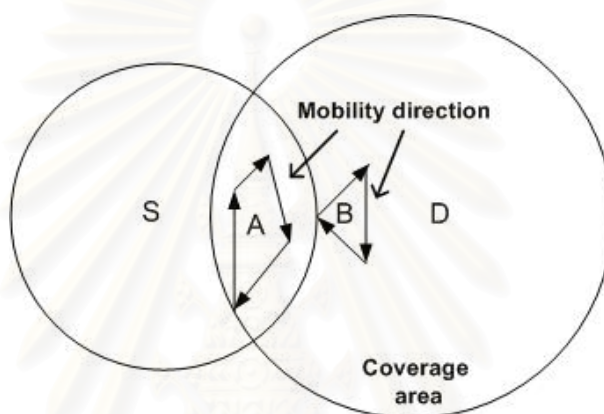
ระบบทดสอบ EWANT (Emulated Wireless Ad Hoc Network Testbed) นี้นำเสนอโดย S. Sanghani, T. X Brown, S. Bhandare และ S. Doshi [13] เพื่อใช้ทดสอบโพรโทคอลจัดเส้นทางใด ๆ โดยมีเป้าหมายคือเป็นระบบทดสอบที่คำนึงถึงการเคลื่อนที่ของโนด รวมถึงปรากฏการณ์ต่าง ๆ เนื่องจากคลื่นวิทยุ การไม่ขึ้นต่ออุปกรณ์ฮาร์ดแวร์และซอฟต์แวร์ที่ใช้งาน และสามารถทำการทดลองในสถานการณ์เดิมซ้ำได้ ซึ่งโดยส่วนใหญ่ของระบบทดสอบที่มีการทำมาแล้วนั้นมักจะไม่มีลักษณะเหล่านี้รวมอยู่ด้วยกัน แต่ในการเคลื่อนที่ของโนดในระบบทดสอบนี้ไม่ได้มีการเคลื่อนที่ที่เกิดขึ้นจริง ๆ จะใช้การจำลองขึ้นแทน โดยใช้ตัวมัลติเพลกซ์ (multiplexer, MUX) ต่อระหว่างเครื่องคอมพิวเตอร์กับสายอากาศ และอาศัยสวิทช์เป็นตัวควบคุมเพื่อกำหนดการเคลื่อนที่ รูปที่ 2.14 แสดงระบบทดสอบจริงในห้องปฏิบัติการ ซึ่งจะเห็นว่าเป็น



รูปที่ 2.14 ระบบทดสอบ EWANT

เพียงห้องเล็ก ๆ และแต่ละเครื่องคอมพิวเตอร์ก็วางอยู่ข้าง ๆ กันเท่านั้น โดยระบบทดสอบนี้ประกอบด้วยแล็ปท็อป

คอมพิวเตอร์ 4 เครื่องเพื่อเป็นโนด 4 โหนด และใช้แผ่นวงจรต่อประสานเครือข่ายไร้สายเป็น Cisco Aironet LMC 352 ต่อเข้ากับสายอากาศภายนอก และใช้ทองแดงหรือลวดมิกซ์เจอร์เพื่อป้องกันสัญญาณรั่ว (leakage signals) โดยในระบบทดสอบนี้ได้กำหนดรูปแบบการเคลื่อนที่ที่เป็นดังรูปที่ 2.15 ซึ่งทำได้โดยใช้สวิตซ์ในการสลับสายอากาศที่ต่ออยู่ผ่านตัวมัลติเพลกซ์แบบ 1 ต่อ 4 ดังนั้นโนดจะเคลื่อนที่ที่สลับไปมาได้สูงสุดเพียง 4 ตำแหน่งเท่านั้น และใช้โพรโทคอลการจัดเส้นทาง DSR ในการทดลอง โดยที่โนดต้นทาง (source node) คือ โหนด S และโนดปลายทาง (destination node) คือ โหนด D ซึ่งในการทดลองได้ทำการวัดผลกระทบจากการแปรค่าเวลารอการตอบรับ (acknowledge timeouts) ของแต่ละเส้นทางที่เป็นไปได้

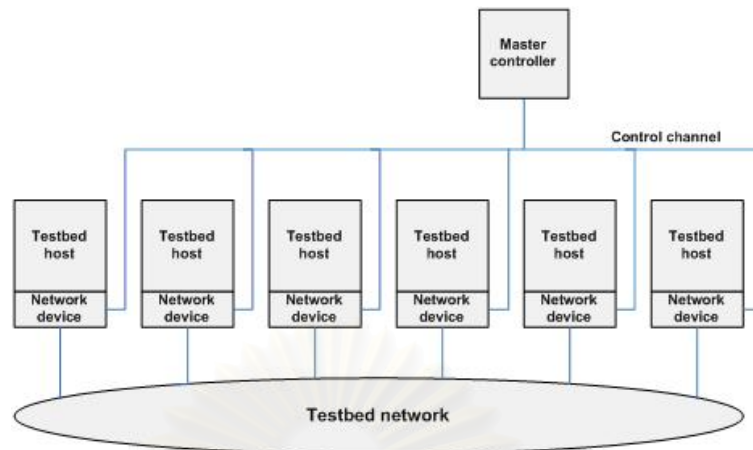


รูปที่ 2.15 ระยะการส่งและรูปแบบการเคลื่อนที่ของโนดในระบบทดสอบ EWANT

2.7.3 ระบบทดสอบของ Y. Zhang และ W. Li

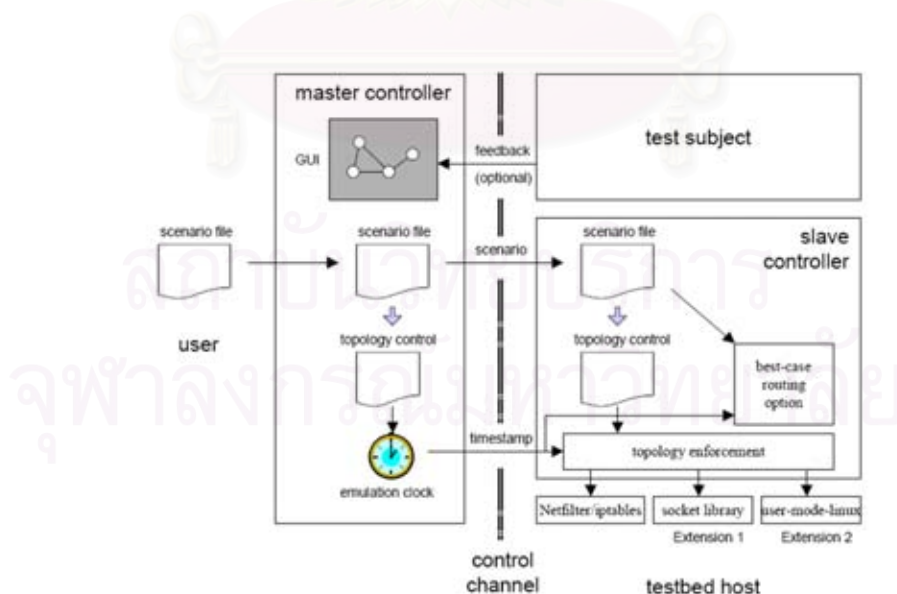
ระบบทดสอบนี้ [14] จำลองสภาพแวดล้อมต่าง ๆ ของเครือข่ายแอตฮอก เพื่อใช้ทดสอบการทำงานในลักษณะต่าง ๆ เช่น การพัฒนาและทดสอบประสิทธิภาพของโพรโทคอลจัดเส้นทางในเครือข่ายแอตฮอก ระบบทดสอบนี้ประกอบด้วยกลุ่มโนดที่ต่อเชื่อมกันผ่านทางระบบเครือข่ายแบบใช้สาย แต่ละโนดไม่มีการเคลื่อนที่จริง จะอาศัยโปรแกรมเลียนแบบการเคลื่อนที่ (mobility emulator) ที่เรียกว่า **MobiEmu** เพื่อจำลองสถานการณ์การเคลื่อนที่แบบต่าง ๆ ของแต่ละโนดแทน และยังสามารถกำหนดจำนวนโนดในเครือข่ายได้ตามต้องการด้วย

ระบบทดสอบนี้จะใช้เครื่องคอมพิวเตอร์ที่ติดตั้งระบบปฏิบัติการลินุกซ์ (Linux operating system) 1 เครื่องแทนโนด 1 โหนด และทุกเครื่องในระบบทดสอบนี้จะต้องการติดตั้งซอฟต์แวร์ควบคุม MobiEmu ไว้เพื่อจำลองการเคลื่อนที่ของโนดในรูปแบบการเคลื่อนที่ (scenario) ต่าง ๆ ซึ่งสามารถจะเลือกได้ ถึงแม้ว่าในทางกายภาพแล้วทุก ๆ โหนดจะสามารถติดต่อกันได้ แต่ MobiEmu จะอาศัยหลักการกรองกลุ่มข้อมูล (packet filtering) เพื่อแยกแต่ละโนดออกจากกันตามรูปแบบการเคลื่อนที่ที่ใช้ในการทดลอง รูปที่ 2.16 แสดงโครงสร้างการเชื่อมต่อของระบบทดสอบ MobiEmu นี้ โดยแต่ละเครื่อง (testbed host) จะเสมือนว่าเป็นแต่ละโนดที่เคลื่อนที่ ซึ่ง MobiEmu นี้จะมีการทำงานเป็นแบบเครื่องหลักและเครื่องลูกข่าย (master/slave) ตัวควบคุมหลัก (master



รูปที่ 2.16 โครงสร้างการเชื่อมต่อของ MobiEmu

controller) จะถูกดำเนินงาน (run) อยู่บนเครื่อง ๆ หนึ่งที่ถูกแยกออกมาต่างหาก ไม่ได้อยู่ในส่วนของระบบทดสอบด้วย ส่วนตัวควบคุมลูกข่าย (slave controller) จะถูกดำเนินงานอยู่บนแต่ละเครื่อง testbed host มีหน้าที่คอยรับคำสั่งจากตัวควบคุมหลัก โดยตัวควบคุมหลักจะเป็นตัวควบคุมทอพอโลยีของระบบที่ทำการทดลอง เมื่อระบบมีการเปลี่ยนแปลงทอพอโลยีเกิดขึ้น ตัวควบคุมหลักจะทำการแพร่สัญญาณเฉพาะกลุ่ม (multicast) ไปบอกตัวควบคุมลูกข่ายนั้น ๆ ผ่านทางช่องสัญญาณควบคุม (control channel) ซึ่งแยกออกจากช่องสัญญาณที่ใช้ในส่วนของระบบทดสอบ เพื่อไม่ให้เกิดการส่งข้อมูลส่วนควบคุมไปรบกวนการทดลอง ซึ่งทำได้โดยใช้แผนวงจรต่อประสานเครือข่ายแยกกัน รูปที่ 2.17 แสดงโครงสร้างภายในของระบบ MobiEmu ซึ่งมีการทำงานดังนี้



รูปที่ 2.17 โครงสร้างของระบบ MobiEmu

2.7.3.1 ตัวควบคุมหลักและส่วนต่อประสานกราฟิกกับผู้ใช้

ตัวควบคุมหลักเป็นส่วนที่สำคัญมากในระบบทดสอบนี้ และที่เครื่องนี้ยังมีส่วนต่อประสานกราฟิกกับผู้ใช้ (graphical user interface, GUI) สำหรับใช้ในการควบคุมการทำงานของระบบทดสอบด้วย โดยตัวควบคุมหลักประกอบด้วยส่วนต่าง ๆ ดังนี้

1. การกำหนดรูปแบบการเคลื่อนที่ (Movement Scenario)

ก่อนการทดลองทุกครั้ง ระบบจะต้องรับข้อมูลจากผู้ใช้ก่อนว่าต้องการใช้รูปแบบการเคลื่อนที่ที่เป็นแบบใดผ่านทาง GUI ซึ่งสามารถทำการโหลดลงไปได้ โดยที่เพิ่มข้อมูลการเคลื่อนที่ (scenario file) นี้จะเป็นรายการของตำแหน่งและการเคลื่อนที่ของทุกโนด พร้อมทั้งเวลานั้น ๆ

2. เวลาในการจำลองและการประสานเวลา (Emulation Clock and Synchronization)

เวลาในการจำลองเป็นเวลาตั้งแต่เริ่มทำการทดลองนับไปเรื่อย ๆ ซึ่งจะคิดที่ตัวควบคุมหลักเท่านั้น โดยที่ตัวควบคุมหลักจะทำการเทียบเวลาในการจำลองนี้กับเวลาที่ถูกระบุไว้ในเพิ่มข้อมูลการเคลื่อนที่ที่ได้โหลดลงไป เมื่อใดก็ตามที่พบว่ามีมีการเปลี่ยนแปลงทอพอโลยีเกิดขึ้น ตัวควบคุมหลักจะทำการแพร่สัญญาณเฉพาะกลุ่มไปยังตัวควบคุมลูกข่าย จากนั้นตัวควบคุมลูกข่ายจึงจะทำการปรับทอพอโลยีใหม่

3. การแสดงภาพการเชื่อมต่อของเครือข่าย (Ad Hoc Network Visualization)

ในระหว่างที่ทำการทดลองอยู่ เพื่อให้ง่ายต่อการตรวจสอบและหาจุดบกพร่องของโปรโตคอลที่พิจารณา ส่วน GUI จะทำการแสดงภาพเครือข่ายในขณะนั้นไปด้วย

2.7.3.2 ตัวควบคุมลูกข่ายและการกรองกลุ่มข้อมูล

ในระบบ MobiEmu ตัวควบคุมลูกข่ายมีหน้าที่บังคับทอพอโลยีให้เป็นตามที่ต้องการ ตัวอย่างเช่น ถ้าทอพอโลยีปัจจุบัน โหนด A อยู่นอกกระยะการส่งของโหนด B โหนด B ก็ทำการบล็อกกลุ่มข้อมูลใด ๆ ที่ส่งมาจากโหนด A ทั้ง ซึ่งสามารถทำได้โดยใช้วิธีการกรองกลุ่มข้อมูล

1. การกรองกลุ่มข้อมูล (Packet Filtering)

การกรองกลุ่มข้อมูลใน MobiEmu นี้ก็ใช้หลักการเดียวกับที่ใช้กันทั่วไปในไฟร์วอลล์ (Firewall) และเนื่องจากในระบบทดสอบนี้ทำงานภายใต้ระบบปฏิบัติการลินุกซ์ จึงได้ใช้ netfilter/iptables ในการกรอง โดยจะตั้งค่ากฎการกรอง (filter rule) ผ่านคำสั่ง iptables ซึ่งกำหนดโดยใช้เลขที่อยู่ MAC (MAC address) เนื่องจากทำได้เร็วกว่าการใช้เลขที่อยู่ไอพี แต่วิธีการกรองโดยใช้เลขที่อยู่ MAC นี้มีข้อจำกัดคือระบบทดสอบจะต้องประกอบด้วย 1 เครือข่ายย่อย (subnet) เท่านั้น เพราะหากมีหลายเครือข่ายย่อย การติดต่อระหว่างเครือข่ายจะต้องผ่านเกตเวย์ ซึ่งจะทำให้เลขที่อยู่ MAC ของโนดต้นทางเปลี่ยนไป แต่หากต้องการจะใช้หลายเครือข่ายย่อยก็สามารถทำได้ โดยเปลี่ยนไปใช้การกรองกลุ่มข้อมูลผ่านทางเลขที่อยู่ไอพีแทน

2. การสื่อสารระหว่างตัวควบคุมหลักและตัวควบคุมลูกข่าย (Master/Slave Communication)

การสื่อสารระหว่างตัวควบคุมหลักและตัวควบคุมลูกข่ายจะผ่านทางช่องสัญญาณควบคุม ซึ่งจะแยกออกมาจากส่วนของเครือข่ายในการทดสอบต่างหาก เพื่อไม่ให้สัญญาณกวนกัน โดยในช่องสัญญาณควบคุม จะทำการติดต่อกันโดยใช้ multicast UDP address และจะมีการรับส่งค่าต่าง ๆ ดังนี้

- *รูปแบบการเคลื่อนที่ (scenario)* ก่อนการทดลองทุกครั้ง ตัวควบคุมหลักต้องทำการแจ้งเครื่องลูกข่ายทุกเครื่องก่อนว่าผู้ใช้เลือกที่จะใช้รูปแบบการเคลื่อนที่แบบใดในการทดลอง
- *Ping* สำหรับตัวควบคุมหลักใช้ในการตรวจสอบสถานะของเครื่องลูกข่ายว่าเครื่องลูกข่ายทุกเครื่องอยู่ในสถานะพร้อมใช้งานหรือไม่
- *Ping reply* สำหรับเครื่องลูกข่ายใช้ในการตอบกลับ ping ของที่ตัวควบคุมหลักส่งไป ซึ่งหากเครื่องลูกข่ายอยู่ในสถานะพร้อมสำหรับการทดลองแล้ว จะทำการแนบ IP และ MAC address ของเครื่องนั้นไปกับ reply message ด้วย
- *เวลาที่ทอพอโลยีเปลี่ยนแปลง (timestamp)* เมื่อทอพอโลยีมีการเปลี่ยนแปลง ตัวควบคุมหลักจะทำการแพร่สัญญาณเฉพาะกลุ่มส่งเวลาในการจำลอง (emulation clock) ไปยังเครื่องลูกข่าย เพื่อใช้ในการตั้งหรือลบกฎการกรองของโหนดนั้น ๆ ใหม่
- *ข้อมูลป้อนกลับ (feedback)* สำหรับเครื่องลูกข่ายใช้ในการส่งข้อมูลกลับไปยังตัวควบคุมหลัก

3. การจัดเส้นทางในเครือข่ายแอดฮอคที่ดีที่สุด (Best Case Ad Hoc Routing)

ในระบบ MobiEmu นี้ได้มีโพรโทคอลจัดเส้นทางรวมอยู่ในระบบด้วย เพื่อเป็นทางเลือกสำหรับผู้ใช้ที่ต้องการจะทำการทดลองในชั้นที่อยู่เหนือชั้นเครือข่ายขึ้นไป โดยเป็นโพรโทคอลจัดเส้นทาง ที่เลือกเส้นทางจากเส้นทางที่สั้นที่สุด (shortest path) ซึ่งจะทำให้การคำนวณจากทุก ๆ เส้นทางที่เป็นไปได้ ดังนั้นเส้นทางที่ได้นี้จะเส้นทางที่ดีที่สุด ไม่มีโพรโทคอลจัดเส้นทางใดที่จะเลือกเส้นทางได้ดีไปกว่าเส้นทางนี้อีกแล้ว ซึ่งในทางปฏิบัติจริงไม่สามารถจะทำการสร้างโพรโทคอลจัดเส้นทางแบบนี้ได้ แต่โพรโทคอลนี้ก็สามารใช้เป็นขอบเขตบน (upper bound) เพื่อเปรียบเทียบกับโพรโทคอลตัวอื่นในการทดลองได้

2.7.4 ข้อสังเกตเกี่ยวกับระบบทดสอบในงานวิจัยที่ผ่านมา

จากทั้ง 3 บทความที่กล่าวมานี้ จะเห็นว่า ในบทความแรกเป็นระบบทดสอบที่มีขนาดใหญ่ สร้างโดยใช้การสื่อสารแบบไร้สาย และโหนดมีการเคลื่อนที่จริง ๆ ซึ่งทำให้สภาพแวดล้อมมีความใกล้เคียงกับระบบจริงมาก แต่ปัญหาที่สำคัญคือ ทำให้ยากต่อการแก้จุดบกพร่อง (debug) ของโพรโทคอลนั้น ๆ เนื่องจากเราไม่สามารถจะควบคุมทุกองค์ประกอบให้เหมือนเดิมในทุกการทดลองได้ เช่น ปรากฏการณ์ต่าง ๆ จากคลื่นวิทยุซึ่งเป็นผลมาจากหลายปัจจัย และการเคลื่อนที่ของโหนดในแต่ละครั้งของการทดลอง ส่วนในบทความที่สองนั้น แม้จะสามารถจำลองสถานการณ์เดิมเพื่อทำการทดลองซ้ำได้ (repeatability) เนื่องจากไม่มีการเคลื่อนที่ของโหนดจริง ๆ และ

ทำการทดลองในห้องปฏิบัติการที่ระยะห่างระหว่างแต่ละเครื่องคอมพิวเตอร์มีค่าคงที่ แต่ในที่นี้ก็ต้องเสียค่าใช้จ่ายในการติดตั้งอุปกรณ์เพิ่มมากขึ้น รวมทั้งการควบคุมรูปแบบการเคลื่อนที่ก็จะต้องมีความซับซ้อนมากหากต้องการให้ได้รูปแบบที่ใกล้เคียงกับในสถานการณ์จริง สำหรับในบทความที่สามเพื่อให้ระบบทดสอบมีคุณสมบัติในการทดลองสถานการณ์เดิมซ้ำได้ และสามารถกำหนดสถานการณ์ ตลอดจนรูปแบบการเคลื่อนที่ที่ใช้ในการทดลองได้ตามต้องการ จึงได้เลือกใช้ระบบทดสอบแบบมีสาย และอาศัยการเลียนแบบสถานการณ์ในเครือข่ายแอตฮอกแทน แต่ก็ยังไม่ใกล้เคียงกับสภาพแวดล้อมจริงเท่าที่ควร เนื่องจากในบทความนี้ไม่ได้คำนึงถึงการจำลองคุณลักษณะของช่องสัญญาณไร้สาย เช่น อัตราเร็วในการส่งกลุ่มข้อมูล หรืออัตราการสูญหายของกลุ่มข้อมูล พิจารณาเพียงการเชื่อมต่อของแต่ละคูโนดเท่านั้น รวมทั้งระบบทดสอบในบทความนี้ยังไม่มีความสามารถในการขยายระบบ (scalability) ได้มากนัก เนื่องจากใช้เครื่องคอมพิวเตอร์หนึ่งเครื่องแทนโนดหนึ่งโนด

จากข้อดีข้อเสียทั้งหมดที่กล่าวมา ในวิทยานิพนธ์นี้จึงเลือกที่จะสร้างระบบทดสอบทั้งสองแบบเพื่อใช้ในการทดลองให้เหมาะสมกับแต่ละสถานการณ์ โดยแบบไร้สายจะเป็นการทดสอบประเมินผลในเครือข่ายแอตฮอกจริง แต่เพื่อให้จัดการได้ง่ายจึงจะทดลองเฉพาะในกรณีที่เครือข่ายมีขนาดเล็ก ส่วนแบบมีสายจะทดสอบในกรณีที่เครือข่ายมีขนาดใหญ่ขึ้น และมีการเปลี่ยนแปลงทอพอโลยี รวมถึงพิจารณาการเคลื่อนที่ของแต่ละโนดด้วย และเพื่อให้มีสภาพแวดล้อมใกล้เคียงความเป็นจริงมากที่สุด ระบบทดสอบนี้จะมีการจำลองคุณลักษณะของช่องสัญญาณไร้สายในส่วนของอัตราเร็วในการส่งกลุ่มข้อมูล และอัตราการสูญหายของกลุ่มข้อมูลด้วย

บทที่ 3

โครงสร้างของระบบทดสอบและกลไกการควบคุมแบบเวลาจริง

ในบทที่ผ่านมาได้กล่าวถึงปัญหาและความสำคัญของการควบคุมจิตเตอร์ในการให้บริการสื่อผสมแบบต่อเนื่อง โดยเฉพาะอย่างยิ่งในเครือข่ายแอตฮอก รวมถึงแนวคิดที่ให้โนดระหว่างทางได้มีส่วนช่วยจัดการควบคุมด้วย และได้กล่าวถึงเหตุผลที่เลือกใช้ระบบทดสอบในการประเมินผลวิธีการที่นำเสนอ ตลอดจนข้อแตกต่างระหว่างระบบทดสอบแบบไร้สายและแบบมีสาย ดังนั้นเนื้อหาในบทนี้จะกล่าวถึงรายละเอียดโครงสร้างของระบบทดสอบพารามิเตอร์ต่าง ๆ ที่เลือกใช้ในระบบทดสอบทั้งแบบไร้สายและแบบมีสาย รวมถึงกลไกการควบคุมแบบเวลาจริงแบบกระจายที่นำเสนอ

3.1 โครงสร้างของระบบทดสอบ

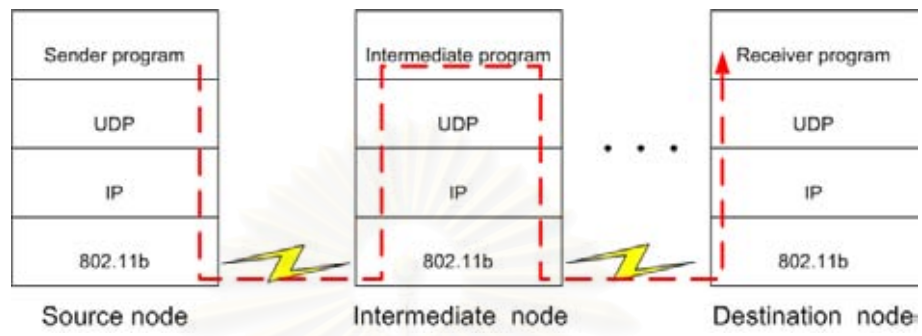
ระบบทดสอบที่พิจารณาในวิทยานิพนธ์นี้แบ่งเป็น 2 ประเภท คือแบบไร้สาย และแบบมีสาย ซึ่งการเลือกใช้งานจะขึ้นอยู่กับความเหมาะสมของรูปแบบสถานการณ์ที่ต้องการทดสอบ โดยอยู่บนพื้นฐานของการทำงานภายใต้ระบบปฏิบัติการลินุกซ์ และแต่ละโนดเชื่อมต่อกันผ่านทางซ็อกเก็ตที่กำหนด โดยโครงสร้างของระบบทดสอบแต่ละประเภทมีรายละเอียดดังนี้

3.1.1 ระบบทดสอบแบบไร้สาย

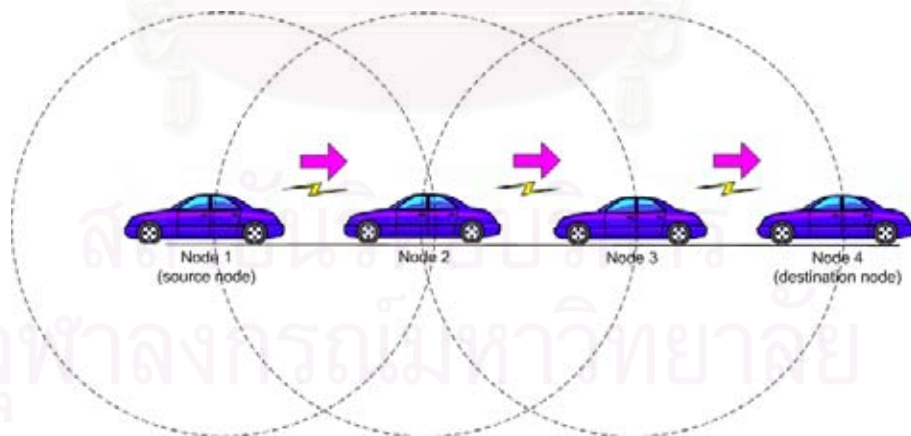
ในระบบทดสอบประเภทนี้จะใช้การเชื่อมต่อผ่านทางเครือข่ายไร้สายในภาวะแอตฮอก (ad hoc mode) ซึ่งปกติแล้วโนดจะสามารถเชื่อมต่อกันได้ภายในหนึ่งช่วงเชื่อมต่อเท่านั้น เนื่องจากอุปกรณ์ยังไม่สามารถรองรับการหาเส้นทางได้ ดังนั้นจึงต้องให้การส่งข้อมูลผ่านทางซ็อกเก็ตที่กำหนดขึ้นเอง เพื่อช่วยในการจำลองทอพอโลยีของเครือข่ายให้สามารถส่งผ่านหลายช่วงเชื่อมต่อได้ ดังรูปที่ 3.1 รูปแบบการทดลองในระบบทดสอบแบบไร้สายนี้แสดงดังรูปที่ 3.2 โดยประกอบไปด้วยโนด 4 โหนด ซึ่งในที่นี้คือโนดบุ๊คคอมพิวเตอร์ที่อยู่ในรถแต่ละคัน และรถทั้ง 4 คันนี้จะแล่นตามกันไปเรื่อย ๆ ในขณะที่มีการส่งข้อมูลจากโนด 1 ไปยังโนด 4 โดยกำหนดให้แต่ละโนดจะมีโนดข้างเคียงคือโนดที่อยู่ติดกันเท่านั้น ซึ่งหมายถึงข้อมูลจะถูกส่งจากโนด 1 ผ่านไปยังโนด 2 ก่อนแล้วจึงค่อยส่งต่อไปยังโนด 3 และโนด 4 ตามลำดับ

3.1.2 ระบบทดสอบแบบมีสาย

เพื่อให้ระบบทดสอบมีความสามารถในการจำลองสถานการณ์เดิมสำหรับทำการทดลองซ้ำได้ และยังคงสภาพแวดล้อมให้เหมือนกับระบบจริงมากที่สุดอยู่ ในระบบทดสอบนี้จึงทำการเลียนแบบคุณลักษณะต่าง ๆ ของช่องสัญญาณไร้สายมาไว้ในเครือข่ายแบบมีสาย ประกอบไปด้วยภาวะเชื่อมต่อ (connectivity) ซึ่งขึ้นอยู่กับระยะการส่งของแต่ละคูโนด ความเร็วในการรับส่งกลุ่มข้อมูลระหว่างคูโนด (link capacity) รวมไปถึงอัตราการสูญหาย

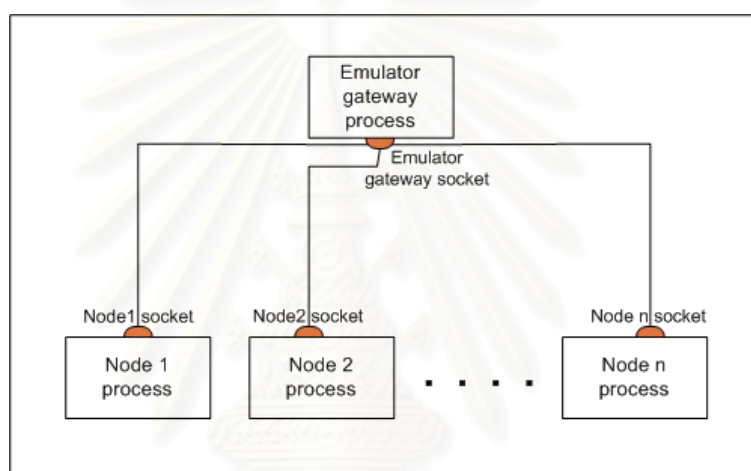


รูปที่ 3.1 โครงสร้างของการส่งข้อมูลที่ใช้ในระบบทดสอบ



รูปที่ 3.2 รูปแบบการทดลองของระบบทดสอบแบบไร้สาย

ของกลุ่มข้อมูล โดยทั้งหมดนี้จะทำการควบคุมผ่านทางโปรแกรมหนึ่งที่กำหนดให้เป็นศูนย์กลางเรียกว่า เกตเวย์เลียนแบบ (emulator gateway) รูปที่ 3.3 แสดงโครงสร้างของระบบทดสอบนี้ จะเห็นว่าทุกโพรเซส (process) ที่ใช้ไม่ว่าจะเป็นของแต่ละโนด หรือของเกตเวย์เลียนแบบจะมีซ็อกเก็ตที่ใช้เชื่อมต่ออยู่ โดยค่าที่เป็นตัวระบุซ็อกเก็ตนี้ก็คือเลขที่อยู่ไอพี (IP address) และหมายเลขช่องทาง (port number) ที่ใช้ ซึ่งในการใช้โพรเซสแทนแต่ละโนดนั้น มีข้อดีคือไม่จำเป็นต้องใช้เครื่องคอมพิวเตอร์หนึ่งเครื่องต่อหนึ่งโนดในการทดลอง ทำให้ถึงแม้จะใช้คอมพิวเตอร์เพียงเครื่องเดียวก็สามารถทดลองในเครือข่ายที่มีขนาดใหญ่ได้ นอกจากนี้โปรแกรมของแต่ละโนดที่ใช้ในระบบทดสอบแบบมีสายนี้ยังสามารถนำไปใช้ในระบบทดสอบแบบไร้สายได้ทันที เพียงแต่ต้องเปลี่ยนเลขที่อยู่ไอพีและหมายเลขช่องทางให้เป็นการเชื่อมต่อโดยตรงแทนที่จะต้องผ่านเกตเวย์เลียนแบบ



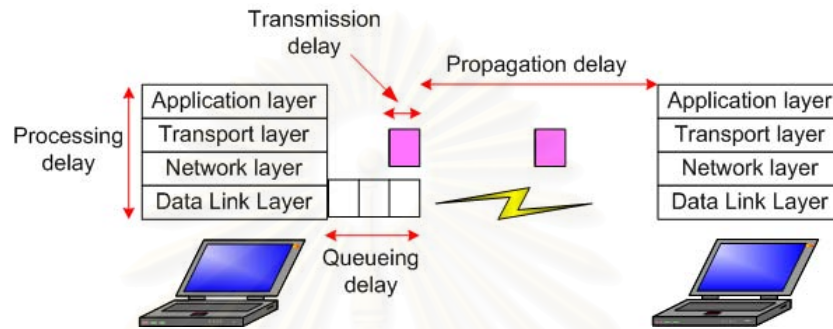
รูปที่ 3.3 โครงสร้างของระบบทดสอบแบบมีสาย

3.1.2.1 หลักการทำงานของเกตเวย์เลียนแบบ

เกตเวย์เลียนแบบจะทำหน้าที่เป็นตัวกลางระหว่างแต่ละโนดที่ต้องการจะส่งข้อมูลถึงกัน โดยทุกกลุ่มข้อมูลที่ต้องการส่งจะต้องผ่านเกตเวย์เลียนแบบนี้เสมอ แต่ละโนดจะไม่สามารถส่งกลุ่มข้อมูลใด ๆ ถึงกันได้โดยตรง ทั้งนี้เพื่อให้สามารถกำหนดทอพอโลยีของเครือข่าย จำลองความเร็วในการรับส่งกลุ่มข้อมูลระหว่างคู่โนด และกำหนดการสูญหายของกลุ่มข้อมูลได้ โดยทอพอโลยีกำหนดได้จากการระบุโนดข้างเคียงของแต่ละโนดที่เวลาต่าง ๆ ซึ่งเกตเวย์เลียนแบบจะส่งต่อกลุ่มข้อมูลออกไปให้เฉพาะโนดที่เป็นโนดข้างเคียงเท่านั้น ส่วนความเร็วในการรับส่งกลุ่มข้อมูลของแต่ละคู่โนด จะทำได้โดยหนดด้วยค่าที่เหมาะสมก่อนที่จะส่งข้อมูลต่อไปยังโนดปลายทาง และการสูญหายของกลุ่มข้อมูลจะถูกกำหนดโดยการสุ่มจากความน่าจะเป็นของการสูญเสียกลุ่มข้อมูลต่อหนึ่งช่วงเชื่อมต่อที่ระบุไว้ ซึ่งข้อมูลเหล่านี้สามารถเก็บบันทึกเป็นแฟ้มลงบันทึกเข้าออก (log file) เพื่อนำมาวิเคราะห์ภายหลังได้

3.1.2.2 ความเร็วของการรับส่งข้อมูลที่นำมาใช้ในระบบทดสอบ

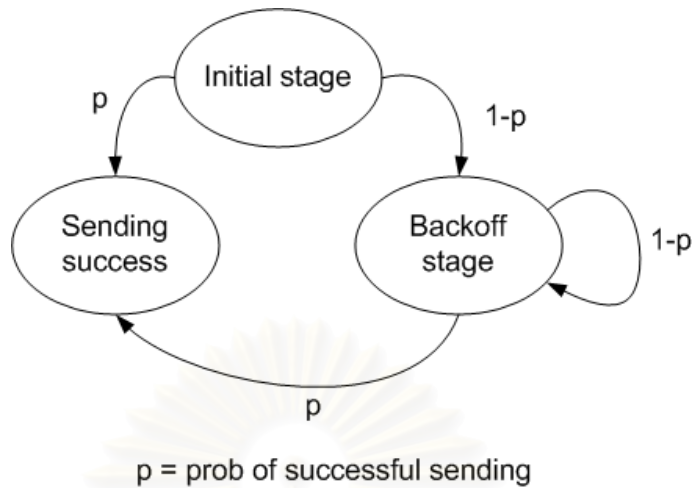
เกตเวย์เลียนแบบในระบบทดสอบนี้จะเลียนแบบความเร็วในการรับส่งของเครือข่ายไร้สาย โดยการหน่วงเวลาเพิ่มให้เท่ากับค่าประวิงเวลาของช่องสัญญาณ ไร้สายก่อนที่ส่งกลุ่มข้อมูลต่อออกไป ซึ่งค่าประวิงเวลาของช่องสัญญาณไร้สายนี้ก็ขึ้นอยู่กับปัจจัยต่าง ๆ มากมาย โดยจากรูปที่ 3.4 จะเห็นว่าสามารถแบ่งได้เป็น 4 ประเภทหลัก ๆ ดังนี้



รูปที่ 3.4 ลักษณะค่าประวิงเวลาของการส่งกลุ่มข้อมูลผ่านเครือข่าย

1. เวลาประวิงการประมวลผล (processing delay) เป็นเวลาที่แต่ละโหนดใช้ในการประมวลผล ซึ่งขึ้นอยู่กับความสามารถของหน่วยประมวลผลกลาง (Central Processing Unit) ของโหนดนั้น ๆ ด้วย
2. เวลาประวิงการแพร่กระจาย (propagation delay) เป็นเวลาที่สัญญาณแพร่กระจายจากตัวส่งไปถึงตัวรับ ซึ่งจะขึ้นอยู่กับระยะห่างระหว่างคูโหนดนั้น
3. เวลาประวิงการส่ง (transmission delay) เป็นเวลาที่ใช้ในการส่งกลุ่มข้อมูลหนึ่ง ๆ โดยขึ้นอยู่กับขนาดของกลุ่มข้อมูล และแบนด์วิธของช่องสัญญาณที่ใช้ส่ง ยิ่งกลุ่มข้อมูลมีขนาดใหญ่หรือแบนด์วิธมีค่าต่ำจะทำให้เวลาประวิงในการส่งมีค่ามาก
4. เวลาประวิงการรอในคิว (queueing delay) เป็นเวลาที่กลุ่มข้อมูลคอยอยู่ในบัฟเฟอร์ก่อนที่จะถูกส่งออกไป ซึ่งจะขึ้นอยู่กับปริมาณทราฟฟิกในขณะนั้น หากมีปริมาณมากเวลาดังกล่าวส่วนนี้ก็จะยิ่งมากด้วย

ซึ่งค่าประวิงเวลาที่พิจารณาในระบบทดสอบนี้จะแบ่งทั้ง 4 ประเภทดังกล่าวออกเป็นสองส่วนหลัก คือ เวลาประวิงการประมวลผล เวลาประวิงการแพร่กระจาย และเวลาประวิงการส่งคิดรวมเป็นส่วนเดียวกัน ซึ่งจะอ้างอิงจากค่าจริงที่วัดได้จากการทดลอง ส่วนเวลาประวิงการรอในคิวจะคิดแยกกันโดยหากค่า back-off ตามมาตรฐาน IEEE 802.11b ดังที่ได้กล่าวไปแล้วในบทที่ 2 ในรูปที่ 3.5 แสดงสเตจที่เกตเวย์เลียนแบบใช้ในการจำลองการส่งกลุ่มข้อมูล โดยค่า p คือความน่าจะเป็นที่แต่ละกลุ่มข้อมูลจะส่งสำเร็จ ถ้า p มีค่ามากจะหมายถึงมีปริมาณทราฟฟิกในเครือข่ายน้อย ทำให้มีโอกาสที่จะส่งสำเร็จมาก กลุ่มข้อมูลจึงไม่ต้องรอในคิวนาน แต่หาก p มีค่าน้อย จะหมายถึงมีปริมาณทราฟฟิกหนาแน่นมากในเครือข่าย ทำให้มีโอกาสที่ส่งแล้วจะเกิดการชนกันมาก จึงต้องรอนานขึ้นกว่าจะส่งแต่ละกลุ่มข้อมูลสำเร็จ ดังนั้นค่าประวิงเวลาที่ใช้ในระบบทดสอบจะคิดจากผลรวมของค่าประวิงเวลาในแต่ละสเตจ

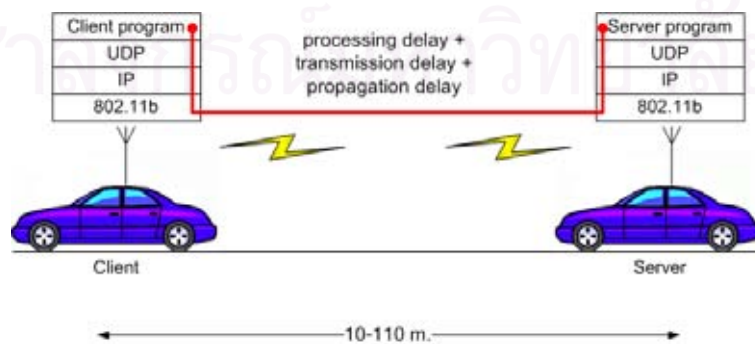


รูปที่ 3.5 แสดงที่เกตเวย์เลียนแบบใช้ในการจำลองการส่งกลุ่มข้อมูล

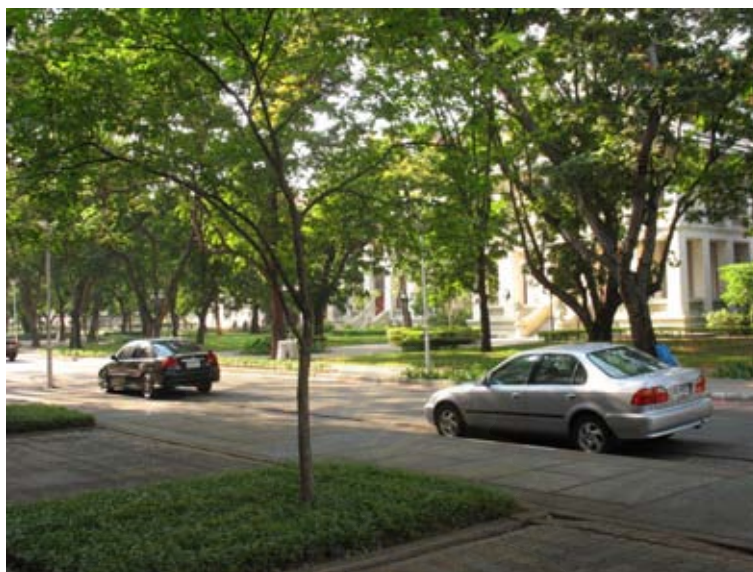
- รูปแบบการทดลองที่ใช้วัดเวลาประวิงการประมวลผล เวลาประวิงการแพร่กระจาย และเวลาประวิงการส่ง

เป็นการทดลองส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์ 2 เครื่องที่ใช้แผ่นวงจรประสานเครือข่ายไร้สาย Intelpro 2100 เชื่อมต่อกันผ่านทางภาวะแอดฮอกตามมาตรฐาน IEEE 802.11b รูปแบบการทดลองจะแสดงดังรูปที่ 3.6 โดยจะใช้โน้ตบุ๊กคอมพิวเตอร์ 2 เครื่องวางอยู่ในรถแต่ละคัน และให้เครื่องหนึ่งทำหน้าที่เป็นลูกข่ายส่งกลุ่มข้อมูลประเภท UDP ไปที่เครื่องแม่ข่ายที่อยู่ในรถอีกคัน เมื่อเครื่องแม่ข่ายได้รับกลุ่มข้อมูลนั้นแล้วจะทำการส่งกลับไปที่เครื่องลูกข่ายทันที ซึ่งในการทดลองจะปรับเปลี่ยนระยะห่างระหว่างรถทั้งสองคันไปทุก ๆ 10 เมตร เริ่มตั้งแต่ 10 เมตรจนถึง 110 เมตร และใช้กลุ่มข้อมูลทั้งหมด 3 ขนาด คือ 512, 1024, 2048 ไบต์ โดยที่ระยะห่างแต่ละตำแหน่งจะส่งกลุ่มข้อมูลขนาดละ 1000 กลุ่มข้อมูลไปด้วยช่วงห่างระหว่างแต่ละกลุ่มข้อมูลเท่ากับ 500 มิลลิวินาที และทำการวัดค่าประวิงเวลาไปกลับ (round-trip time) ของแต่ละกลุ่มข้อมูล รวมถึงอัตราการสูญหายของกลุ่มข้อมูลที่เกิดขึ้นมาวิเคราะห์เพื่อใช้ในระบบทดสอบ ส่วนรูปที่ 3.7 แสดงภาพขณะทำการทดลอง

โดยผลของค่าประวิงเวลาไปกลับที่วัดได้ เมื่อนำมาหารสองเพื่อประมาณเป็นค่าประวิงเวลาสำหรับการส่งทางเดียวจะได้ดังรูปที่ 3.8 และเมื่อนำมาคิดเป็นความเร็วในการส่งโดยคำนวณจากขนาดของกลุ่มข้อมูล จะได้ความเร็ว



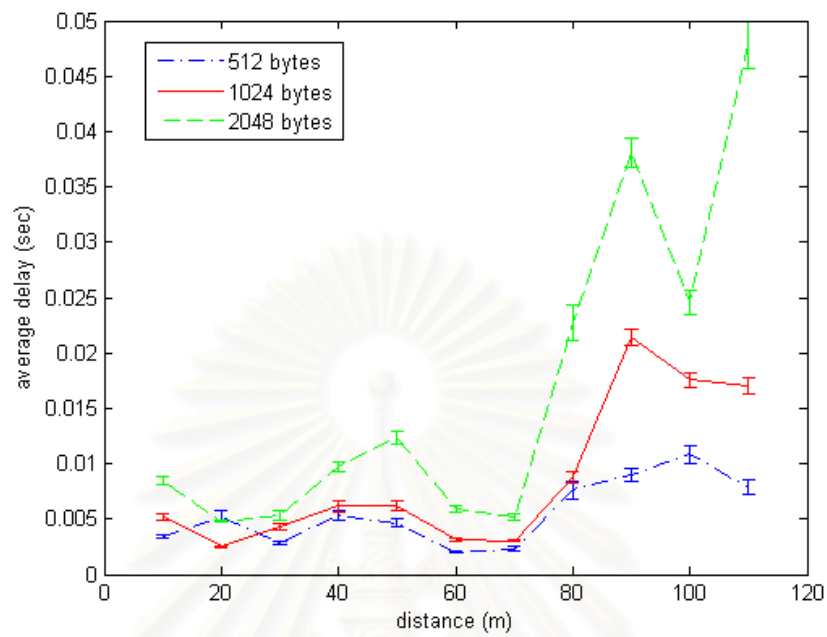
รูปที่ 3.6 รูปแบบการทดลองวัดค่าเวลาประวิงการประมวลผล การแพร่กระจาย และการส่ง



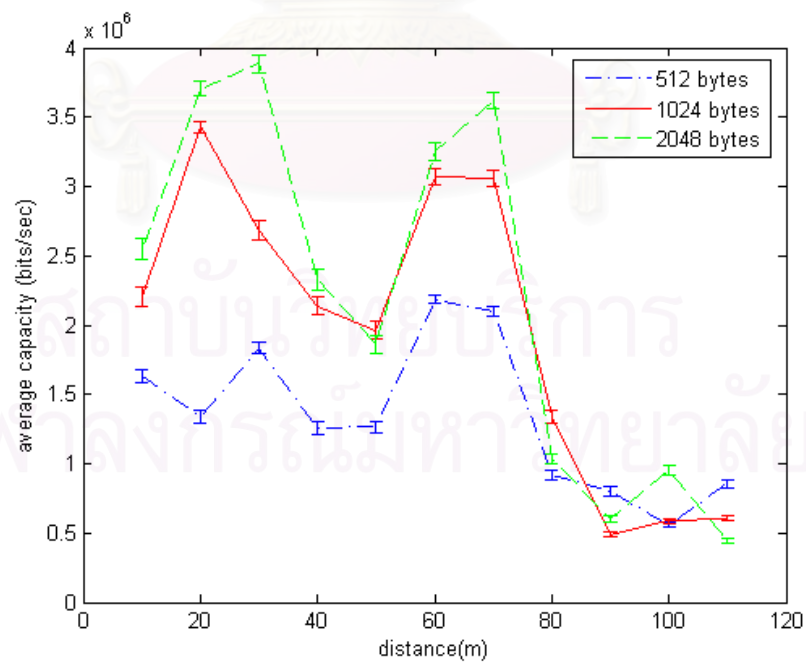
รูปที่ 3.7 ภาพขณะทดลองวัดค่าเวลาประวิงการประมวลผล การแพร่กระจาย และการส่ง

เฉลี่ยในการส่งข้อมูลดังรูปที่ 3.9 และความเร็วสูงสุดดังรูปที่ 3.10 ส่วนในรูปที่ 3.11 แสดงอัตราการสูญเสียกลุ่มข้อมูลที่วัดได้ โดยในรูปที่ 3.9 จะเห็นว่าที่ระยะห่าง 40-50 เมตรความเร็วเฉลี่ยของการส่งกลุ่มข้อมูลทั้ง 3 ขนาดมีค่าลดลงมาและเพิ่มกลับขึ้นไปใหม่ที่ระยะ 60 เมตร ที่เป็นเช่นนี้คาดว่าเนื่องมาจากที่ระยะ 40-50 เมตรมีสัญญาณรบกวนบางอย่างอยู่ เช่น สัญญาณจากแอคเซสพอยต์ที่ติดตั้งอยู่บริเวณนั้น ส่งผลให้ความเร็วเฉลี่ยที่ต่ำลง ส่วนในรูปที่ 3.10 จะเห็นว่าความเร็วสูงสุดที่ได้ของกลุ่มข้อมูลแต่ละขนาดมีค่าต่างกันมาก นั้นหมายถึงว่าค่าประวิงเวลาไม่ได้เพิ่มขึ้นแบบเป็นเชิงเส้นตามขนาดของกลุ่มข้อมูล ซึ่งเมื่อมาพิจารณาที่แต่ละประเภทของค่าประวิงเวลาแล้ว พบว่าเวลาประวิงการแพร่กระจายถือว่าน้อยมากเมื่อเทียบกับเวลาประวิงชนิดอื่น ๆ จึงสามารถตัดทิ้งได้ ส่วนเวลาประวิงการส่งก็เพิ่มขึ้นตามขนาดของกลุ่มข้อมูลแบบเชิงเส้นอยู่แล้ว ดังนั้นจึงเหลือเพียงเวลาประวิงการประมวลผลเท่านั้นที่จะเป็นส่วนสำคัญที่ทำให้ค่าประวิงเวลารวมเพิ่มขึ้นแบบไม่เป็นเชิงเส้นได้ หรือกล่าวอีกนัยหนึ่งว่าเวลาประวิงการประมวลผลเป็นค่าที่มีความสำคัญต่อค่าประวิงเวลารวมที่เกิดขึ้น ไม่สามารถตัดทิ้งได้ ซึ่งโดยปกติแล้วค่าประวิงเวลาประมวลผลจะมีค่าคงที่ในแต่ละขนาดของกลุ่มข้อมูล

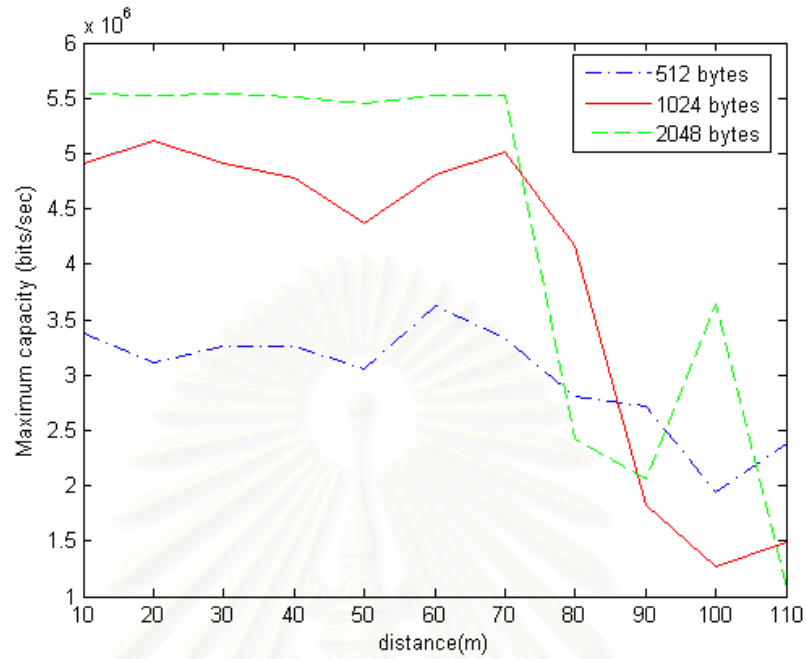
เมื่อนำรูปที่ 3.8-3.11 มาวิเคราะห์โดยเทียบกับมาตรฐานในตารางที่ 2.1 ที่กำหนดไว้ว่าความเร็วสูงสุดจะแบ่งออกเป็นช่วง ๆ ตามความแรงของสัญญาณที่ได้รับ จะเห็นว่าความเร็วสูงสุดที่ได้จากการทดลองก็แบ่งออกได้เป็น 2 ช่วง โดยช่วงแรกอยู่ในระยะทางตั้งแต่ 0-70 เมตร และช่วงที่สองตั้งแต่ 70-110 เมตร แต่ค่าที่ได้เมื่อเทียบกับมาตรฐานแล้วถือว่าต่ำกว่ามาก ทั้งนี้เนื่องจากค่าตามมาตรฐานคิดความเร็วการรับส่งข้อมูลที่สามารถทำได้สูงสุดในชั้นกายภาพเท่านั้น แต่ในความเป็นจริงแล้วกลุ่มข้อมูลจะต้องถูกประมวลผลและส่งผ่านชั้นอื่น ๆ ด้วย ซึ่งในการทดลองนี้คำนวณค่าประวิงเวลานับตั้งแต่ส่งจากชั้นการประยุกต์ (application layer) ที่ต้นทางจนถึงชั้นการประยุกต์ที่ปลายทาง และจากรูปที่ 3.11 จะเห็นว่าระยะตั้งแต่ 100 เมตรขึ้นไป อัตราการสูญเสียกลุ่มข้อมูลมีค่ามากจนอาจส่งผลกระทบต่อคุณภาพของข้อมูลสื่อผสมแย่เกินกว่าจะยอมรับได้ ดังนั้นในวิทยานิพนธ์นี้จึงไม่เพิ่มระยะ



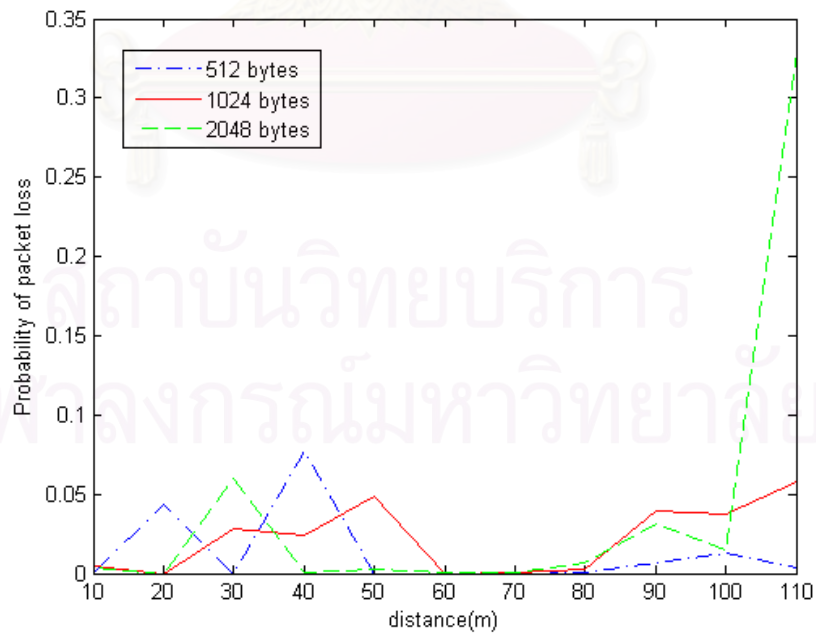
รูปที่ 3.8 ค่าประวิงเวลาเฉลี่ยของการส่งกลุ่มข้อมูลในแต่ละขนาดระหว่างคูโนดที่ระยะห่างต่าง ๆ



รูปที่ 3.9 ความเร็วเฉลี่ยของการส่งกลุ่มข้อมูลในแต่ละขนาดระหว่างคูโนดที่ระยะห่างต่าง ๆ



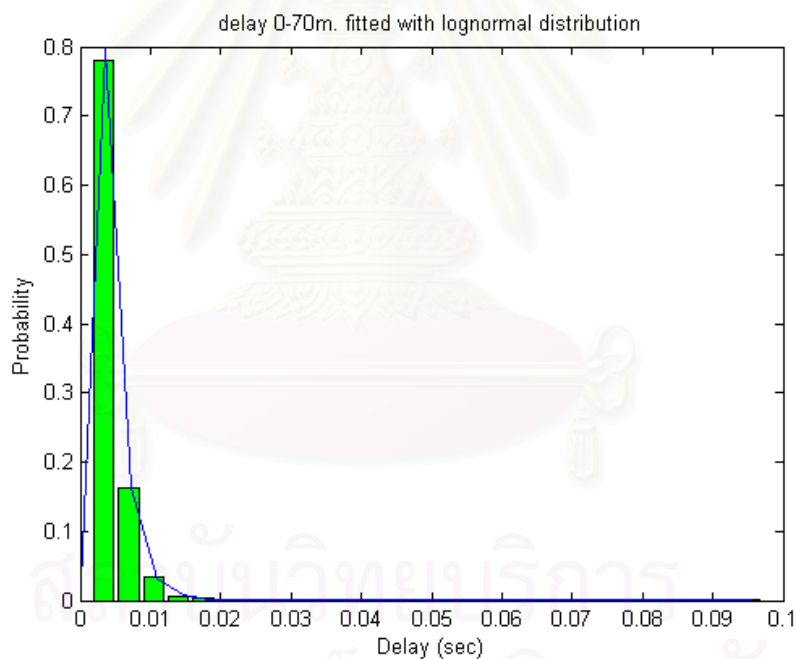
รูปที่ 3.10 ความเร็วสูงสุดของการส่งกลุ่มข้อมูลในแต่ละขนาดระหว่างคูโนดที่ระยะห่างต่าง ๆ



รูปที่ 3.11 อัตราการสูญเสียของกลุ่มข้อมูลในแต่ละขนาดระหว่างคูโนดที่ระยะห่างต่าง ๆ

ห่างในการทดลองอีก โดยเลือกที่จะส่งในระยะสูงสุดเพียง 100 เมตรเท่านั้น

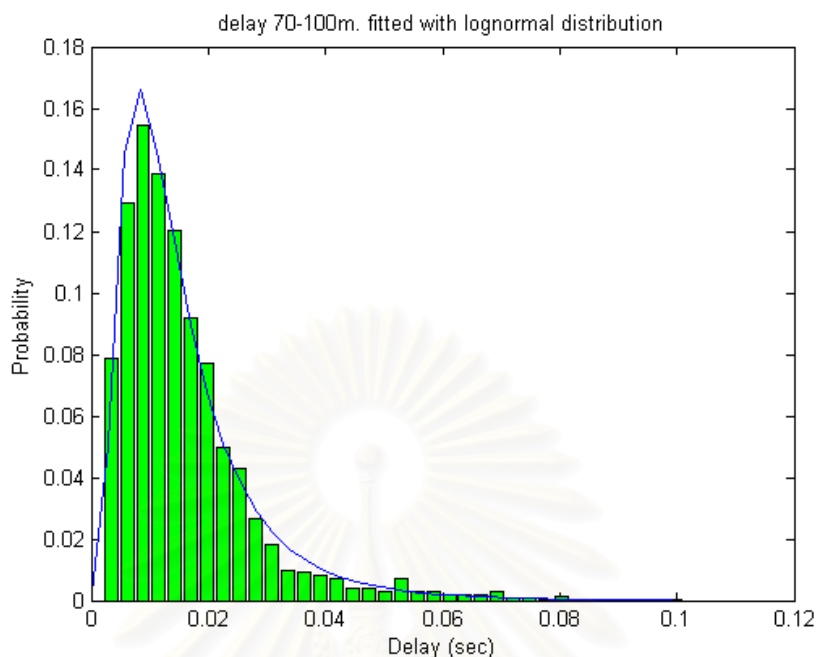
จากค่าประวิงเวลาที่วัดได้ในแต่ละช่วง เมื่อนำอันตรภาคชั้น (histogram) มาวิเคราะห์หาการแจกแจง (distribution) แล้ว พบว่าเมื่อลองเทียบกับการแจกแจงหลาย ๆ แบบ ค่าประวิงเวลาที่ได้มีการกระจายตัวใกล้เคียงกับลักษณะการแจกแจงล็อกปรกติ (lognormal distribution) มากที่สุด จากรูปที่ 3.10 จะเห็นว่าที่ระยะ 0-70 เมตร จะมีความเร็วสูงสุดของการส่งกลุ่มข้อมูลอยู่ในระดับหนึ่ง และจะลดลงมาเป็นอีกระดับหนึ่งในระยะไกลออกไป เนื่องมาจากมีการปรับลักษณะมอดูเลชันให้เหมาะสมกับความแรงของสัญญาณที่ได้รับ ดังนั้นรูปแบบของค่าประวิงเวลาทั้งสองกลุ่มนี้ก็ต่างกันด้วย จึงต้องแยกการแจกแจงของค่าประวิงเวลาที่ใช้ในระบบทดสอบออกเป็นสองกลุ่ม ในรูปที่ 3.12 และ 3.13 แสดงลักษณะของการแจกแจงของกลุ่มข้อมูลขนาด 1024 ไบต์ในช่วง 0-70 เมตร และ 70-100 เมตรตามลำดับ โดยในช่วง 0-70 เมตร จะมี $\mu = -5.66005$ และ $\sigma = 0.556374$ ซึ่งมีค่าคลาดเคลื่อนมาตรฐาน (standard error) เท่ากับ 6.7×10^{-3} และ 4.74×10^{-3} ตามลำดับ ส่วนในช่วง 70-100 เมตร จะมี $\mu = -4.37188$ และ $\sigma = 0.671991$ โดยที่ค่าคลาดเคลื่อนมาตรฐานเท่ากับ 12.4×10^{-3} และ 8.79×10^{-3} ตามลำดับ



รูปที่ 3.12 อันตรภาคชั้นของค่าประวิงเวลาในช่วง 0-70 เมตรเทียบกับการแจกแจงล็อกปรกติ

3.1.2.3 การเคลื่อนที่ของโหนดและกระบวนการเลือกเส้นทาง

ในระบบทดสอบนี้จะใช้แบบจำลองการเคลื่อนที่ตามกันของรถดั่งที่ได้กล่าวไปแล้วในบทที่ 2 ซึ่งทุกโหนดจะมีการเคลื่อนที่ไปในทิศทางเดียวกัน ส่วนในการตัดสินใจเลือกเส้นทางจะถือว่าทุกโหนดทราบเส้นทางไปยังโหนดอื่น ๆ ในเครือข่าย และไม่ต้องมีกระบวนการร้องขอเส้นทางเมื่อทอพอโลยีมีการเปลี่ยนแปลง แต่สามารถสลับไปใช้เส้นทาง



รูปที่ 3.13 อันตรภาคชั้นของค่าประวิงเวลาในช่วง 70-100 เมตรเทียบกับการแจกแจงล็อกปรกติ

ใหม่ได้ทันที ซึ่งโพรโทคอลจัดเส้นทางส่วนใหญ่มักจะเลือกเส้นทางที่มีค่าประวิงเวลาน้อยที่สุดเป็นอันดับแรก ดังนั้นในระบบทดสอบนี้จะเลือกเส้นทางจากเส้นทางที่ใช้ช่วงเชื่อมต่อที่น้อยที่สุด โดยมีสมมติฐานว่าเป็นเส้นทางที่มีค่าประวิงเวลาน้อยที่สุด

3.2 กลไกการควบคุมแบบเวลาจริงแบบกระจาย (Distributed Real-time Control Mechanism)

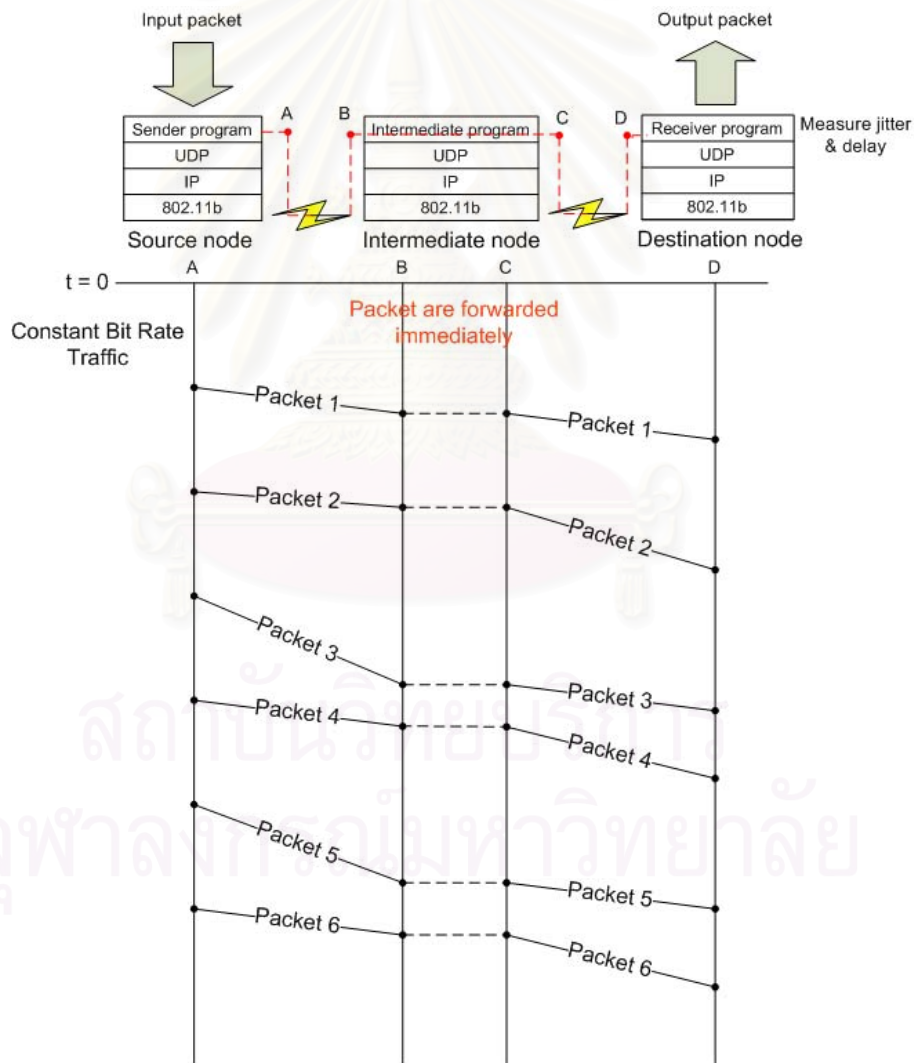
วิทยานิพนธ์นี้ได้นำเสนอกลไกควบคุมแบบเวลาจริงแบบกระจาย 3 วิธี โดยทั้งหมดเป็นการควบคุมที่โนดระหว่างทางเพื่อช่วยลดจัตเตอร์ที่โนดปลายทาง ซึ่งในวิธีแรกจะใช้ได้เฉพาะกับทราฟฟิกที่มีอัตราบิตคงที่เท่านั้น ส่วนสองวิธีหลังจะใช้ได้กับทั้งทราฟฟิกที่มีอัตราบิตคงที่ และอัตราบิตแปรผันได้ โดยแต่ละวิธีมีหลักการดังนี้

3.2.1 การควบคุมแบบเวลาจริงแบบกระจายโดยการควบคุมขีดเริ่มเปลี่ยนของบัฟเฟอร์เพื่อการปรับให้เป็นปกติ (Distributed Real-time Control with Regulating Buffer Threshold : DRC-RBT)

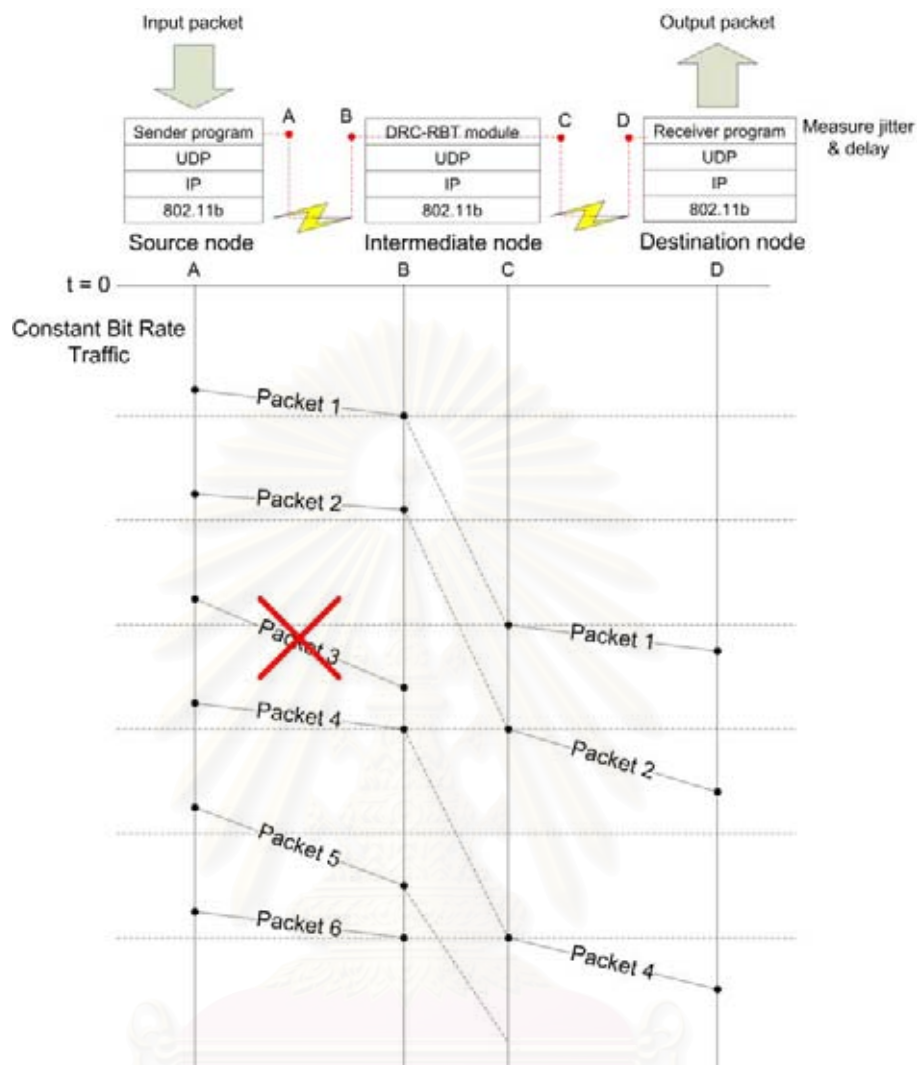
วิธีนี้อาศัยหลักการควบคุมจัตเตอร์ที่โนดระหว่างทางโดยการเก็บกลุ่มข้อมูลเข้ามาไว้ในบัฟเฟอร์ก่อน และทำการจับเวลาว่าควรส่งกลุ่มข้อมูลแรกที่ได้รับเข้ามาออกไปที่เวลาไหน ซึ่งเวลานี้ก็ขึ้นอยู่กับค่าขีดเริ่มเปลี่ยน (threshold) ที่กำหนดไว้ และเมื่อถึงเวลาดังกล่าวมอดูลควบคุมการรับส่งก็จะส่งกลุ่มข้อมูลนั้นออกไป พร้อมทั้งเริ่มทยอยส่งต่อกลุ่มข้อมูลอื่น ๆ ที่ตามมาออกไปด้วยช่วงเวลาที่เหมาะสม โดยขึ้นอยู่กับลำดับของกลุ่มข้อมูลด้วย เช่น หากมีการ

สูญหายของกลุ่มข้อมูลเกิดขึ้นในช่วงใด ช่วงเวลาที่โหนดระหว่างทางจะส่งกลุ่มข้อมูลต่อก็จะชดเชยเวลานั้นไปด้วยโดยคำนวณจากระยะห่างระหว่างกลุ่มข้อมูล (time interval) ดังนั้นในวิธี DRC-RBT นี้จะใช้ได้ในกรณีที่ข้อมูลถูกส่งมาด้วยอัตราบิตคงที่เท่านั้น

เพื่อให้เข้าใจได้ชัดเจนขึ้น ในรูปที่ 3.14 และ 3.15 แสดงตัวอย่างแผนภาพเวลา (timing diagram) ของการส่งแบบปกติที่ไม่มีการควบคุมใด ๆ และการส่งที่มีการควบคุมตามวิธี DRC-RBT ตามลำดับ โดยในตัวอย่างนี้กำหนดให้ซัดเริ่มเปลี่ยนเท่ากับ 2 จะเห็นว่าเมื่อโหนดระหว่างทางได้รับกลุ่มข้อมูลที่หนึ่งแล้ว จะรอจนกระทั่งถัดไปอีกสองช่วงเวลา (time slot) จึงจะส่งกลุ่มข้อมูลที่หนึ่งต่อไป หลังจากนั้นก็ค่อยส่งกลุ่มข้อมูลที่ตามมาต่อไปด้วยระยะเวลาห่างเท่า ๆ กัน แต่เมื่อกลุ่มข้อมูลที่สามสูญหายไป ก็จะชดเชยเวลาในช่วงนั้นก่อนที่จะส่งกลุ่มข้อมูลที่สี่ ทั้งนี้ก็เพื่อให้อัตราของกลุ่มข้อมูลที่ถูกส่งออกไปใกล้เคียงกับที่โหนดต้นทางมากที่สุด



รูปที่ 3.14 แผนภาพเวลาของการส่งที่ไม่มีการควบคุมใด ๆ



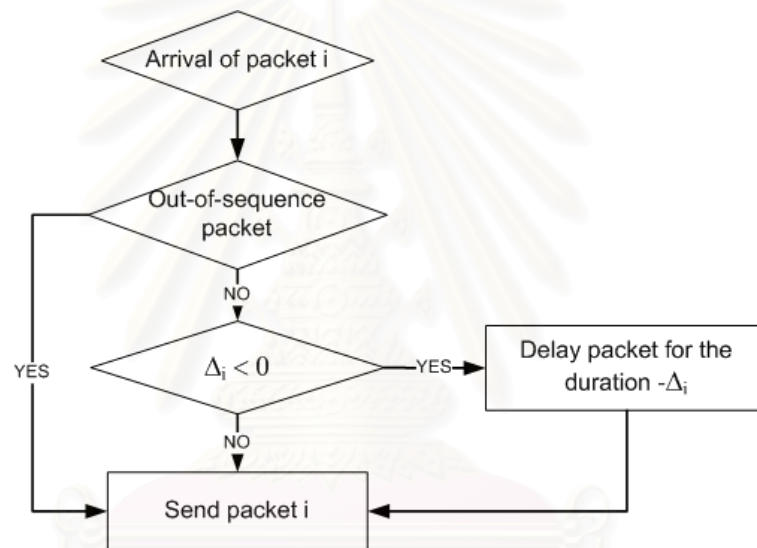
รูปที่ 3.15 แผนภาพเวลาของการส่งที่ควบคุมโดย DRC-RBT

3.2.2 การควบคุมแบบเวลาจริงแบบกระจายโดยใช้ป้ายระบุค่าประวิงเวลา (Distributed Real-time Control with Delay Time Tag : DRC-DTT)

ในวิธีนี้จะเป็นการปรับปรุงวิธี DRC-RBT โดยหลีกเลี่ยงค่าประวิงเวลาที่เกิดจากการเก็บกลุ่มข้อมูลไว้ในบัฟเฟอร์ก่อนที่จะส่งต่อไป ดังนั้นการทำงานจะคล้ายกับวิธี DRC-RBT มาก ต่างกันเพียงแค่กระบวนการตัดสินใจในการส่งต่อข้อมูลเท่านั้น โดยแทนที่จะทำการเก็บกลุ่มข้อมูลไว้ในบัฟเฟอร์ก่อน วิธีนี้จะทำการดูว่าช่วงเวลาระหว่างการส่งกลุ่มข้อมูล (sending interval) ที่โน้ตต้นทางซึ่งแนบมากับแต่ละกลุ่มข้อมูลเป็นเท่าไร จากนั้นจึงทำการตรวจสอบว่ากลุ่มข้อมูลนั้นมาถึงเร็วหรือช้าเกินไปเมื่อเทียบกับเวลาดังกล่าว โดยใช้เวลาที่ส่งกลุ่มข้อมูลก่อนหน้าออกไปเป็นเวลาอ้างอิง หากพบว่ามาช้าเกินไปก็จะทำการส่งต่อทันที แต่หากพบว่ามาเร็วไป ถ้าส่งต่อไปทันทีจะทำให้ค่าจิตเตอร์มีค่ามาก ก็ทำการหน่วงเวลาจนกระทั่งถึงเวลาที่เมื่อส่งออกไปแล้วจะทำให้จิตเตอร์ของกลุ่มข้อมูลนั้นเป็นศูนย์แล้วจึงค่อยส่งออกไป เพื่อทำให้ช่วงเวลาระหว่างการส่งกลุ่มข้อมูลของแต่ละโหนดระหว่างทางใกล้เคียงกับที่

โนดต้นทางมากที่สุด ซึ่งขั้นตอนการทำงานนี้แสดงดังรูปที่ 3.16 แต่สำหรับกรณีที่มีการสูญหายของกลุ่มข้อมูลเกิดขึ้น โนดระหว่างทางจะส่งกลุ่มข้อมูลถัดไปทันที โดยไม่มีการประมวลผลใด ๆ เนื่องจากไม่สามารถไปเทียบกับกลุ่มข้อมูลที่มีลำดับก่อนหน้าได้

ให้ R_i และ S_i เป็นเวลาที่โนดระหว่างทางได้รับและเวลาที่ส่งกลุ่มข้อมูล i ตามลำดับ ส่วน S_i^* เป็นเวลาที่โนดต้นทางทำการส่งกลุ่มข้อมูล i ออกไป ส่วนในรูปที่ 3.17 แสดงแผนภาพเวลาของการส่งที่ใช้การควบคุมวิธี DRC-DTT ซึ่งกลุ่มข้อมูลถูกส่งมาด้วยอัตราคงที่ตามตัวอย่างในรูปที่ 3.14 โดยจากรูปจะเห็นว่าจิตเตอร์เฉลี่ยที่โนดปลายทางมีค่าน้อยลง ขณะที่ค่าประวิงเวลาของแต่ละกลุ่มข้อมูลมีแนวโน้มเพิ่มขึ้นเรื่อย ๆ เนื่องจากค่าประวิงเวลาที่มีค่าสูงของกลุ่มข้อมูลหนึ่งจะมีผลถึงค่าประวิงเวลาของกลุ่มข้อมูลอื่น ๆ ที่ตามมาด้วย แต่หากมีการสูญหายของกลุ่มข้อมูลเกิดขึ้น ค่าประวิงเวลาก็จะมีค่าลดต่ำลงได้

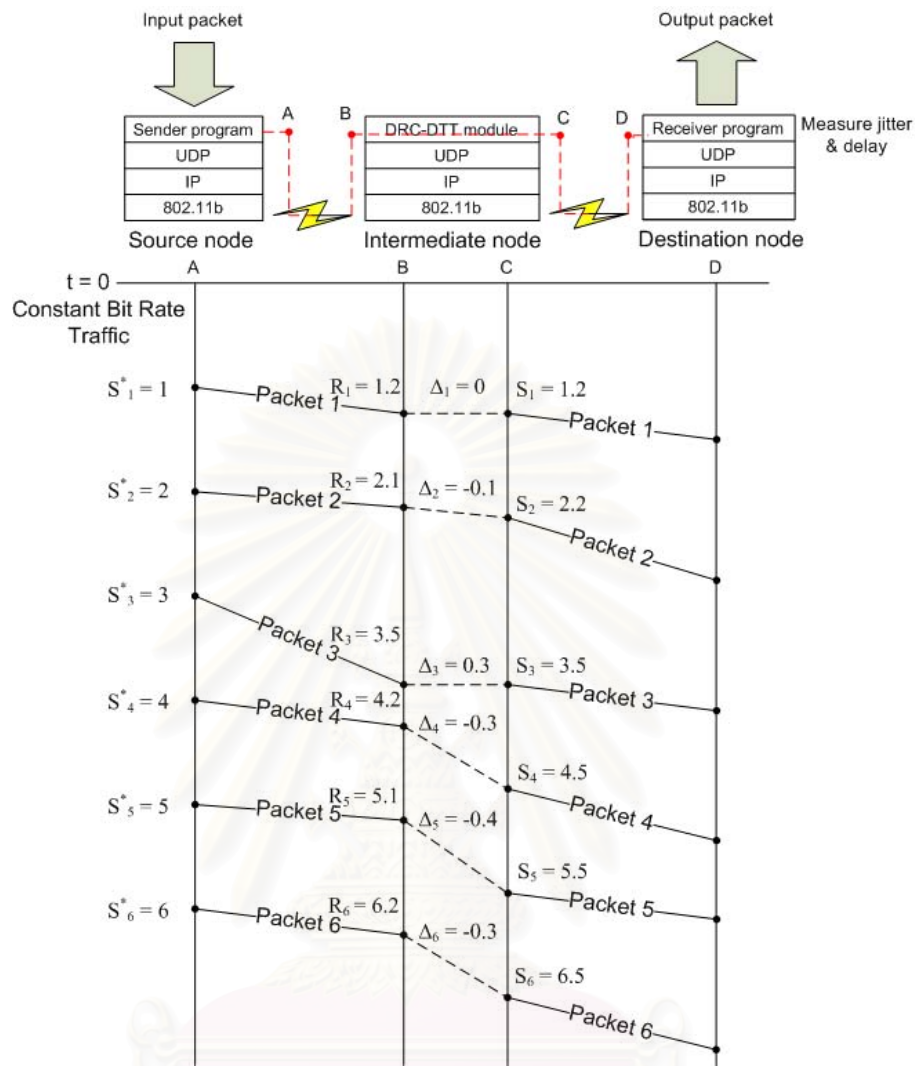


R_i : Receiving time at each intermediate node of packet i
 S_i : Sending time at each intermediate node of packet i
 S_i^* : Sending time at source node of packet i
 $\Delta_i = (R_i - S_{i-1}) - (S_i^* - S_{i-1}^*)$

รูปที่ 3.16 กระบวนการตัดสินใจของ DRC-DTT

3.2.3 การควบคุมแบบเวลาจริงแบบกระจายที่ปรับตัวได้ (Adaptive Distributed Real-time Control : ADRC)

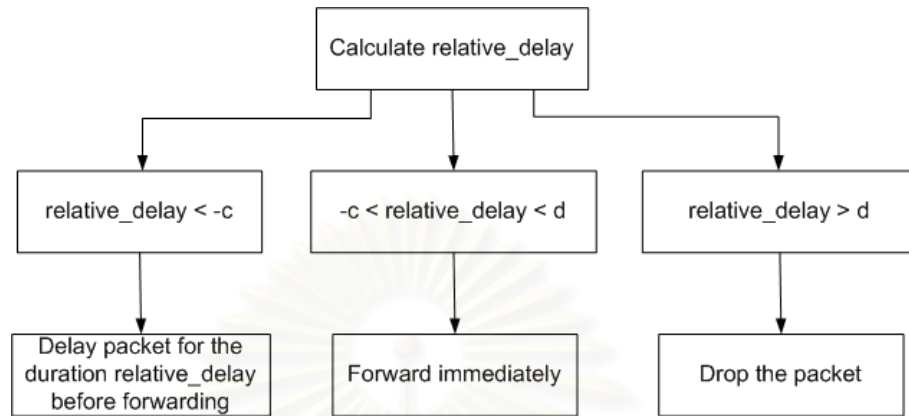
ในวิธี ADRC นี้ จะเป็นการปรับปรุงเสถียรภาพของวิธี DRC-DTT ซึ่งมีปัญหาที่ค่าประวิงเวลาของกลุ่มข้อมูลหนึ่ง ๆ ที่สูงผิดปกติสามารถส่งผลกระทบต่อกรเลือกกำหนดช่วงเวลาสำหรับกลุ่มข้อมูลที่ตามมาทั้งหมดได้ โดยแทนที่จะหนดเวลาเพิ่มจากการเทียบกับค่าประวิงเวลาของกลุ่มข้อมูลก่อนหน้าเพียงกลุ่มข้อมูลเดียว วิธีนี้จะใช้การนำค่าเฉลี่ยเคลื่อนที่ถ่วงน้ำหนักเลขชี้กำลัง (exponentially weighted moving average: EWMA) ของค่าประวิง



รูปที่ 3.17 แผนภาพเวลาของการส่งที่ควบคุมโดย DRC-DDT

เวลาทุก ๆ กลุ่มข้อมูลมาเปรียบเทียบแทน นั่นคือเวลาที่จะหน่วงเพิ่มเข้าไปจะอ้างอิงจากค่าประวิงเวลาของกลุ่มข้อมูลส่วนใหญ่ หากมีกลุ่มข้อมูลอื่นใดอันหนึ่งที่มีค่าประวิงเวลาพุ่งสูงขึ้นมา ก็จะไม่ส่งผลกระทบต่อมากนัก และเนื่องจากวิธีนี้ไม่ได้อ้างอิงจากเวลาของกลุ่มข้อมูลลำดับก่อนหน้าเหมือนในวิธี DRC-DDT หากมีการสูญหายของกลุ่มข้อมูลเกิดขึ้น ก็สามารถจัดการกับกลุ่มข้อมูลที่ตามมาได้ตามปกติ นอกจากนี้วิธี ADRC ยังนำแนวคิดของวิธีตรวจหาจิตเตอร์ [26] มาประยุกต์รวมด้วย คือกลุ่มข้อมูลใดที่มีจิตเตอร์มากก็ไม่ควรถูกส่งต่อไป เพราะถึงแม้จะไปถึงโนดปลายทางก็ไม่สามารถนำมาใช้ประโยชน์ได้ เนื่องจากจิตเตอร์มีค่าเกินจากค่าที่ยอมรับได้สำหรับข้อมูลสื่อผสมแล้ว อีกทั้งยังเป็นการช่วยลดความคับคั่งในเครือข่ายลงอีกด้วย วิธี ADRC มีหลักการทำงานดังแสดงในรูปที่ 3.18 โดยที่ *relative_delay* หาได้จากสมการ (3.1) *delay* และ *ave_delay* หาได้จากสมการ (3.2) และ (3.3) ตามลำดับ เมื่อ R_i เป็นเวลาที่โนดระหว่างทางได้รับกลุ่มข้อมูลที่ i ส่วน S_i เป็นเวลาที่โนดต้นทางส่งกลุ่มข้อมูลที่ i ออกมา ซึ่งเวลาดังกล่าวนี้จะถูกแนบมากับแต่ละกลุ่มข้อมูล ส่วน d เป็น *relative_delay* สูงสุดที่โนดยังยอมส่ง

กลุ่มข้อมูลนั้นต่อไป และ c เป็นค่า $relative_delay$ ต่ำสุดที่สามารถยอมรับได้โดยไม่ทำการหน่วงเวลาเพิ่ม



รูปที่ 3.18 กระบวนการตัดสินใจของวิธี ADRC

$$relative_delay = delay - ave_delay \quad (3.1)$$

$$delay = R_i - S_i \quad (3.2)$$

$$ave_delay = (1 - w) * ave_delay + w * delay \quad (3.3)$$

ซึ่งในวิธีนี้ถึงแม้จะคิดค่าประวิงเวลาที่คนละโนดกัน แต่ก็ไม่จำเป็นต้องมีการประสานเวลา (synchronisation) เพื่อให้เวลาของทุกโนดตรงกัน เพียงแต่ต้องอยู่ภายใต้สมมติฐานว่าเวลาเลื่อน (drift time) ของสัญญาณนาฬิกาที่แต่ละโนดมีค่าเท่า ๆ กัน นั่นคือเวลาที่ผิดพลาดไปของแต่ละโนดมีค่าคงที่ตลอด

สำหรับกรณีที่มีกลุ่มข้อมูลเข้ามาไม่ตรงตามลำดับ (out-of-sequence) อันเนื่องมาจากมีการเปลี่ยนแปลงเส้นทาง หรือมีการส่งกลุ่มข้อมูลใหม่ที่ขึ้นควบคุมการเข้าถึงขึ้น กระบวนการทั้ง 3 วิธีจะจัดการโดยเรียงตามลำดับที่มาถึงก่อน ส่วนกลุ่มข้อมูลลำดับก่อนหน้าที่ส่งมาถึงช้ากว่า เมื่อโนดระหว่างทางได้รับแล้วก็จะส่งต่อออกไปโดยไม่มีผลกระทบผลใด ๆ เกิดขึ้น

โดยหลักการควบคุมแบบเวลาจริงทั้ง 3 วิธีที่ได้กล่าวมา จะนำไปทดลองเปรียบเทียบประสิทธิภาพกับการส่งโดยไม่มี การควบคุมใด ๆ ในระบบทดสอบทั้งสองแบบ ซึ่งผลของจัตเตอร์และค่าประวิงเวลาที่โนดปลายทาง รวมถึงอัตรา การสูญหายของกลุ่มข้อมูลจะได้กล่าวถึงในบทที่ 4 ต่อไป

บทที่ 4

ผลการทดสอบ

เนื้อหาในบทนี้จะกล่าวถึงผลการประเมินประสิทธิภาพของการควบคุมแบบเวลาจริงที่เสนอทั้ง 3 วิธีในสถานการณ์ต่าง ๆ พร้อมทั้งศึกษาปัจจัยที่มีผลกระทบต่อจิตเตอร์ และค่าประวิงเวลาที่โนดปลายทาง ซึ่งในวิทยานิพนธ์นี้ได้แบ่งการทดลองออกเป็น 5 แบบ โดยใน 2 แบบแรกจะเป็นการทดลองปรับเปลี่ยนค่าพารามิเตอร์ภายในของวิธี ADRC และ DRC-RBT เพื่อหาค่าที่เหมาะสมสำหรับนำมาใช้งาน ส่วนอีก 2 แบบ จะเป็นการศึกษาผลกระทบเนื่องมาจากขนาดของเครือข่าย อัตราการสูญเสียกลุ่มข้อมูล และการเคลื่อนที่ของโนด โดยที่ในแต่ละการทดลองได้เลือกใช้ประเภทของระบบทดสอบต่างกันไป ขึ้นอยู่กับความเหมาะสมของสถานการณ์

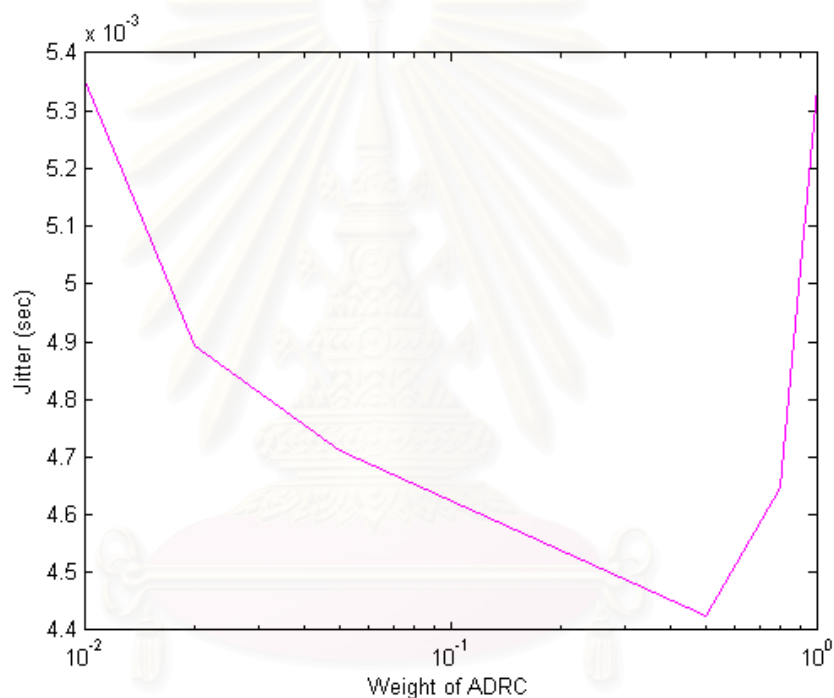
คำจำกัดความและค่าที่ถูุกำหนดของพารามิเตอร์หลัก ๆ ที่ใช้ในการทดลองมีดังนี้

- l = อัตราการสูญเสียกลุ่มข้อมูลต่อหนึ่งช่วงเชื่อมต่อ
- p = ความน่าจะเป็นที่เกตเวย์เลียนแบบจะส่งกลุ่มข้อมูลสำเร็จ
- w = ค่าถ่วงน้ำหนักในวิธี ADRC
- d = ซีดเริ่มเปลี่ยนของการตัดสินใจในวิธี ADRC ว่าจะตัดกลุ่มข้อมูลนั้นทิ้งหรือไม่
- c = ค่า *relative_delay* ต่ำสุดที่สามารถยอมรับได้โดยจะไม่ทำการหน่วงเวลาเพิ่ม
- $threshold$ = ซีดเริ่มเปลี่ยนของวิธี DRC-RBT
- ช่วงเวลา ระหว่างข้อมูลแต่ละเฟรมของกราฟฟิกแบบอัตราบิตแปรผัน ได้มีค่าเท่ากับ 66.7 มิลลิวินาที เนื่องจากส่งด้วยอัตราเร็ว 15 เฟรมต่อวินาที โดยมีค่าคงที่ในทุกการทดลอง
- ช่วงเวลา ระหว่างแต่ละกลุ่มข้อมูลในกราฟฟิกแบบอัตราบิตคงที่มีค่าเท่ากับ 61.9 มิลลิวินาที ซึ่งหามาจากอัตราเร็วเฉลี่ยของกราฟฟิกแบบอัตราบิตแปรผันได้ โดยมีค่าคงที่ในทุกการทดลอง
- ช่วงเวลา ระหว่างแต่ละกลุ่มข้อมูลในช่วงที่มีการส่งเป็นชุดของกราฟฟิกแบบอัตราบิตแปรผัน ได้มีค่าเท่ากับ 3 มิลลิวินาที ซึ่งถือเป็นเวลาที่ใช้ในการบีบอัดข้อมูลในแต่ละเฟรม โดยมีค่าคงที่ในทุกการทดลอง
- จำนวนกลุ่มข้อมูลที่ส่งต่อการทดลองหนึ่งครั้งในระบบทดสอบแบบมีสายเท่ากับ 12095 กลุ่มข้อมูล ใช้เวลาประมาณ 12 นาที
- จำนวนกลุ่มข้อมูลที่ส่งต่อการทดลองหนึ่งครั้งในระบบทดสอบแบบไร้สายเท่ากับ 4838 กลุ่มข้อมูล ใช้เวลาประมาณ 5 นาที

4.1 การทดลองเพื่อหาค่าถ่วงน้ำหนักสำหรับวิธี ADRC ที่เหมาะสม

การทดลองนี้จะเป็นการปรับเปลี่ยนค่าถ่วงน้ำหนัก (w) ที่ใช้ในการคำนวณหาค่าประวิงเวลาเฉลี่ยสำหรับวิธี ADRC เพื่อหาค่าที่เหมาะสมกับการใช้งาน โดยที่กำหนดให้กลุ่มข้อมูลถูกส่งผ่าน 50 ช่วงเชื่อมต่อ ในขณะที่ทุกโหนดไม่มีการเคลื่อนที่, $d = 0.5$ วินาที, $c = 2$ มิลลิวินาที, $p = 0.7$, $l = 0.005$ และจะทำการปรับเปลี่ยนค่า w ตั้งแต่ 0.01 ไปจนถึง 1 รูปที่ 4.1-4.4 แสดงจิตเตอร์เฉลี่ยและค่าประวิงเวลาเฉลี่ยในกราฟฟิกแบบอัตราบิตคงที่และอัตราบิตแปรผันได้ ตามลำดับ

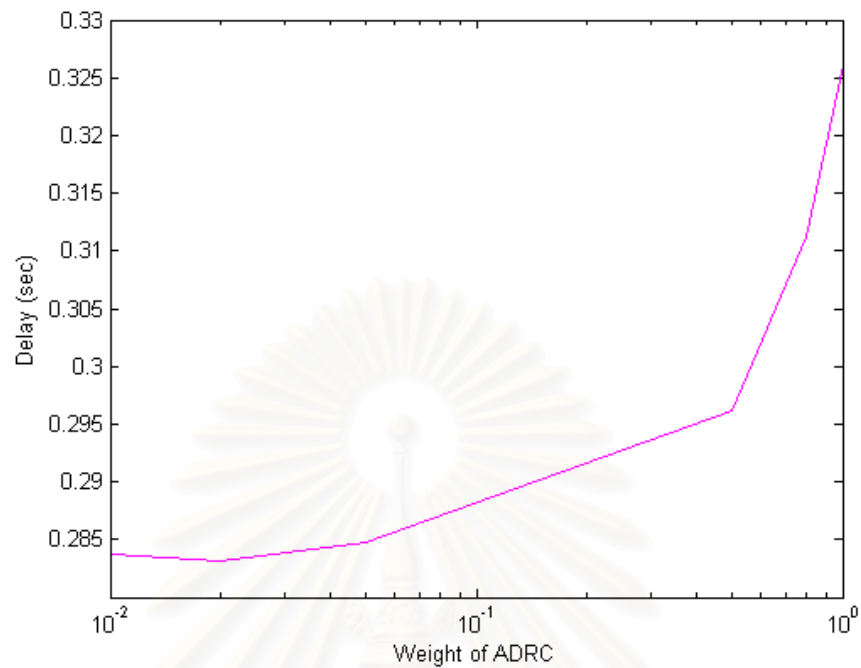
ค่าถ่วงน้ำหนักที่มีค่าน้อยจะหมายถึงระบบนั้นมีความทนทานต่อการรบกวนจากกลุ่มข้อมูลที่มีค่าประวิงเวลา



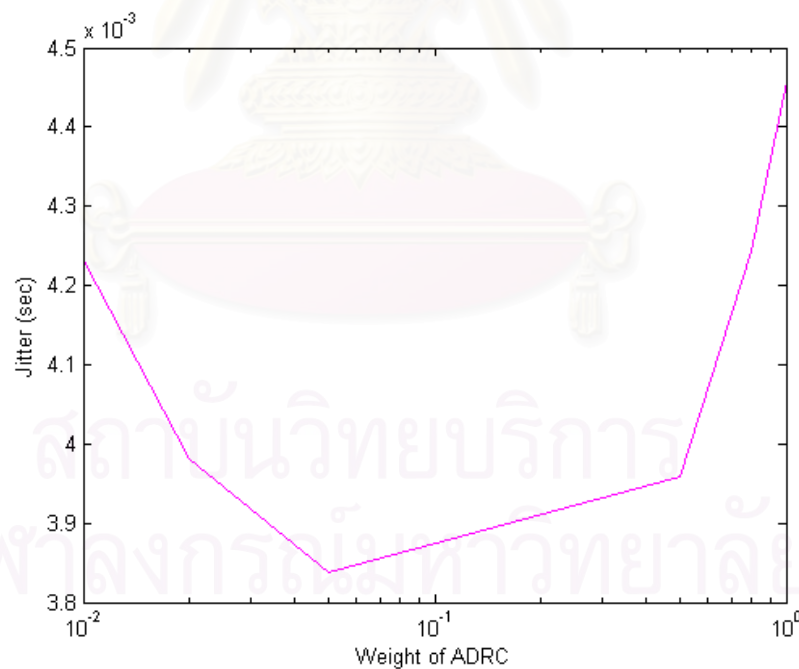
รูปที่ 4.1 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิตคงที่ในวิธี ADRC เมื่อปรับเปลี่ยนค่าถ่วงน้ำหนัก

สูงผิดปกติได้มาก เนื่องจากค่าประวิงเวลาที่สูงผิดปกติของกลุ่มข้อมูลส่วนน้อยไม่ส่งผลถึงค่าประวิงเวลาเฉลี่ยมากนัก แต่ขณะเดียวกันหากใช้ในเครือข่ายที่ค่าประวิงเวลาของแต่ละกลุ่มข้อมูลมีการเปลี่ยนแปลงบ่อย เช่นเครือข่ายที่โหนดมีการเคลื่อนที่ ระบบนี้ก็จะมีความสามารถในการปรับตัวได้ช้า ในทางกลับกันค่าถ่วงน้ำหนักที่มีค่ามาก ถึงแม้ระบบจะปรับตัวเข้ากับสถานการณ์ในขณะนั้นได้เร็ว แต่ก็จะมีเสถียรภาพเท่าที่ควร เนื่องจากการรบกวนสามารถส่งผลกระทบต่อระบบได้มาก ดังนั้นในการเลือกค่าถ่วงน้ำหนักจึงมีความสำคัญและส่งผลถึงประสิทธิภาพของวิธี ADRC นี้มาก

จากรูปที่ 4.1-4.4 จะเห็นว่าค่าถ่วงน้ำหนักที่เหมาะสมต้องมีค่าไม่น้อยหรือมากเกินไป ซึ่งในวิทยานิพนธ์นี้ได้เลือกใช้ค่าเท่ากับ 0.02 เนื่องจากทำให้ค่าประวิงเวลามีค่าต่ำ ในขณะที่จิตเตอร์ก็ไม่สูงจนเกินไป



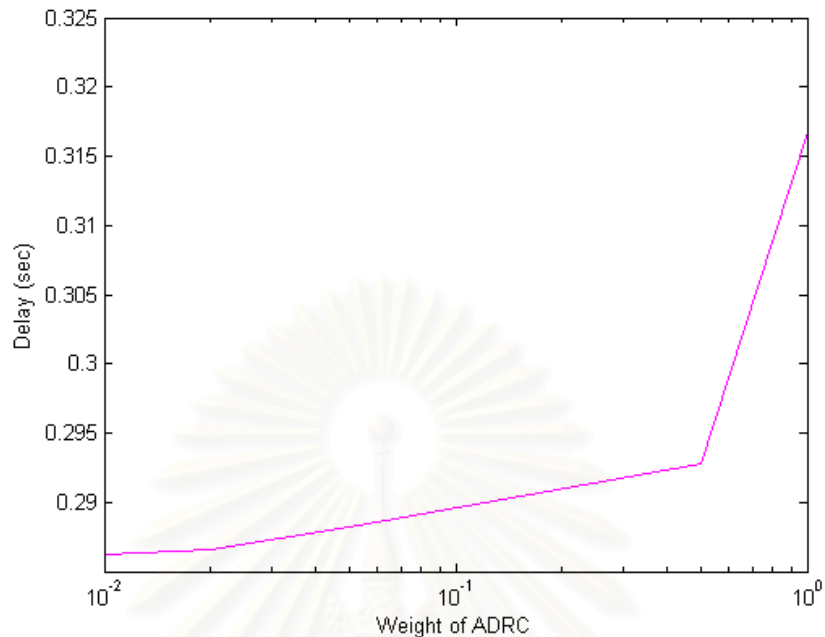
รูปที่ 4.2 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิดคงที่ในวิธี ADRC เมื่อปรับเปลี่ยนค่าถ่วงน้ำหนัก



รูปที่ 4.3 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้ในวิธี ADRC เมื่อปรับเปลี่ยนค่าถ่วงน้ำหนัก

4.2 การทดลองเพื่อหาค่าขีดเริ่มเปลี่ยนสำหรับวิธี DRC-RBT ที่เหมาะสม

การทดลองนี้จะเป็นการปรับเปลี่ยนค่าขีดเริ่มเปลี่ยน (*threshold*) ที่ใช้ในวิธี DRC-RBT โดยที่กำหนดให้กลุ่มข้อมูลถูกส่งออกไปด้วยอัตราบิดคงที่ผ่าน 50 ช่วงเชื่อมต่อ ในขณะที่ทุกโนดไม่มีการเคลื่อนที่, $p = 0.7$, $l =$



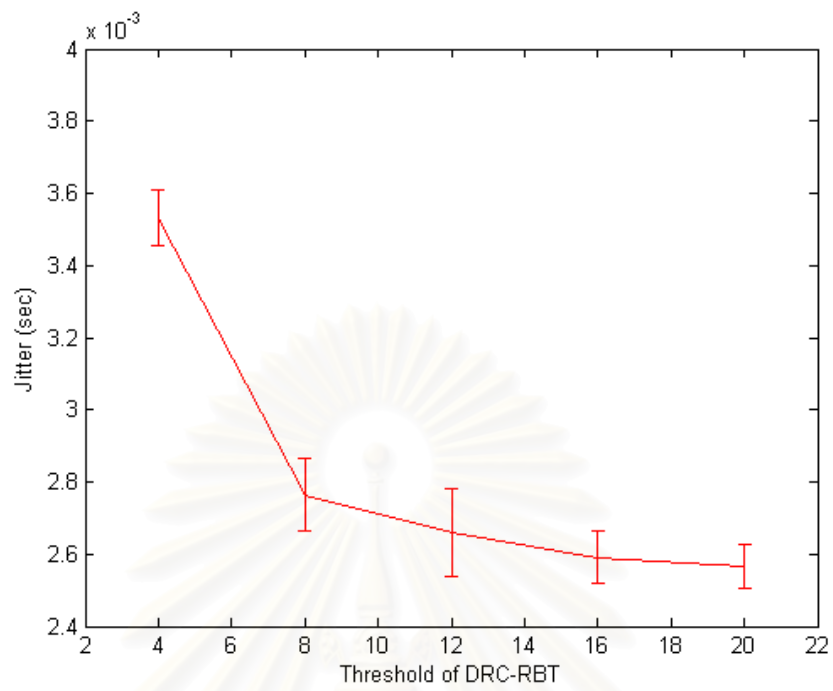
รูปที่ 4.4 ค่าประวิงเวลาเฉลี่ยของกราฟฟิคที่มีอัตราบิดแปรผันได้ในวิธี ADRC เมื่อปรับเปลี่ยนค่าถ่วงน้ำหนัก

0.005 รูปที่ 4.5 และ 4.6 แสดงจิตเตอร์เฉลี่ยและค่าประวิงเวลาเฉลี่ยเมื่อมีการปรับเปลี่ยนขีดเริ่มเปลี่ยนจาก 4 จนถึง 20

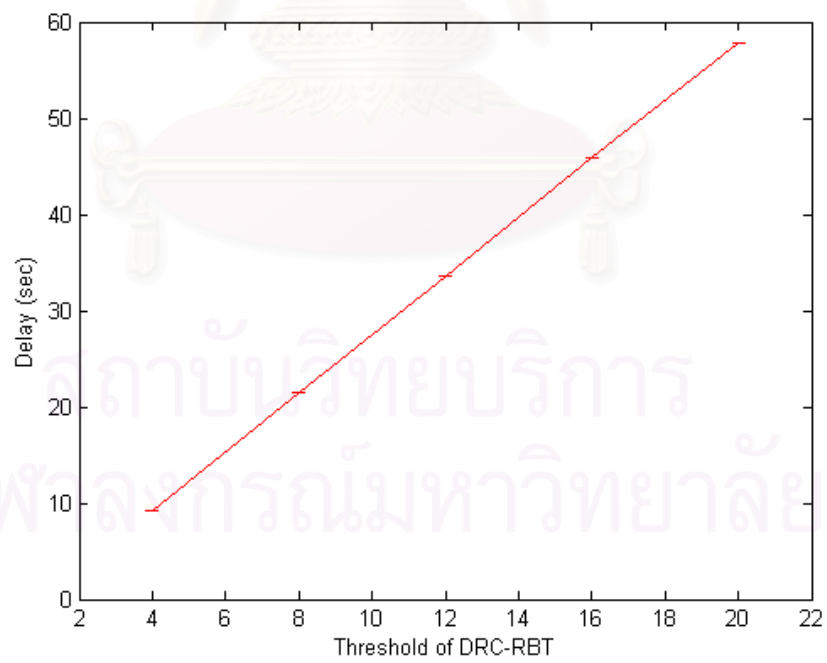
จะเห็นว่าขีดเริ่มเปลี่ยนที่มีค่าน้อย จะไม่สามารถช่วยจัดการจิตเตอร์ได้ดีเท่าที่ควร และเมื่อขีดเริ่มเปลี่ยนมีค่ามากขึ้น จิตเตอร์ก็จะต่ำลง ในขณะที่ค่าประวิงเวลาก็มีค่าเพิ่มขึ้น แต่จิตเตอร์จะต่ำลงได้จนถึงระดับหนึ่งเท่านั้น ซึ่งหลังจากนี้ไม่ว่าจะเพิ่มขีดเริ่มเปลี่ยนอีกเท่าใดก็ไม่สามารถช่วยให้จิตเตอร์ต่ำกว่านี้ได้ โดยจากรูปที่ 4.5 พบว่าหลังจากค่าขีดเริ่มเปลี่ยนเท่ากับ 8 เป็นต้นไป จิตเตอร์จะลดลงในอัตราที่ไม่มากนัก ขณะที่ค่าประวิงเวลายังคงเพิ่มขึ้นอย่างเป็นเชิงเส้นอยู่ ดังนั้นค่าขีดเริ่มเปลี่ยนที่เหมาะสมกับการนำไปใช้งานจึงควรเป็นค่าที่ช่วยลดจิตเตอร์ได้ในขณะที่ค่าประวิงเวลาไม่สูงจนเกินไปนัก ซึ่งในวิทยานิพนธ์นี้ได้เลือกใช้ค่าขีดเริ่มเปลี่ยนเท่ากับ 10

4.3 ผลกระทบจากขนาดของเครือข่าย

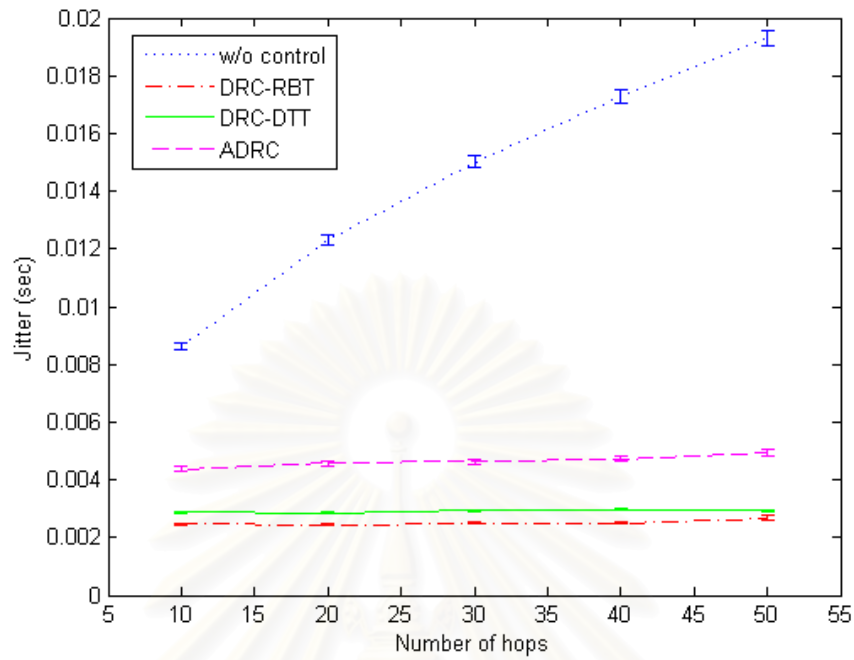
ในการทดลองนี้จะใช้ระบบทดสอบแบบมีสายในการประเมินผล เนื่องจากสามารถเพิ่มขนาดของเครือข่ายได้ตามต้องการ โดยกำหนดให้ทุกโหนดไม่มีการเคลื่อนที่ และจะปรับเปลี่ยนจำนวนช่วงเชื่อมต่อที่กลุ่มข้อมูลต้องเดินทางจากโหนดต้นทางไปถึงโหนดปลายทางตั้งแต่ 10 ถึง 50 ช่วงเชื่อมต่อ, $w = 0.02$, $d = 0.5$ วินาที, $c = 2$ มิลลิวินาที. $threshold = 10$, $p = 0.7$, $l = 0$ รูปที่ 4.7-4.10 แสดงจิตเตอร์เฉลี่ยและค่าประวิงเวลาเฉลี่ยของกราฟฟิคแบบอัตราบิดคงที่และอัตราบิดแปรผันได้ ตามลำดับ ส่วนรูปที่ 4.11-4.14 แสดงการเปรียบเทียบจิตเตอร์และค่าประวิงเวลาของแต่ละวิธีเมื่อส่งผ่าน 50 ช่วงเชื่อมต่อในกราฟฟิคแบบอัตราบิดคงที่และอัตราบิดแปรผันได้ ตามลำดับ โดย



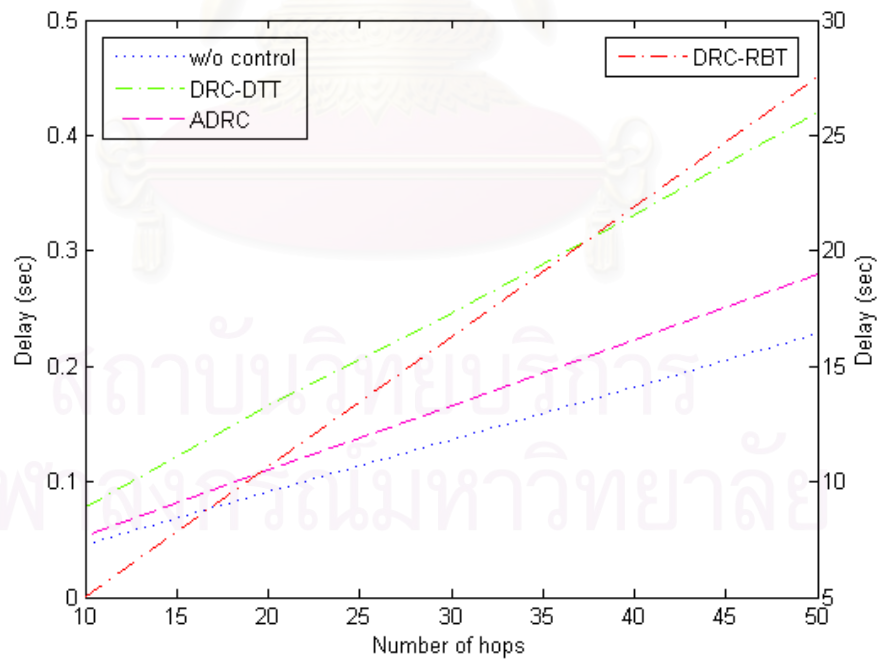
รูปที่ 4.5 จิตเตอร์เฉลี่ยเมื่อปรับเปลี่ยนค่าขีดเริ่มเปลี่ยน



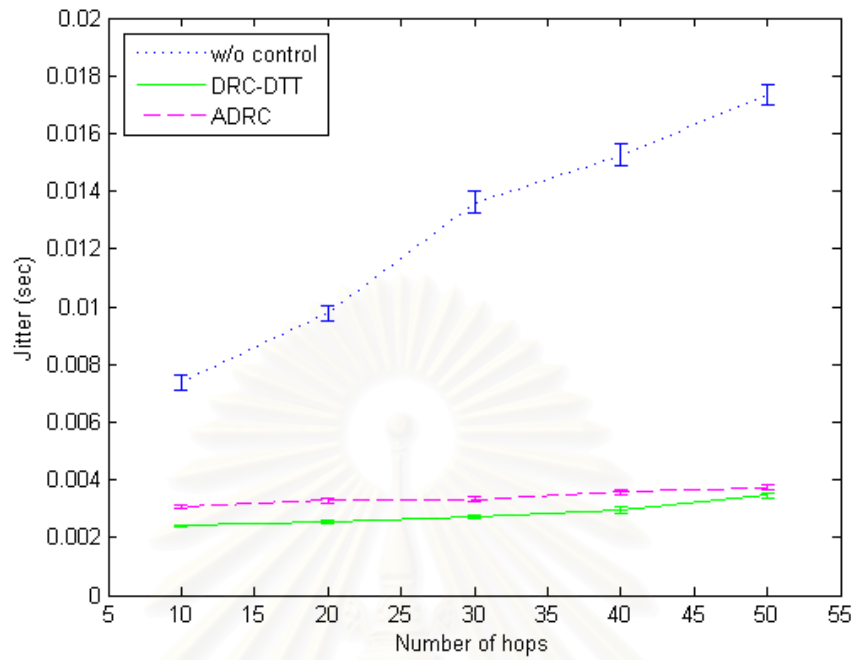
รูปที่ 4.6 ค่าประวิงเวลาเฉลี่ยเมื่อปรับเปลี่ยนค่าขีดเริ่มเปลี่ยน



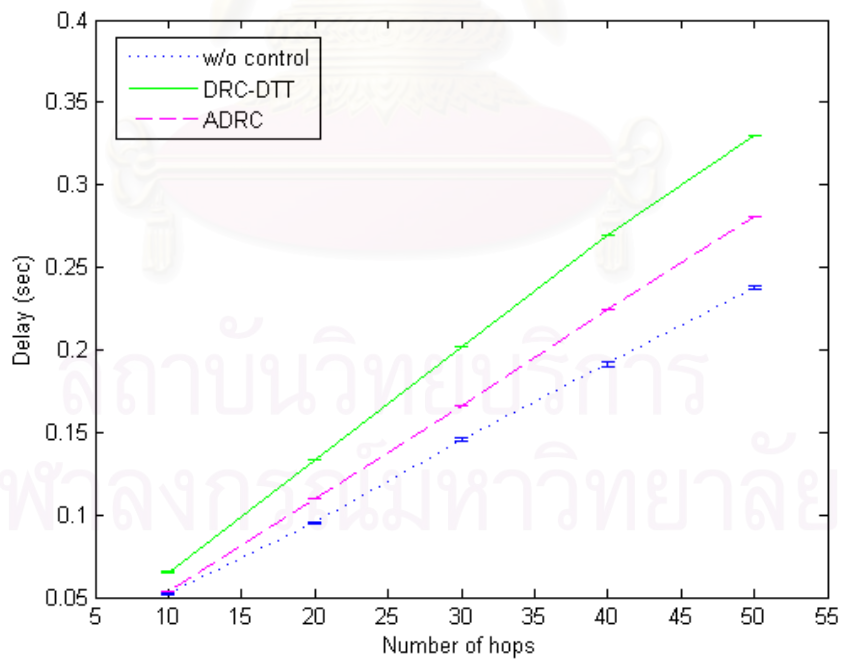
รูปที่ 4.7 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดคงที่เมื่อปรับเปลี่ยนจำนวนช่วงเชื่อมต่อ



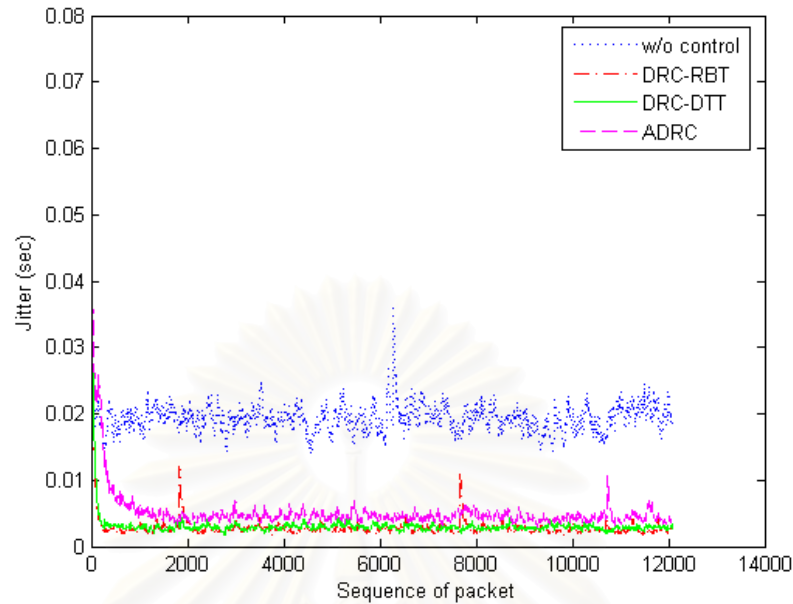
รูปที่ 4.8 ค่าประวิงเวลาเฉลี่ยของกราฟฟิกที่มีอัตราบิดคงที่เมื่อปรับเปลี่ยนจำนวนช่วงเชื่อมต่อ



รูปที่ 4.9 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้เมื่อปรับเปลี่ยนจำนวนช่วงเชื่อมต่อ



รูปที่ 4.10 ค่าประวิงเวลาเฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้เมื่อปรับเปลี่ยนจำนวนช่วงเชื่อมต่อ

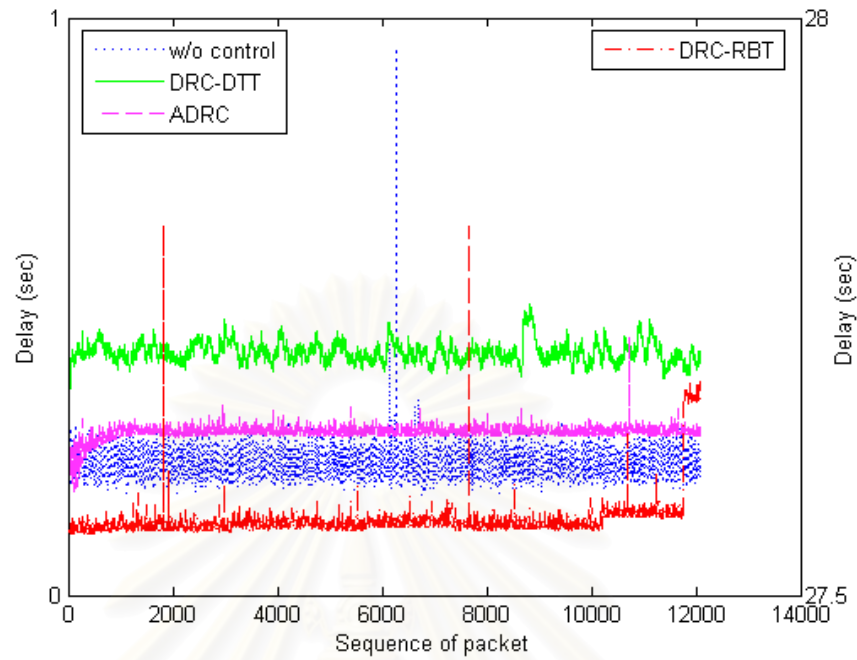


รูปที่ 4.11 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิตคงที่เพื่อเปรียบเทียบการควบคุมแบบเวลาจริงในแต่ละวิธี

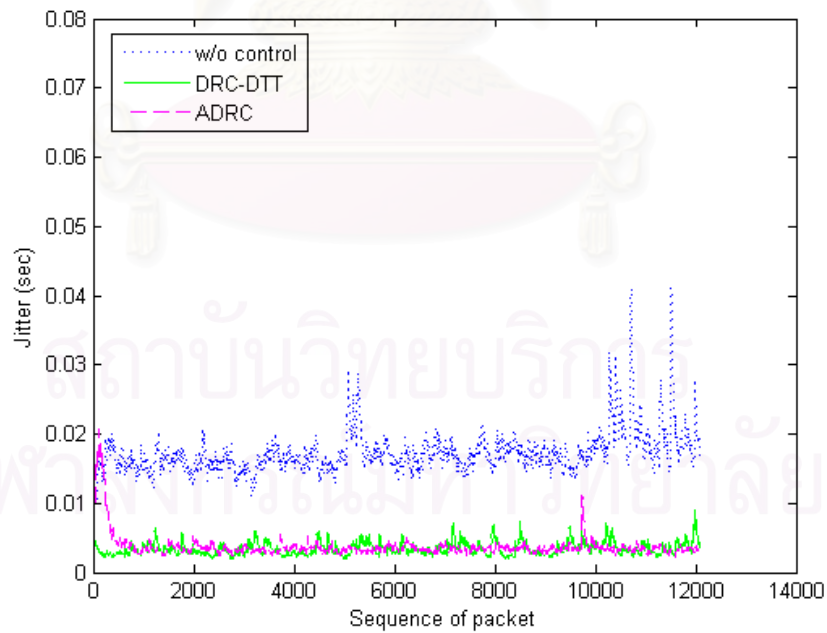
ในกราฟฟิกแบบอัตราบิตคงที่ กรณีที่ไม่มีการควบคุมใด ๆ, DRC-RBT, DRC-DTT และ ADRC จะมีค่าจิตเตอร์เฉลี่ยเท่ากับ 19.3, 2.7, 2.9 และ 5 มิลลิวินาที ตามลำดับ ส่วนกราฟฟิกแบบอัตราบิตแปรผันได้ กรณีที่ไม่มีการควบคุมใด ๆ, DRC-DTT และ ADRC จะมีค่าจิตเตอร์เฉลี่ยเท่ากับ 17.3, 3.5 และ 3.7 มิลลิวินาที ตามลำดับ

จะเห็นว่าเมื่อจำนวนช่วงเชื่อมต่อมีค่ามากขึ้น จิตเตอร์และค่าประวิงเวลาของกรณีที่ไม่มีการควบคุมใด ๆ จะมีความมากขึ้นด้วย แต่จิตเตอร์ที่เพิ่มขึ้นนี้จะไม่เพิ่มแบบเป็นเชิงเส้น เนื่องจากเมื่อต้องเดินทางผ่านช่วงเชื่อมต่อมากขึ้น จิตเตอร์ของแต่ละช่วงเชื่อมต่อก็จะมีโอกาสชดเชยกันไปได้ ส่วนจิตเตอร์ของอีก 3 วิธีจะไม่เพิ่มขึ้นตามขนาดของเครือข่าย เนื่องจากทุกโหนดระหว่างทางได้ช่วยจัดการควบคุมอยู่ โดยวิธี DRC-RBT จิตเตอร์จะมีค่าน้อยที่สุด แต่ค่าประวิงเวลาจะมีค่าสูงขึ้นมา และสามารถใช้ได้เฉพาะกับกราฟฟิกที่มีอัตราบิตคงที่ รองลงมาคือวิธี DRC-DTT ซึ่งจะมีจิตเตอร์มากกว่า DRC-RBT เล็กน้อย แต่ค่าประวิงเวลามีค่าต่ำกว่ามาก ส่วนวิธี ADRC จะมีจิตเตอร์สูงกว่าทั้งวิธี DRC-DTT และ ADRC แต่ค่าประวิงเวลาจะมีค่าต่ำกว่า ดังนั้นในเครือข่ายที่มีอัตราการสูญเสียกลุ่มข้อมูลต่ำวิธี DRC-RBT จะควบคุมจิตเตอร์ได้ดีที่สุด หากยอมรับได้กับค่าประวิงเวลาที่มีค่าเพิ่มขึ้นมามาก แต่หากไม่ต้องการให้ค่าประวิงเวลามีค่ามากเกินไป สามารถเลือกใช้วิธี DRC-DTT และ ADRC ที่ควบคุมจิตเตอร์ได้ดีรองลงมาตามลำดับได้

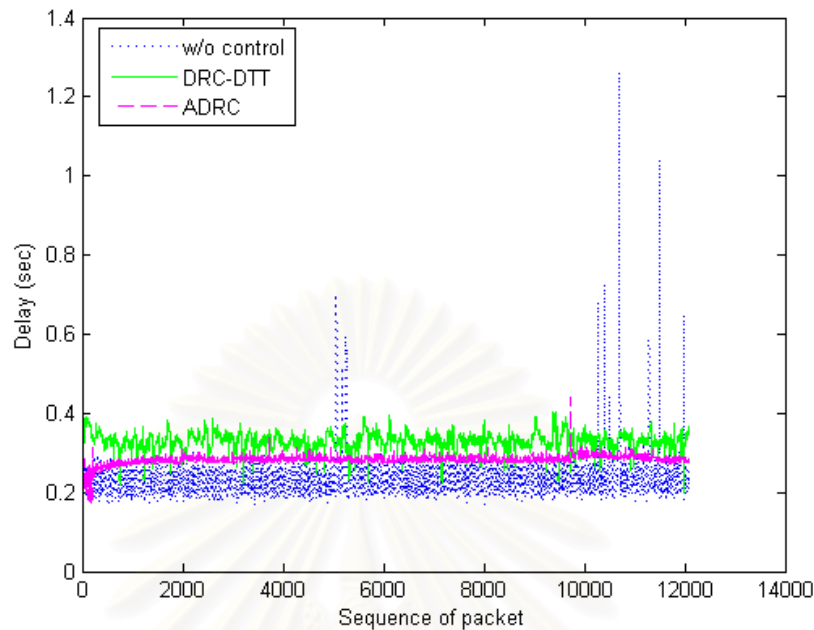
เมื่อนำผลของกราฟฟิกแบบอัตราบิตคงที่และอัตราบิตแปรผันได้มาเปรียบเทียบกัน จะเห็นว่าในกราฟฟิกแบบอัตราบิตคงที่ที่มีจิตเตอร์มากกว่าแบบอัตราบิตแปรผันได้ แต่ค่าประวิงเวลาจะมีค่าใกล้เคียงกัน ที่เป็นเช่นนี้เนื่องจากว่าในกราฟฟิกแบบอัตราบิตแปรผันได้มีการส่งกลุ่มข้อมูลออกไปเป็นชุด ๆ (burst) ซึ่งกลุ่มข้อมูลในแต่ละชุดนี้จะมีค่าประวิงเวลาต่างกันไม่มาก เพราะส่งออกไปในช่วงเวลาที่ใกล้เคียงกัน ทำให้ความแปรผันของค่าประวิงเวลาในส่วน



รูปที่ 4.12 ค่าประวิงเวลาเฉลี่ยของกราฟฟิกที่มีอัตราบิตคงที่เพื่อเปรียบเทียบการควบคุมแบบเวลาจริงในแต่ละวิธี



รูปที่ 4.13 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิตแปรผันได้เพื่อเปรียบเทียบการควบคุมแบบเวลาจริงในแต่ละวิธี



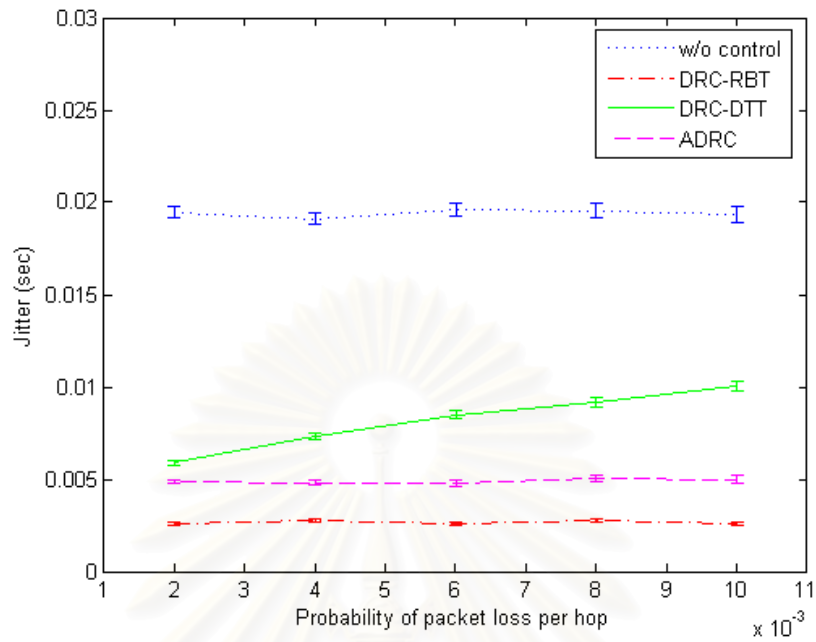
รูปที่ 4.14 ค่าประวิงเวลาเฉลี่ยของกราฟฟิกที่มีอัตราบิตแปรผันได้ เพื่อเปรียบเทียบการควบคุมแบบเวลาจริงในแต่ละวิธี

นี้มีค่าน้อย ซึ่งต่างจากกราฟฟิกแบบอัตราบิตคงที่ที่ค่าประวิงเวลาของทุกกลุ่มข้อมูลไม่ขึ้นต่อกัน ทำให้ค่าประวิงเวลามีโอกาสแปรผันไปได้มากกว่า ดังนั้นเมื่อหาค่าเฉลี่ยออกมาแล้วจิตเตอร์ของกราฟฟิกแบบอัตราบิตคงที่จึงมีค่ามากกว่า

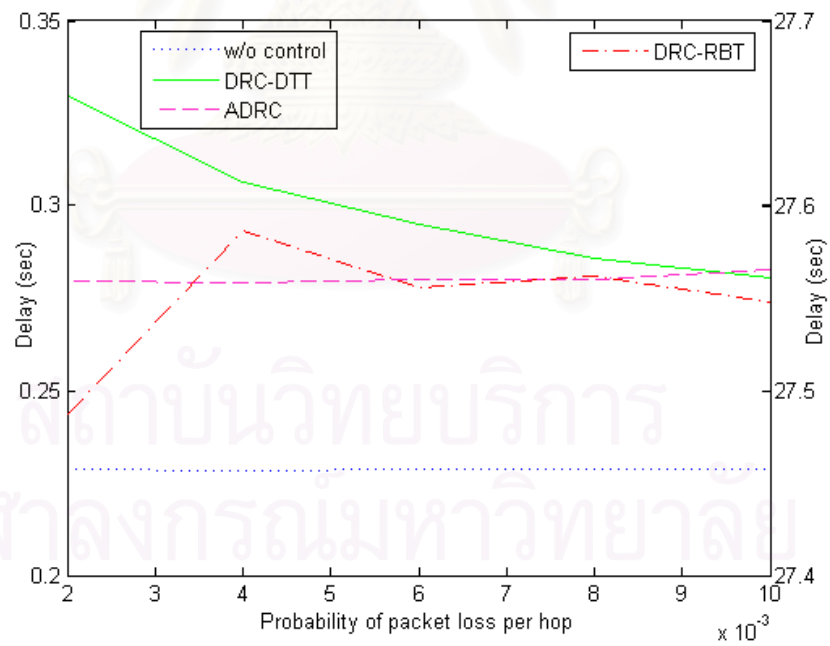
4.4 ผลกระทบจากอัตราการสูญเสียกลุ่มข้อมูล

ในการทดลองนี้จะใช้ระบบทดสอบแบบมีสายในการประเมินผลเช่นกัน เนื่องจากสามารถกำหนดอัตราการสูญเสียกลุ่มข้อมูลตามที่ต้องการได้ โดยกำหนดให้ทุกโหนดไม่มีการเคลื่อนที่ และมีจำนวนช่วงเชื่อมต่อที่กลุ่มข้อมูลต้องเดินทางจากต้นทางไปปลายทางเท่ากับ 50 ช่วงเชื่อมต่อ ซึ่งในการทดลองจะปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูลต่อหนึ่งช่วงเชื่อมต่อ (l) จาก 0.002 จนถึง 0.01, $w = 0.02$, $d = 0.5$ วินาที, $c = 2$ มิลลิวินาที. $threshold = 10$, $p = 0.7$ รูปที่ 4.15-4.18 แสดงจิตเตอร์เฉลี่ยและค่าประวิงเวลาเฉลี่ยของกราฟฟิกแบบอัตราบิตคงที่และอัตราบิตแปรผันได้ ตามลำดับ

จะเห็นว่าเมื่ออัตราการสูญเสียกลุ่มข้อมูลมีค่ามากขึ้น จะไม่มีผลต่อจิตเตอร์และค่าประวิงเวลาในกรณีที่ไม่มีการควบคุมใด ๆ รวมถึงวิธี DRC-RBT และ ADRC ส่วนในวิธี DRC-DTT จิตเตอร์จะมีค่าสูงขึ้นเรื่อย ๆ ตามอัตราการสูญเสียกลุ่มข้อมูลที่เพิ่มขึ้น เนื่องจากในวิธีนี้ใช้หลักการเทียบเวลาที่ต้องหน่วงจากกลุ่มข้อมูลก่อนหน้า หากมีกลุ่มข้อมูลใดเกิดการสูญหาย กลุ่มข้อมูลที่ตามมาจะถูกส่งต่อออกไปทันที โดยไม่มีการประมวลผลใด ๆ ดังนั้นหากเครือข่ายมีอัตราการสูญเสียกลุ่มข้อมูลมาก ก็จะทำให้มีกลุ่มข้อมูลที่ถูกส่งต่อออกไปโดยไม่ได้จัดการควบคุมจิต

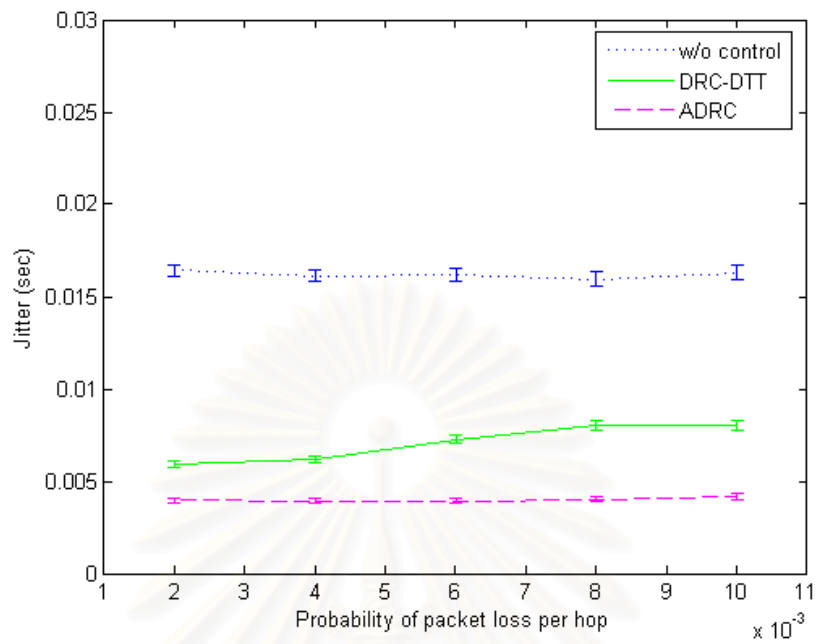


รูปที่ 4.15 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดงที่เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูล

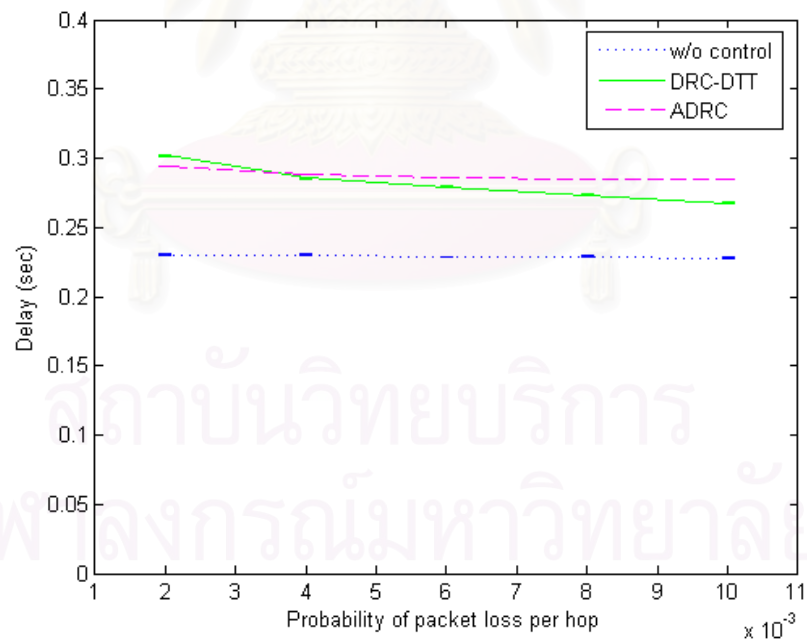


รูปที่ 4.16 ค่าประวิงเวลาเฉลี่ยของกราฟฟิกที่มีอัตราบิดงที่เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูล

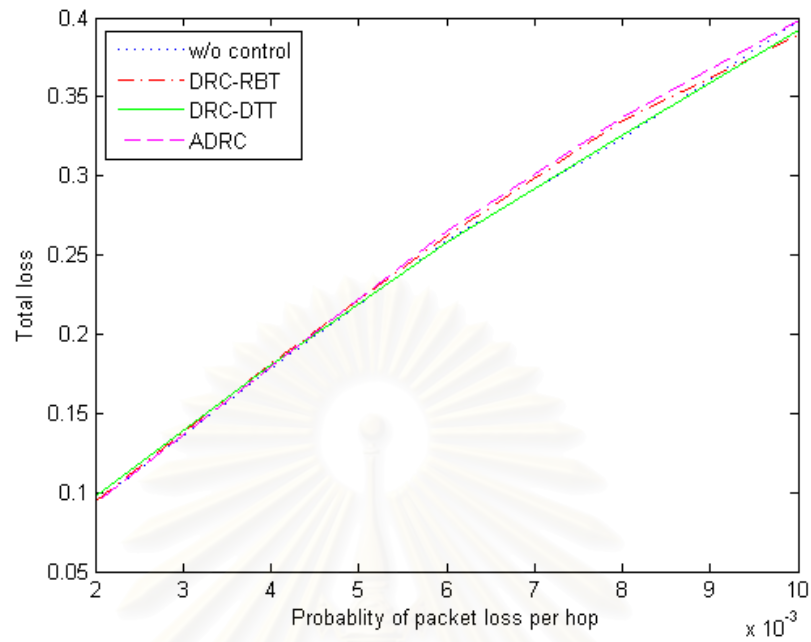
เตอร์มากด้วย จิตเตอร์จึงมีค่ามากขึ้น แต่ปัญหานี้ไม่เกิดขึ้นกับวิธี ADRC เนื่องจากกระบวนการตัดสินใจไม่ได้ขึ้นอยู่กับค่าประวิงเวลาของกลุ่มข้อมูลอันใดอันหนึ่ง แต่อ้างอิงจากค่าประวิงเวลาเฉลี่ยของทุกกลุ่มข้อมูลแทน และตั้งที่



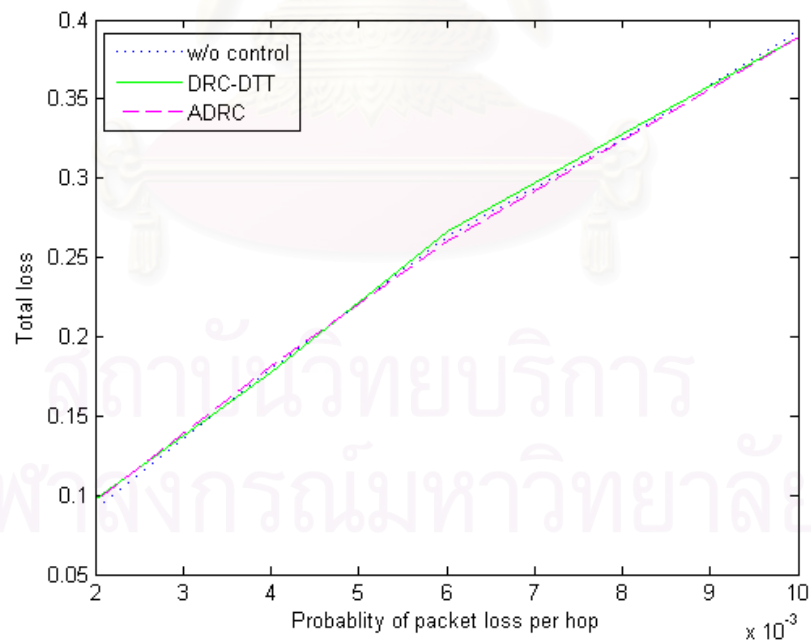
รูปที่ 4.17 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูล



รูปที่ 4.18 ค่าประวิงเวลาเฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูล



รูปที่ 4.19 อัตราการสูญเสียกลุ่มข้อมูลตลอดเส้นทางของกราฟฟิกที่มีอัตราบิดคงที่เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูลต่อหนึ่งช่วงเชื่อมต่อ



รูปที่ 4.20 อัตราการสูญเสียกลุ่มข้อมูลตลอดเส้นทางของกราฟฟิกที่มีอัตราบิดแปรผันได้เมื่อปรับเปลี่ยนอัตราการสูญเสียกลุ่มข้อมูลต่อหนึ่งช่วงเชื่อมต่อ

ได้กล่าวไว้ว่าวิธี ADRC ใช้หลักการตัดกลุ่มข้อมูลที่มีค่าประวิงเวลาสูงผิดปกติออกไป ในรูปที่ 4.19 และ 4.20 จึงได้แสดงอัตราการสูญเสียกลุ่มข้อมูลที่เกิดขึ้นตลอดเส้นทางเปรียบเทียบทั้ง 4 วิธีของกราฟฟิกแบบอัตราบิดงที่และอัตราบิดแปรผันได้ ตามลำดับ โดยจะเห็นว่าอัตราการสูญเสียกลุ่มข้อมูลของทุกวิธีมีค่าใกล้เคียงกันมาก โดยเฉพาะวิธี ADRC ที่ถึงแม้จะไม่ได้ตัดกลุ่มข้อมูลใดทิ้ง ก็ยังช่วยลดค่าจิตเตอร์ลงได้มาก

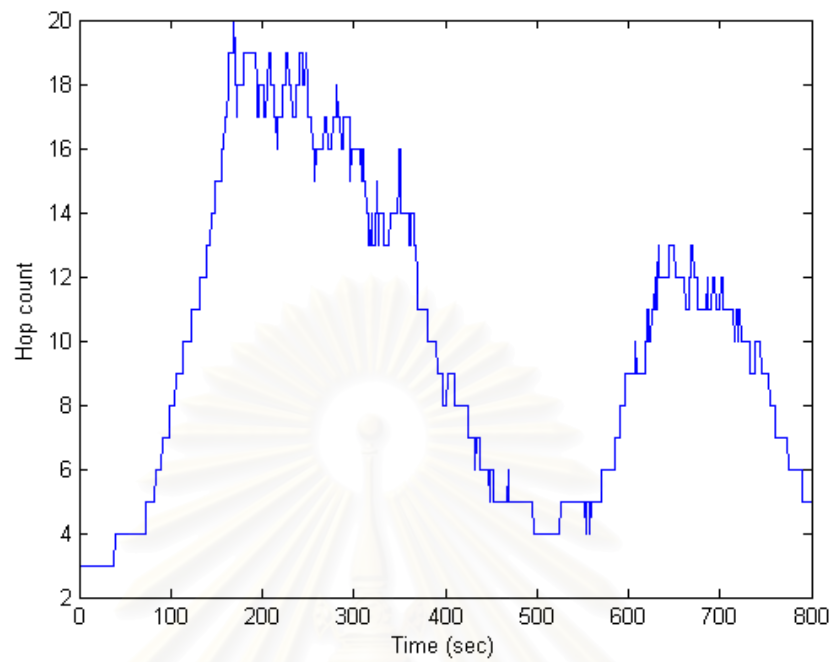
4.5 ผลกระทบจากการเคลื่อนที่ของโนด

การทดลองนี้จะใช้ทั้งระบบทดสอบแบบมีสายและไร้สายในการประเมินผล ในระบบทดสอบแบบมีสายจะเน้นศึกษาผลกระทบที่มีต่อจิตเตอร์จากการเปลี่ยนเส้นทางของโนด ส่วนในระบบทดสอบแบบไร้สายจะเน้นศึกษาผลกระทบจากสภาพแวดล้อมที่เปลี่ยนแปลง โดยจะเปรียบเทียบจิตเตอร์และค่าประวิงเวลาระหว่างกรณีที่โนดมีการเคลื่อนที่ และไม่มีการเคลื่อนที่ ในขณะที่ทอพอโลยีไม่มีการเปลี่ยนแปลง

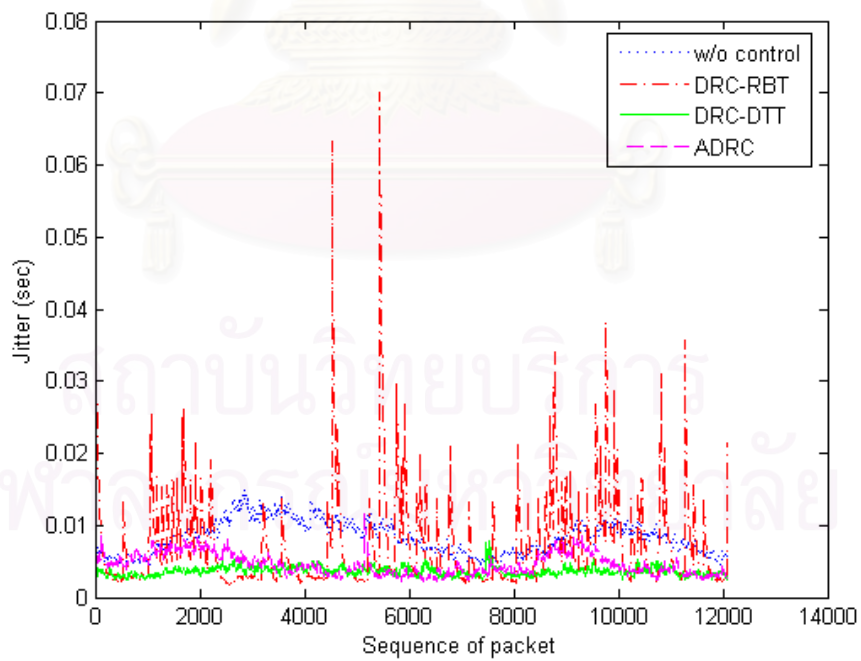
4.5.1 ผลกระทบจากการเปลี่ยนเส้นทางการส่งข้อมูล

การทดลองนี้จะทำในระบบทดสอบแบบมีสาย เนื่องจากสามารถควบคุมการเคลื่อนที่ให้เป็นไปตามที่ต้องการได้ โดยในการทดลองจะประกอบด้วย 30 โนด ซึ่งในที่นี้หมายถึงรถ 30 คันแล่นตามกันตามแบบจำลองการเคลื่อนที่ตามกันของรถดังที่ได้กล่าวไปแล้วในบทที่ 2 $w = 0.02$, $d = 0.5$ วินาที, $c = 2$ มิลลิวินาที. $threshold = 10$, $l = 0.005$ รูปที่ 4.21 แสดงจำนวนช่วงเชื่อมต่อการส่งข้อมูลที่เวลาต่าง ๆ และรูปที่ 4.22-4.25 แสดงจิตเตอร์เฉลี่ยและค่าประวิงเวลาที่วัดได้ของกราฟฟิกแบบอัตราบิดงที่และแบบอัตราบิดแปรผันได้ ตามลำดับ โดยในกราฟฟิกแบบอัตราบิดงที่ ค่าจิตเตอร์เฉลี่ยของกรณีที่ไม่มีการควบคุมใด ๆ, DRC-RBT, DRC-DTT และ ADRC เท่ากับ 8.2, 7.4, 3.6 และ 4.6 มิลลิวินาที ตามลำดับ ส่วนกราฟฟิกแบบอัตราบิดแปรผันได้ จิตเตอร์เฉลี่ยของกรณีที่ไม่มีการควบคุมใด ๆ, DRC-DTT และ ADRC จะมีค่าเท่ากับ 5.8, 2.9 และ 3.2 มิลลิวินาที ตามลำดับ

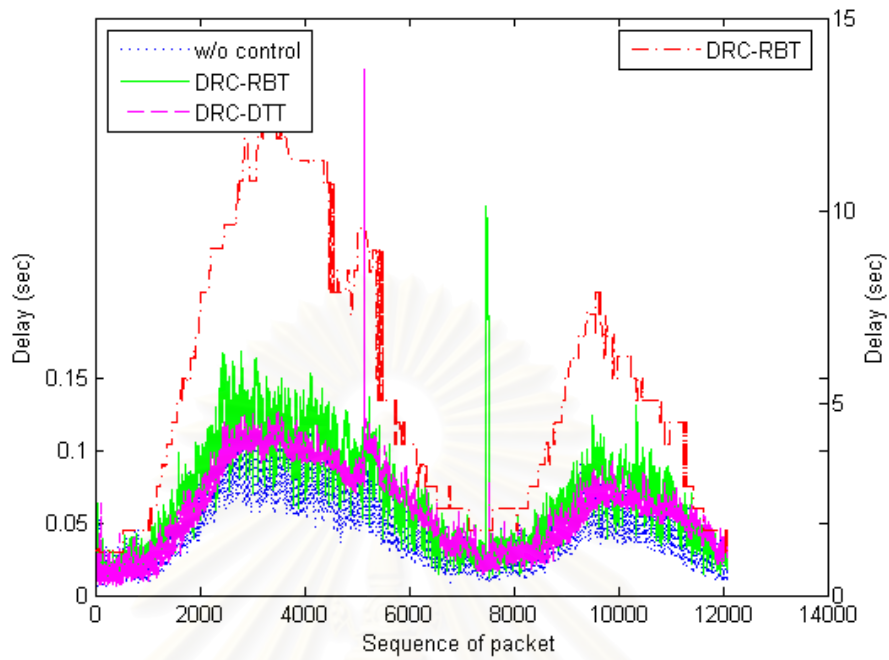
จะเห็นว่าเมื่อมีการเปลี่ยนแปลงเส้นทางเกิดขึ้น วิธี DRC-RBT จากเดิมที่ควบคุมจิตเตอร์ได้ดีที่สุด กลายเป็นทำให้จิตเตอร์มีค่ามากกว่ากรณีที่ไม่มีการควบคุมใด ๆ เนื่องจากในวิธีนี้เหมาะสำหรับการเปลี่ยนเส้นทางที่จำนวนช่วงเชื่อมต่อยังคงเท่าเดิมอยู่ ซึ่งเป็นไปได้ยากในทางปฏิบัติจริง ดังนั้นวิธี DRC-RBT จึงเหมาะสำหรับกรณีที่โนดไม่มีการเคลื่อนที่เท่านั้น ส่วนวิธี ADRC จิตเตอร์ที่โนดปลายทางก็จะมีค่าสูงขึ้นเมื่อเทียบกับกรณีที่ไม่มีการเปลี่ยนเส้นทาง เนื่องจากเมื่อมีการเปลี่ยนเส้นทางจะทำให้จำนวนช่วงเชื่อมต่อที่กลุ่มข้อมูลต้องเดินทางผ่านเปลี่ยนแปลงอยู่ตลอดเวลา ซึ่งส่งผลถึงค่าประวิงเวลาของแต่ละกลุ่มข้อมูลที่โนดระหว่างทางได้รับก็มีค่าไม่แน่นอนด้วย จึงทำให้ค่าประวิงเวลาเฉลี่ยของแต่ละโนดระหว่างใช้ในการตัดสินใจมีค่าไม่แน่นอนเช่นกัน เพราะปกติแล้ววิธี ADRC จะต้องใช้ช่วงระยะเวลาหนึ่งในการปรับค่าประวิงเวลาเฉลี่ยให้มีค่าที่เหมาะสมสำหรับใช้ในการตัดสินใจ แต่สำหรับวิธี DRC-DTT นั้นการเปลี่ยนเส้นทางจะไม่ส่งผลถึงจิตเตอร์เท่าใดนัก เนื่องจากกระบวนการตัดสินใจอ้างอิงจากค่าประวิงเวลาของกลุ่มข้อมูลก่อนหน้าเท่านั้น อย่างไรก็ตาม ทั้งวิธี ADRC และ DRC-DTT ก็ยังสามารถลดจิตเตอร์ลงไปได้มากเมื่อเทียบกับกรณีที่ไม่มีการควบคุมใด ๆ เลย



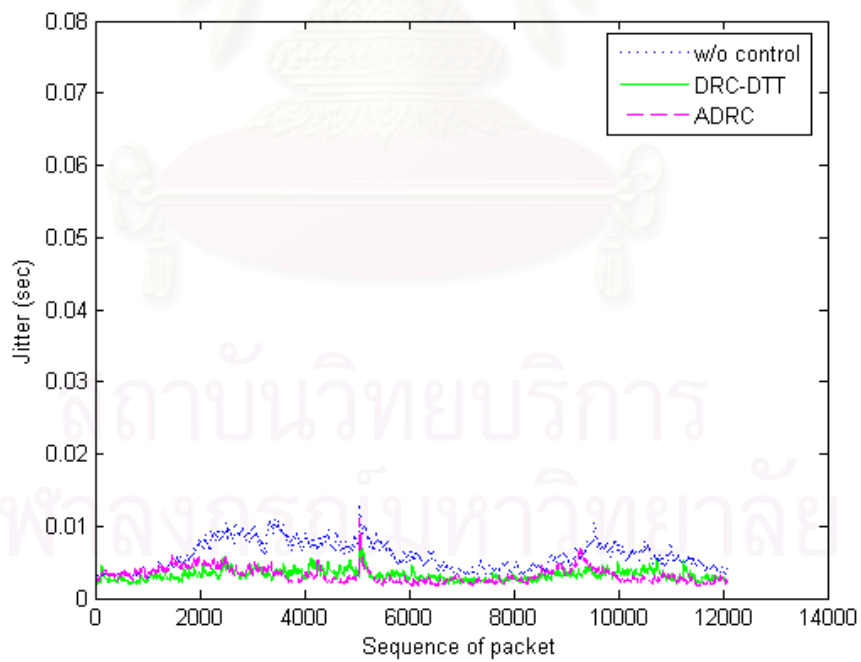
รูปที่ 4.21 จำนวนช่วงเชื่อมต่อที่เวลาต่าง ๆ ของรูปแบบการเคลื่อนที่ที่พิจารณา



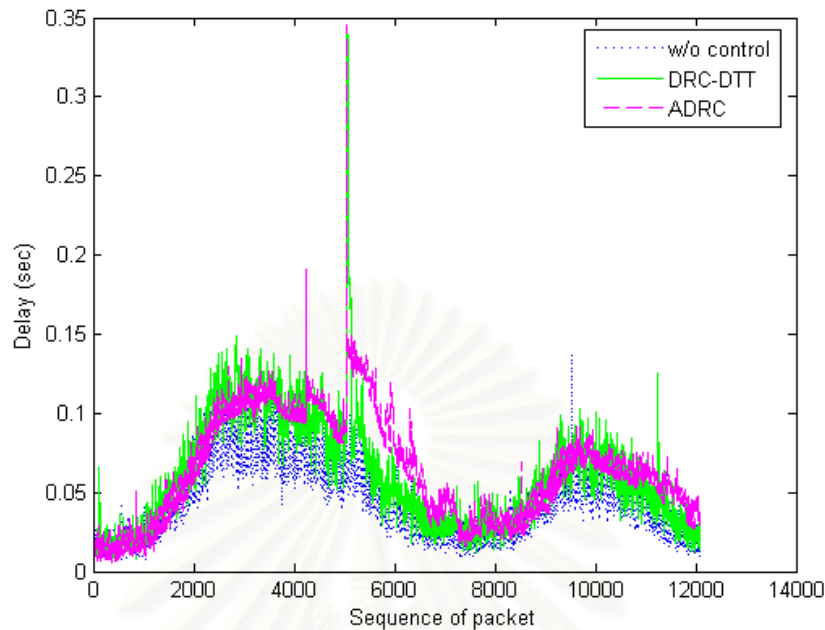
รูปที่ 4.22 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิตคงที่ในระบบทดสอบแบบมีสายในกรณีที่ไม่เคลื่อนที่



รูปที่ 4.23 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิดคงที่ในระบบทดสอบแบบมีสายในกรณีทีโนดเคลื่อนที่



รูปที่ 4.24 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้ในระบบทดสอบแบบมีสายในกรณีทีโนดเคลื่อนที่



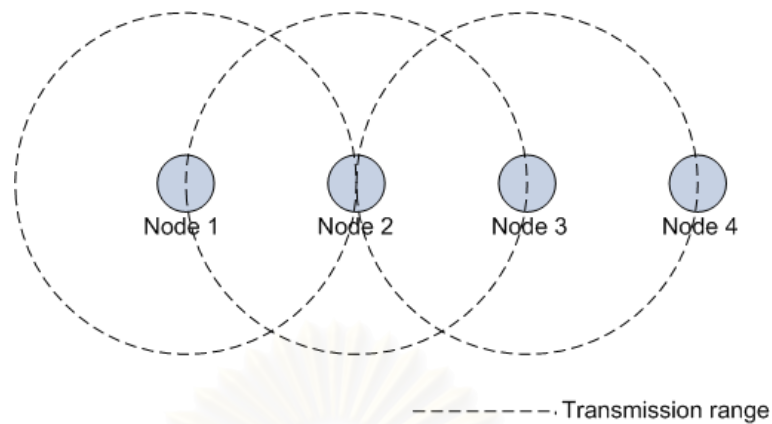
รูปที่ 4.25 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิตแปรผันได้ในระบบทดสอบแบบมีสายในกรณีที่ไม่เคลื่อนที่

4.5.2 ผลกระทบจากสภาพแวดล้อมที่เปลี่ยนแปลง

การทดลองนี้จะทำในระบบทดสอบแบบไร้สายเพื่อวัดผลในสภาพแวดล้อมจริง โดยประกอบไปด้วย 4 โหนดเรียงกัน โหนดต้นทางคือโหนด 1 โหนดปลายทางคือโหนด 4 แต่การส่งข้อมูลจะต้องส่งผ่านโหนด 2 และ 3 เสมอ รูปที่ 4.26 แสดงทอพอโลยีที่พิจารณาในการทดลองนี้ ซึ่งอุปกรณ์ที่ใช้จะประกอบไปด้วยโน้ตบุ๊กคอมพิวเตอร์ 4 เครื่อง แต่ละเครื่องทำงานภายใต้ระบบปฏิบัติการลินุกซ์ Mandrake 10.1 และ Fedora Core 3 ส่วนแผ่นวงจรประสานเครือข่ายไร้สายที่ใช้เป็น Intelpro 2100 และ Intelpro 2200 โดยที่จะทดลองใน 2 รูปแบบ คือโหนดอยู่กับที่ และ โหนดมีการเคลื่อนที่ และทั้งสองรูปแบบจะใช้พารามิเตอร์ในการทดลองดังนี้ $w = 0.02$, $d = 5$ วินาที, $c = 2$ มิลลิวินาที. $threshold = 10$

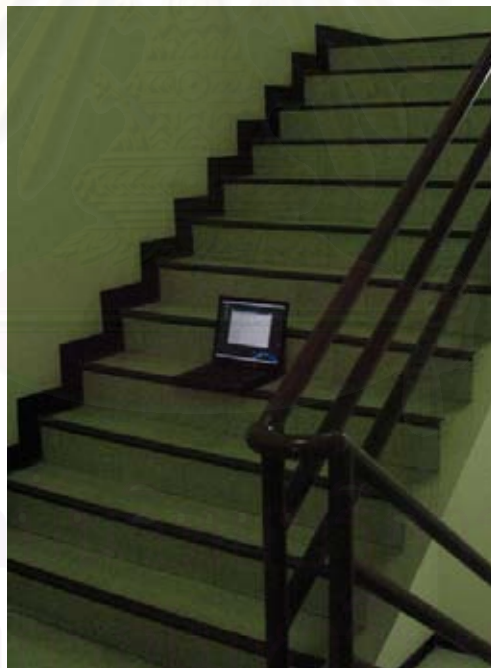
4.5.2.1 สภาพแวดล้อมสำหรับการทดลองที่โหนดไม่เคลื่อนที่

ในการทดลองนี้ใช้โน้ตบุ๊กคอมพิวเตอร์ 4 เครื่องวางไว้ที่บริเวณบันไดทั้งหมด 4 ชั้น เพื่อให้แต่ละโหนดติดต่อกันได้เฉพาะโหนดข้างเคียงเหมือนกับทอพอโลยีที่กำหนดไว้ รูปที่ 4.27 แสดงภาพขณะทำการทดลอง ส่วนรูปที่ 4.28-4.31 แสดงจitterเฉลี่ยและค่าประวิงเวลาที่วัดได้ของกราฟฟิกแบบอัตราบิตคงที่และแบบอัตราบิตแปรผันได้ ตามลำดับ โดยในกราฟฟิกแบบอัตราบิตคงที่ กรณีที่ไม่มีการควบคุมใด ๆ, DRC-RBT, DRC-DTT และ ADRC จะมีค่าจitterเฉลี่ยเท่ากับ 1.5, 0.254, 0.291 และ 1.2 มิลลิวินาที ตามลำดับ และค่า jitter free delay ซึ่งเป็นค่าประวิงเวลานับตั้งแต่กลุ่มข้อมูลถูกส่งออกมาจากโหนดต้นทางจนกระทั่งนำออกแสดงผล เท่ากับ 0.463, 1.126,



รูปที่ 4.26 ทอพอโลยีที่พิจารณาในระบบทดสอบแบบไร้สาย

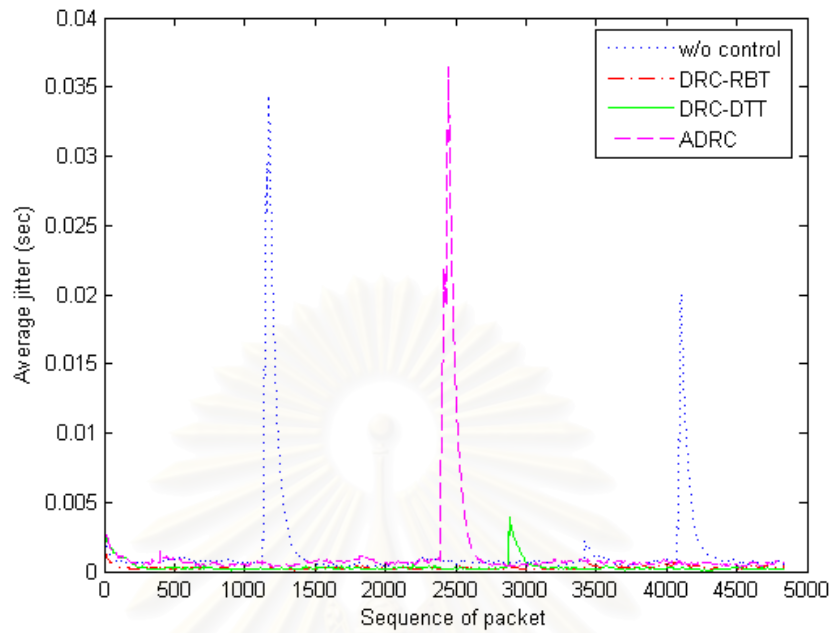
0.406 และ 0.476 วินาที ตามลำดับ ส่วนกราฟฟิกแบบอัตราบิตแปรผันได้ กรณีที่ไม่มีการควบคุมใด ๆ, DRC-DTT และ ADRC จะมีค่าจitterเฉลี่ยเท่ากับ 2.9, 0.471 และ 2.2 มิลลิวินาที ตามลำดับ และ jitter free delay เท่ากับ 0.444, 0.666 และ 0.429 วินาที ตามลำดับ



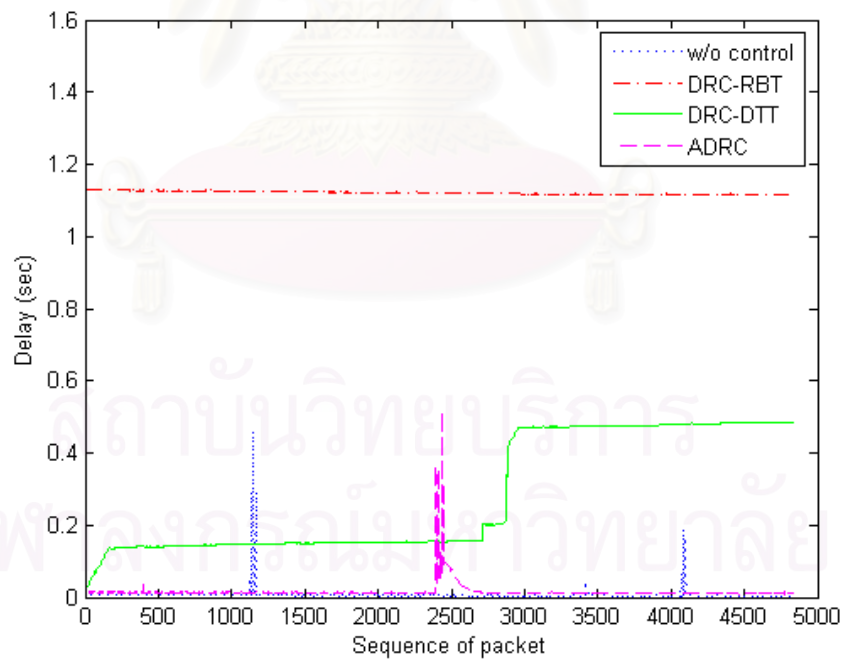
รูปที่ 4.27 ภาพขณะทดลองในระบบทดสอบแบบไร้สายกรณีที่โน้ตไม่เคลื่อนที่

4.5.2.2 สภาพแวดล้อมสำหรับการทดลองที่โน้ตมีการเคลื่อนที่

การทดลองนี้จะใช้โน้ตบุ๊คคอมพิวเตอร์วางอยู่บนรถ 4 คัน โดยที่รถทั้ง 4 คันจะเล่นตามกันไปเรื่อย ๆ ดังที่ได้กล่าวไปแล้วในบทที่ 3 ซึ่งสภาพแวดล้อมระหว่างการทดลองจะเปลี่ยนแปลงตลอดเวลา รูปที่ 4.32 แสดงภาพขณะ

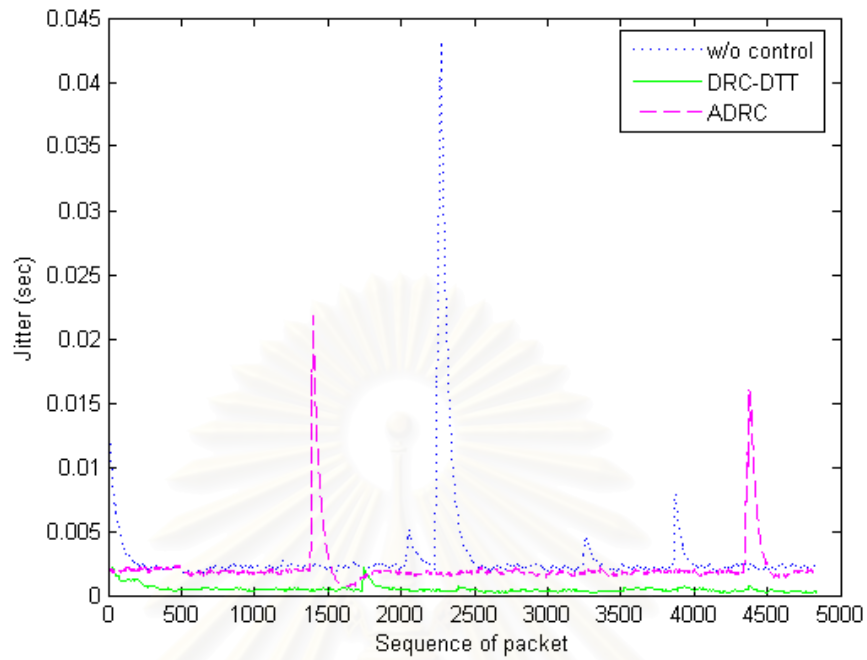


รูปที่ 4.28 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิตคงที่ในระบบทดสอบแบบไร้สายที่ไม่มีการเคลื่อนที่

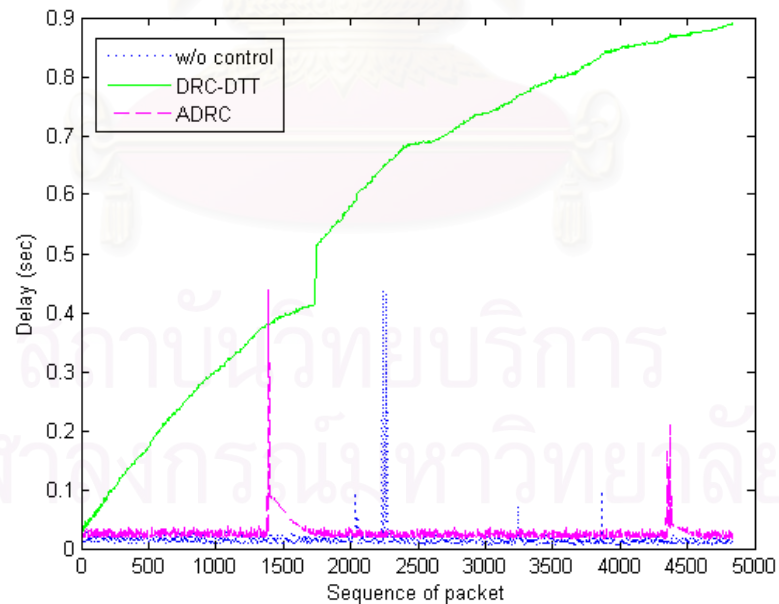


รูปที่ 4.29 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิตคงที่ในระบบทดสอบแบบไร้สายที่ไม่มีการเคลื่อนที่

ทำการทดลอง ส่วนรูปที่ 4.33-4.36 แสดงจิตเตอร์เฉลี่ยและค่าประวิงเวลาที่วัดได้ของกราฟฟิกแบบอัตราบิตคงที่ และแบบอัตราบิตแปรผันได้ ตามลำดับ โดยในกราฟฟิกแบบอัตราบิตคงที่ กรณีที่ไม่มีการควบคุมใด ๆ, DRC-RBT,



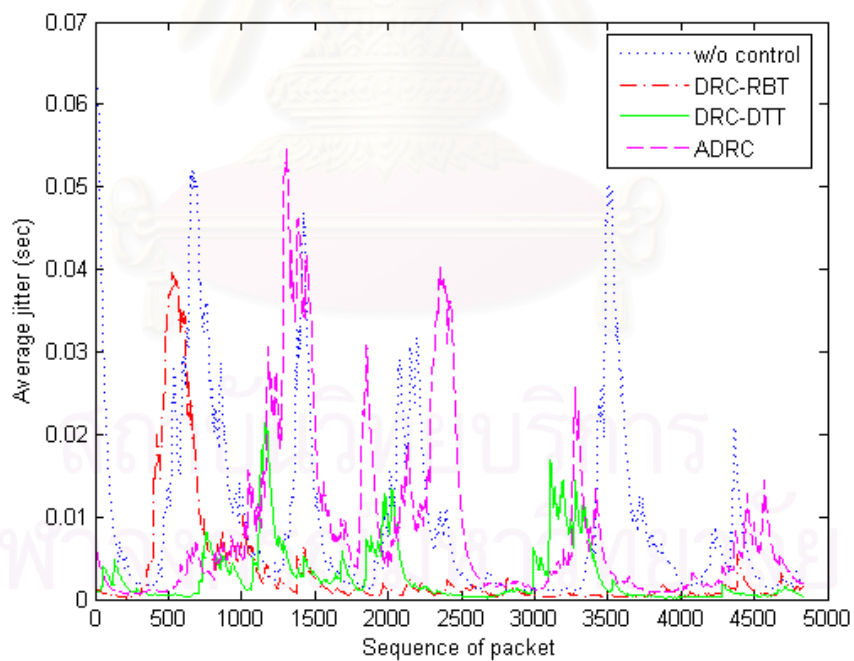
รูปที่ 4.30 จิตเตอร์เฉลี่ยของทราฟฟิกที่มีอัตราบิตแปรผันได้ในระบบทดสอบแบบไร้สายที่ไม่มีการเคลื่อนที่



รูปที่ 4.31 ค่าประวิงเวลาของทราฟฟิกที่มีอัตราบิตแปรผันได้ในระบบทดสอบแบบไร้สายที่ไม่มีการเคลื่อนที่

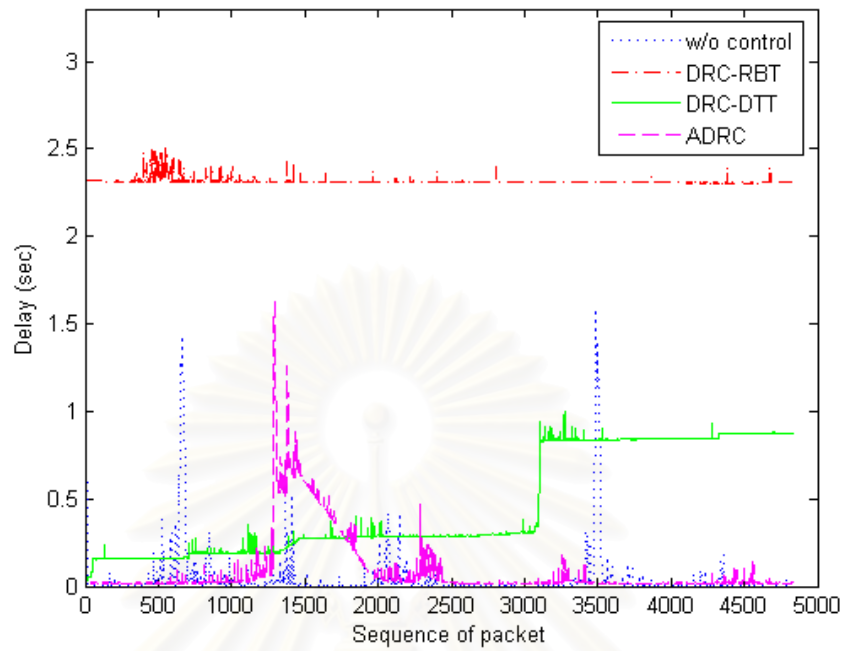


รูปที่ 4.32 ภาพขณะทดลองในระบบทดสอบแบบไร้สายกรณีที่โนดเคลื่อนที่โดยใช้รถ 4 คันขับตามกัน

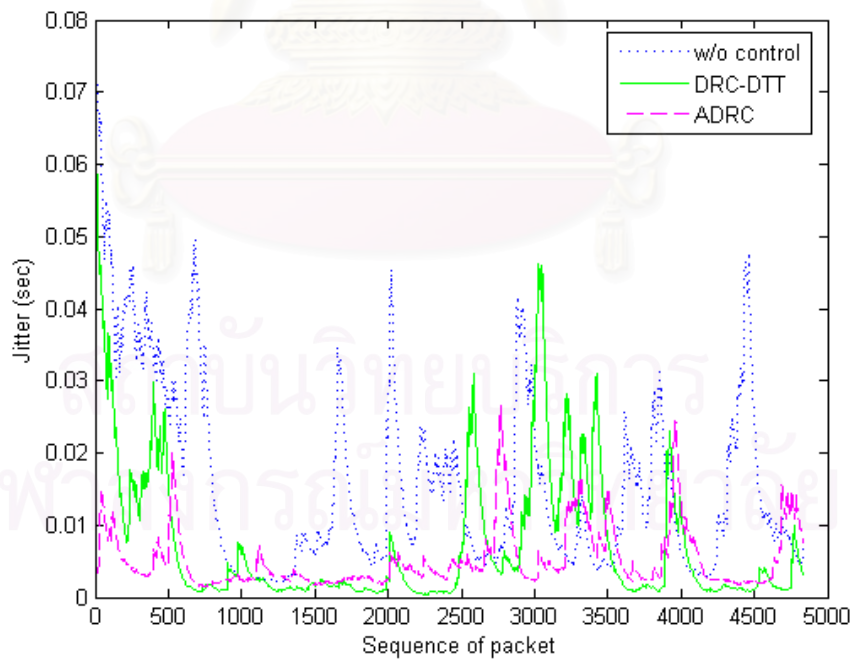


รูปที่ 4.33 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิตคงที่ในระบบทดสอบแบบไร้สายที่โนดเคลื่อนที่

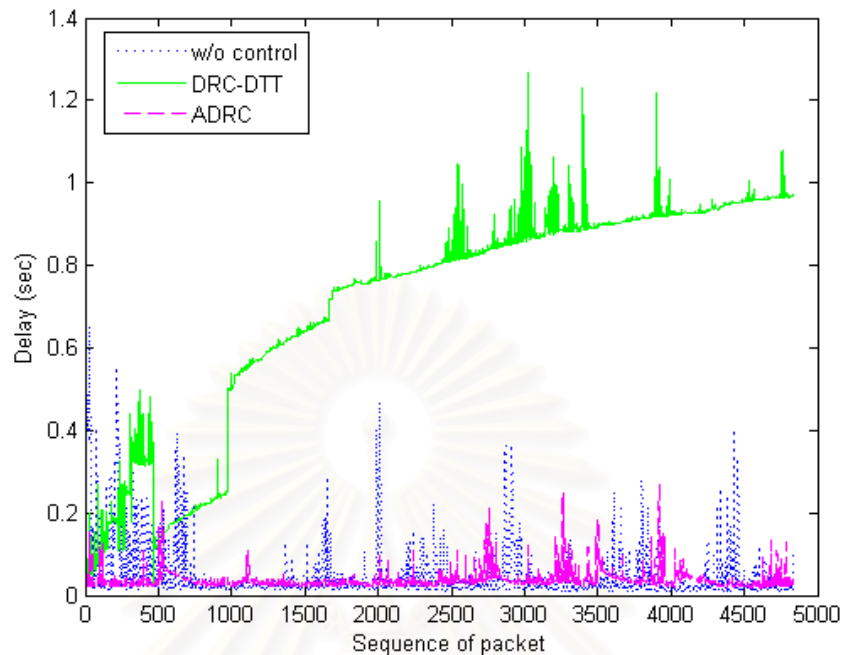
DRC-DTT และ ADRC จะมีค่าจิตเตอร์เฉลี่ยเท่ากับ 9.4, 2.9, 2.8 และ 8.1 มิลลิวินาที ตามลำดับ และ jitter free delay เท่ากับ 0.556, 2.473, 0.673 และ 0.383 วินาที ตามลำดับ ส่วนกราฟฟิกแบบอัตราบิตแปรผันได้



รูปที่ 4.34 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิดคงที่ในระบบทดสอบแบบไร้สายที่โนดเคลื่อนที่



รูปที่ 4.35 จิตเตอร์เฉลี่ยของกราฟฟิกที่มีอัตราบิดแปรผันได้ในระบบทดสอบแบบไร้สายที่โนดเคลื่อนที่



รูปที่ 4.36 ค่าประวิงเวลาของกราฟฟิกที่มีอัตราบิตแปรผันได้ในระบบทดสอบแบบไร้สายที่โนดเคลื่อนที่

กรณีที่ไม่มีการควบคุมใด ๆ, DRC-DTT และ ADRC จะมีค่าจัตเตอร์เฉลี่ยเท่ากับ 14.7, 6.7 และ 5.4 มิลลิวินาที ตามลำดับ และ jitter free delay เท่ากับ 0.486, 0.94 และ 0.233 วินาที ตามลำดับ

4.5.2.3 ผลการทดลองและการวิเคราะห์

เมื่อพิจารณาที่ค่าเฉลี่ยของจัตเตอร์ในแต่ละวิธีที่ทดลองเปรียบเทียบกันแล้ว จะเห็นว่าค่าที่ได้ไม่แตกต่างกันมากนัก เนื่องจากการทดลองในระบบไร้สายจริง ๆ มีข้อจำกัดอยู่หลายอย่าง ทำให้ไม่สามารถใช้เวลาในการทดลองได้นานเพียงพอที่จะนำมาเปรียบเทียบกันได้ ดังนั้นการประเมินผลในการทดลองนี้จะเน้นในลักษณะของการวิเคราะห์ว่าแต่ละวิธีมีความสามารถในการจัดการได้ดีเพียงใดเมื่อต้องประสบกับสถานการณ์ต่าง ๆ

จากผลการทดลองทั้งแบบที่โนดไม่เคลื่อนที่และแบบโนดเคลื่อนที่ เมื่อเปรียบเทียบกันแล้วจะเห็นว่าเมื่อมีการเคลื่อนที่ทำให้สภาพแวดล้อมระหว่างการทดลองเปลี่ยนไป จะส่งผลให้ค่าประวิงเวลาที่มีความแปรผันสูงขึ้นด้วย แต่ในจังหวะที่ค่าประวิงเวลาพุ่งสูงขึ้นแล้วลดลงอย่างรวดเร็ว หากไม่มีการจัดการใด ๆ เลย จัตเตอร์ในจังหวะนั้นก็จะมีค่าสูง แต่จากกราฟจะเห็นว่า วิธี ADRC จะจัดการให้ค่าประวิงเวลาค่อย ๆ ลดลงมาเป็นลำดับ ซึ่งส่งผลให้จัตเตอร์มีค่าสม่ำเสมอ ส่วนในวิธี DRC-DTT จะเห็นว่าจัตเตอร์มีค่าสูงเพียงในช่วงที่มีกลุ่มข้อมูลที่มีค่าประวิงเวลาสูงเกิดขึ้น แต่หลังจากนั้นก็พยายามทำให้ค่าประวิงเวลาของกลุ่มข้อมูลที่ตามมามีค่าสม่ำเสมอ เพื่อให้จัตเตอร์มีค่าต่ำ โดยหากเปรียบเทียบจัตเตอร์และค่าประวิงเวลาระหว่างกราฟฟิกแบบอัตราบิตคงที่และอัตราบิตแปรผันได้แล้ว จะเห็นว่าในกราฟฟิกแบบอัตราบิตแปรผันได้จะมีจัตเตอร์น้อยกว่า ในขณะที่ค่าประวิงเวลามีค่าสูงกว่า เนื่องจากรูปแบบของ

กราฟฟิกแบบอัตราบิดแปรผันได้นี้มีความไม่แน่นอนของค่าประวิงเวลาในแต่ละกลุ่มข้อมูลมากกว่า จึงมีโอกาที่กลุ่มข้อมูลจะถูกห้วงเวลาออกไปมากกว่าด้วย ทำให้ค่าประวิงเวลามีค่าสูงกว่าและจิตเตอร์มีค่าน้อยกว่ากราฟฟิกแบบอัตราบิดคงที่



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 บทสรุป

วิทยานิพนธ์ฉบับนี้ได้นำเสนอกลไกควบคุมแบบเวลาจริงสำหรับการรับส่งข้อมูลสื่อผสมในเครือข่ายแอตชอก โดยมีจุดประสงค์เพื่อให้จัดเตอร์ที่โนดปลายทางมีค่าน้อยลง ขณะที่ค่าประวิงเวลาอยู่ในช่วงที่สามารถยอมรับได้ และได้ทำการทดสอบกลไกที่นำเสนอในระบบทดสอบ 2 รูปแบบ คือแบบไร้สายในกรณีเครือข่ายมีขนาดเล็ก รวมถึงไม่มีการเปลี่ยนแปลงทอพอโลยี ส่วนในแบบมีสายจะทดสอบในกรณีที่เครือข่ายมีขนาดใหญ่ขึ้น และใช้แบบจำลองการเคลื่อนที่ตามกันของรถ เพื่อให้มีสถานการณ์ใกล้เคียงกับการนำไปใช้งานจริง

กลไกการควบคุมแบบเวลาจริงที่นำเสนอในวิทยานิพนธ์นี้ใช้แนวคิดที่ให้โนดระหว่างทางมีส่วนช่วยในการจัดการควบคุมจัดเตอร์ ซึ่งประกอบไปด้วย 3 วิธี คือ DRC-RBT, DRC-DTT และ ADRC โดยการประเมินผลจะมุ่งไปที่การศึกษาผลกระทบอันเนื่องมาจากปัจจัยต่าง ๆ ต่อจัดเตอร์ และหาสถานการณ์ที่เหมาะสมในการนำแต่ละกลไกการควบคุมที่นำเสนอไปใช้งาน ซึ่งสามารถสรุปได้ดังนี้

- DRC-RBT

วิธีนี้สามารถใช้ได้กับทราฟฟิกที่มีอัตราบิตคงที่เท่านั้น โดยเหมาะสำหรับใช้กับเครือข่ายที่โนดไม่เคลื่อนที่ และเป็นระบบที่ให้ความสำคัญกับจัดเตอร์เป็นหลัก โดยไม่สนใจค่าประวิงเวลามากนัก เนื่องจากวิธีนี้จะทำให้ค่าประวิงเวลาจะมีค่าสูงขึ้นมาก

- DRC-DTT

วิธีนี้สามารถใช้ได้กับทั้งทราฟฟิกแบบอัตราบิตคงที่และอัตราบิตแปรผันได้ โดยจะให้จัดเตอร์ที่มีค่าต่ำในกรณีที่เครือข่ายมีอัตราการสูญเสียกลุ่มข้อมูลไม่มากนัก ในขณะที่ค่าประวิงเวลาก็ไม่สูงขึ้นจนเกินไป แต่มีข้อเสียคือระบบจะไม่ค่อยเสถียร เนื่องจากค่าประวิงเวลาที่มีค่าสูงของหนึ่งกลุ่มข้อมูลจะส่งกระทบถึงค่าประวิงเวลาของกลุ่มข้อมูลอื่น ๆ ที่ตามมาทั้งหมด

- ADRC

วิธีนี้สามารถใช้ได้กับทั้งทราฟฟิกแบบอัตราบิตคงที่และอัตราบิตแปรผันได้เช่นเดียวกับวิธี DRC-DTT โดยในกรณีที่เครือข่ายมีอัตราการสูญเสียกลุ่มข้อมูลน้อย วิธีนี้จะให้จัดเตอร์ที่มีค่าสูงกว่า DRC-DTT ในขณะที่มีค่าประวิงเวลาน้อยกว่า แต่หากเป็นเครือข่ายที่มีอัตราการสูญเสียกลุ่มข้อมูลมาก วิธีนี้จะจัดการได้ดีกว่า DRC-DTT มาก ในขณะที่ค่าประวิงเวลาก็มีค่าน้อยกว่าด้วย นอกจากนี้ปัญหาจากกลุ่มข้อมูลที่มีค่าประวิงเวลาสูงผิดปกติที่เกิดใน DRC-DTT ยังไม่มีผลต่อวิธี ADRC นี้ด้วย แต่สำหรับกรณีที่โนดมีการเคลื่อนที่ กระบวนการทำงานของวิธีนี้อาจจะตัดสินใจได้ไม่ทันในช่วงแรกของการเปลี่ยนเส้นทาง เนื่องจากค่าประวิงเวลาเฉลี่ยของแต่ละโนดระหว่างทางใช้

อ้างอิงไม่เหมาะสมที่จะนำมาเปรียบเทียบ ดังนั้นจึงต้องใช้เวลาระยะหนึ่งในการปรับเปลี่ยนให้เข้าที่ อย่างไรก็ตามวิธีนี้ก็ยังคงช่วยลด jitter ลงได้มาก

จากบทสรุปของทั้ง 3 วิธีที่กล่าวมาทั้งหมด สามารถสรุปรวมได้ดังตารางที่ 5.1 ซึ่งจะเห็นว่าแต่ละวิธีก็เหมาะสมกับสถานการณ์ที่ต่างกัน ดังนั้นในการนำไปใช้งานจริงนั้นจะขึ้นอยู่กับว่าต้องการนำไปใช้กับสถานการณ์ในรูปแบบใด

ตารางที่ 5.1 สรุปเปรียบเทียบ jitter และค่าประวิงเวลาของกลไกควบคุมทั้ง 3 วิธีในแต่ละสถานการณ์

รูปแบบสถานการณ์		จิตเตอร์	ค่าประวิงเวลา
static	lossless	CBR	$ADRC < DRC-DTT < DRC-RBT$
		VBR	$ADRC < DRC-DTT$
	lossy	CBR	$ADRC < DRC-DTT < DRC-RBT$
		VBR	$ADRC < DRC-DTT$
moving	lossless	CBR	$ADRC < DRC-DTT < DRC-RBT$
		VBR	$ADRC < DRC-DTT$
	lossy	CBR	$ADRC < DRC-DTT < DRC-RBT$
		VBR	$ADRC < DRC-DTT$

5.2 ข้อเสนอแนะ

หัวข้อที่ควรศึกษาและวิจัยต่อไปในอนาคตคือ

1. การกำหนดรูปแบบการแจกแจงของค่าประวิงเวลาในระบบทดสอบแบบมีสายให้เหมาะสมมากขึ้น

ถึงแม้การแจกแจงของค่าประวิงเวลาที่ใช้ในวิทยานิพนธ์นี้จะสามารถใช้ในการทดสอบเพื่อเปรียบเทียบและประเมินผลแต่ละวิธีได้ แต่ค่าที่ใช้ก็อาจจะยังไม่ถูกต้องตามสถานการณ์จริงเท่าที่ควร ดังนั้นหากสามารถพัฒนาให้มีรูปแบบที่ใกล้เคียงกับความเป็นจริงมากยิ่งขึ้น ก็จะทำให้ระบบทดสอบมีประสิทธิภาพดียิ่งขึ้น

2. ระบบปฏิบัติการที่ใช้ในระบบทดสอบ

ระบบปฏิบัติการที่ใช้ในวิทยานิพนธ์นี้เป็นระบบปฏิบัติการลินุกซ์แบบทั่วไป จึงทำให้ยังมีปัญหาจากค่าประวิงเวลาประมวลผล (processing time) ที่เกิดขึ้นเมื่อเรียกใช้บางฟังก์ชันของระบบปฏิบัติการอยู่ เช่น ฟังก์ชัน sleep ซึ่งอาจส่งผลถึงค่าที่ทำการวัดในระบบทดสอบได้ เนื่องจาก jitter และค่าประวิงเวลาที่วัดก็ถือว่าอยู่ในระดับที่น้อยมาก ดังนั้นหากเลือกใช้ระบบปฏิบัติการที่สามารถตอบสนองแบบเวลาจริงได้ ก็จะทำให้ระบบทดสอบมีประสิทธิภาพดียิ่งขึ้น

3. การขยายขนาดของระบบทดสอบแบบมีสาย

ระบบทดสอบแบบมีสายในวิทยานิพนธ์นี้ใช้เครื่องคอมพิวเตอร์เพียงเครื่องเดียว ซึ่งก็สามารถรองรับการทดสอบในเครือข่ายที่มีโหนดจำนวนมากได้ แต่หากทำการพัฒนาต่อไปโดยนำกิกะบิตสวิตช์เข้ามาเชื่อมต่อ ก็จะทำให้ขยายขนาดของเครือข่ายที่สามารถจำลองขึ้นในระบบทดสอบออกไปได้มากยิ่งขึ้น



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

- [1] The Internet Engineering Task Force (IETF).d Moblie Ad-hoc Network (MANET) Working Group. Available from: <http://www.ietf.org/html.charters/manet-charter.html>. (18 November 2004)
- [2] P. Horrell. Intelligence: Behold the All-Seeing, Self-Parking, Safety-Enforcing, Networked Automobile. Popular Science, Special section: The Future of The Car, Available from: <http://www.popsci.com/popsci/futurecar/article/0,20967,679165,00.html>. (September 2004)
- [3] M. Rabel, A. Schmeiser and H. P. Grobmann. Ad-hoc In-Car Networking Concept. Intelligent Transportation Systems. (September 2005): 363-368
- [4] C. Perkins and E. Royer. Ad-hoc On-Demand Distance Vector (AODV) Routing. RFC 3561. (July 2003)
- [5] D. B. Johnson, D. A. Maltz and Y. C. Hu. The Dynamic Source Routing Protocol for Mobile Ad-hoc Networks (DSR). Internet Draft, IETF Mobile Ad-hoc networks. draft-ietf-manet-dsr-10.txt. (July 2004)
- [6] C. Chaudet , D. Dhoutaut and I. G. Lassous. Performance Issues with IEEE 802.11 in Ad-hoc Networking. IEEE Communications Magazines. 43, 7. (July 2005): 110-116
- [7] S. Chakrabarti and A. Mishra. QoS Issues in Ad-hoc Wireless Networks. IEEE Communications Magazines. 39, 2. (February 2001): 142-148
- [8] CMU Monarch Group. ns-2. Available from: <http://www.isi.edu/nsnam/ns/>. (11 July 2004)
- [9] GloMoSim. Available from: <http://pcl.cs.ucla.edu/projects/glomosim/>. (11 July 2004)
- [10] H. Zhang and Z. Jiang. Theoretic Analysis of Two Broadcasting Protocols in Mobile Ad-hoc Networks. American Control Conference 2004. 4. (June 2004): 3599-3604
- [11] D. A. Maltz and J. Broch. Lessons from a Full-Scale Multihop Wireless Ad-hoc Network Testbed. IEEE Personal Communications. (February 2001): 8-15

- [12] H. Lundgren, D. Lundgren, J. Nielsen, E. Nordstrom and C. Tschudin. A Large-scale Testbed for Reproducible Ad-hoc Protocol Evaluations. IEEE Wireless Communications and Networking Conference (WCNC2002). 1. (March 2002): 412-418
- [13] S. Sanghani, T. X. Brown, S. Bhandare and S. Doshi. EWANT: The Emulated Wireless Ad-hoc Network Testbed. IEEE Wireless Communications and Networking Conference (WCNC). (March 2003)
- [14] Y. Zhang and W. Li. An Integrated Environment for Testing Mobile Ad-hoc Networks. Proceedings of 3rd ACM International Symposium on Mobile Ad-hoc Networking and Computing (MobiHoc'02). (June 2002): 104-111
- [15] M. Puzar and T. Plagemann. NEMAN: A Network Emulator for Mobile Ad-hoc Networks. Telecommunications 2005, ConTEL 2005. 1. (June 2005): 155-161
- [16] P. Mahadevan, A. Rodriguez, D. Becker and A. Vahdat. MobiNet: A Scalable Emulation Infrastructure for Ad-hoc and Wireless Networks. UCSD Technical Report CS2004-0792. (2004)
- [17] Y. Ito, S. Tasaka and Y. Fukuta. Psychometric Analysis of The Effect of End-to-End Delay on User-Level QoS in Live Audio-Video Transmission. IEEE International Conference on Communication, ICC2004, 4. (June 2004): 2214-2220
- [18] S. P. Chan and C. W. Kok. Bitrate Adaptation Flow Control for Multimedia-on-Demand. IEEE International Conference (ICC). 4. (April 2002): 2503-2507
- [19] R. W. Rothery. Car Following Models. Traffic Flow Theory. Transportation Research Board, Special Report. Chap 4. (1992): 165
- [20] Video Codec for Audio visual Services at px 64 kb/s ITU-T Recommendation H.261. (March 1993)
- [21] Video Coding of Low Bitrate Communication ITU-T Recommendation H.263. (March 1996)
- [22] Information technology - Coding of audio-visual objects - Part 10: Advanced Video Coding. ISO/IEC 14496-10
- [23] Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s. ISO/IEC 11172

- [24] Information technology - Generic coding of moving pictures and associated audio information.
ISO/IEC 13818
- [25] Information technology - Coding of audio-visual objects. ISO/IEC 14496
- [26] S. Chan, C. Kok and A. K. Wong. Multimedia Streaming Gateway With Jitter Detection.
IEEE Transactions on Multimedia. 7, 3. (June 2005)
- [27] IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. (November 1997)
- [28] J. Hunter, V. Witana and M. Antoniadou. A Review of Video Streaming over The Internet.
DSTC Technical Report, TR97-10. (August 1997)
- [29] T. Camp, J. Boleng and V. Davies. A Survey of Mobility for Ad-hoc Network Research.
Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad-hoc Networking: Research, Trends and Applications. (2002): 483-502
- [30] M. Sanchez and P. Manzoni. Anejos: A Java Based Simulator for Ad-hoc Networks. Future Generation Computer systems. (2001): 573–583
- [31] D. B. Johnson and D. A. Maltz. Dynamic Source Routing in Ad-hoc Wireless Networks.
Mobile Computing. (1996): 153–181
- [32] D. B. Johnson, D. A. Maltz and Y. C. Hu. The Dynamic Source Routing Protocol for Mobile Ad-hoc Networks (DSR). Internet Draft, IETF Mobile Ad-hoc networks. draft-ietf-manet-dsr-10.txt. (July 2004)
- [33] C. Perkins. IP Mobility Support. RFC 2002. (October 1996)



ภาคผนวก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทความทางวิชาการจากวิทยานิพนธ์

- [1] พนิดา วีระวุฒิพล, S. B. Wibowo, และ เซวาน์ดิศ อัสวกุล. ระบบทดสอบโครงข่ายแอดฮอกไร้สายสำหรับประเมินการควบคุมจิตเตอร์ในการส่งทราฟฟิกแบบอัตราบิตคงที่. การประชุมวิชาการทางวิศวกรรมไฟฟ้า (EECON) ครั้งที่ 28.



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ระบบทดสอบโครงข่ายแอดฮอคไร้สายสำหรับประเมินการควบคุมจitterในการส่งทราฟฟิกแบบอัตราบิดงที่

Wireless Ad-hoc Network Testbed for Jitter Control Evaluation in CBR Traffic Transmission

พนิดา วีระวุฒิพล, Sigit Basuki Wibowo และ ชาอดิต อัสวกุล
ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ถนนพญาไท เขตปทุมวัน กรุงเทพฯ 10330

E-mail: panida.ve@student.chula.ac.th, sigit.b@student.chula.ac.th, chaodit.a@chula.ac.th

บทคัดย่อ

บทความนี้นำเสนอการ สร้างระบบทดสอบ (testbed) เพื่อประเมินผลจากการควบคุมจitterในโครงข่ายแอดฮอคไร้สาย ซึ่งใช้หลักการ ประวิง เวลา ก่อน ที่ โหนด ระหว่าง ทาง จะ ส่ง กลุ่ม ข้อมูล (packet) ที่รับมาต่อออกไป เพื่อให้ค่าจitterที่โหนดปลายทางมีค่าน้อยที่สุด โดยโครงข่ายที่พิจารณาประกอบด้วย 4 โหนด และเชื่อมต่อกันตามมาตรฐาน 802.11b ผ่านทางซ็อกเก็ต (socket) UDP ซึ่งในบทความนี้ได้แสดงผลจากการทดลองในระบบทดสอบ พร้อมทั้งผลจากการจำลองโดยใช้โปรแกรมคอมพิวเตอร์ OMNET++ เพื่อเปรียบเทียบกันด้วย โดยผลที่ได้จากทั้ง 2 การทดลองสามารถยืนยันได้ว่า วิธีการควบคุมจitterที่นำเสนอถึงแม้ว่าจะทำให้ค่าประวิงเวลามีค่าเพิ่มขึ้น แต่ก็ทำให้ค่าจitterลดลงไปมากด้วยเช่นกัน

คำสำคัญ: โครงข่ายแอดฮอคไร้สาย, ระบบทดสอบ, การควบคุมจitter

Abstract

This paper proposes testbed implementation for evaluation of distributed jitter control in wireless ad-hoc network. The jitter control algorithm works by adding some delays of packets at intermediate node before being forwarded to the next node. It is intended to minimize end-to-end jitter. This testbed consists of four nodes connected by 802.11b standard and uses UDP socket to send packets. The testbed result is also compared to that of OMNET++ simulation. Both test platforms confirm that the distributed jitter control algorithm, while increasing the average delay, can decrease end-to-end jitter significantly.

Keywords: wireless ad-hoc network, testbed, jitter control

1 บทนำ

ทุกวันนี้เทคโนโลยีไร้สายได้มีการพัฒนาและเข้ามามีบทบาทกับมนุษย์มากขึ้น ไม่ว่าจะเป็นโครงข่ายเซลลูลาร์ โครงข่ายแลนไร้สาย ที่ใช้งานกันอยู่อย่างแพร่หลายในปัจจุบัน รวมถึงโครงข่ายแอดฮอคที่นักวิจัยจำนวนมากกำลังให้ความสนใจ [1, 2] ที่ผ่านมามีงานเกี่ยวกับโครงข่ายแอดฮอคที่อยู่ระหว่างการกำหนดมาตรฐานและงานวิจัยที่สำคัญ เช่น โพรโทคอลจัดเส้นทาง [3, 4] การเข้าถึงช่องสัญญาณ การควบคุมคุณภาพการให้บริการ [5] ซึ่งมีวิธีการต่าง ๆ ที่นักวิจัยทำการนำเสนอ แต่ปัญหาที่สำคัญคือ จะสามารถทดสอบวิธีการนั้น ๆ ว่ามีประสิทธิภาพและเหมาะสมกับการใช้งานในระบบจริงได้อย่างไร

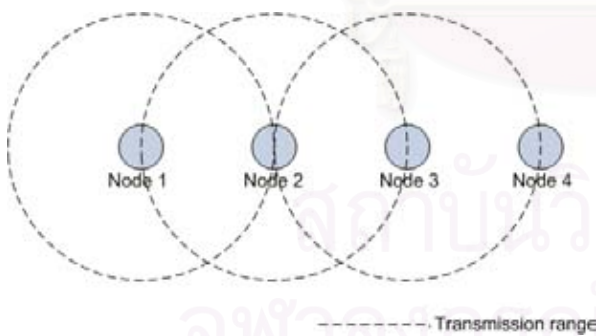
การทดสอบประสิทธิภาพของวิธีการต่าง ๆ ที่นักวิจัยนำเสนอ นั้น สามารถแบ่งได้เป็น 3 วิธีหลัก ๆ คือ การจำลองโดยใช้โปรแกรมคอมพิวเตอร์ [6, 7, 8], การวิเคราะห์ทางคณิตศาสตร์ และการสร้างระบบทดสอบ [9, 10, 11] ซึ่งสองวิธีแรกเป็นวิธีที่งานวิจัยส่วนใหญ่นิยมใช้กัน เนื่องจากวิธีแรกเป็นวิธีที่ทำได้ง่าย และสามารถทำการจัดการระบบให้เป็นอย่างที่ต้องการได้ ซึ่งรวมถึงการกำหนดหรือตัดบางพารามิเตอร์ออกเพื่อให้ระบบจัดการได้ง่ายขึ้น เช่นเดียวกับในวิธีที่สองที่ต้องใช้ การตั้งสมมติฐานเหมือนกันด้วยเหตุนี้จึงทำให้สองวิธีนี้อาจจะให้ผลที่คลาดเคลื่อนจากระบบจริงได้ ดังนั้นในบทความนี้จึงได้เลือกใช้การสร้างระบบทดสอบ เพราะถึงแม้ว่าจะมีความซับซ้อนมากกว่าและต้องใช้ต้นทุนสูงกว่า แต่ก็สามารถจำลองสภาพแวดล้อมต่าง ๆ ได้ใกล้เคียงกับระบบจริงมากกว่าเช่นกัน

ในการให้บริการสื่อผสม (multimedia) ในปัจจุบันถือว่าเป็นที่นิยมอย่างมาก ไม่ว่าจะเป็นการเรียกดูข้อมูลภาพเคลื่อนไหว หรือการประชุมผ่านทางเครือข่าย โดยคุณภาพของการให้บริการก็ขึ้นอยู่กับหลาย ๆ ปัจจัย ทั้งการสูญหายของกลุ่มข้อมูล ค่าประวิงเวลา และจitter ซึ่งถือว่ามีผลสำคัญมาก เนื่องจากหากจitterมีค่ามาก จะทำให้ข้อมูลภาพหรือเสียงนั้น ๆ ไม่ต่อเนื่อง ส่งผลให้ผู้ใช้บริการเกิดความไม่พอใจได้ และที่ผ่านมามีงานวิจัยส่วนใหญ่ได้มุ่งเน้นไปที่การแก้ปัญหาของจitterในโครงข่ายอินเทอร์เน็ตแบบมีสายเท่านั้น [12, 13] ยังไม่มีงานวิจัยใดที่พิจารณาปัญหาของจitterโดยตรงในโครงข่ายแอดฮอคบริการ สื่อ ผสม ที่ ใช้ กัน อยู่ ทุกวันนี้ ใน โครงข่าย อิน เทอร์ เน็ต

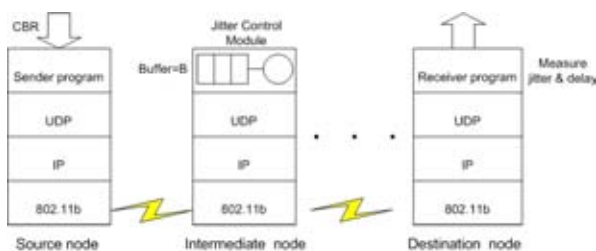
นั้น ใช้วิธีควบคุมจitterเตอร์ที่โหนดปลายทาง โดยการเก็บกลุ่มข้อมูลจำนวนหนึ่งไว้ในบัฟเฟอร์ก่อนแล้วค่อยนำออกแสดงผล ซึ่งทำได้ง่าย เนื่องจากต้องมีการจัดการที่โหนดปลายทางเท่านั้น แต่การจัดการบัฟเฟอร์ในลักษณะนี้อาจไม่เหมาะกับโครงข่ายแอตฮอกเนื่องจากเป็นโครงข่ายไร้สาย ค่าประวิงเวลาที่มีความไม่แน่นอนสูง ประกอบกับอุปกรณ์ที่ใช้ไม่ได้มีเพียงเครื่องคอมพิวเตอร์เท่านั้น อาจจะมีอุปกรณ์ขนาดเล็กอื่น ๆ ที่มีขนาดบัฟเฟอร์ที่จำกัด ดังนั้นงานวิจัยในบทความนี้จึงได้สร้างระบบทดสอบเพื่อประเมินผลจากการใช้หลักการควบคุมจitterเตอร์แบบกระจาย (distributed jitter control) โดยให้โหนดระหว่างทางสามารถช่วยจัดการควบคุมจitterเตอร์ได้ ซึ่งทำให้โหนดปลายทางไม่จำเป็นต้องรับภาระหนักเกินไป และเพื่อให้ระบบจัดการได้ง่ายขึ้นระบบทดสอบนี้จึงทำการพิจารณารูปแบบของทราฟฟิกที่ส่งเป็นแบบการส่งด้วยอัตราคงที่ (Constant bit rate: CBR)

2 โครงสร้างของระบบทดสอบ

ระบบทดสอบที่พิจารณาประกอบไปด้วย 5 โหนด เป็นโหนดในโครงข่าย 4 โหนด และอีก 1 โหนดใช้สำหรับสร้างสัญญาณรบกวนโหนดอื่น ๆ โดยที่โหนดทั้งหมดเชื่อมต่อกันผ่านทางโครงข่ายแอตฮอกไร้สาย และกำหนดให้ว่าแต่ละโหนดจะมีโหนดข้างเคียงคือโหนดที่อยู่ติดกันเท่านั้น ดังแสดงในรูปที่ 1 ส่วนแล็ปท็อปคอมพิวเตอร์ที่ใช้ในระบบทดสอบนี้ประกอบด้วย IBM 4 เครื่อง และ Compaq 1 เครื่องที่มีการติดตั้งระบบปฏิบัติการลินุกซ์ Fedora core3 และ Mandrake 10.1 ส่วนการ์ดแลนไร้สายที่ใช้เป็น Intelpro 2100 และ Intelpro 2200 ซึ่งเชื่อมต่อกันด้วยมาตรฐาน 802.11b โดยที่การรับส่งข้อมูลในการทดลองทั้งหมดจะกระทำผ่านทางซ็อกเก็ต และโปรโตคอลที่ใช้คือ UDP ดังแสดงในรูปที่ 2



รูปที่ 1: รูปแบบของโครงข่ายที่พิจารณา



รูปที่ 2: โครงสร้างของระบบทดสอบ

3 การคำนวณจitterเตอร์ (Jitter Calculation)

จitterเตอร์ของแต่ละกลุ่มข้อมูล สามารถคำนวณได้จากสมการด้านล่างนี้ โดยที่ S_i คือ เวลาที่โหนดต้นทางส่งกลุ่มข้อมูลที่ i ออกไป และ R_i คือ เวลาที่โหนดปลายทางได้รับกลุ่มข้อมูลที่ i

$$Jitter = |(R_j - S_j) - (R_i - S_i)| = |(R_j - R_i) - (S_j - S_i)|$$

4 หลักการควบคุมจitterเตอร์

ในระบบ ทดสอบ นี้ อาศัย หลักการ ควบคุม จิตเตอร์ ที่ โหนด ระหว่างทาง โดยการเก็บกลุ่มข้อมูลเข้าบัฟเฟอร์ซึ่งมีค่าขีดเริ่มเปลี่ยน (threshold) เท่ากับ B โดยในระบบทดสอบนี้กำหนดให้ $B = 10$ กลุ่มข้อมูล เมื่อจitterเตอร์จะทำหน้าที่เก็บกลุ่มข้อมูลที่เข้ามาจนกระทั่งจำนวนกลุ่มข้อมูลสะสม k มีค่าเกินจากขีดเริ่มเปลี่ยน B หลังจากนั้นจึงเริ่มทยอยส่งต่อกลุ่มข้อมูลออกไปด้วยอัตราคงที่ $= 1/\tau$ กลุ่มข้อมูล/มิลิวินาที เพื่อทำให้ค่าจitterเตอร์มีค่าน้อยที่สุดเท่าที่จะเป็นไปได้ แต่ในขณะเดียวกันค่าประวิงเวลาก็จะมีค่าเพิ่มขึ้นตามไปด้วย แต่ยังคงอยู่ในระดับที่ยอมรับได้ เช่น ในการเรียกดูภาพเคลื่อนไหว ถึงแม้ว่าจะรอนานกว่าเดิมเล็กน้อยในช่วงต้น แต่จังหวะการเล่นต่อเนื่อง ก็สามารถทำให้ผู้ใช้บริการพอใจได้มากกว่า หลักการควบคุมจitterเตอร์ดังที่กล่าวมาสามารถแสดงการทำงานได้ดังนี้

initialization: $i = 0, j = 0, k = 0$

if (receive packet = yes)

 buffer[j] = packet

 j = j+1; k = k+1;

skip the lower loop until $k = B$ for the first time

if ($k \neq 0$)

 send packet in buffer[i]

 wait for τ ms.

 i = i+1; k = k-1;

/* recur the whole process */

โดย k เป็นจำนวนกลุ่มข้อมูลที่เก็บอยู่ในบัฟเฟอร์ ในขณะที่ i และ j เป็นลำดับของกลุ่มข้อมูลที่โหนดส่งต่อและได้รับมา ตามลำดับ ส่วน N เป็นจำนวนกลุ่มข้อมูลทั้งหมดที่โหนดต้นทางต้องการส่ง

5 ผลการทดลอง

การทดลองในระบบทดสอบนี้ทำขึ้นเพื่อประเมินผลว่าวิธีการควบคุมจitterเตอร์ที่กล่าวมาข้างต้น สามารถลดค่าจitterเตอร์และค่าประวิงเวลาได้มากน้อยเพียงใด โดยจะแสดงผลเปรียบเทียบกันระหว่างการทดลองที่โหนด ระหว่างทาง มีการ ควบคุม จิตเตอร์ และการทดลองที่ส่งต่อกลุ่มข้อมูลไปเลยทันที รวมทั้งการวิเคราะห์ถึงปัจจัยที่อาจส่งผลกระทบต่อค่าจitterเตอร์ที่ปลายทางได้ ดังนั้นในบทความนี้จึงได้แบ่งการทดลองออกเป็น 2 การทดลองหลัก ๆ ดังนี้

5.1 ผลกระทบจากสัญญาณรบกวน

การทดลองนี้ทำขึ้นเพื่อวิเคราะห์ว่าเมื่อมีการรบกวนเกิดขึ้นที่แต่ละโหนด จะมีผลกระทบต่อค่าจัตเตอร์หรือไม่ ซึ่งทำได้โดยการส่งกลุ่มข้อมูลขนาด 1024 ไบต์ ไปด้วยอัตรา 200 มิลลิวินาทีต่อหนึ่งกลุ่มข้อมูล จากโหนด 1 ไปที่โหนด 4 และใช้คอมพิวเตอร์อีกเครื่องหนึ่งทำการถ่ายโอนข้อมูลผ่าน FTP ไปที่แต่ละโหนดในขณะที่กำลังส่งกลุ่มข้อมูลกันอยู่ ดังแสดงในรูปที่ 3 ซึ่งจะทำให้มีการแย่งใช้ช่องสัญญาณกันมากขึ้น ส่งผลให้ค่าประวิงเวลาของแต่ละกลุ่มข้อมูลมีความไม่แน่นอนสูงขึ้น และจัตเตอร์มีค่ามากขึ้นด้วย โดยผลการทดลองเปรียบเทียบแสดงในตารางที่ 1 โดยผลการทดลองทั้งหมดที่แสดงในบทความนี้เป็นค่าเฉลี่ยที่เกิดจากการทดลองจำนวนหลายครั้งเพื่อให้ได้ค่าที่เชื่อถือได้ ซึ่งจะเห็นว่าในกรณีที่มีการถ่ายโอนข้อมูลอยู่



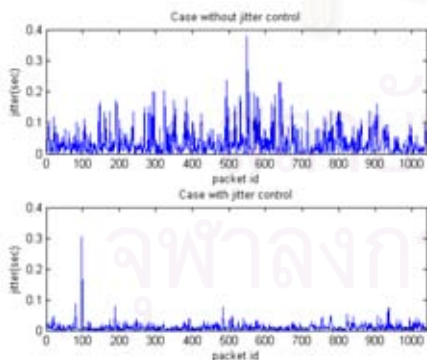
รูปที่ 3: โครงสร้างทางกายภาพของระบบทดสอบที่ใช้ทดลอง

ตารางที่ 1: ผลเปรียบเทียบในกรณีที่มีและไม่มีการรบกวนด้วย FTP

รูปแบบการทดลอง	จัตเตอร์(ms)		ค่าประวิงเวลา (sec)	
	FTP	no FTP	FTP	no FTP
ไม่ควบคุมจัตเตอร์	33.34	2.77	0.043	0.002
ควบคุมจัตเตอร์	10.04	4.08	4.14	3.81

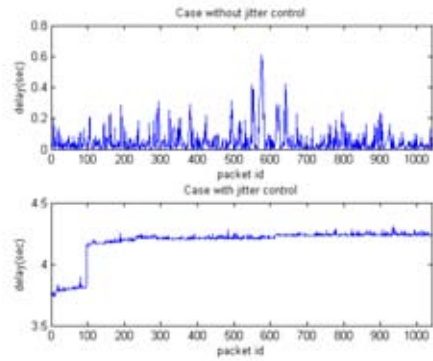
ด้วย จะทำให้ค่าจัตเตอร์และค่าประวิงเวลามีค่ามากขึ้น ส่วนในรูปที่ 4 และ 5 จะแสดงค่าจัตเตอร์และค่าประวิงเวลาที่วัดได้จากการทดลองหนึ่งที่มีการรบกวน เปรียบเทียบในแต่ละกลุ่มข้อมูล ตามลำดับ

เพื่อให้เห็นความแตกต่างได้ชัดเจนยิ่งขึ้น ในการทดลองนี้



รูปที่ 4: จัตเตอร์ที่วัดได้ในแต่ละกลุ่มข้อมูล

ยังได้มีการนำผลจากการจำลองโดยใช้โปรแกรมคอมพิวเตอร์ OMNET++ [8] ซึ่งเป็นโปรแกรมจำลองชนิดเหตุการณ์เต็มหน่วย (discrete event simulator) มาเปรียบเทียบกับ โดยใช้คำสั่งเริ่มเปลี่ยน B และอัตราการส่งข้อมูลเหมือนกับที่ใช้ในระบบทดสอบ และได้ผลดังแสดงในตารางที่ 2 จะเห็นว่าผลที่ได้จากทั้งระบบทดสอบและการ



รูปที่ 5: ค่าประวิงเวลาที่วัดได้ในแต่ละกลุ่มข้อมูล

ตารางที่ 2: ผลที่ได้จากจำลองโดยใช้ omnet++

รูปแบบการทดลอง	จัตเตอร์(ms)	ค่าประวิงเวลา (sec)
ไม่ควบคุมจัตเตอร์	36.85	0.123
ควบคุมจัตเตอร์	20.9	4.57

จำลองโดยใช้ OMNET++ ต่างก็ให้แนวโน้มไปในทิศทางเดียวกัน คือ กรณีที่มีการควบคุมจัตเตอร์สามารถลดค่าจัตเตอร์ลงได้จริง แต่ส่วนของค่าที่ได้อาจจะไม่ใกล้เคียงกันมากนัก เนื่องจากใน OMNET++ มีการก่อกำเนิดค่าประวิงเวลาโดยใช้การสุ่มแบบแจกแจงเอกกรุป (uniform distribution) ภายในช่วง ๆ หนึ่ง ซึ่งอาจจะมากกว่าช่วงที่ปรับเปลี่ยนในระบบทดสอบได้ เพราะในขณะที่ทดลองสภาพแวดล้อมไม่ได้เปลี่ยนแปลงมากนัก

5.2 ผลกระทบจากขนาดของกลุ่มข้อมูล

การทดลองนี้ทำขึ้นเพื่อวิเคราะห์ว่าขนาดของแต่ละกลุ่มข้อมูลที่ส่งมีผลต่อจัตเตอร์หรือไม่ โดยในการทดลองได้ทำการแบ่งกลุ่มข้อมูลออกเป็น 3 ขนาด คือ 512, 1024, 2048 ไบต์ ส่งจากโหนด 1 ไปโหนด 4 ด้วยอัตรา 200 มิลลิวินาทีต่อหนึ่งกลุ่มข้อมูลเช่นเดียวกับการทดลองที่ 1 ซึ่งได้ผลดังแสดงในตารางที่ 3

โดยผลที่ได้จากทั้ง 2 การทดลอง แสดงให้เห็นว่าเมื่อขนาด

ตารางที่ 3: ผลที่วัดได้เมื่อส่งด้วยกลุ่มข้อมูลขนาดต่าง ๆ

ขนาดกลุ่มข้อมูล (ไบต์)	ไม่ควบคุมจัตเตอร์		ควบคุมจัตเตอร์	
	จัตเตอร์ (ms)	ค่าประวิงเวลา (ms)	จัตเตอร์ (ms)	ค่าประวิงเวลา (sec)
512	12.94	36.72	7.1	3.82
1024	29.44	45.16	6.99	3.93
2048	33.88	50.36	10.4	3.98

ของกลุ่มข้อมูลมีค่ามากขึ้น และมีการรบกวนเกิดขึ้น จัตเตอร์จะมีค่าเพิ่มขึ้นตามไปด้วย ซึ่งหากใช้วิธีการควบคุมจัตเตอร์จะสามารถลดค่าจัตเตอร์ที่โหนดปลายทางลงได้มาก ถึงแม้ว่าในวิธีที่ไม่มีการควบคุมจัตเตอร์จะมีจัตเตอร์ไม่มากอยู่แล้ว แต่ในโครงข่ายที่พิจารณา มีการส่งผ่านกลุ่มข้อมูลไปเพียงแค่ 3 ช่วงเชื่อมต่อก่อนนั้น จัตเตอร์จึงยังมีค่าน้อยอยู่ แต่ในโครงข่ายที่ใช้งานกันจริง ๆ นั้น กลุ่มข้อมูลอาจต้องส่งผ่านช่วงเชื่อมต่อที่มากกว่านี้หลายเท่า รวมถึงขนาดของกลุ่มข้อมูลและ

สัญญาณรบกวนที่มีมากกว่าในระบบทดสอบนี้มาก ดังนั้นการควบคุมจิตเตอร์จึงมีความสำคัญและจำเป็นมาก ส่วนค่าประวิงเวลาที่เห็นว่ามีค่าเพิ่มขึ้นจากเดิมมากนั้น เกิดจากการกำหนดค่าที่ใช้ในการทดลอง โดยหากกำหนดให้ค่าซีดีเริ่มเปลี่ยน B มีค่าน้อยกว่านี้ และเพิ่มอัตราการส่งกลุ่มข้อมูลให้เร็วกว่านี้ ค่าประวิงเวลาจะมีค่าต่ำลงได้ โดยตารางที่ 4 แสดงผลที่ได้เมื่อกำหนดให้ค่า B เท่ากับ 2 และ 6 ซึ่งจะเห็นว่าค่าประวิงเวลามีค่าลดลงไปมาก แต่หากกำหนดให้ค่า B น้อยเกินไป ก็จะทำให้ประสิทธิภาพในการช่วยลดจิตเตอร์ลดลงไปด้วย

ตารางที่ 4: ผลที่ได้เมื่อปรับเปลี่ยนค่าซีดีเริ่มเปลี่ยน

รูปแบบการทดลอง	จิตเตอร์(ms)	ค่าประวิงเวลา (sec)
ไม่ควบคุมจิตเตอร์	15.7	0.018
ซีดีเริ่มเปลี่ยน B=2	7.4	0.412
ซีดีเริ่มเปลี่ยน B=6	3.5	2.03

6 สรุป

ในบทความนี้ นำเสนอ การ สร้าง ระบบ ทดสอบ เพื่อ ประเมินผลจากการใช้หลักการควบคุมจิตเตอร์แบบกระจายในโครงข่ายแอตซอก พร้อมทั้งเปรียบเทียบผลที่ได้จากการจำลองโดยใช้โปรแกรม OMNET++ โดยผลจากทั้ง 2 วิธีทดสอบสอดคล้องกันว่าถึงแม้วิธีควบคุมจิตเตอร์ที่นำเสนอจะทำให้ค่าประวิงเวลาเพิ่มขึ้น แต่ก็สามารถลดจิตเตอร์ลงได้มาก ซึ่งส่งผลให้บริการข้อมูลภาพหรือเสียงมีความต่อเนื่องในการเล่น ทำให้ผู้ใช้บริการได้รับความพอใจมากกว่า

ระบบ ทดสอบ ที่ ได้ นำเสนอ ใน บทความ นี้ มี การ จำลองสภาพแวดล้อมทุกอย่างใกล้เคียงกับโครงข่ายจริงมาก โดยเหมาะสมสำหรับการประเมินผลในโครงข่ายที่มีขนาดเล็กถึงขนาดกลาง แต่อาจจะยากต่อการทดลองในกรณีที่มีโหนดมีการเคลื่อนที่ เนื่องจากข้อจำกัดหลาย ๆ อย่างทั้งทางด้านงบประมาณ และความยากต่อการจัดการควบคุมระบบ ดังนั้นเพื่อให้รองรับความสามารถในส่วนนี้ แผนงานในปัจจุบันที่กำลังดำเนินการพัฒนาต่อคือ การสร้างระบบทดสอบสำหรับโครงข่ายแอตซอกที่เชื่อมต่อกันโดยใช้โครงข่ายแบบมีสาย ซึ่งเป็นระบบที่เราสามารถควบคุมได้ และสามารถขยายขนาดของโครงข่ายทดลองจนกำหนดการเคลื่อนที่ของแต่ละโหนดได้ตามต้องการ รวมทั้งชนิดของทราฟฟิกที่พิจารณาจะเป็นแบบการส่งด้วยอัตราไม่คงที่ เพื่อให้มีสถานการณ์ใกล้เคียงกับระบบจริงมากยิ่งขึ้น

เอกสารอ้างอิง

[1] M. Krunz, A. Muqattash, S. Lee, "Transmission power control in wireless ad hoc networks: challenges, solutions and open issues," *IEEE Network*, Volume 18, Issue 5, pp.8 - 14, Sept.-Oct. 2004.

[2] N. Milanovic, M. Malek, A. Davidson, and V. Milutinovic, "Routing and Security in Mobile Ad Hoc

Networks," *IEEE Computer*, Volume 37, Issue 2, p-p.61 - 65, Feb 2004.

- [3] C. Perkins and E. Royer, "Ad hoc On-Demand Distance Vector (AODV) Routing," in *RFC 3561*, July 2003.
- [4] D. B. Johnson, D. A. Maltz and Y. C. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," *Internet Draft, IETF Mobile Ad hoc networks*, draft-ietf-manet-dsr-10.txt, July 2004.
- [5] H. Sun and H.D. Hughes, "Adaptive QoS Routing Based on Prediction of Local Performance in Ad Hoc Networks," *Wireless Communications and Networking*. IEEE, Volume 2, March 2003.
- [6] CMU Monarch Group, The Network Simulator: ns-2, <http://www.isi.edu/nsnam/ns/>
- [7] GloMoSim, <http://pcl.cs.ucla.edu/projects/glomosim/>
- [8] A. Varga, "Object Modular Network Testbed in C++," www.omnetpp.org
- [9] D. A. Maltz and J. Broch, "Lessons from a Full-Scale Multihop Wireless Ad Hoc Network Testbed," *IEEE Personal Communications*, pp.8-15, February 2001.
- [10] S. Sanghani, T. X Brown, S. Bhandare and S. Doshi, "EWANT: The Emulated Wireless Ad Hoc Network Testbed," *IEEE Wireless Communications and Networking Conference (WCNC)*, 16-20 March 2003.
- [11] H. Lundgren, D. Lundberg, J. Nielsen, E. Nordstrom and C. Tschudin, "A Large-scale Testbed for Reproducible Ad hoc Protocol Evaluations," *IEEE Wireless Communications and Networking Conference (WCNC)*, 2002
- [12] D. Verma, H. Zhang, and D. Ferrari, "Delay Jitter Control for Real-Time Communication in a Packet Switching Network," *IEEE TRICOMM'91*, pp. 35-43, April 1991.
- [13] S. F. Bush, A. Kulkarni, S. Evans and L. Galup, "Active Jitter Control," *7th International IS&N Conference, Intelligence in Services and Networks (ISN)'00*, February 23-25, 2000.

ประวัติผู้เขียนวิทยานิพนธ์

พนิดา วีระวุฒิพล เกิดเมื่อวันที่ 18 มกราคม พ.ศ. 2525 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาชั้นมัธยมศึกษาจากโรงเรียนช่างตากูร์สคอนแวนท์ในปี พ.ศ. 2541 จากนั้นได้เข้าศึกษาต่อที่คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยเกษตรศาสตร์ จนสำเร็จหลักสูตรวิศวกรรมศาสตรบัณฑิตในปี พ.ศ. 2545 และได้เข้าทำงานในตำแหน่ง System engineer ที่บริษัทแห่งหนึ่งเป็นเวลา 1 ปี ก่อนที่จะเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต ณ จุฬาลงกรณ์มหาวิทยาลัย จนสำเร็จการศึกษาในปี พ.ศ. 2549



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย