การประมาณแบนด์วิดท์แบบปรับตัวได้ของทีซีพีเพื่อควบคุมความคับคั่งในอินเทอร์เน็ตผ่านดาวเทียม

นายศราวุธ ไชยชนะ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2550

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

ADAPTIVE BANDWIDTH ESTIMATION IN TCP FOR CONTROLLING CONGESTION OVER SATELLITE INTERNET

Mr. Sarawuth Chaichana

ศูนย์วิทยทรัพยากร
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
Chulalongkorn University

Academic Year 2007

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การประมาณแบนด์วิดท์แบบปรับตัวได้ของทีซีพีเพื่อควบคุมความคับคั่งในอินเทอร์เน็ตผ่านดาวเทียม
โดย	นายศราวุธ ไชยชนะ
สาขาวิชา	วิศวกรรมไฟฟ้า
อาจารย์ที่ปรึกษา	รองศาสตราจารย์ ดร.ประสิทธิ์ ทีมพุฒิ

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ศาสตราจารย์ ดร.ประสิทธิ์ ประพิณมงคลการ)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์
(รองศาสตราจารย์ ดร.ประสิทธิ์ ทีมพุฒิ)

..... กรรมการ
(รองศาสตราจารย์ ดร.โยธิน เปรมปราณีรัชต์)

..... กรรมการ
(ดร.ไกรสร อัญชลีวรพันธุ์)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ศราวุธ ไชยชนะ : การประมาณแบนด์วิดท์แบบปรับตัวได้ของทีซีพีเพื่อควบคุมความคับคั่งในอินเทอร์เน็ตผ่านดาวเทียม (ADAPTIVE BANDWIDTH ESTIMATION IN TCP FOR CONTROLLING CONGESTION OVER SATELLITE INTERNET) อ. ที่ปรึกษา : รศ.ดร.ประสิทธิ์ ทีฆพุฒิ, 90 หน้า.

วิทยานิพนธ์ฉบับนี้นำเสนอวิธีการประมาณแบนด์วิดท์แบบปรับตัวได้ของทีซีพีเพื่อควบคุมความคับคั่งข้อมูลในช่องสัญญาณอินเทอร์เน็ตผ่านดาวเทียม ซึ่งได้นำแนวทางกลไกลประมาณโดยใช้ประโยชน์จากสัญญาณการตอบรับ(ACK) ที่เป็นตัวบ่งชี้อัตราปริมาณข้อมูลซึ่งส่งผ่านสำเร็จและได้รับโดยปลายทาง มาทำการวัดและเปรียบเทียบค่าเฉลี่ยให้เป็นเกณฑ์ความจุช่องสัญญาณ เมื่อเกิดการคับคั่งขึ้นก็จะนำค่าเกณฑ์ประมาณที่ได้มาปรับค่าขนาดความคับคั่งวินโดว์และจุดอิ่มตัวทางด้านผู้ส่งใหม่ เพื่อทำการส่งข้อมูลเข้าสู่ช่องสัญญาณ จากผลการจำลองในโครงข่ายแสดงให้เห็นว่าเมื่อทำการเปรียบเทียบสมรรถนะกับวิธีการควบคุมความคับคั่งมาตรฐาน พบว่าวิธีนี้เป็นแนวทางหนึ่งในการปรับปรุงสมรรถนะของ TCP ในรูปแบบปลายทางไปยังปลายทางที่ช่วยแก้ปัญหาการลดขนาดของวินโดว์และจุดอิ่มตัวของช่องสัญญาณลงอย่างรวดเร็ว ทำให้อัตราการส่งผ่านในช่องสัญญาณมีประสิทธิภาพยิ่งขึ้น ซึ่งสามารถพิสูจน์ได้โดยการจำลอง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา วิศวกรรมไฟฟ้า ลายมือชื่อนิสิต.....ศราวุธ ไชยชนะ.....
สาขาวิชา วิศวกรรมไฟฟ้า ลายมือชื่ออาจารย์ที่ปรึกษา.....
ปีการศึกษา 2550

4870480921 : MAJOR ELECTRICAL ENGINEERING

KEY WORD: CONGESTION CONTROL / BANDWIDTH ESTIMATION / TCP / CWND / ACK

SARAWUTH CHAICHANA : ADAPTIVE BANDWIDTH ESTIMATION IN TCP FOR CONTROLLING CONGESTION OVER SATELLITE INTERNET. THESIS ADVISOR : ASSOC. PROF. DR.PRASIT TEEKAPUT, Ph.D. 90 pp.

The propose of this thesis is to control TCP congestion over satellite internet channel with bandwidth estimation technique. The bandwidth is estimated by measure and average the rate of returning ACKs at the time that implies amount of significant capacity. When congestion occurs, bandwidth estimated will be used to adjust dynamic gain for setting new the congestion window and the slow start threshold at TCP sender before input data channel. The simulation results when compared with standard congestion control presented to technically improve TCP congestion control in term end to end will avoid rapid window and threshold reductions. Experimental technique reveal improvements in throughput performance of satellite channel.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Department Electrical Engineering Student's signature Sarawuth Chaichana

Field of study Electrical Engineering Advisor's signature P. Teekaput

Academic year 2007

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความช่วยเหลืออย่างดียิ่งของอาจารย์ที่ปรึกษาวิทยานิพนธ์ รศ.ดร.ประสิทธิ์ ฑีฆพุฒิ ซึ่งได้ให้คำแนะนำและข้อคิดเห็นต่างๆที่เป็นประโยชน์ในการทำวิจัยด้วยดีตลอดมา ขอขอบคุณคณะกรรมการสอบวิทยานิพนธ์ คุณครู อาจารย์ทุกๆท่านในอดีตและปัจจุบันที่ได้อบรมสั่งสอนวิชาความรู้อันมีค่าแก่ผู้วิจัย เพื่อนนิสิตสาขาวิชาวิศวกรรมไฟฟ้าสื่อสารทุกท่านที่เป็นกำลังใจและให้ความช่วยเหลือในทุกด้าน รวมทั้งภาควิชาวิศวกรรมไฟฟ้าและห้องปฏิบัติการวิจัยระบบโทรคมที่เอื้อเฟื้อสถานที่และสิ่งอำนวยความสะดวกในการทำวิจัย

สุดท้ายนี้ ผู้วิจัยใคร่ขอกราบขอบพระคุณบิดา-มารดา ซึ่งให้การสนับสนุนและให้กำลังใจแก่ผู้วิจัยอย่างดียิ่งตลอดจนสำเร็จการศึกษา

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย…..... ง

บทคัดย่อภาษาอังกฤษ..... จ

กิตติกรรมประกาศ..... ฉ

สารบัญ..... ช

สารบัญภาพ..... ฌ

สารบัญตาราง..... ฎ

บทที่

1.บทนำ..... 1

1.1 ความเป็นมาและความสำคัญของปัญหา..... 1

1.2 วัตถุประสงค์ของการวิจัย..... 2

1.3 ขอบเขตของการวิจัย..... 2

1.4 คำจำกัดความ..... 3

1.5 ประโยชน์ที่คาดว่าจะได้รับ..... 3

1.6 วิธีดำเนินการวิจัย..... 3

1.7 ลำดับขั้นตอนในการเสนอผลการวิจัย..... 4

2.เอกสารและงานวิจัยที่เกี่ยวข้อง..... 5

2.1 โปรโตคอลทีซีพี (Transmission Control Protocol)..... 5

2.1.1 รูปแบบส่วนหัวของTCP (TCP Header Format)..... 6

2.1.2 ระบบการควบคุมของTCP..... 9

2.1.3 การควบคุมการคับคั่งของTCP..... 10

2.2 มาตรฐานการควบคุมความคับคั่งของ TCP..... 12

2.2.1 Slow Start..... 12

2.2.2 Congestion Avoidance..... 13

2.2.3 Fast Retransmission..... 15

2.2.4 Fast Recovery..... 16

2.3 แบบแผนควบคุมความคับคั่งของทีซีพีรุ่นต่างๆ..... 17

2.4 ข้อตกลงเกี่ยวกับทราฟฟิกสำหรับแพ็คเกจสวิตชิ่ง(Traffic Contract)..... 19

2.4.1 ประเภทของการบริการ(Service Categories)..... 19

หน้า

2.4.2 ตัวแปรแสดงคุณภาพของการให้บริการ..... 22
2.4.3 การควบคุมการร้องขอเพื่อสร้างการเชื่อมต่อ..... 26
2.4.4 การจัดสรรแบนด์วิดท์แบบปรับตัวได้..... 27
2.5 ระบบสื่อสารดาวเทียม..... 29
2.5.1 ประเภทของดาวเทียม 31
2.5.2 โครงข่ายดาวเทียมอินเทอร์เน็ต..... 32
2.6 ผลกระทบที่เกิดกับโปรโตคอล TCP บนโครงข่ายดาวเทียม..... 41
3.วิธีดำเนินการวิจัย..... 44
3.1 การวัดแบนด์วิดท์รูปแบบปลายทาง(end to end)..... 45
3.1.1 การวัดโดยใช้พื้นฐานของสัญญาณการตอบรับ (ACK based)..... 45
3.1.2 การกรองอัตราการรับ ACK..... 47
3.1.3 ค่าการประมาณแบนด์วิดท์กับการปรับขนาดวินโดว์..... 49
3.2 แบบจำลองการทำงาน..... 51
3.3 การหาสมรรถนะของ TCP..... 52
4. ผลการจำลองและการวิเคราะห์..... 54
4.1 สมรรถนะของกลไกลการประมาณแบนด์วิดท์..... 54
4.2 การปรับขนาดของ CWND,SSTHRESH และ GOODPUT ของ TCP บนโครงข่ายดาวเทียม..... 57
4.3 อัตราการส่งผ่านข้อมูลในช่องสัญญาณเมื่อเทียบอัตราผิดพลาดบิตและช่วงเวลาเดินทาง..... 61
4.4 ความสามารถในการทำงานร่วมกับแบบแผน TCP ที่ต่างกัน..... 62
4.5 ผลการวิเคราะห์..... 64
5. สรุปผลการวิจัยและข้อเสนอแนะ..... 68
5.1 สรุปผลการวิจัย..... 68
5.2 ข้อเสนอแนะ..... 68
รายการอ้างอิง..... 70
ภาคผนวก..... 73
ประวัติผู้เขียน..... 90

สารบัญภาพ

รูปที่		หน้า
1	รูปแบบส่วนหัวของ TCP	6
2	การทำงานของ TCP เมื่อเทียบชั้นการจำลองใน OSI	8
3	การใช้เทคนิค sliding window ในการนำพาข้อมูล	9
4	การเปรียบเทียบภาระ (Load) กับ ความล่าช้า (Delay) และอัตราส่งผ่าน	11
5	การปรับขนาด cwnd ของอัลกอริทึม slow start และ congestion avoidance	15
6	การปรับขนาด cwnd และ ssthresh เมื่อหมดคาบเวลาการรอ ACK	16
7	ไดอะแกรมกลไกลการทำงานมาตรฐานการควบคุมการคับคั่ง	17
8	องค์ประกอบพื้นฐานของระบบสื่อสารดาวเทียม	30
9	ตัวอย่างการใช้เทคนิคความถี่ซ้ำ	31
10	เปรียบเทียบระดับความสูงของดาวเทียม 3 วงโคจร GEO,MEO และ LEO จากผิวโลก	32
11	โครงข่ายดาวเทียมอินเทอร์เน็ตโครงสร้างแบบ bent-pipe	34
12	โครงข่ายดาวเทียมอินเทอร์เน็ตโครงสร้างแบบ OBS และ ISL	35
13	โครงข่ายดาวเทียมอินเทอร์เน็ตโครงสร้างแบบ DBS	35
14	โครงข่ายระบบดาวเทียมอินเทอร์เน็ตและการใช้งาน	40
15	กลไกลการควบคุมการคับคั่งมาตรฐานเมื่อเกิดการลดขนาด cwnd	44
16	(ก)อัตราการส่งผ่านข้อมูล(throughput) (ข)ขนาดความคับคั่งของวินโดว์(cwnd) ของ TCP-Reno เมื่อเกิดการคับคั่งในช่องสัญญาณ	45
17	ระบบการทำงานของการวัดแบนด์วิดท์โดยใช้พื้นฐานสัญญาณการตอบรับ	46
18	ชุด ACKs ในช่องสัญญาณกับปริมาณข้อมูลที่ส่งผ่านสำเร็จ	47
19	การสุ่มปริมาณข้อมูลเมื่อเทียบกับแกนเวลา	47
20	ช่วงระยะเวลาสุ่มที่มีการสุ่มเสมือน(virtual sample)แทรกอยู่	49
21	การทำงานของกลไกลการประมาณแบนด์วิดท์ที่แหล่งส่งข้อมูล	50
22	แผนภูมิการทำงานของกลไกลการประมาณแบนด์วิดท์เมื่อได้รับชุด ACKs ซ้ำ	51
23	โครงสร้างแบบจำลอง Peer to Peer	52
24	โครงสร้างในการทำงานร่วมกันของแบบแผน TCP เดียวกัน	53
25	โครงข่ายแบบการจำลองสมรรถนะการประมาณแบนด์วิดท์	55

รูปที่ หน้า

26 เปรียบเทียบสมรรถนะการประมาณแบนด์วิดท์กับค่าแบนด์วิดท์จริงในช่องสัญญาณ (τ = 500ms)..... 55

27 เปรียบเทียบสมรรถนะการประมาณแบนด์วิดท์กับค่าแบนด์วิดท์จริงในช่องสัญญาณ (τ = 10ms)..... 56

28 เปรียบเทียบสมรรถนะการประมาณแบนด์วิดท์กับค่าแบนด์วิดท์จริงในช่องสัญญาณ (τ = 120ms)..... 56

29 โครงข่ายแบบการจำลองการปรับขนาด cwnd และ ssthresh ในช่องสัญญาณ..... 57

30 การปรับขนาดของ cwnd และ ค่า ssthresh ในช่องสัญญาณเมื่อใช้การควบคุมการคับคั่งมาตรฐาน(ช่องสัญญาณไม่มีอัตราบิตผิดพลาด)..... 58

31 การปรับขนาดของ cwnd และ ค่า ssthresh ในช่องสัญญาณเมื่อใช้กลไกลการประมาณ (ช่องสัญญาณไม่มีอัตราบิตผิดพลาด)..... 58

32 การปรับขนาดของ cwnd และ ค่า ssthresh ในช่องสัญญาณเมื่อใช้การควบคุมการคับคั่งมาตรฐาน(ช่องสัญญาณมีอัตราบิตผิดพลาด 0.1%)..... 59

33 การปรับขนาดของ cwnd และ ค่า ssthresh ในช่องสัญญาณเมื่อใช้กลไกลการประมาณ (ช่องสัญญาณมีอัตราบิตผิดพลาด 0.1%)..... 59

34 การปรับขนาดของ cwnd , ssthresh และค่า goodput ในช่องสัญญาณเมื่อใช้กลไกลการควบคุมการคับคั่งมาตรฐาน (ช่องสัญญาณมีอัตราบิตผิดพลาด 0.5%)..... 60

35 การปรับขนาดของ cwnd , ssthresh และค่า goodput ในช่องสัญญาณเมื่อใช้กลไกลการควบคุมการคับคั่งมาตรฐาน (ช่องสัญญาณมีอัตราบิตผิดพลาด 0.5%)..... 60

36 อัตราการส่งผ่านข้อมูลเมื่ออัตราผิดพลาดบิตในช่องสัญญาณเพิ่มขึ้น..... 61

37 อัตราการส่งผ่านข้อมูลเมื่อช่วงเวลาการเดินทางในช่องสัญญาณเพิ่มขึ้น..... 62

38 โครงข่ายแบบการจำลองความสามารถในการทำงานร่วมของทีซีพีแบบแผนต่างกัน..... 62

39 ค่าเฉลี่ยอัตราการส่งผ่านข้อมูลเมื่อจำนวนการเชื่อมต่อ TCP ของการควบคุมการคับคั่งมาตรฐานในช่องสัญญาณเพิ่มขึ้น (ไม่มีอัตราบิตผิดพลาดในช่องสัญญาณ)..... 63

40 ค่าเฉลี่ยอัตราการส่งผ่านข้อมูลเมื่อจำนวนการเชื่อมต่อ TCP ของการควบคุมการคับคั่งมาตรฐานในช่องสัญญาณเพิ่มขึ้น (ช่องสัญญาณมีอัตราบิตผิดพลาด 1%)..... 63

สารบัญตาราง

ตารางที่ หน้า

1. พารามิเตอร์จำลองสมรรถนะการประมาณแบนด์วิดท์..... 55
2. พารามิเตอร์จำลองการปรับขนาด cwnd และ ssthresh ในช่องสัญญาณ..... 57
3. พารามิเตอร์จำลองอัตราการส่งผ่านเมื่อเทียบอัตราผิดพลาดบิตและช่วงเวลาเดินทาง 61
4. พารามิเตอร์จำลองความสามารถในการทำงานร่วมของทีซีพี..... 63
5. ผลการวิเคราะห์ค่าคาบเวลาของตัวกรอง..... 64
6. ผลการวิเคราะห์และเปรียบเทียบค่า ssthresh และ cwnd ในช่องสัญญาณ..... 64
7. เปรียบเทียบอัตราส่งผ่านข้อมูลเมื่อเปลี่ยนแปลงอัตราผิดพลาดบิต..... 65
8. เปรียบเทียบอัตราส่งผ่านข้อมูลเมื่อเปลี่ยนแปลงค่าเวลาเดินทาง..... 66
9. เปรียบเทียบสมรรถนะความสามารถในการทำงานร่วมกับแบบแผนTCP ต่างกัน..... 66

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

TCP (Transmission Control Protocol) [1] กลายเป็นโปรโตคอลสื่อสารที่ครอบคลุมการประยุกต์ใช้งานระบบมัลติมีเดียในปัจจุบันทราฟฟิคของอินเทอร์เน็ตส่วนใหญ่ถูกนำพาโดยโปรโตคอล TCP ซึ่งจัดอยู่ในชั้นทรานสปอร์ตเลเยอร์โดยให้บริการรองรับการเชื่อมต่อในรูปแบบปลายทางต่อปลายทาง (end to end) อย่างสมบูรณ์ โดยมีกลไกลส่งผ่านแพ็คแกจที่สร้างความเชื่อมั่นและจัดลำดับในการส่งข้อมูล เริ่มแรก TCP ถูกออกแบบมาสำหรับใช้งานบนเครือข่ายแบบมีสายซึ่งมีอัตราการผิดพลาดบิตต่ำ สาเหตุหลักที่ทำให้เกิดแพ็คเกจสูญหายส่วนใหญ่นั้นจะมาจากการคับคั่งข้อมูลเป็นสำคัญ TCP จึงได้มีการปรับปรุงการนำพาข้อมูลและอัลกอริทึมการควบคุมความคับคั่งขึ้น [2] ซึ่งใช้พื้นฐานการปรับขนาดวินโดว์ที่มีการเพิ่มแบบบวกและลดแบบทวีคูณ(AIMD) [3] กลไกลการนำพาข้อมูลที่ใช้การปรับขนาดวินโดว์อนุญาตให้ผู้ส่งสามารถส่งข้อมูลต่อไปได้โดยขึ้นอยู่กับการรับสัญญาณการตอบกลับ (acknowledgment) เพื่อแสดงให้ทราบว่าแพ็คเกจลำดับสุดท้ายประสบผลสำเร็จในการรับโดยผู้รับ หากมีการสูญหายของข้อมูลเกิดขึ้น เช่นผู้ส่งได้รับสัญญาณการตอบกลับซ้ำ (duplicate ACKs) หรือสิ้นสุดเวลาการส่งซ้ำ (retransmission time out) จะไปกระตุ้นกลไกลควบคุมการคับคั่งโดยผู้ส่งจะลดขนาดวินโดว์ที่ส่งผ่านข้อมูล หรือ congestion window (cwnd) ในส่วนของ TCP ลงเป็นผลให้อัตราการส่งผ่านลดต่ำลงเพื่อหลีก เลี่ยงการคับคั่งข้อมูลในช่องสัญญาณ

เมื่อเทคโนโลยีอินเทอร์เน็ตก้าวหน้ามากขึ้น มีการให้บริการอย่างแพร่หลายและครอบคลุมมากขึ้นด้วยโครงข่ายไร้สายแบบต่างๆ รวมทั้งในระบบโครงข่ายดาวเทียมที่สามารถจัดสรรช่องสัญญาณความจุสูงแบบครอบคลุมได้โดยง่าย โปรโตคอล TCP กลายเป็นโปรโตคอลสำคัญที่ถูกนำมาใช้ในการนำพาข้อมูลในระบบดาวเทียมอินเทอร์เน็ต [4] และได้มีการพัฒนาการควบคุมการคับคั่งของ TCP ขึ้นจากเดิม [5],[6] เพื่อลดผลกระทบที่เกิดขึ้นจากการใช้งานบนช่องสัญญาณดาวเทียม ซึ่งมีคุณลักษณะอัตราผิดพลาดบิตและการประวิงของสัญญาณเนื่องจากระยะทางสูง เป็นผลให้เกิดปัญหาในการควบคุมการคับคั่งของข้อมูลในช่องสัญญาณไม่สามารถแก้ไขด้วยวิธีการเช่นเดียวกับโครงข่ายแบบมีสาย โดยงานวิจัยต่างๆที่ผ่านมาได้นำเสนอวิธีการควบคุมความคับคั่ง[7],[8] อย่างไรก็ตามวิธีการเหล่านี้ก็ไม่เหมาะสมเพียงพอกับอินเตอร์เน็ตผ่านดาวเทียมซึ่งมีการประวิงของสัญญาณระยะไกลและแบนด์วิดท์ขนาดใหญ่ และใน [8] ต้องมีการปรับเปลี่ยนระบบประมวลผลของโหนดที่ตำแหน่งผู้ส่ง โหนดผู้รับและทุกเราท์เตอร์กลาง ซึ่งนำไปใช้งานในระบบจริงได้ยาก

วิทยานิพนธ์นี้จึงมีจุดประสงค์ที่จะนำเสนอการปรับปรุงประสิทธิภาพการควบคุมความคับคั่งของ TCP ในโครงข่ายอินเทอร์เน็ตผ่านดาวเทียม ด้วยการนำเทคนิคการประมาณความเป็นไปได้ของแบนด์วิดท์ที่ใช้หลักการพื้นฐานของข้อมูลสัญญาณการตอบกลับในช่องสัญญาณมาใช้ เพื่อควบคุมความคับคั่งข้อมูลของโปรโตคอล TCP บนโครงข่ายดาวเทียมอินเทอร์เน็ต โดยวิธีที่นำเสนอสามารถทำงานโดยใช้การปรับปรุงเฉพาะส่วนประมวลผลของ TCP ทางด้านผู้ส่งเท่านั้น ดังนั้นจึงมีความเป็นไปได้สูงในการนำไปใช้ในโครงข่ายจริง

1.2 วัตถุประสงค์ของการวิจัย

1. ศึกษากลไกลการทำงานของโปรโตคอล TCP ในการควบคุมความคับคั่งของข้อมูลบนโครงข่ายดาวเทียม
2. ศึกษาอัลกอริทึมที่มีการพัฒนาขึ้นมาใช้กับโปรโตคอล TCP เพื่อควบคุมความคับคั่งข้อมูลในระบบอินเทอร์เน็ตผ่านดาวเทียม
3. ปรับปรุงอัลกอริทึมการควบคุมความคับคั่งโดยใช้วิธีการที่นำเสนอ และประเมิณสมรรถนะการทำงานโดยเปรียบเทียบกับกลไกลมาตรฐานที่ใช้งานแพร่หลายในปัจจุบัน เพื่อเป็นแนวทางในการนำไปประยุกต์ใช้งาน

1.3 ขอบเขตของการวิจัย

1. ทำการจำลองความคับคั่งของโปรโตคอล TCP ในระบบอินเทอร์เน็ตผ่านดาวเทียมบนโครงข่ายดาวเทียมค้างฟ้า (GEO) ที่มีรูปแบบการเชื่อมต่อแบบปลายทางไปยังปลายทาง (end to end) เนื่องจากเป็นโครงข่ายดาวเทียมที่ได้รับผลกระทบจากการคับคั่งข้อมูลมากที่สุด
2. วิเคราะห์ประสิทธิภาพการทำงานของ TCPโดยใช้การจำลองผ่านโครงข่ายดาวเทียมพื้นฐานแบบ Peer to Peer ซึ่งมีโครงสร้างการประมวลผลที่ไม่ซับซ้อนในการสวิตซ์แพ็คเกจ
3. ทำการเปรียบเทียบสมรรถนะกลไกลที่นำเสนอกับกลไกลการควบคุมการคับคั่งมาตรฐาน ด้วยพารามิเตอร์ ค่า cwnd, ssthresh, goodput, friendliness

1.4 คำจำกัดความที่ใช้ในการวิจัย

การควบคุมความคับคั่ง (Congestion control) คือ การควบคุมการคับคั่งของข้อมูลในช่องสัญญาณ

การประมาณแบนด์วิดท์ (Bandwidth estimation) คือ การประมาณความเป็นไปได้ของแบนด์วิดท์ในช่องสัญญาณ

สัญญาณการตอบรับ (ACK) คือ สัญญาณการตอบรับจากด้านผู้รับไปยังด้านส่งผู้เพื่อแสดงให้ด้านส่งทราบว่าข้อมูลมาถึงยังผู้รับแล้ว

ขนาดความคับคั่งวินโดว์ (cwnd) คือขนาดความคับคั่งของวินโดว์ที่ด้านผู้ส่งใช้ในการบรรจุข้อมูลส่งไปยังด้านผู้รับ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ความรู้ความเข้าใจในระบบโครงข่ายดาวเทียมอินเทอร์เน็ตและการทำงานของโปรโตคอล TCP
2. ได้ทราบถึงปัจจัยที่ทำให้เกิดปัญหาความคับคั่งข้อมูลของ TCP บนโครงข่ายดาวเทียม
3. วิธีการแก้ปัญหาความคับคั่งข้อมูลของ TCP ที่เกิดขึ้นในโครงข่ายดาวเทียมอินเทอร์เน็ต
4. ได้แนวทางในการพัฒนาอัลกอริทึมเพื่อควบคุมความคับคั่งของ TCP บนช่องสัญญาณดาวเทียมอินเทอร์เน็ต

1.6 วิธีดำเนินการวิจัย

1. ศึกษาพื้นฐานระบบการสื่อสารดาวเทียมในรูปแบบต่างๆ และโครงสร้างหลักการทำงานของโปรโตคอล TCP
2. ศึกษาปัจจัยที่ทำให้เกิดความคับคั่งข้อมูลของโปรโตคอล TCP ในระบบดาวเทียมอินเทอร์เน็ต
3. ศึกษาและค้นคว้าวิธีการในการควบคุมความคับคั่งของโปรโตคอล TCP ในระบบดาวเทียมอินเทอร์เน็ต และงานวิจัยที่เกี่ยวข้อง เพื่อเป็นแนวทางในการทำวิจัย
4. ศึกษาและค้นคว้าวิธีการนำการประมาณแบนด์วิดท์มาใช้งานบนช่องสัญญาณดาวเทียม
5. กำหนดขอบเขตของงานวิจัย เพื่อระบุเป้าหมายที่แน่นอนชัดเจนของงานวิจัยที่จะดำเนินการต่อไป และให้สอดคล้องกับข้อกำหนดของระบบที่จำลองขึ้นมา
6. การตั้งสมมุติฐาน เพื่อคาดการณ์ถึงความเป็นไปได้ของเป้าหมายของงานวิจัยตามหลักการทางทฤษฎี
7. ออกแบบจำลองของระบบเพื่อทดสอบวิธีการประมาณแบนด์วิดท์ของในช่องสัญญาณดาวเทียมอินเทอร์เน็ตที่สอดคล้องกับข้อกำหนดของแบบจำลอง
8. การทดสอบความถูกต้องของการจำลองระบบ เขียนโปรแกรมเพื่อประมวลผลตรวจสอบความเป็นไปได้จริงของผลที่ได้จากการประมวลผลจากโปรแกรม
9. วิเคราะห์ผลเมื่อคำนึงถึงเงื่อนไขต่างๆ หรือเหตุการณ์ที่อาจเกิดขึ้นได้

10. นำเสนอและสรุปผลการวิจัยการนำไปประยุกต์ใช้งาน
11. เรียบเรียงผลงานวิจัย และจัดทำรายงานฉบับสมบูรณ์

1.7 ลำดับขั้นตอนในการเสนอผลการวิจัย

การจำลองผลในวิทยานิพนธ์นี้ทำการจำลองเพื่อหาสมรรถนะของการประมาณแบนด์วิดท์แบบปรับตัวได้ในช่องสัญญาณโดยได้จัดแบ่งหัวข้อเพื่อแสดงขั้นตอนของผลการจำลองในการประเมินประสิทธิภาพดังนี้

1. สมรรถนะของกลไกลการประมาณแบนด์วิดท์ในช่องสัญญาณ
2. การปรับขนาดของ CWND, SSTHRESH และค่า goodput ของ TCP บนโครงข่ายดาวเทียม
3. อัตราการส่งผ่านข้อมูลในช่องสัญญาณเมื่อเทียบอัตราการผิดพลาดบิตและช่วงเวลาเดินทาง
4. ความสามารถในการทำงานร่วมกับแบบแผนทีซีพีที่ต่างกันในช่องสัญญาณ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

แนวคิดและทฤษฎี

2.1 โปรโตคอลทีซีพี (Transmission Control Protocol)

โปรโตคอล TCP เป็นโปรโตคอลที่มีคุณสมบัติการส่งชุดข้อมูลเป็นไบต์ (byte stream) จัดเป็นโปรโตคอลที่อยู่ในระดับชั้นทรานสปอร์ตเลเยอร์โดยมีความน่าเชื่อถือในการส่งข้อมูล หน้าที่โดยหลักคือ สร้างเส้นทางการติดต่อที่มีการเชื่อมต่อโดยสมบูรณ์ (connection-oriented) ก่อนที่จะมีการส่งข้อมูล, แบ่งข้อมูลที่มาจาก Application Layer ให้มีขนาดที่เหมาะสม, ใส่ Sequence Number (SEQ) เพื่อจัดลำดับข้อมูลที่ส่งไป, ตรวจสอบความถูกต้องของการรับส่งข้อมูล, จัดสรร Buffer ที่เหมาะสมกับการรับส่งข้อมูล และจัดข้อมูลให้เป็นชุดเดียวกันก่อนส่งให้ Application Layer โดย TCP มีความสามารถในการรับประกันการส่งข้อมูลให้ถึงปลายทางได้โดยอาศัยกระบวนการดังต่อไปนี้

1). กระบวนการตรวจสอบ (Checksums)

ทุกเซกเมนต์จะมีฟังก์ชัน checksum อยู่ ซึ่งจะทำหน้าที่ตรวจจับความเสียหายเนื่องจากการผิดพลาดเนื่องจากช่องสัญญาณในการส่งข้อมูลทั้งส่วนหัว TCP และส่วนของชุดข้อมูล

2). กระบวนการตรวจจับความซ้ำซ้อนของชุดข้อมูล

โครงข่ายการสวิทซ์แบบแพ็คเกจอาจทำให้เกิดแพ็คเกจซ้ำได้ โดยมากจะเกิดขึ้นจากการกระบวนการส่งซ้ำข้อมูลที่เกิดข้อผิดพลาดโดยที่แพ็คเกจชุดแรกอาจถูกประวิงเวลาและแพ็คเกจชุดที่สองถูกส่งไปอย่างเนื่องทำให้ผู้ รับสามารถรับแพ็คเกจนี้ซ้ำกันได้ TCP จะคอยติดตามชุดข้อมูลที่ได้รับเพื่อที่จะยกเลิกชุดข้อมูลที่ได้รับไปแล้ว

3). กระบวนการส่งซ้ำ (Retransmissions)

เป็นกระบวนการรับประกันการส่งข้อมูลบน TCP ซึ่งจะส่งแพ็คเกจที่สูญหายไปใหม่อีกครั้งหนึ่ง การส่งชุดข้อมูลเมื่อถึงผู้รับแล้ว ผู้รับจะส่งสัญญาณตอบกลับแบบบวก (positive acknowledgement) กลับไปบอกผู้ส่งว่าการส่งชุดข้อมูลชุดนั้นสำเร็จ แต่หากการส่งชุดข้อมูลไปนั้นไม่สำเร็จ จะทราบได้จากการขาดสัญญาณการตอบกลับ (ACK) หรือ หมดคาบเวลา (timeout) และทำให้เกิดกระบวนการส่งซ้ำ

4). กระบวนการลำดับชุดข้อมูล (Sequencing)

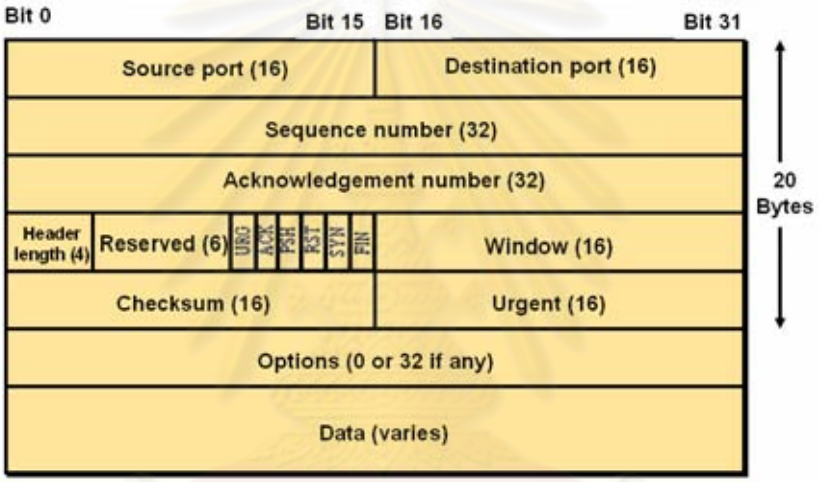
โครงข่ายการสวิทซ์แบบแพ็คเกจ สามารถเกิดชุดแพ็คเกจที่ผู้รับได้รับโดยไม่ได้เรียงตามลำดับหมายเลขได้ TCP จะทำหน้าที่จัดเรียงหมายเลขชุดข้อมูลใหม่ ก่อนที่จะส่งต่อไปที่เลเยอร์ระดับสูงขึ้นไป

5). ตัวนับเวลา (Timers)

TCP มีการนำความสามารถของตัวนับเวลามาใช้ได้ทั้งแบบ static และ dynamic เช่นใช้ในการนับเวลาเมื่อรอผู้รับตอบสัญญาณการตอบรับ (ACK) กลับมาในช่วงเวลาที่กำหนด หากเกินช่วงเวลาที่กำหนด TCP จะเข้าสู่กระบวน การส่งซ้ำเป็นต้น

2.1.1 รูปแบบส่วนหัวของ TCP (TCP Header Format)

TCP เซกเมนต์ คือ รูปแบบส่วนหัว (header) ของ TCP รวมกับ TCP หนึ่งแพ็คเกจ แสดงไว้ดังรูปที่ 1 ขนาดของรูปแบบส่วนหัวไม่รวมตัวเลือก (options) มีขนาดเท่ากับ 20 ไบต์



รูปที่ 1 รูปแบบส่วนหัวของ TCP

อธิบายรายละเอียดการทำงานแต่ละส่วนได้ดังนี้

Source Port

ตัวเลข 16 บิต ระบุถึงโปรแกรมประยุกต์ที่ TCP เซกเมนต์ถูกส่งมาจากฝั่งผู้ส่งโดยตัวเลขพอร์ตแบ่งได้เป็น 3 ช่วง คือ well-known ports (0 ถึง 1023), registered ports (1024 ถึง 49151) และ private ports (49152 ถึง 65535) การกำหนดหมายเลขพอร์ตเพื่อให้ TCP สามารถระบุการเชื่อมต่อกับโปรแกรมประยุกต์ในแอพพลิเคชั่นเลเยอร์ได้อย่างถูกต้อง ตัวอย่างเช่น TELNET server จะถูกกำหนดขึ้นให้เป็น well-known port หมายเลข 23 เมื่อรวมหมายเลขพอร์ตและ หมายเลข IP จะแสดงถึงการมีอยู่หนึ่งเดียว (unique) ของการส่งข้อมูลในโปรโตคอล TCP บนโลกอินเทอร์เน็ต

Destination Port

ตัวเลข 16 บิต ระบุถึงโปรแกรมประยุกต์ทางฝั่งผู้รับ มักจะเป็นหมายเลขเดียวกันกับหมายเลขพอร์ตต้นทาง

Sequence Number

ตัวเลข 32 บิต ระบุหมายเลขของแพ็คเกจ เพื่อให้ผู้รับสามารถแยกแยะแพ็คเกจและเรียงแพ็คเกจเข้าด้วยกันได้ มีจำนวนหมายเลขเท่ากับ 2^{32} -1 หมายเลข และจะถูกกลับไปนับ 0 ใหม่ เมื่อกำหนดจนครบหมายเลขทั้งหมด Acknowledgement Number ตัวเลข 32 บิต ระบุถึงข้อมูลชุดต่อไปที่ผู้รับคาดว่าจะได้รับ ดังนั้นหมายเลขนี้จะมากกว่าหมายเลขชุดข้อมูลที่ได้รับแล้ว หมายเลขนี้จะถูกใช้ก็ต่อเมื่อ บิตควบคุม ACK ถูกเปิดใช้

Header Length

ตัวเลข 4 บิตนี้ บอกความยาวของส่วนหัวของ TCP โดยทั่วไปใน ส่วนหัวของ TCP 1เซกเมนต์จะมีขนาด 20 ไบต์ เมื่อไม่มีส่วนของ options แต่สามารถมีขนาดได้สูงสุด 60 ไบต์ Header Length จำเป็นต้องมี เพราะในกรณีที่ใช้ options ถ้าไม่มี Header Length จะไม่สามารถรู้ได้ว่ามีการใช้ส่วนของ options ไปกี่ไบต์ ในมาตรฐาน TCP จะเรียก Header Length อีกชื่อหนึ่งว่า “data offset”

Reserved

ตัวเลข 6 บิต นี้สำรองไว้ใช้ในอนาคต

Control Bits

ตารางที่ 1 รายละเอียดส่วน Control Bits

Urgent Pointer (URG)	ถ้าบิตนี้ถูกเปิดใช้ ผู้รับจะแปลความหมายในส่วนฟิลด์ Urgent Pointer
Acknowledgement (ACK)	ถ้าบิตนี้ถูกเปิดใช้ จะมีกระบวนการส่งสัญญาณการตอบรับจะมีการใช้งาน
Push Function (PSH)	ถ้าบิตนี้ถูกเปิดใช้ ผู้รับจะส่งข้อมูลชุดนั้นไปที่โปรแกรมประยุกต์ทันที ตัวอย่างการใช้บิตนี้ คือ การส่ง Control BREAK ที่โปรแกรมประยุกตNต้องการทันที
Reset the Connection (RST)	ถ้าบิตนี้ถูกเปิดใช้ จะเป็นสัญญาณที่บอกให้ปลายทางทราบว่าต้นทางได้ยกเลิกการติดต่อ และให้คิวข้อมูลถูกนำกลับไปเริ่มต้นนับใหม่
Synchronize (SYN)	ใช้ในกระบวนการเริ่มต้นของการติดต่อ ในช่วงเริ่มการสร้างเส้นทาง เพื่อบอกปลายทางถึงความพยายามที่ต้นทางจะทำการ "synchronize"
No More Data from Sender (FIN)	ถ้าบิตนี้ถูกเปิดใช้ จะเป็นการบอกปลายทางว่า ต้นทางได้ส่งข้อมูลจนหมดแล้ว

Window

ตัวเลข 16 บิต ระบุการใช้ ขนาดของวินโดว์ในการควบคุมการนำพาข้อมูล (flow control)ในการบอก ตัวเลขนี้บอกถึงปริมาณขนาดของวินโดว์ ที่ปลายทางยอมรับก่อนที่จะส่ง ACK มีขนาดสูงสุด

65,535 ไบต์ แต่หากใช้ร่วมกับ options ที่เรียกว่า “window scale” จะสามารถเพิ่มขนาดของวินโดว์ได้อีก

Checksum

ใช้ในการตรวจสอบความผิดพลาดของชุดข้อมูล โดยมีความสามารถอยู่ในระดับหนึ่ง

Urgent pointer

ตัวเลข 16 บิตนี้ จะบอกปลายทางเมื่อไบต์สุดท้ายของชุดข้อมูลเร่งด่วนในเซกเมนต์สิ้นสุด

Options

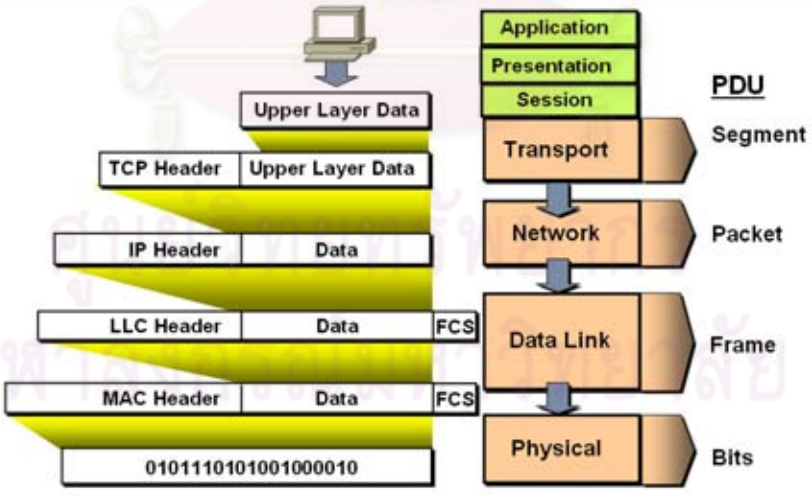
ใช้เป็นฟังก์ชันเพิ่มเติม ส่วนนี้จะมีขนาดไม่แน่นอน แต่จะมีขนาดได้ไม่เกิน 40 ไบต์ เนื่องจากขนาดของฟิลด์ header length มี 4 บิต โดยทั่วไปจะเป็น maximum segment size (MMS) ใช้ในการบอกต้นทางถึงปริมาณข้อมูลที่ปลายทางสามารถรับได้ ส่วน options อื่นๆ จะใช้ในเทคนิคการทำ flow control หรือ congestion control

Padding

เนื่องจาก options ทำให้ส่วนหัวของ TCP มีขนาดไม่แน่นอน ทำให้ต้องมีฟิลด์ที่บอกจุดสิ้นสุดของส่วน header ประกอบไปด้วยศูนย์จำนวน 8 บิต ก่อนที่จะเข้าสู่ส่วนของข้อมูล

Data

เป็นส่วนของข้อมูลที่รับมาจากโปรแกรมประยุกต์ส่งไปที่โปรแกรมประยุกต์ปลายทาง เมื่อรวมเข้ากับส่วนหัว TCP จะถูกเรียกเป็นเซกเมนต์ ถ้าเป็นเซกเมนต์ของสัญญาณการตอบรับ (ACK) จะไม่มีส่วนนี้



รูปที่ 2 การทำงานของ TCP เมื่อเทียบชั้นการจำลองใน OSI

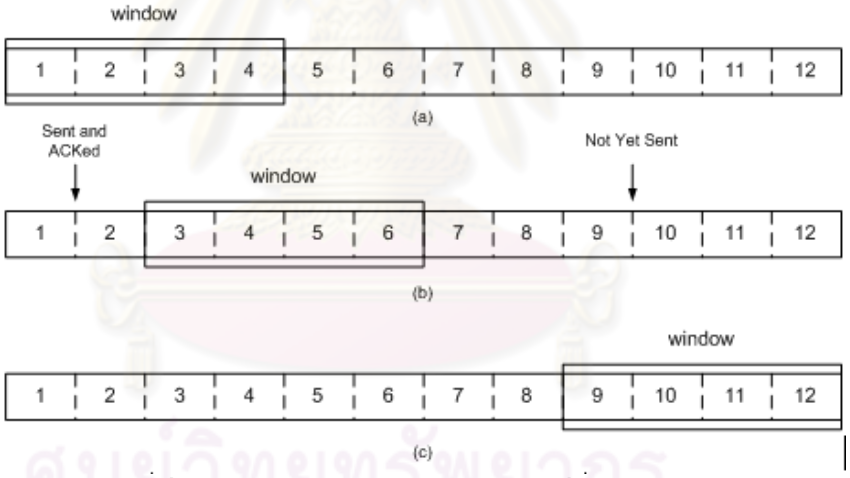
โดยโครงสร้างของ Protocol Data Unit (PDU) เมื่อเทียบกับชั้นการจำลอง OSI ในโครงข่ายแสดงการทำงานของ โปรโตคอล TCP ได้ดังรูปที่ 2 โดยการทำงานข้อมูลจากชั้น Application,

Presentation และ Session เมื่อมาถึงยังชั้นของ Transport โปรโตคอล TCP จะทำการแบ่งข้อมูลที่จัดส่งมาให้เป็นชุดมีขนาดที่เหมาะสมโดยเรียกเป็นเซกเมนต์ข้อมูล หลังจากนั้นจะเพิ่มส่วนหัวของ TCP ซึ่งมีขนาด 20 ไบต์ โดยที่ส่วนหัวมีรายละเอียดต่างๆดังที่กล่าวมา (รวมทั้ง Sequence Number (SEQ) เพื่อจัดลำดับข้อมูลที่ส่งไปพร้อมกับส่วน Checksum ที่ตรวจสอบความถูกต้องของชุดข้อมูล) หลังจากนั้นส่งไปยังชั้น Network เพื่อทำการเพิ่มส่วนหัวของ IP ต่อไป

2.1.2 ระบบการควบคุมของ TCP

1). การควบคุมการนำพาข้อมูล (Flow Control)

มีจุดประสงค์หลักในการควบคุมอัตราการส่งข้อมูลจากต้นทางไปยังปลายทาง เพื่อให้ได้อัตราการส่งข้อมูลที่ความเร็วสูงและให้ได้สมรรถนะที่ดี โดยโปรโตคอลได้ใช้เทคนิคการปรับขนาดวินโดว์ (Sliding window) เพื่อควบคุมอัตราการไหลของข้อมูลระหว่างการแลกเปลี่ยนข้อมูลระหว่างต้นทางและปลายทาง โดยฝั่งผู้รับจะตั้งค่าขนาดวินโดว์ที่ต้องการ (RCVWND) ไปในส่วนสัญญาณการตอบกลับ (acknowledgement) เพื่อแจ้งให้ผู้ส่งทราบโดยมีขนาดเป็นไปตามขนาดบัฟเฟอร์ที่ว่างอยู่ เมื่อผู้ส่งทำการส่งข้อมูลมายังผู้รับจะได้ไม่เกิดการโอเวอร์โหลดของข้อมูลที่บัฟเฟอร์ขึ้น



รูปที่ 3 การใช้เทคนิค sliding window เพื่อนำพาข้อมูล

การควบคุมการนำพาข้อมูลนี้มีลักษณะการทำงานเช่นเดียวกับโปรโตคอลในลำดับชั้น data link เพื่อกำหนดจังหวะให้กับฝั่งส่ง เพื่อเป็นการขยายวินโดว์เพื่อเพิ่มขนาดบัฟเฟอร์ ใน TCP อัตราการส่งจะถูกกำหนดโดยอัตราการได้รับของ ACK ก่อนหน้า

2). การควบคุมการคับคั่ง (Congestion Control)

มีจุดประสงค์หลักในการควบคุมความคับคั่งข้อมูลในโครงข่ายโดยผู้ส่งสามารถปรับเปลี่ยนขนาดการคับคั่งวินโดว์ (CWND) เพื่อป้องกันว่าผู้ส่งจะไม่ส่งข้อมูลจนเกิดโอเวอร์โหลดบนโครงข่าย ในขณะที่ RCVWND ใช้เพื่อป้องกันไม่ให้ผู้ส่งส่งข้อมูลมามากจนโอเวอร์โหลดที่บัฟเฟอร์ผู้รับ ผู้ส่ง

สามารถที่จะส่งข้อมูลด้วยค่าน้อยที่สุดระ หว่าง RCVWND และ CWND ได้อย่างมีประสิทธิภาพโดยไม่ต้องมีการส่งสัญญาณตอบกลับ

3). การควบคุมการผิดพลาด (Error Control)

ส่วนนี้เป็นส่วนสำคัญที่สร้างความน่าเชื่อถือในการทำงานของโปรโตคอล จะประกอบด้วยการตรวจสอบและแก้ไขการผิดพลาด โดย TCP จะใช้สัญญาณการตอบกลับของแพ็คเกจ, การนับเวลา และการส่งข้อมูลซ้ำเป็นส่วนประกอบในการควบคุมความผิดพลาด การตอบกลับจะใช้รูปแบบการตอบกลับแบบสะสม (cumulative ACK) ซึ่งหมายความว่าเมื่อมีแพ็คเกจเกิดการสูญหายขึ้น มันจะป้องกันไม่ให้การตอบกลับดำเนินการต่อไป และวินโดว์ไม่สามารถเพิ่มขนาดได้จนกว่าแพ็คเกจที่สูญหายจะได้รับการแก้ไข ในกลไกลการปรับขนาดของวินโดว์ ที่จริงแล้วเป็นการเชื่อมโยงการควบคุมการนำพาข้อมูล การควบคุมการคับคั่งและการควบคุมการผิดพลาดเข้าด้วยกัน กลไกลนี้ประสิทธิภาพการทำงานจะลดลงเมื่อเกิดผลกระทบจากการสูญหายของข้อมูลซึ่งเกิดคับคั่งและการผิดพลาดของแพ็คเกจในโครงข่ายพร้อมๆกัน

2.1.3 การควบคุมความคับคั่งของ TCP

ความคับคั่งเกิดขึ้นเมื่อมีความต้องการใช้ทรัพยากรของโครงข่ายมากเกินกว่าที่สามารถใช้ได้ เมื่อภาระของผู้ใช้มีจำนวนมากเกินกว่าความจุที่โครงข่ายจะสามารถรองรับได้ โครงข่ายก็จะเกิดความคับคั่ง ในรูปที่ 4 แสดงสมรรถนะของโครงข่ายเทียบกับภาระที่เพิ่มขึ้น จากรูปแสดงค่าอัตราการส่งผ่านและความล่าช้าเทียบกับภาระ ขณะเมื่อมีภาระต่ำ อัตราการส่งผ่านจะเพิ่มขึ้นหรือลดลงตามภาระ แต่ค่าเวลาการตอบสนอง (response time) หรือ ค่าความล่าช้า (delay) จะเพิ่มขึ้นไม่มาก ขณะเมื่อภาระมากเท่ากับความจุของโครงข่าย ภาระที่เพิ่มขึ้นจะทำให้ค่าความล่าช้าเพิ่มขึ้นตามไปด้วยแต่อัตราการส่งผ่านจะไม่เพิ่มขึ้น การเพิ่มขึ้นมาจากการล่าช้าของสัญญาณเนื่องมาจากการเข้าคิวในโครงข่าย จุดที่ภาระมากเท่ากับความจุของโครงข่าย เรียกว่าจุด “knee” เมื่อภาระมากเกินไปจะเป็นผลให้ค่าอัตราการส่งผ่านลดลงและค่าความล่าช้าเพิ่มขึ้นอย่างมาก เพราะเนื่องมาจากคิวเกิดการล้นจึงทำให้เกิดการส่งแพ็คเกจใหม่ จุดที่ทำให้อัตราการส่งผ่านลดลงเรียกว่าจุด “cliff” โดยวัตถุประสงค์ของการหลีกเลี่ยงความคับคั่ง (congestion avoidance) คือการรักษาการทำงานของโครงข่ายให้อยู่ที่จุด “knee” ขณะที่วัตถุประสงค์ของการควบคุมความคับคั่ง (congestion control) คือรักษาจุดการทำงานของโครงข่ายให้อยู่ทางด้านซ้ายมือของจุด “cliff” โดย Jain [9] ได้อธิบายถึง 2 องค์ประกอบของการหลีกเลี่ยงความคับคั่ง ดังนี้

1). Network policies

มีหลายองค์ประกอบของโครงข่ายที่สามารถวัดระดับภาระ Jain แสดงให้เห็นว่าความยางคิวเฉลี่ยเป็นสิ่งที่แสดงให้เห็นขนาดภาระของโครงข่าย จากนั้น Jain ยังได้เสนอให้ใช้บิทในส่วนหัวของแพ็คเกจเป็นตัวชี้ว่าเกิดความคับคั่งโดยถูกเซตที่เราท์เตอร์ แหล่งกำเนิดจะใช้บิตที่ป้อนกลับมานี้ทำการ

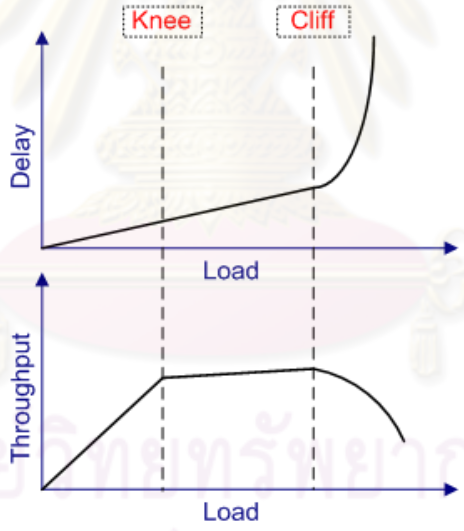
เพิ่มหรือลดอัตราการส่ง (หรือขนาดของวินโดว์) TCP ได้ใช้การเพิ่มค่าความล่าช้า [10] และ การสิ้นสุดเวลา timeout [11] เป็นเครื่องแสดงว่าขณะนี้เกิดความคับคั่ง การป้อนกลับให้ผู้ใช้งานไม่ว่าจะเป็นต้นทางหรือปลายทางจากโครงข่ายเพื่อเป็นการปรับตัวหรืออัตราการส่งให้เหมาะสมต่อโครงข่ายขณะนั้นๆ

2). End system policies

วัตถุประสงค์ของการศึกษาที่ end system เพื่อให้การทำงานอยู่ในสถานะอยู่ตัว (steady state) โครงข่ายที่อยู่ในสถานะอยู่ตัวจะมีจำนวนแพ็คเกจที่อยู่ในโครงข่ายเท่ากับความจุของโครงข่าย (network capacity) ที่กำหนดให้เป็นผลคูณของแบนด์วิดท์กับความล่าช้า (bandwidth-delay product) [12]

$$Capacity\ (bits) = bandwidt\ (bit/\sec) \times round\ trip\ time\ (\sec) \qquad (2.1)$$

สำหรับการหลีกเลี่ยงความคับคั่งของระบบ มีวิธีการ โดยหลักคือ ทำอย่างไรจึงจะทำให้โครงข่ายอยู่ในสถานะอยู่ตัวอย่างรวดเร็ว และ ทำอย่างไรที่จะรักษาสภาวะโครงข่ายให้อยู่ในสถานะอยู่ตัวถึงแม้ว่าจะเกิดการเปลี่ยนแปลงสภาวะของโครงข่าย



รูปที่ 4 การเปรียบเทียบ ภาระ (Load) กับ ความล่าช้า (Delay) และอัตราการส่งผ่าน (Throughput)

ระบบต้นทางจะรู้สภาพโครงข่ายที่มีการเปลี่ยนแปลงได้จากกลไกลการป้อนกลับ เมื่อได้รับการป้อนกลับจากโครงข่าย ระบบต้นทางก็จะทำการตัดสินใจแล้วทำการเปลี่ยนอัตราการส่งข้อมูลให้เพิ่มขึ้นหรือลดลง และระยะเวลาในการเปลี่ยนอัตราส่งข้อมูล ขณะที่เกิดความคับคั่งขึ้นจะเกิดการสูญหายของแพ็คเกจขึ้น ดังนั้นจึงต้องพยายามให้โครงข่ายเข้าสู่สถานะอยู่ตัวอย่างรวดเร็วที่สุด และให้ระบบต้นทางได้ทำการส่งแพ็คเกจที่เกิดสูญหายมาใหม่

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ความคับคั่งในโครงข่ายอินเทอร์เน็ตสร้างปัญหาใหญ่ให้กับระบบ คือทำให้อัตราการส่งผ่านในช่องสัญญาณลดลง และเพิ่มเวลาในการตอบสนอง ในโครงข่ายแบบสวิตซ์ชุดข้อมูล (packet switching) จะทำการเลือกเส้นทางโดยการหาเส้นทางแบบพลวัต (dynamic routing) ที่สามารถกระจายความคับคั่งออกไป ไม่ให้มารวมกันอยู่ที่จุดเดียว แต่วิธีการเหล่านี้เหมาะที่ใช้กับโครงข่ายภาคพื้นดินที่มีช่องสัญญาณเชื่อมต่อของ load ที่หลากหลาย โดยมีปริมาณของข้อมูลที่เพิ่มขึ้นและลดลงอย่างรวดเร็ว รวมทั้งระบบการประมวลผลประสิทธิภาพสูงมากกว่าในระบบโครงข่ายดาวเทียม แต่ตัวโปรโตคอล TCP มีเครื่องมือที่สามารถนำมาใช้ควบคุมความคับคั่งโดยใช้กลไกการปรับขนาดของวินโดว์ (sliding window) ซึ่งเหมาะสมกับโครงข่ายในรูปแบบปลายทางไปยังปลายทาง (end to end) และถูกนำมาพัฒนาเป็นกลไกการทำงานที่แพร่หลายขึ้น

2.2 มาตรฐานการควบคุมความคับคั่งของ TCP

กลไกการควบคุมความคับคั่งเป็นกลไกควบคุมหนึ่งในการสร้างความน่าเชื่อถือของ TCP โดยให้ความเชื่อมั่นว่าข้อมูลที่ส่งไปยังปลายทางได้รับครบถ้วนไม่เกิดการโอเวอร์โหลดบนโครงข่าย มีความสำคัญในการนำมาใช้ในโครงข่ายเพื่อเพิ่มประสิทธิภาพในการส่งผ่านข้อมูลที่ดี มาตรฐานการควบคุมความคับคั่งของทีซีพี [13] ใช้อัลกอริทึมมาตรฐาน 4 ตัว คือ Slow Start, Congestion Avoidance, Fast Retransmit และ Fast Recovery ซึ่งอัลกอริทึมเหล่านี้จะถูกนำมาใช้เพื่อทำการปรับปริมาณข้อมูลที่สามารถป้อนเข้าไปในโครงข่ายและส่งซ้ำชุดข้อมูลในส่วนที่มีการสูญหายเกิดขึ้น

ผู้ส่ง TCP ใช้ตัวแปรที่มีความสำคัญ 2 ตัวในกระบวนการควบคุมความคับคั่ง ตัวแปรแรกคือขนาดความคับคั่งวินโดว์ (cwnd) ซึ่งเสมือนเป็นขอบเขตบนของปริมาณข้อมูลที่ผู้ส่งสามารถป้อนเข้าไปในโครงข่ายก่อนที่จะได้รับสัญญาณการตอบรับ (ACK) ค่าของ cwnd ถูกจำกัดโดยขนาดวินโดว์ที่ประกาศไว้โดยผู้รับ ขนาดของความคับคั่งวินโดว์สามารถเพิ่มขนาดขึ้นหรือลดขนาดลงในระหว่างการส่งผ่านข้อมูลโดยขึ้นกับการอ้างอิงปริมาณคับคั่งในโครงข่ายปัจจุบัน ตัวแปรที่ 2 คือ slow start threshold (ssthresh) ตัวแปรนี้เป็นตัวที่กำหนดว่าจะใช้อัลกอริทึมใดในการเพิ่มค่าของ cwnd ถ้า cwnd มีค่าน้อยกว่า ssthresh อัลกอริทึม Slow Start จะถูกนำมาใช้เพื่อเพิ่มค่าของ cwnd ถ้าหากว่า cwnd มีค่ามากกว่าหรือเท่ากับ ssthresh อัลกอริทึม congestion avoidance ก็จะถูกนำมาใช้ ในสภาวะเริ่มต้น ssthresh คือขนาดของวินโดว์ที่ประกาศไว้โดยผู้รับ เมื่อตรวจพบความคับคั่งข้อมูลขึ้น ค่า ssthresh จะถูกตั้งขึ้นใหม่ โดยอัลกอริทึมควบคุมความคับคั่งมาตรฐานทั้ง 4 แสดงรายละเอียดได้ดังนี้

2.2.1 Slow Start

slow start เป็นกระบวนการบน TCP ที่นำไปใช้ในการควบคุมอัตราการส่งข้อมูล บนฝั่งของผู้ส่งซึ่งจะมีความสัมพันธ์โดยตรงกับอัตราสัญญาณการตอบรับ (ACK) ที่ผู้รับส่งกลับมา หรืออาจกล่าวได้ว่า

slow start มีหน้าที่ในการประเมินหาแบนด์วิดท์ของช่องสัญญาณ เพราะก่อนทำการส่งข้อมูล ผู้ส่งไม่ทราบข้อมูลใดๆ เกี่ยวกับช่องสัญญาณเลย ดังนั้นวิธีที่เหมาะสมกว่าคือการค่อยๆเพิ่มขนาดของวินโดว์ทีละน้อย โดยเป็นไปตามสมการ

$$awnd = \min[credit, cwnd] \quad (2.2)$$

เมื่อ awnd คือวินโดว์ที่ได้รับอนุญาติ มีหน่วยเป็นเซกเมนต์ เป็นจำนวนที่ TCP ให้ส่งได้โดยไม่ต้องได้รับ ACK

credit คือจำนวนของ credit ยังไม่ได้ใช้ ซึ่งได้รับจาก ACK ล่าสุด มีหน่วยเป็นเซกเมนต์เมื่อแต่ละ ACK ที่เข้ามาโดยคำนวณด้วย window / segment size (จำนวนของข้อมูลที่ TCP จะตอบรับ)

slow start จึงเป็นกระบวนการที่ขาดไม่ได้ในการเริ่มส่งข้อมูลเมื่อการติดต่อครั้งแรกเริ่มต้นขึ้น อัลกอริทึม slow start จะตั้งให้ขนาดการคับคั่งวินโดว์ (cwnd) มีค่าเริ่มต้นเป็น 1 เท่าของขนาดเซกเมนต์ที่มากที่สุด (maximum segment size) แล้วส่งไปยังผู้รับ เมื่อมี ACK ตอบกลับมาจากผู้รับ ผู้ส่งจะเพิ่ม cwnd ขึ้นอีก 1เซกเมนต์ ในทุกครั้งที่ได้รับ ACK ตอบกลับมา ดังนั้นผู้ส่งสามารถมีขนาดของวินโดว์อย่างน้อยเท่ากับค่าต่ำสุดระหว่างขนาดการคับคั่งวินโดว์ที่กำหนดขึ้นโดยผู้ส่ง กับ วินโดว์ที่กำหนดขึ้นโดยผู้รับ และขนาดของวินโดว์ที่ส่งออกไปจริงเรียก ว่า transmission window จะเห็นได้ว่า slow start ซึ่งมักทำให้เข้าใจผิดว่าเป็นการส่งข้อมูลแบบช้าๆ แต่ทีจริงแล้วไม่จำเป็นต้องส่งข้อมูลช้าๆ เสมอไป เมื่อโครงข่ายยังไม่เกิดความคับคั่ง หรือยังมีเวลาการตอบสนองของโครงข่ายที่ดี เช่น สามารถเพิ่มขนาดของวินโดว์เป็น 2 เซกเมนต์ หลังจากได้ ACK จากการส่งในครั้งแรก และเพิ่มขนาดของวินโดว์ เป็น 4 เซกเมนต์ หลังจากได้ ACK จากการส่งในครั้งสอง และเพิ่มขนาดของวินโดว์เป็น 8 เซกเมนต์ หลังจากได้รับ ACK กล่าวคือมีอัตราการเพิ่มเป็นสองเท่า อย่างนี้ไปเรื่อยๆ จนกระทั้งถึงค่าสูงสุดที่ค่า ssthresh โดยขนาด cwnd ที่เพิ่มขึ้นนี้จะเพิ่มขึ้นเป็นรูปแบบเอ็กโพแนนเชียลเมื่อเทียบกับรอบเวลาการเดินทางระหว่างผู้ส่งและผู้รับ

2.2.2 Congestion Avoidance

การเกิดความคับคั่งในโครงข่ายเกิดได้จากการป้อนข้อมูลเข้าไปในโครงข่ายมากจนเกินความจุของช่อง สัญญาณหรือเปลี่ยนโครงข่ายจากที่มีความจุช่องสัญญาณขนาดใหญ่ไปยังโครงข่ายที่มีความจุสัญญาณขนาดเล็กกว่าหรือเกิดจากสัญญาณอินพุตหลายสัญญาณเข้ามาในเราต์เตอร์และเอาท์พุตมีความจุต่ำกว่าสัญญาณอินพุทรวม เป็นผลให้เกิดความคับคั่งได้ กระบวนการ Congestion Avoidance เป็นกระบวนการที่คอยหลีกเลี่ยงการเกิดความคับคั่งหลังจากความจุของโครงข่ายมีค่าอยู่ในระดับนึง (อ้างอิงโดยใช้ค่า ssthresh) ในระหว่างกระบวนการนี้ ฝั่งส่งจะเพิ่ม cwnd ขึ้น $\frac{1}{cwnd}$ เมื่อได้รับ ACK ซึ่งเสมือนการเพิ่ม cwnd ขึ้น 1 MSS สำหรับทุกๆรอบเวลาการเดินทาง (round trip time หรือ RTT) การเพิ่มขนาด cwnd จะเป็นรูปแบบเชิงเส้นซึ่งช่วยให้ฝั่งส่งตรวจสอบความเป็นไปได้ของแบนด์วิดท์อย่างช้าๆ และ

cwnd จะถูกลดลงครึ่งหนึ่งของค่าปัจจุบันเมื่อผู้ส่งได้รับสัญญาณการตอบกลับซ้ำ (DupACKs) โดยที่จำนวนครั้งปกติอยู่ที่ 3 ครั้ง ณ จุดนี้ TCP จะสรุปว่าแพ็คเกจเกิดการสูญหายเนื่องจากการคับคั่งบนช่องสัญญาณ

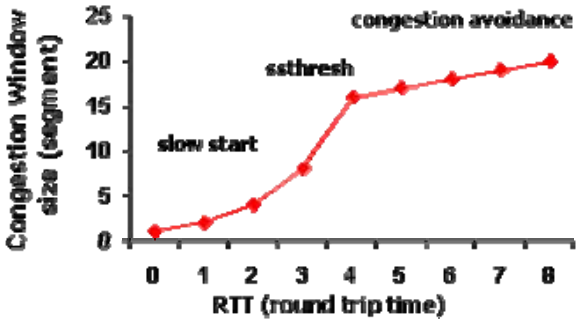
การสูญหายของแพ็คเกจอาจเกิดขึ้นได้จากสองกรณี คือ เกิดขึ้นจากความคับคั่งของโครงข่ายจุดใดจุดหนึ่งในช่วงต้นทางถึงปลายทาง หรือเกิดจากความเสียหายของสัญญาณข้อมูลในระหว่างทาง (link error) ข้อสัณนิษฐานเบื้องต้นเมื่อเกิดการสูญหายของแพ็คเกจ คือ ให้มีสาเหตุมาจากความคับคั่งในโครงข่าย เพราะความน่าจะเป็นในการเกิดความเสียหายของสัญญาณข้อมูลมีต่ำมาก จุดบ่งชี้การสูญหายของแพ็คเกจอยู่สองประการ คือ หมดคาบเวลาการรอ ACK ทางฝั่งผู้ส่ง (timeout) หรือ ผู้ส่งได้รับสัญญาณการตอบกลับซ้ำ

เมื่อ timeout ขึ้น ต้นทางจะว่างหรือไม่ส่งข้อมูลประมาณ 1 รอบเวลาทำงาน แพ็คเกจที่มีการ timeout สมมติว่ามีการสูญหายแพ็คเกจเนื่องมาจากความคับคั่ง ช่วงเวลาที่ว่างไม่ส่งข้อมูลจนถึงเวลา timeout จะต้องมีช่วงเวลาที่เพียงพอสำหรับแก้ไขความคับคั่ง เวลา time out ไม่ได้เป็นเพียงแค่เครื่องบ่งบอกว่าเกิดการคับคั่งเท่านั้น ยังเป็นเสมือนตัว trigger ให้แหล่งกำเนิดทำการส่งข้อมูลใหม่ด้วยโดย เมื่อเกิดการ timeout แล้วแหล่งกำเนิดจะตั้งค่า ssthresh ให้มีค่าเป็นครึ่งหนึ่งของ cwnd โดยกำหนดไว้เป็น

$$ssthresh = \max\left\{2, \min\left\{\frac{cwnd}{2}, rcvwnd\right\}\right\} \tag{2.3}$$

เมื่อ cwnd<ssthresh และแหล่งกำเนิดเข้าสู่ระยะ Slow Start ดังนั้นแหล่งกำเนิดจึงเริ่มทำการส่งข้อมูลใหม่ (retransmit) และ เพิ่ม cwnd ทีละหนึ่งแพ็กเกจทุกครั้งที่ได้รับ ACK

ถ้ามีการสูญเสียแพ็คเกจหนึ่งและถ้าบัฟเฟอร์ด้านรับไม่พอ ด้านส่งจะได้รับ ACK ซ้ำ และกู้ข้อมูลเดิมที่สูญเสียกลับมาจากความคับคั่งโดยกระบวนการ Fast Retransmission และ Fast Recovery ตามลำดับ โดยทั่วไป TCP จะมีการกำหนดค่า retransmission timeout ไว้ค่าหนึ่งโดยแหล่งกำเนิด TCP จะประมาณค่ารอบเวลาการทำงาน (round trip time:RTT) ของการเชื่อมต่อโดยการวัดเวลา ระหว่างการส่งแพ็คเกจจนถึงได้รับ ACK ของแพ็คเกจกลับมา โดยถูกคำนวณเป็นฟังก์ชั่นโดยประมาณของค่าเฉลี่ยและค่า mean-deviation ของ RTT เมื่อเกิดการสูญเสียเนื่องจากความคับคั่ง เนื่องจากตัวนับเวลาของ TCP นั้นมีค่าหยาบอาจทำให้มีการสูญเสียเวลาในขณะที่รอการ trig ก่อนที่แหล่งกำเนิดส่งข้อมูลออกไปตามจำนวนที่ window อนุญาติให้ส่งได้ แหล่งกำเนิดไม่ยอมส่งข้อมูลใหม่เมื่อได้รับ ACKs ซึ่งเป็นผลให้ช่องสัญญาณอาจจะว่าเป็นระยะเวลานานและทำให้การใช้งานไม่มีประสิทธิภาพ



รูปที่ 5 การปรับขนาด cwnd ของอัลกอริทึม slowstart และ congestion avoidance

2.2.3 Fast Retransmission

เนื่องจาก ค่า Retransmission time out (RTO) ถูกใช้โดยด้านผู้ส่งเพื่อกำหนดเวลาในการส่งเซกเมนต์ซ้ำนั้นโดยปกติจะมีค่ามากกว่าเวลารอบเวลาการทำงานจริง(RTT)มาก เป็นผลให้การประเมินค่าไม่แม่นยำเนื่องจาก RTO ถูกคำนวณโดยพื้นฐานการคาดเดา RTT ที่ผ่านไปแล้ว ถ้าในโครงข่ายมีเวลาประวิงเพิ่มขึ้นค่าประเมินของ RTT จะน้อยกว่าค่า RTT จริง และทำนองเดียวกันถ้าความหน่วงเกิดการแกว่งตัวขึ้นลงอย่างรวดเร็วค่าประเมินของ RTT ยิ่งเกิดความไม่น่าเชื่อถือ จากสาเหตุเหล่านี้ ถ้าเซกเมนต์สูญหาย TCP อาจจะช้าที่จะส่งชุดข้อมูลซ้ำ สมมติว่า A ส่งเซกเมนต์จำนวนหนึ่งแล้ว เซกเมนต์แรกในจำนวนนี้ได้สูญหายไปขณะที่ send window ยังไม่เต็ม และ RTO ยังไม่หมดเวลา A ก็จะยังคงส่งต่อไปโดยไม่ต้องรอรับ ACK ที่ B ได้รับข้อมูลทุกๆ เซกเมนต์ ยกเว้นเซกเมนต์แรก แต่ต้องเก็บทุกเซกเมนต์ ที่เข้ามาลงบัฟเฟอร์จนกระทั่งได้รับเซกเมนต์ที่หายไป B ไม่สามารถยกเลิกบัฟเฟอร์ของตัวเองได้จนกว่าจะได้รับเซกเมนต์ที่หายไป ถ้าการส่งซ้ำของเซกเมนต์ที่หายไปจาก A ใช้เวลานาน ทำให้ B มีบัฟเฟอร์ไม่พอที่จะเก็บเซกเมนต์ที่เข้ามาทำให้เกิดการคัดเซกเมนต์ทิ้ง จึงได้มีการนำเสนอวิธีการ Fast Retransmission และ Fastrecovery ขึ้นเพื่อแก้ไขปัญหาดังกล่าว

Fast Retransmission ใช้ประโยชน์จากฎเกณฑ์ใน TCP ที่ว่า ถ้า TCP ได้รับเซกเมนต์ผิดลำดับ TCP จะส่ง ACK สำหรับเซกเมนต์สุดท้ายที่ได้รับเรียงลำดับออกไปทันที และจะยังคงส่ง ACK นี้ซ้ำในทุกๆเซกเมนต์ที่เข้ามา จนกระทั่ง segment ที่หายไปนั้นเข้ามาเติมช่องว่างในบัฟเฟอร์สำเร็จ

เมื่อผู้ส่งได้รับสัญญาณการตอบรับซ้ำ (duplicate ACKs) จะตีความว่าอาจมีสาเหตุมาจากการ

1) การเรียงหมายเลขชุดข้อมูล (sequence number) ที่ทางฝั่งผู้รับไม่สามารถเรียงกันได้ภายในเวลาที่ผู้รับใช้ในการเรียงหมายเลขที่มีค่าไม่มากนักหรือเซกเมนต์มีการหน่วงไปผิดลำดับ

2) เซกเมนต์ข้อมูลเกิดการสูญหาย

ให้มีการสรุปไว้ว่า หากผู้ส่งได้รับ ACK ซ้ำหนึ่งหรือสองครั้ง สาเหตุน่ามาจากการเรียงหมายเลขยังไม่สมบูรณ์เท่านั้น แต่หากได้รับ ACKs ซ้ำมากกว่าสามครั้ง สรุปว่ามีสาเหตุมาจากการสูญ

หายของแพ็คเกจ ดังนั้นหากผู้ส่งได้รับ ACKs ซ้ำสามครั้ง ผู้ส่งจะทำการส่งชุดข้อมูลที่แจ้งข้อผิดพลาดซ้ำ (retransmission) ทันทีโดยไม่รอให้ถึงการหมดคาบเวลา timeout

2.2.4 Fast Recovery

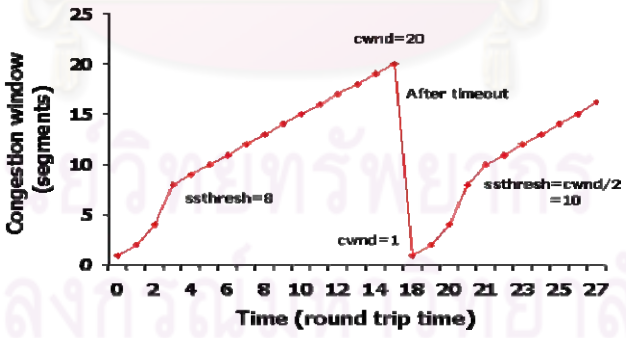
เมื่อ TCP ทำการส่งเซกเมนต์ซ้ำโดยใช้กระบวนการ Fast Retransmission เป็นข้อบ่งชี้ว่าทางผู้ส่งได้รับรู้ว่ามีเซกเมนต์นั้นหายไป ทั้งที่ยังไม่หมดเวลารอการตอบรับเซกเมนต์นั้น ดังนั้นผู้ส่งควรทำการหลีกเลี่ยงความแออัดที่เกิดขึ้น โดยวิธีการ Slow Start / Congestion Avoidance เมื่อเกิด time out คือ ตั้งค่าให้ ssthresh = cwnd/2 และให้ cwnd มีค่า=1 และเริ่มขั้นตอน exponential slow start จนกระทั่ง cwnd = ssthresh และเพิ่ม cwnd เป็นแบบเชิงเส้น แต่วิธีกการนี้ยังไม่ดีพอในกรณีที่มี ACK ซ้ำตอบมามากๆ ดังนั้นจึงได้มีการเสนอกลไกล Fast Recovery ที่มีหลักการทำงานดังนี้

1. เมื่อได้รับ ACK ซ้ำ 3 ครั้ง
   a. ตั้งค่า ssthresh = cwnd/2
   b. ส่งซ้ำเซกเมนต์ข้อมูลที่หายไป
   c. ตั้งค่า cwnd = ssthresh+3

เหตุผลของการบวก 3 เข้ากับค่า ssthresh คือเป็นจำนวนของเซกเมนต์ที่ส่งไปยังปลายทางแล้ว

2. ในแต่ละครั้งที่ได้รับ ACK ซ้ำเพิ่มขึ้น (ของเซกเมนต์เดิม) ให้เพิ่ม cwnd ขึ้นอีก 1 และส่ง segment ข้อมูลต่อไปถ้ามีการบวกนี้ เพราะรู้ว่ามีจำนวน segment ที่ส่งไปและได้รับแล้วเพิ่มอีก 1

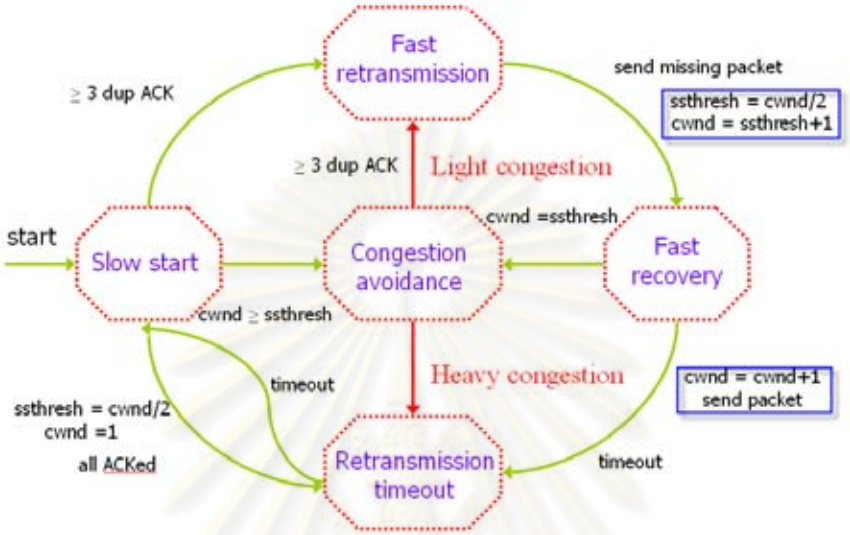
3. เมื่อได้รับ ACK ถัดไปที่เป็นการตอบรับเซกเมนต์ใหม่ ให้ตั้งค่า cwnd=ssthresh



รูปที่ 6 การปรับขนาดค่า cwnd และ ssthresh เมื่อหมดคาบเวลาการรอ ACK

เมื่อ fast retransmission ใช้ปรับขนาดการคับคั่งวินโดว์โดยจะตั้งค่า ssthresh เป็นขนาดครึ่งนึงของ cwnd เกิดขึ้น โดยมีข้อบ่งชี้จากการรับ duplicate ACK ทำให้ผู้ส่งรู้ว่าการส่งข้อมูลยังคงส่งไปที่ผู้รับได้ โดยไม่เกิดความคับคั่ง เนื่องจากได้ปรับตั้งค่า ssthresh ขึ้นใหม่แล้วและ duplicate ACK จะเกิดขึ้นได้เมื่อผู้รับได้รับ TCP เซกเมนต์ที่ส่งไปแล้ว นั้นเป็นสิ่งยืนยันได้ว่าไม่มีการเกิดความคับคั่งใน

โครงข่าย กระบวนการfast recovery จึงเพิ่มความเร็วในขั้นตอนแก้ไขข้อมูล ซึ่งจะพิจารณาDupACKs เป็นแบบ ACKs ปกติและขยายขนาด cwnd ขึ้น กระบวนการ fast retransmit จะสิ้นสุดเมื่อฝั่งส่งได้รับการตอบกลับของ ACK ปกติของลำดับข้อมูลก่อนหน้านี้ที่มีการสูญหาย ทำให้แทนที่จะลดอัตราการส่งข้อมูลและกลับไปที่กระบวนการ slow start ก็ทำเพียงเข้าสู่กระบวนการ congestion avoidance



รูปที่ 7 ไดอะแกรมกลไกลการทำงานมาตรฐานการควบคุมการคับคั่ง

2.3 แบบแผนควบคุมความคับคั่งของทีซีพีรุ่นต่างๆ

โปรโตคอลทีซีพีเป็นโปรโตคอลที่มีรูปแบบการทำงานซับซ้อนและได้มีการพัฒนาขึ้นมาอย่างต่อเนื่อง หลังจาก TCP Tahoe ได้ถูกพัฒนาขึ้นจากเวอร์ชั่น 4.3 BSD (Berkeley Software Distribution) เมื่อปี คศ.1988 ต่อมาหลังจากนั้นได้พัฒนาเป็น TCP Reno ในปี 1990 จนประสบผลสำเร็จและมีการใช้งานกันเป็นที่แพร่หลาย และยังคงมีการพัฒนารุ่นต่างๆตามมาอย่างต่อเนื่อง เพื่อให้มีประสิทธิภาพในการทำงานร่วมกับโครงข่ายต่างๆ โดยแสดงรุ่นที่มีความสำคัญได้ดังนี้

TCP Tahoe [2]

ถูกพัฒนาขึ้นในปี คศ.1988 โดยอ้างอิงด้วยกลไกลการควบคุมความคับคั่งของ Van Jacobson โดยใช้หลักการ conservation of packets เช่น ถ้าการเชื่อมต่อทำงานในความจุที่เป็นไปได้ของช่องสัญญาณ เมื่อแพ็คเกจหนึ่งไม่สามารถป้อนเข้าไปในโครงข่าย แพ็คเกจนั้นก็จะถูกปลดออกจากชุดข้อมูล พัฒนาขึ้นโดยใช้พื้นฐานของ ACK เพื่อเป็นสัญญาณ clock ให้ผู้ส่งทำการส่งออกของแพ็คเกจ อย่างไรก็ตามยังพบปัญหาในการสร้างความเชื่อมั่นเพื่อรักษาดุลยภาพของข้อมูลในช่องสัญญาณ

TCP Reno [14]

ถูกพัฒนาขึ้นใน คศ.1990 โดยยังคงใช้พื้นฐานการทำงานจาก Tahoe เช่น slow starts และ การส่งซ้ำเมื่อหมดคาบเวลา มีการพัฒนาขึ้นโดยเพิ่มกลไกลประมวลผลที่มีความสามารถยิ่งขึ้น กรณีเมื่อเกิดการสูญหายแพ็คเกจจะสามารถตรวจพบได้อย่างรวดเร็วโดยไม่ทำการตัดกระบวนการส่งข้อมูล (เข้าสู่ขั้นตอนสิ้นสุดคาบเวลาการส่ง) ประกอบด้วยกลไกลที่สำคัญ 4 กลไกล คือ slow start, congestion avoidance, fast retransmit และ fast recovery โดย TCP จะใช้ตัวแปรสำคัญสองตัวคือ congestion window(cwnd) และ slow-start threshold (ssthresh) เพื่อกำหนดปริมาณข้อมูลที่ควรป้อนเข้าสู่โครงข่าย ข้อได้เปรียบของ TCP Reno คือการส่งซ้ำที่มีการปรับตัว และกลไกลการควบคุมความคับคั่ง

TCP SACK [15]

เป็น TCP ที่เพิ่มส่วน Selective Acknowledgment เป็นส่วนขยายจาก TCP Reno โดยยังคงใช้กลไกลของ slow-start และ fast-retransmit ของ TCP Reno มีการเพิ่มส่วนของ SACK option ที่บรรจุจำนวนของ SACK blocks ซึ่งจะแสดงความไม่ต่อเนื่องและการจัดลำดับของชุดข้อมูลที่ผู้รับได้รับ TCP SACK ยังคงใช้พื้นฐานการควบคุมความคับคั่งของ TCP Reno มีข้อได้เปรียบที่สามารถทำการกู้ข้อมูลจากการสูญหายของแพ็คเกจจำนวนมากได้โดยไม่ต้องเข้าสู่ขั้นตอนการสิ้นสุดคาบเวลาส่ง ซึ่งจะช่วยปรับปรุงอัตราการส่งผ่านในช่องสัญญาณที่มีอัตราผิดพลาดบิตมากได้

TCP Vegas [16]

พัฒนาขึ้นโดย มหาวิทยาลัยอริโซน่า university of Arisona ในปีคศ.1994 โดยปรับปรุงมาจากโค้ดของ Reno พัฒนาด้วยทฤษฎีการวัดแบบ proactive ที่มีประสิทธิภาพมากกว่าวิธี reactive ที่ใช้งานอยู่ โดยจะควบคุมความคับคั่งของวินโดว์ด้วยวิธีการสำรวจการเปลี่ยนแปลงของรอบเวลาการทำงานของเซกเมนต์ข้อมูลที่ได้ส่งไป และจะทำการกำหนดบัฟเฟอร์พิเศษให้กับผู้ส่งเพื่อใช้ส่งข้อมูลในสภาวะปัจจุบัน ในกลไกล slow start จะอนุญาติให้ผู้ส่งเพิ่ม cwnd แบบ exponential ในบางรอบเวลาเท่านั้น ทำให้เกิดปัญหาการล่าช้าในการส่งข้อมูลกับช่องสัญญาณที่มีเวลาการประวิงสูง ประสิทธิภาพของ TCP Vegas ในโครงข่ายแบบมีสายได้ถูกทดสอบกันอย่างแพร่หลายและให้ผลเป็นที่น่าพอใจ แต่ในโครงข่ายดาวเทียมที่แสดงใน [17] พบว่าประสิทธิภาพยังคงด้อยกว่า TCP Reno เมื่อช่องสัญญาณมีการประวิงเวลาเดินทางสูง

TCP Newreno [13]

พัฒนาเพิ่มเติมจาก reno โดยพัฒนากระบวนการ fast retransmit และ fast recovery ให้มีประสิทธิภาพดีขึ้นเพื่อให้สามารถตรวจสอบแพ็คเกจที่สูญหายจำนวนมากได้ ในกลไกล fast retransmit มีการทำงานเช่นเดียวกับ reno แต่ในกลไกล fast recovery จะอนุญาติให้ส่งซ้ำแพ็คเกจจำนวนมากได้

โดยจะยังไม่ออกจากกระบวนการ fast recovery จนกระทั่งทุกชุดข้อมูลที่ส่งในกระบวนการนี้ได้รับการตอบกลับ และผู้ส่งจะเข้าสู่กระบวนการ fast retransmission ก็ต่อเมื่อเกิดการสิ้นสุดคาบเวลา หรือผู้ส่งได้รับ ACK จำนวน 3 ครั้ง

TCP Peach [18]

TCP Peach ถูกออกแบบมาสำหรับโครงข่ายดาวเทียม โดยมีกลไกล 4กลไกลคือ sudden start, congestion avoidance, fast retransmission และ rapid recovery โดยกลไกล congestion avoidance และ fast retransmission ยังอ้างอิงกับมาตรฐานของ TCP เดิม แต่ได้ทำการปรับปรุง slow start เป็น sudden start และ fast recovery เป็น rapid recovery ใช้ dummy segment เข้าเป็นตัวช่วยในการทำงาน ซึ่งเป็น segment ที่มีความสำคัญต่ำในชั้น IP และ เป็น segment ที่ไม่มีข้อมูลบรรจุภายใน ถูกสร้างขึ้นโดยผู้ส่งเพื่อทำหน้าที่ประเมิณหาทรัพยากรในโครงข่าย ถ้าในโครงข่ายเกิดความคับคั่งขึ้น แพ็คเกจในชั้น IP ที่บรรจุ segment นี้ จะถูกขจัดออกไปเป็นอันดับแรก โดยไม่ทำให้อัตราส่งผ่านของผู้ส่งลดลงเนื่องจากข้อมูลไม่ได้ถูกขจัดไป ถ้าในโครงข่ายไม่มีความคับคั่ง segment ส่วนนี้สามารถส่งไปถึงยังผู้รับ โดยผู้รับสามารถทำการแยก data และ dummy segment ออกจากกันได้

2.4 ข้อตกลงเกี่ยวกับทราฟฟิกสำหรับแพ็กเกจสวิตชิ่ง(Traffic Contract)

การให้บริการที่แตกต่างกันบนระบบโครงข่ายแพ็กเกจสวิตชิ่งที่ให้บริการการเชื่อมต่อแบบต่อเนื่อง สำหรับรูปแบบการใช้งาน (Application) ที่แตกต่างกันนั้น จำเป็นต้องมีข้อตกลงของทราฟฟิกที่เป็นมาตรฐานเดียวกันเพื่อให้ใช้เครือข่ายร่วมกันได้ โดยข้อตกลงทราฟฟิกต่างๆ มีดังนี้

- การแยกประเภทของการให้บริการ(Service Categories)
- ความต้องการคุณภาพการบริการ (QoS Requirment)
- ตัวแปรแสดงคุณลักษณะของทราฟฟิก (Traffic Decriptor)
- ข้อกำหนดที่ทราฟฟิกควรปฏิบัติตาม (Confermance Definition)

2.4.1 ประเภทของการบริการ (Service Categories)

การให้บริการการรับประกันคุณภาพการบริการ (Quality of Service) แก่รูปแบบการใช้งานที่แตกต่างกันเป็นสิ่งที่ยากและซับซ้อน ตัวอย่างเช่น สัญญาณเสียง (Voice) ซึ่งต้องการการรับประกันเวลาประวิง (Time Delay) ในการสื่อสารแต่กลับไม่ต้องการการรับประกันการสูญหายของแพ็กเกจแต่สำหรับข้อมูล (Data) กลับต้องการรับประกันการสูญหายของแพ็คเกจโดยไม่คำนึงถึงการรับประกันเวลาประวิงของการส่งมากนัก ในขณะที่รูปแบบการใช้งานอื่นๆอาจจะต้องการคุณภาพการบริการทั้งการสูญหายของข้อมูลและเวลาประวิงในการส่งข้อมูลด้วย ดังนั้นเพื่อให้ง่ายต่อการจัดการจึงได้แบ่ง

ประเภทของการบริการออกเป็น 4 ประภท โดยแบ่งตามระดับความต้องการคุณภาพการบริการที่แตกต่างกัน

- การให้บริการด้วยอัตราความเร็วคงที่ (Constant Bit Rate:CBR)
- การให้บริการด้วยอัตราความเร็วไม่คงที่ (Variable Bit Rate:VBR)
- การให้บริการด้วยอัตราความเร็วตามสภาพพร้อมใช้งาน (Available Bit Rate:ABR)
- การให้บริการด้วยอัตราความเร็วที่ไม่มีการรับประกัน (Unspecified Bit Rate:UBR)

สำหรับการให้บริการด้วย CBR และ VBR นั้น จะมีการสำรองแบนด์วิดท์สำหรับการเชื่อมต่อนั้นในกระบวนการการควบคุมการร้องขอทรัพยากรเพื่อสร้างการเชื่อมต่อ (Call Admission Control:CAC) โดยจะสำรองแบนด์วิดท์เท่ากันตลอดระยะการเชื่อมต่อ โดยปกติการเชื่อมต่อจะไม่ได้ใช้แบนด์วิดท์ทั้งหมดตลอดเวลา ดังนั้นเพื่อเพิ่มประสิทธิภาพการใช้แบนด์วิดท์จึงได้เกิดการให้บริการแบบปรับเปลี่ยนค่าแบนด์วิดท์ได้ในระหว่างการเชื่อมต่อ นั่นก็คือการให้บริการทราฟฟิกแบบ ABR และแบบ UBR

การให้บริการด้วยอัตราความเร็วคงที่ (CBR Service)

สำหรับการใช้งานแบบกำหนดระยะเวลาโต้ตอบ เช่น ข้อมูลวีดีโอ (Video) และ ออดิโอ (Audio) นั้นต้องการรับประกันคุณภาพการบริการในเรื่องของเวลาประวิง และการรับประกันการกระเพื่อมของเวลาประวิง (Jitter) เป็นอย่างมาก ซึ่งการใช้งานรูปแบบนี้ทราฟฟิกจะมีลักษณะขึ้นๆ ลงๆ ไม่สม่ำเสมอ (Bursty) แต่เครือข่ายจะทำการสำรองแบนด์วิดท์ให้ด้วยอัตราความเร็วสูงสุดตลอดระยะเวลาการเชื่อมต่อ ถึงแม้ว่าการเชื่อมนั้นจะไม่จำเป็นที่จะใช้แบนด์วิดท์ด้วยอัตราความเร็วสูงสุดตลอดเวลา ดังนั้นเพื่อเพิ่มประสิทธิภาพการใช้แบนด์วิดท์จึงได้เกิดการให้บริการแบบปรับเปลี่ยนค่าแบนด์วิดท์ได้ในระหว่างการเชื่อมต่อ นั่นก็คือการให้บริการทราฟฟิกแบบ ABR และ UBR

การให้บริการด้วยอัตราเร็วคงที่ (CBR)

สำหรับการใช้งานแบบกำหนดระยะเวลาโต้ตอบ เช่น ข้อมูลวีดีโอ(Video) และ ออดิโอ(Audio) นั้นต้องการรับประกันคุณภาพการบริการในเรื่องของเวลาประวิง และการรับประกันการกระเพื่อมของเวลาประวิง (Jitter) เป็นอย่างมาก ซึ่งการใช้งานในรูปแบบนี้ทราฟฟิกจะมีลักษณะขึ้นๆ ลงๆ ไม่สม่ำเสมอ (Bursty) แต่เครือข่ายจะทำการสำรองแบนด์วิดท์ให้ด้วยอัตราความเร็วสูงสุดตลอดระยะเวลาการเชื่อมต่อ ถึงแม้ว่าการเชื่อมต่อนั้นไม่จำเป็นที่จะใช้แบนด์วิดด้วยอัตราความเร็วสูงสุดตลอดก็ตาม

การให้บริการด้วยอัตราความเร็วคงที่นั้น ได้รับการออกแบบมาเพื่อการใช้งานในรูปแบบที่มีการกำหนดเวลาโต้ตอบ ซึ่งต้องการคุณภาพการบริการที่มีอัตราการสูญหายของแพ็คเกจและเวลาประวิงต่ำ ค่าคุณลักษณะทราฟฟิกที่ต้องการใช้คืออัตราความเร็วสูงสุด (Peak Rate) โดยที่ทราฟฟิกอาจจะไม่ใช้แบนด์วิดท์ทั้งหมดที่เครือข่ายสำรองให้ แต่เมื่อใดที่ทราฟฟิกต้องการใช้แบนด์วิดท์มากๆ ก็จะสามารถ

ใช้งานได้ทันที ทั้งนี้เพื่อให้ได้รับคุณภาพการบริการเรื่องเวลาประวิงและการกระเพื่อมของเวลาประวิงนั่นเอง โดยปกติแล้วการให้บริการรูปแบบนี้จะมีการสำรองแบนด์วิดท์ด้วยอัตราความเร็วสูงสุดของทราฟฟิก ทำให้ระบบเครือข่ายไม่ได้รับมัลติเพล็กซ์ซิ่งเกนสำหรับการให้บริการรูปแบบนี้

ประเภทการให้บริการด้วยอัตราความเร็วคงที่ถูกกำหนดขึ้นมาสำหรับการสื่อสารผ่านสายลวดทองแดงหรือสายใยแก้ว โดยข้อมูลแต่ละบิตจะถูกส่งเข้าในสายสื่อสารทางปลายด้านหนึ่งและออกมาที่ปลายอีกด้านหนึ่งของสายโดยไม่มีการตรวจสอบความผิดพลาด ไม่มีการควบคุมการไหลหรือกระบวนการอื่นใดเกิดขึ้นเลย วิธีการนี้ได้รับความนิยมในการนำไปใช้กับระบบโทรศัพท์ปัจจุบัน ทั้งนี้เนื่องจากการส่งข้อมูลประเภทการสนทนาด้วยเสียงของมนุษย์นั้นไม่ต้องการความถูกต้องบริบูรณ์ แต่ต้องการความเร็วอย่างสม่ำเสมอในการรับ-ส่งข้อมูลและเป็นประเภทที่ใช้จังหวะสัญญาณนาฬิกาจังหวะเดียวกัน ตัวอย่างของข้อมูลได้แก่การส่งสัญญาณเสียงและภาพเคลื่อนไหวที่ไม่ผ่านการบีบอัดข้อมูล

การให้บริการด้วยอัตราความเร็วไม่คงที่(VBR Service)

การให้บริการนำส่งข้อมูลด้วยอัตราเร็วไม่คงที่แบ่งออกเป็น 2 รูปแบบคือ แบบกำหนดเวลาโต้ตอบ (Real-Time:rt-VBR) และแบบไม่กำหนดเวลาโต้ตอบ (Non Real-Time:nrt-VBR)

- แบบกำหนดเวลาโต้ตอบ(Real-time):rt-VBR)

การให้บริการแบบกำหนดเวลาในการตอบโต้ถูกออกแบบมาสำหรับทราฟฟิกที่มีอัตราความเร็วในการรับส่ง-ข้อมูลที่ไม่คงที่ และมีการกำหนดเวลาสำหรับข้อมูลที่จะต้องได้รับภายในเวลาอันจำกัด เช่น การให้บริการประชุมทางอิเล็กทรอนิกส์ที่ต้องส่งสัญญาญข้อมูลภาพเคลื่อนไหวแบบมีการบีบอัดข้อมูล เป็นต้น เนื่องจากเทคโนโลยีที่มีใช้ในปัจจุบันคือ MPEG (Moving Picture Expert Group) และอื่นๆได้ใช้เทคนิคการส่งภาพโดยการสร้างภาพเบื้องต้น (Based frame) ตามด้วยชุดของภาพที่เป็นภาพเฉพาะส่วนที่มีการเคลื่อนไหว ดังนั้นอัตราการส่งข้อมูลจึงไม่คงที่ระบบจึงต้องมีความสามารถในการส่งชุดข้อมูลเหล่านี้ด้วยความเร็วแตกต่างกันโดยไม่ทำให้เกิดการสะดุดของภาพที่ปรากฏขึ้นที่ผู้ใช้นั่นคือจะต้องมีการบริหารการถ่วงเวลา ในระบบอย่างเข้มงวด

- แบบไม่กำหนดเวลาโต้ตอบ (Non Real-time:nrt-VBR)

การให้บริการแบบไม่กำหนดเวลาในการโต้ตอบจะไม่มีการกำหนดเวลาการโต้ตอบที่เข้มงวดมากนัก เช่นการส่งภาพเคลื่อนไหวไปพร้อมกับจดหมายอิเล็คทรอนิกส์ หรือการใช้โปรแกรมประเภทการค้นหาเวป(Web browser) ซึ่งข้อมูลที่ถูกส่งมาอาจเกิดการชะงักได้เป็นครั้งคราว

การให้บริการด้วยอัตราความเร็วตามสภาพพร้อมใช้งาน (ABR Service)

การให้บริการด้วยอัตราความเร็วตามสภาพพร้อมใช้งานหรือ ABR ได้รับการออกแบบมาใช้งานในระบบเครือข่ายที่มีปริมาณการไหลเวียนค่อนข้างสูงทำให้ไม่ทราบอัตราการส่งข้อมูลที่แน่นอน ตัวอย่างเช่น องค์กรที่มีการใช้สายเช่าจำนวนหนึ่ง โดยปกติอัตราความเร็วของสายเช่าจะต้องมากพอที่จะ

สามารถใช้งานในช่วงที่มีการส่งข้อมูลในปริมาณสูงสุดในแต่ละวัน แต่ในช่วงเวลาอื่นสายเช่าแทบไม่ได้ใช้งานเลย

บริการแบบ ABR ช่วยแก้ปัญหานี้เพื่อไม่ต้องใช้สายเช่าที่มีช่องสัญญาณกว้างมาก ที่มีราคาแพงมากด้วย เช่นอาจใช้สายเช่าที่มีช่องสัญญาณขนาด 5 ล้านบิตต่อวินาทีสำหรับองค์กรที่มีการส่งสัญญาณสูงสุดที่ 10 ล้านบิตต่อวินาที ระบบจะสามารถรับประกันในการส่งข้อมูลขนาด 5 ล้านบิตต่อวินาทีตลอดเวลาโดยจะสามารถให้บริการส่งข้อมูลขนาด 10 ล้านบิตต่อวินาทีเมื่อต้องการใช้และระบบเครือข่ายมีแบนด์วิดท์เหลือและให้บริการได้ แต่ก็ไม่รับประกันให้ว่าจะสามารถส่งข้อมูลได้ด้วยอัตราเร็วสูงสุดที่ต้องการหรือไม่

บริการแบบ ABR เป็นบริการเดียวที่มีความสามารถในการส่งข้อมูลกลับไปยังผู้ส่งเพื่อขอให้ส่งข้อมูลช้าลงเมื่อเกิดความคับคั่งขึ้นในระบบเครือข่าย ถ้าผู้ส่งทำตามคำแนะนำแล้วจะทำให้อัตราการสูญหายของข้อมูลอยู่ในเกณฑ์ต่ำ ข้อมูลที่รอการนำส่งจะอยู่ในสภาพเตรียมพร้อมซึ่งสจะถูกนำส่งในทันทีที่สามารถทำได้และจะต้องรอต่อไปถ้ายังไม่สามารนำส่งข้อมูลได้

โดยทั่วไปการส่งข้อมูลแบบ ABR จะใช้สำหรับการส่งข้อมูลที่ต้องการการรับประกันคุณภาพการบริการ เช่น การใช้งานมีโอกาสที่จะเกิดการสูญหายของข้อมูลต่ำและมีความผิดพลาดน้อย (Low Probability of Loss and Error) โดยบางรูปแบบการใช้งานก็มีความต้องการการรับประกันเวลาประวิงด้วย แต่จะไม่ให้ความสำคัญมากเท่ากับการสูญหายและความผิดพลาดของข้อมูล ดังนั้นแหล่งกำเนิดข้อมูลจึงมีการปรับเปลี่ยนอัตราความเร็วทราฟฟิกตามอัตราการเกิดความคับคั่งในเครือข่ายเพื่อให้เกิดการสูญหายของแพ็คเกจน้อย และมีการใช้แบนด์วิดท์ที่เหมาะสม

การให้บริการด้วยอัตราความเร็วที่ไม่มีการรับประกัน (UBR Service)

การบริการแบบสุดท้ายคือการให้บริการด้วยอัตราเร็วที่ไม่มีการรับประกันหรือ UBR ซึ่งจะไม่มีการรับประกันอัตราความเร็วในการส่งข้อมูล และไม่บอกสถานะความคับคั่งของระบบ โดยสวิตซ์ทุกตัวจะรับแพ็คเกจที่ส่งมาทั้งหมดและจะจัดส่งแพ็คเกจเหล่านั้นในทันทีที่ทำได้ ถ้าเกิดความคับคั่งขึ้นแพ็คเกจแบบ UBR จะถูกลบทิ้งโดยไม่มีการแจ้งกลับไปยังผู้ส่งและระบบก็ไม่ได้คาดหวังว่าผู้ส่งจะส่งข้อมูลช้าลง บริการประเภทนี้เป็นที่น่าสนใจเพราะมีอัตราค่าบริการที่ถูกที่สุด จึงเหมาะสำหรับโปรแกรมประยุกต์ที่มีการควบคุมข้อผิดพลาดด้วยตัวเอง เช่น การจัดส่งแฟ้มข้อมูลผ่านเครือข่าย การส่งจดหมายอิเล็คทรอนิคส์ เป็นต้น

2.4.2 ตัวแปรแสดงคุณภาพของการให้บริการ (Quality of Service Parameters)

คุณภาพการให้บริการจะถูกแสดงด้วยตัวแปรต่างๆที่แสดงถึงประสิทธิภาพความต้องการของการเชื่อมต่อนั้น ตัวแปรที่ใช้วัดประสิทธิภาพของเครือข่ายโดยทำการพิจารณาระหว่างโหนดปลายทาง (End System) กับโครงข่าย (Network) แบ่งออกเป็น 2 กลุ่มคือ

ตัวแปรประสิทธิภาพที่อยู่ในข้อตกลงทราฟฟิก

- อัตราการสูญหายของข้อมูล (Packet Loss)
- เวลาประวิงมากที่สุดของการนำส่ง (Maximum Transfer Delay)
- การกระเพื่อมมากที่สุดของเวลาประวิง (Packet Delay Deviation)

ตัวแปรประสิทธิภาพที่ไม่อยู่ในข้อตกลงทราฟฟิก

- อัตราการผิดพลาดของข้อมูล (Packet Error Ratio)
- อัตราการปฏิเสธการเชื่อมต่อ (Blocking Ratio)
- อัตราการส่งข้อมูลผิดเนื่องจากความผิดพลาดของส่วนหัวของแพ็คเกจ (Misinsertion Rate)

อัตราการสูญหายข้อมูล (Packet Loss Ratio:PLR)

การสูญหายของแพ็คเกจข้อมูลเกิดจากการล้นจากบัฟเฟอร์เนื่องจากมีการมาถึงพร้อมๆกันของทราฟฟิกที่อัตราความเร็วสูง (Bursty Traffic) จากหลายการเชื่อมต่อ และอัตราการสูญหายของข้อมูลอาจจะเกิดจากความผิดพลาดของอุปกรณ์ และการป้องกันไม่ให้เกิดความเสียหายกับสวิตซ์โดยอัตราการสูญหายของข้อมูลจะพิจารณาสำหรับแต่ละการเชื่อมต่อได้ด้วยสมการ(2.4)

$$PLR = \frac{LostPacket}{TotalTrasmittedPackets} \tag{2.4}$$

เมื่อแพ็คเกจที่สูญหาย (Loss Packet) ไปประกอบด้วย

- แพ็คเกจที่เดินทางไม่ถึงปลายทาง
- แพ็คเกจที่ส่งผิดเนื่องจากความผิดพลาดของส่วนหัวของแพ็คเกจ
- แพ็คเกจข้อมูลที่เกิดความเพี้ยน

และแพ็คเกจทั้งหมดที่ถูกส่ง (Total Transmitted Packets) คือจำนวนแพ็คเกจทั้งหมดที่ส่งไปภายในระยะเวลาหนึ่ง โดยอัตราการสูญหายของแพ็คเกจจะไม่รวมถึงแพ็คเกจที่ไม่ปฏิบัติตามค่าแสดงคุณลักษณะของทราฟฟิกที่ได้ประกาศไว้

เวลาประวิงของการส่งข้อมูล (Delay Parameters)

การวัดค่าเวลาประวิงในการส่งข้อมูล(Transfer Delay) นั้นวัดจากเวลาที่ผ่านระหว่างเวลาที่แพ็คเกจเดินทางออกจากแหล่งกำเนิดข้อมูล และแพ็คเกจเดินทางมาถึงปลายทาง เวลาประวิงในการนำส่งข้อมูลของเครือข่ายเป็นผลรวมของเวลาประวิงในการนำส่งข้อมูลในแต่ละโหนด ซึ่งประกอบด้วย

- เวลาประวิงที่เกิดจากการรอคิวและการส่งข้อมูลภายในโหนด (Internal queuing and transmission delay) คือเวลาที่ต้องการในการรอคิวและเวลาของการส่งบิตข้อมูลบนการ

เชื่อมต่อภายในโหนดนั้น โดยเวลาประวิงนี้จะขึ้นอยู่กับปริมาณทราฟฟิกในเครือข่ายและชนิดของอัลกอริทึมที่นำมาใช้จัดการคิว ซึ่งโครงสร้างภายในสวิตซ์ที่ต่างกันก็จะมีค่าเวลาประวิงของการสวิตซ์แพ็คเกจที่แตกต่างกันด้วย

- เวลาประวิงที่เกิดจากการรอคิวบนบัฟเฟอร์ขาออก (External Queuing) คือเวลาที่ใช้ในการรอคิวในบัฟเฟอร์บนการเชื่อมต่อขาออก (Output Link) โดยเวลาประวิงนี้จะขึ้นอยู่กับปริมาณทราฟฟิกในเครือข่ายและชนิดของอัลกอริทึมที่นำมาใช้ในการจัดการคิว
- เวลาประวิงที่เกิดจากการส่งบิตข้อมูลสู่ภายนอกโหนด (Transmission Delay) คือเวลาที่ใช้ในการส่งบิตข้อมูลบนการเชื่อมต่อขาออกจากโหนดสู่สื่อสัญญาณ โดยเวลาประวิงนี้จะขึ้นอยู่กับความเร็วในการส่งของโหนดและความเร็วของสายสื่อสัญญาณ
- เวลาประวิงของการแพร่กระจายบนสื่อสัญญาณ (Propagation Delay) คือเวลาที่ต้องการใช้สำหรับการแพร่ไปบนสายส่งสัญญาณ (Physical Media) โดยปัจจัยสำคัญสำหรับเวลาประวิงของการแพร่กระจายบนสื่อสัญญาณคือเวลาที่สัญญาณเดินทางระหว่างโหนด 2 โหนด ซึ่งขึ้นอยู่กับสื่อกลางที่ใช้และชนิดของสัญญาณแม่เหล็กไฟฟ้า เวลาประวิงของการแพร่กระจายบนสื่อสัญญาณสามารถคำนวณได้จากสมการ เมื่อ Propagation factor มีค่าระหว่าง 0.2 และ 0.6 ด้วยสมการ (2.5)

$$\text{Propagation Delay} = \frac{\text{Distance}}{\text{Propagation Factor} \times \text{Speed of Light}} \qquad (2.5)$$

- เวลาประวิงที่เกิดจากการประมวลผล(Processing Delay) เวลาที่เกิดจากการประมวลผลแพ็กเก็ต เช่น การวิเคราะห์และปรับปรุงส่วนหัวของแพ็กเกจ การหาข้อมูลเส้นทางตารางข้อมูลเส้นทาง (Routing Table) เป็นต้น

ตัวแปรแสดงลักษณะทราฟฟิก (Traffic Descriptors)

สำหรับตัวแปรที่ใช้แสดงคุณลักษณะของแต่ละทราฟฟิกที่เกิดจากแหล่งกำเนิดข้อมูลหนึ่งประกอบไปด้วย

- อัตราความเร็วสูงสุด (Peak Rate)
- อัตราความเร็วเฉลี่ย (Mean Rate)
- อัตราความเร็วต่ำสุด (Minimum Rate)
- กลุ่มข้อมูลขนาดใหญ่ที่สุดที่ถูกส่งต่อเนื่องกันด้วยอัตราความเร็วในการนำส่งสูงสุด (Maximum Burst Size)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

อัตราความเร็วสูงสุด (Peak Rate)

ตัวแปรอัตราความเร็วสูงสุด คือการกำหนดอัตราความเร็วสูงสุดที่ผู้ใช้วางแผนหรือคาดว่าจำเป็นต้องใช้ซึ่งอาจจะต่ำกว่าค่าสูงสุดที่เป็นไปได้ เช่น สายสื่อสารมีความสามารถในการส่งข้อมูลที่ความเร็วสูงสุด 1 ล้านแพ็คเกจต่อวินาที หรือ 1 แพ็คเกจในทุกๆ 1 ไมโครวินาที แต่ผู้ใช้อาจกำหนดอัตราความเร็วสูงสุดเป็น 250000 แพ็คเกจต่อวินาที หรือ 1 แพ็คเกจในทุกๆ 4 ไมโครวินาทีก็ได้

อัตราความเร็วเฉลี่ย (Mean Rate)

เป็นตัวแปรระบุความเร็วโดยเฉลี่ยของแพ็คเกจของการเชื่อมต่อ ตัวแปรนี้ใช้กับการให้บริการแบบ VBR โดยผู้ส่งจะต้องมีการประกาศตัวแปรในกระบวนการร้องขอจัดตั้งเส้นทางโดยอัตราความเร็วเฉลี่ยสจะแสดงความสัมพันธ์ได้ดังสมการ

$$MeanRate = \frac{\lambda\gamma}{\lambda + \mu} \tag{2.6}$$

เมื่อใช้แบบจำลองแหล่งจ่ายข้อมูลแบบเปิด-ปิด (On-Off Source Model) และตัวแปรต่างๆคือ

$MeanRate$ คืออัตราเร็วเฉลี่ยของทราฟฟิก มีหน่วย บิตต่อวินาที

$1/\lambda$ คือช่วงเวลาปิดเฉลี่ย มีหน่วยเป็น วินาที

$1/\mu$ คือช่วงเวลาเปิดเฉลี่ย มีหน่วยเป็น วินาที

γ คืออัตราความเร็วสูงสุด มีค่าคงที่ มีหน่วยเป็น บิตต่อวินาที

อัตราความเร็วต่ำสุด (Minimum Bit Rate)

คือการกำหนดค่าความเร็วต่ำสุดในการส่งข้อมูลที่ผู้ใช้สามารถยอมรับได้ และระบบเครือข่ายจะต้องสามารถรับประกันข้อมูลในอัตราความเร็วนี้ได้ มิฉะนั้นจะต้องปฏิเสธการจัดตั้งวงจรเสมือนนั้น สำหรับการให้บริการแบบ ABR จะมีการเปลี่ยแปลงอัตราความเร็วสูงสุด ถ้ายินยอมให้อัตราความเร็วต่ำสุดมีค่าเท่ากับศูนย์แล้ว การให้บริการแบบ ABR จะมีลักษณะเดียวกันกับบริการแบบ UBR

กลุ่มข้อมูลขนาดใหญ่ที่สุดที่ถูกส่งต่อเนื่องด้วยอัตราความเร็วสูงสุด (Maximum Burst Size:MBS)

คือขนาดของกลุ่มแพ็คเกจข้อมูลมากที่สุดที่ผู้ส่งสามารถส่งได้ที่อัตราความเร็วสูงสุด มีหน่วยเป็นบิต หรือแพ็คเกจ ซึ่งค่า MBS นี้จะสัมพันธ์กับช่วงเวลาที่แหล่งจ่ายหยุดการส่งแพ็คเกจข้อมูลด้วย สำหรับแบบจำลองแหล่งจ่ายข้อมูลแบบ เปิด-ปิด ความสัมพันธ์แสดงได้ดังสมการ(2.7)

$$MBS = \left(\frac{\gamma}{\mu}\right) \times \left(\frac{\lambda + \mu}{\mu}\right) \tag{2.7}$$

2.4.3 การควบคุมการร้องขอทรัพยากรเพื่อสร้างการเชื่อมต่อ (Call Admission Control:CAC)

หนึ่งในเรื่องที่สำคัญซึ่งยังไม่มีการแก้ไขปัญหาอย่างเป็นที่น่าพอใจก็คือ กระบวนการควบคุมการร้องขอทรัพยากรเพื่อสร้างการเชื่อมต่อ (Call Admission Control) เมื่อผู้ใช้ (User) ร้องขอการเชื่อมต่อใหม่ เครือข่ายจะต้องทำการตัดสินใจว่าจะยอมรับหรือปฏิเสธการเชื่อมต่อนั้น และถ้าเครือข่ายยอมรับการเชื่อมต่อ เครือข่ายจะต้องทำการสร้างเส้นทางและสำรองทรัพยากรหรือแบนด์วิดท์สำหรับการเชื่อมต่อนั้น สำหรับเครือข่ายเซอร์กิจสวิตซิ่งจะมีการแสดงผลของกระบวนการควบคุมการร้องขอทรัพยากรเพื่อสร้างการเชื่อมต่อในรูปของอัตราการปฏิเสธการร้องขอเชื่อมต่อกับอัตราการร้องขอทั้งหมด (Cannection Blocking Ratio) ซึ่งจะเกิดขึ้นเมื่อปริมาณแบนด์วิดท์ของการร้องขอมีค่ามากกว่าปริมาณแบนด์วิดท์ที่สามารถใช้งานได้

เนื่องจากทราฟฟิกแบบ VBR มีอัตราส่งที่ไม่แน่นอนและความต้องการแบนด์วิดท์เฉลี่ยจะเปลี่ยนแปลงไปตามจำนวนแหล่งจ่ายทราฟฟิก ดังนั้นจึงเป็นเรื่องยากในการแสดงลักษณะทราฟฟิกของการเชื่อมต่อนั้น ปัญหานี้มีการแก้ไขด้วยการสำรองแบนด์วิดท์ด้วยอัตราความเร็วสูงสุดการเชื่อมต่อ (Deterministic Multiplexing) ซึ่งจะเหมือนกับการให้บริการแบบ CBR อย่างไรก็ตามโดยอาศัยคุณสมบัติของทราฟฟิกแบบ VBR ซึ่งมีอัตราส่งที่ไม่แน่นอนแต่อยู่ภายใต้ขอบเขตที่จำกัด เมื่อมีการนำส่งร่วมกันของการเชื่อมต่อแบบ VBR จำนวนมากๆ อัตราแบนด์วิดท์ที่ต้องการสำหรับการนำส่งทราฟฟิกทั้งหมดนั้นมักจะมีค่าต่ำกว่าผลรวมปริมาณอัตราแบนด์วิดท์สูงสุดของทราฟฟิกทั้งหมดอยู่มาก ซึ่งวิธีการนี้ช่วยลดปริมาณการใช้แบนด์วิดท์ในระบบได้ แต่อาจจะทำให้เกิดการสูญหายแพ็คเกจหรือเกิดการประวิงเวลาที่มากขึ้นได้เมื่อแต่ละทราฟฟิกมีการนำส่งข้อมูลด้วยอัตราความเร็วสูงสุดพร้อมกัน โดยคุณภาพการบริการที่ลดลงนี้ถูกแสดงด้วยตัวแปรคุณภาพการบริการ (QoS Parameter)

จุดประสงค์ของการควบคุมการร้องขอทรัพยากรเพื่อสร้างการเชื่อมต่อก็คือ เพื่อรักษาคุณภาพการให้บริการของการเชื่อมต่อที่ให้บริการอยู่แล้วในระบบเครือข่าย ณ ขณะใดๆ โดยจำกัดจำนวนของการเชื่อมต่อการใช้งานเครือข่ายและพิจารณาปริมาณทรัพยากรที่เหลืออยู่กับปริมาณที่การร้องขอเชื่อมต่อต้องการ ซึ่งการพิจารณาว่าเครือข่ายจะยอมรับหรือปฏิเสธการร้องขอการเชื่อมต่อนั้นจำเป็นต้องพิจารณาหลายองค์ประกอบ เช่น พิจารณาจากพฤติกรรมของทราฟฟิกนั้น (Traffic Behavior) โดยดูจากค่าแสดงคุณลักษณะทราฟฟิก ได้แก่ อัตราการส่งข้อมูลสูงสุด อัตราการส่งข้อมูลเฉลี่ย ความยาวมากที่สุดของช่วงที่มีการส่งด้วยอัตราความเร็วสูงสุด และอัตราการส่งข้อมูลต่ำสุด การพิจารณาจากความต้องการคุณภาพการบริการของผู้ใช้ (Quality of Service Requirement) ได้แก่ อัตราการสูญหายของแพ็คเกจข้อมูลที่ยอมรับได้ เวลาประวิงที่ยอมรับได้ และค่าการกระเพื่อมของเวลาประวิงที่ยอมรับได้ เป็นต้น นอกจากนี้ยังมีการพิจารณาจากสถานะปัจจุบันของเครือข่าย เป็นการวัดสถานการณ์ใช้งานจริงของเครือข่ายในขณะนั้น โดยการวัดประสิทธิภาพของการใช้เครือข่ายเช่น การวัดอัตราการสูญหายของแพ็คเกจข้อมูล การวัดเวลาที่เกิดขึ้นในแต่ละโหนด การวัดความยาวลำดับคิว เป็นต้น แล้วนำข้อมูล

เหล่านี้ไปใช้พิจารณาเพื่อตัดสินใจ โดยเครือข่ายจะยอมรับการเชื่อมต่อเมื่อการเชื่อมต่อนั้นไม่มีผลกระทบใดๆ ต่อผู้ใช้งานอื่นๆในเครือข่าย

2.4.4 การจัดสรรแบนด์วิดท์แบบปรับตัวได้ (Dynamic Bandwidth Allocation)

การจัดสรรแบนด์วิดท์แบบปรับตัวได้นี้ ระบบเครือข่ายจะเป็นผู้ทำการจัดสรรอัตราแบนด์วิดท์ใหม่สำหรับทราฟฟิกใดๆ โดยแบ่งออกเป็นแบบที่มีการคำนวณและจัดการแบนด์วิดท์ก่อนการนำส่งหรือในระหว่างการนำส่งข้อมูล สำหรับแบบการคำนวณและจัดสรรแบนด์วิดท์ก่อนการนำส่งข้อมูลนั้นจะทำการเปลี่ยนปริมาณแบนด์วิดท์ตามแกนเวลา โดยใช้หลักการรู้ทราฟฟิกล่วงหน้าทั้งหมด ทำให้สามารถวางแผนในการจัดสรรปริมาณแบนด์วิดท์ตามลักษณะทราฟฟิกได้ โดยการใช้ประโยชน์จากการใช้บัฟเฟอร์ปลายทางซึ่งช่วยป้องกันให้ทราฟฟิกไปยังผู้รับอย่างราบรื่น

การจัดสรรแบนด์วิดท์แบบปรับตัวได้ แบบที่มีการคำนวณและจัดสรรแบนด์วิดท์ในระหว่างการนำส่งข้อมูล ซึ่งเป็นแนวทางในวิทยานิพนธ์นี้ วิธีการจะทำการวัดค่าต่างๆเพื่อใช้ในการทำนายลักษณะทราฟฟิกช่วงเวลาถัดไป เช่น การวัดอัตราการสูญหายของแพ็คเกจในขณะนั้น จากนั้นระบบเครือข่ายจะเป็นผู้ทำการจัดสรรอัตราการแบนด์วิดท์ใหม่แก่ทราฟฟิก โดยจะมีการพิจารณาทุกๆช่วงเวลาคงที่ค่าหนึ่ง ซึ่งจำนวนครั้งการปรับเปลี่ยนปริมาณแบนด์วิดท์จะขึ้นอยู่กับค่าที่วัดได้ข้างต้น สำหรับเทคนิคการวัดแบ่งออกเป็น 2 วิธี คือ การวัดค่าคุณลักษณะของแหล่งจ่าย และการวัดคุณภาพการให้บริการในขณะนั้น ซึ่งแต่ละวิธีต่างมีข้อดี ข้อเสีย ความซับซ้อนและประสิทธิภาพที่ต่างกันไปจากบทความวิจัย สามารถสรุปวิธีการหลักๆได้ดังนี้

การวัดคุณลักษณะของแหล่งจ่าย (Source Measurement)

หลักการของวิธีการนี้ คือการสังเกตุลักษณะของเฟรมข้อมูล ปัจจุบันเพื่อทำนายลักษณะของเฟรมถัดไป ด้วยสมการคณิตศาสตร์ เช่น neural network เป็นต้น ซึ่งมีข้อเสียคือ เป็นการคำนวณที่ซับซ้อน และจำนวนของข้อมูลควบคุมมีส่วนสำคัญมากต่อความถูกต้องและแม่นยำของผลการทำนาย

การวัดคุณภาพการให้บริการ (QoS Measurement)

Errin[19]เสนอวิธีการจัดสรรแบนด์วิดท์แบบปรับตัวได้ โดยการวัดอัตราการสูญหายของแพ็คเกจข้อมูลในขณะนั้น เทียบกับค่าอัตราการสูญหายแพ็คเกจข้อมูลที่ผู้ใช้ยอมรับได้ ถ้าอัตราการสูญหายที่เกิดขึ้นในขณะนั้นมากกว่าอัตราการสูญหายที่ผู้ใช้ยอมรับได้ ระบบเครือข่ายจะทำการเพิ่มปริมาณแบนด์วิดท์ให้แก่ทราฟฟิก ในทางตรงกันข้ามถ้าอัตราการสูญหายที่เกิดขึ้นในขณะนั้นน้อยกว่าอัตราการสูญหายที่ผู้ใช้สามารถยอมรับได้ ระบบอาจจะลดปริมาณแบนด์วิดท์ของทราฟฟิกนั้นลง เพื่อนำแบนด์วิดท์ที่ไม่ถูกใช้งานไปให้บริการแก่ผู้ใช้รายอื่นๆได้ ด้วยหลักการนี้ทำให้ระบบเครือข่ายมีประสิทธิภาพการใช้งานสูงขึ้น สำหรับหลักการของ Errin แสดงได้ดังต่อไปนี้

$$\alpha_{n+1} = \alpha_n + \frac{K}{\tau} \times \ln \frac{P_n}{Q_l} \tag{2.8}$$

$$\tau = \begin{cases} 1...if & P_{\mathrm{n}} > Q_l \\ 2...if & P_{\mathrm{n}} \le Q_l \end{cases} \tag{2.9}$$

$$P_n = \frac{L_n}{A_n} \tag{2.10}$$

เมื่อ P_n คืออัตราการสูญหายของแพ็คเกจในช่วงเวลาลำดับที่ n

Q_l คืออัตราการสูญหายของแพ็คเกจที่ผู้ใช้ยอมรับได้

α_n คืออัตราการแบนด์วิดท์ ณ ช่วงเวลาปัจจุบัน

α_{n+1} คืออัตราแบนด์วิดท์สำหรับช่วงเวลาถัดไป

K คือค่าคงที่ใช้ขยายผลของอัตราความคลาดเคลื่อน

A_n คือจำนวนของแพ็คเกจที่มาถึงในช่วงเวลาลำดับที่ n

L_n คือจำนวนของแพ็คเกจที่สูญหายไปในช่วงเวลาลำดับที่ n

เนื่องจากวิธีของ Errin ใช้อัตราการสูญหายของข้อมูลเปรียบเทียบกับอัตราการสูญหายที่ผู้ใช้สามารถยอมรับได้ ในการปรับเปลี่ยนปริมาณแบนด์วิดท์ กรณีที่เกิดการสูญหายมากกว่าอัตราการสูญหายที่ผู้ใช้สามารถยอมรับได้ ระบบเครือข่ายจะต้องเพิ่มปริมาณแบนด์วิดท์ให้แก่ทราฟฟิกเพื่อลดอัตราการสูญหายให้เป็นไปตาม QoS ที่กำหนดไว้ ซึ่งช่วงเวลาที่ระบบเครือข่ายวัดค่าอัตราการสูญหายของแพ็คเกจได้มากกว่าอัตราการสูญหายที่ผู้ใช้ยอมรับได้นี้ แพ็คเกจได้มีการสูญหายไปเรียบร้อยแล้วก่อนที่ระบบเครือข่ายจะมีการปรับเพิ่มปริมาณแบนด์วิดท์ให้แก่ทราฟฟิก ทำให้อัตราการสูญหายที่เกิดขึ้นจริงมีค่ามากกว่าอัตราการสูญหายที่ผู้ใช้สามารถยอมรับได้ ดังนั้นในกรณีนี้ระบบเครือข่ายจึงไม่สามารถรับประกันอัตราการสูญหายแพ็คเกจได้จริง

จึงได้มีงานวิจัย[20] ที่นำเสนอการคำนวณค่าแบนด์วิดท์ที่เหมาะสมที่ให้การรับประกันคุณภาพการบริการทั้งอัตราการสูญหายของข้อมูลและเวลาประวิงในการนำส่งข้อมูลซึ่งเหมาะสมกับการสื่อสารแบบการกำหนดเวลาโต้ตอบ ซึ่งมีรูปแบบการคำนวณที่ไม่ซับซ้อนและสามารถนำไปใช้งานได้จริง นั่นคือสำหรับบัฟเฟอร์ใดๆ เมื่อความยาวคิวที่เกิดขึ้นจริงมีค่ามากกว่าค่าความยาวคิวเฉลี่ยที่ผู้ใช้สามารถยอมรับได้ อาจทำให้เวลาประวิงของทราฟฟิกมีค่าเกินกว่าที่ผู้ใช้กำหนด ดังนั้นเมื่อเกิดเหตุการณ์เช่นที่กล่าวมาขึ้น ระบบเครือข่ายจะต้องเพิ่มปริมาณอัตราการจัดสรรแบนด์วิดท์แก่ทราฟฟิกนั้น เพื่อทำให้ความยาวคิวในบัฟเฟอร์ลดลง หลักการดังกล่าวสามารถสรุปเป็นวิธีการคำนวณค่าอัตราการจัดสรรแบนด์วิดท์แบบปรับตัวได้ ดังสมการ (2.11)

$$\alpha_{t+1} = \alpha_t + K \times \ln \frac{Q_t}{Q_D} \qquad (2.11)$$

เมื่อ α_{t+1} คืออัตราการแบนด์วิดท์ที่เหมาะสมสำหรับช่วงเวลาถัดไป (บิตต่อวินาที)

α_t คืออัตราการแบนด์วิดท์ในช่วงเวลาปัจจุบัน (บิตต่อวินาที)

Q_t คือความยาวคิวที่ ณ เวลาปัจจุบัน (บิต)

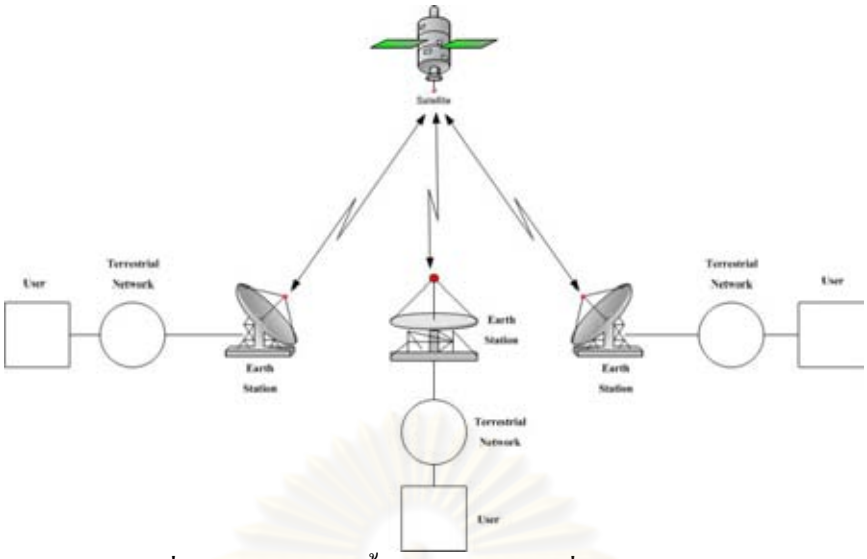
K คือค่าคงที่ใช้ขยายผลของอัตราความคลาดเคลื่อน (บิตต่อวินาที)

$\overline{Q}_D$ คือค่าความยาวคิวเฉลี่ยในบัฟเฟอร์ สำหรับการรับประกัน QoS (บิต)

อัตราแบนด์วิดท์ที่เหมาะสมที่คำนวณได้สำหรับช่วงเวลาถัดไปจะนำมาใช้เพื่อช่วยเพิ่มประสิทธภาพการใช้แบนด์วิดท์เพื่อให้อัตราการจัดสรรแบนด์วิดท์มีความสอดคล้องกันกับทราฟฟิกจริงมากขึ้น และเมื่อความยาวคิวที่เวลาปัจจุบัน t มีค่ามากกว่าค่าความยาวคิวมากที่สุดที่ผู้ใช้สามารถยอมรับได้ นั่นคือเกิดการ Bursty ในบัฟเฟอร์นั้น ระบบเครือข่ายจะต้องทำการเพิ่มอัตราการจัดสรรแบนด์วิดท์เพื่อเร่งให้แพ็คเกจถูกส่งออกมาจากบัฟเฟอร์โดยเร็ว มิเช่นนั้นแพ็คเกจอาจจะล้นบัฟเฟอร์และเกิดการสูญหายของข้อมูลได้ ในทางตรงกันข้ามถ้าความยาวคิวที่เวลา t มีค่าน้อยกว่า $\overline{Q}_D$ นั่นแสดงว่ามีการใช้แบนด์วิดท์ที่น้อยกว่าที่ทำการจัดสรรไว้กับระบบเครือข่าย ดังนั้นระบบเครือข่ายจะทำการลดอัตราการแบนด์วิดท์ลงตามอัตราความแตกต่างนั้นเพื่อปรับให้สอดคล้องกับอัตราการใช้งานจริงมากที่สุด ซึ่งช่วยเพิ่มประสิทธิภาพการใช้ทรัพยากรเครือข่ายและทำให้ระบบเครือข่ายสามารถรองรับผู้ใช้ได้จำนวนมากขึ้นด้วย

2.5 ระบบสื่อสารดาวเทียม (Satellite Systems)

ระบบสื่อสารดาวเทียม [21] มีส่วนประกอบพื้นฐานคือดาวเทียมที่ลอยอยู่ในอวกาศซึ่งดาวเทียมนี้จะสื่อสารเชื่อมโยงกับสถานีภาคพื้นดิน ดังแสดงในรูปที่ 8 มีหลักการทำงานคือผู้ใช้ส่งสัญญาณเบสแบนด์ (Baseband) ไปยังสถานีภาคพื้นดินโดยเครือข่ายสื่อสารภาคพื้นดิน (Terrestrial Network) ซึ่งอาจจะเป็นสายโทรศัพท์หรือสายสัญญาณที่สร้างขึ้นมาสำหรับการนี้โดยเฉพาะ เมื่อมาถึงสถานีภาคพื้นดินสัญญาณจะถูกมอดูเลต (Modulate) และขยายส่งขึ้นไปยังตัวดาวเทียม เมื่อดาวเทียมได้รับสัญญาณแล้วจะกรองเอาสัญญาณเฉพาะในช่วงความถี่ที่ต้องการแล้วจึงขยายสัญญาณและส่งกลับลงมายังระบบสื่อสารภาคพื้นดินอีกครั้งหนึ่งโดยทำหน้าเป็นตัวทวนซ้ำสัญญาณ(repeater) สถานีภาคพื้นดินเมื่อได้รับสัญญาณจากดาวเทียมนี้จะทำกระบวนการที่ย้อนกลับกับตอนที่ส่งสัญญาณขึ้นไปบนดาวเทียมคือจะขยายสัญญาณที่รับมาแล้วทำการดีมอดูเลต (Demodulate) กลับไปเป็นสัญญาณเบสแบนด์ และส่งต่อไปยังผู้ใช้โดยผ่านทางเครือข่ายการสื่อสารภาคพื้นดิน (Terrestrial Network)

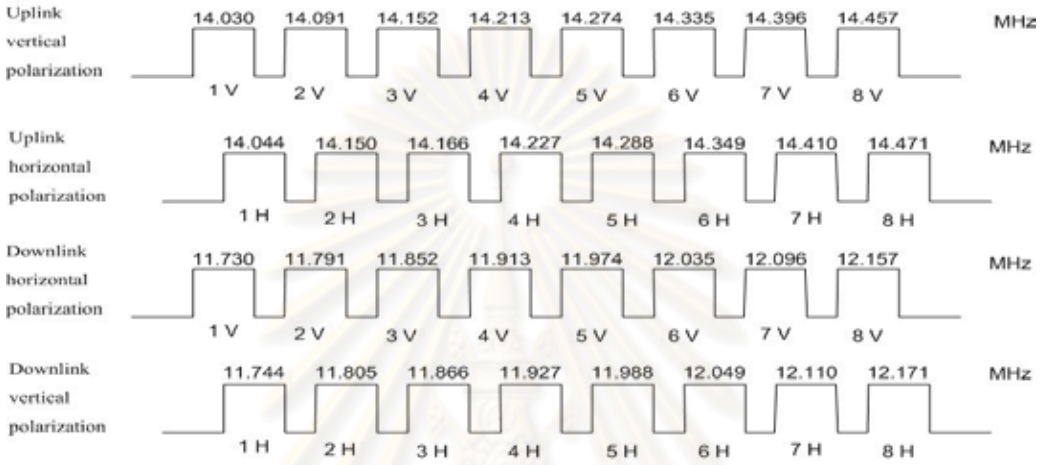


รูปที่ 8 องค์ประกอบพื้นฐานของระบบสื่อสารดาวเทียม

ปัจจุบันดาวเทียมสื่อสารทางธุรกิจทั่วๆ ไป จะมีขอบเขตในการใช้งานขนาดแบนด์วิดท์ (Bandwidth) ของสัญญาณขาขึ้นและขาลงประมาณ 500 MHz และแถบความถี่ที่นิยมใช้กันมากที่สุดคือย่าน C-band 6/4 GHz โดยสัญญาณขาขึ้นจะมีความถี่คลื่นพาห์ (Carrier) ในช่วง 5.725 ถึง 7.075 GHz และสัญญาณขาลงจะมีความถี่คลื่นพาห์ ในช่วง 3.4 ถึง 4.8 GHz การใช้งานในย่านความถี่ 6/4 นี้ ในปัจจุบันนี้มีการใช้งานกันมากขึ้นเรื่อยๆ ทั่วโลก ถ้าปล่อยให้มีการใช้งานในย่านนี้มากขึ้นต่อไปอีก ก็จะทำให้เกิดปัญหาทางด้านสัญญาณที่จะเกิดการรบกวนขึ้นได้ ดังนั้นจึงได้มีการเริ่มใช้งานในย่านความถี่ Ku-band 14/12 GHz ขึ้น โดยที่ความถี่ขาขึ้นจะอยู่ในช่วง 12.75 ถึง 14.8 GHz และความถี่ขาลงอยู่ในช่วง 10.7 ถึง 12.3 GHz ความถี่ในย่านนี้มีแนวโน้มว่าจะมีการใช้งานมากขึ้นในอนาคต แต่ปัญหาหนึ่งที่เป็นข้อเสียเปรียบในการใช้งานย่านความถี่นี้คือปัญหาการลดทอนสัญญาณเนื่องจากฝน (Rain induced attenuation) ซึ่งมีผลกระทบมากกว่าย่านความถี่ C-band นอกจากนี้ความถี่ในย่านความถี่ 30/20 GHz ได้มีการนำมาใช้งานบ้างแล้ว ซึ่งประกอบด้วยความถี่ขาขึ้นในช่วง 27.5 ถึง 31 GHz และความถี่ขาลงในช่วง 18.1 ถึง 21.2 GHz

สำหรับในย่านความความถี่ C-band และ Ku-band ได้แบ่งเป็นส่วนย่อยๆ ที่เรียกว่าทรานส์พอนเดอร์แบนด์วิดท์ (Transponder Bandwidth) ตัวอย่างเช่นดาวเทียมการสื่อสารพาณิชย์ดวงหนึ่งมี 8 ทรานส์พอนเดอร์ (Transponder) โดยในแต่ละทรานส์พอนเดอร์มีขนาดแบนด์วิดท์เท่ากับ 54 MHz และจุดศูนย์กลางความถี่ของแต่ละทรานส์พอนเดอร์จะห่างกัน 61 MHz (เพื่อเหลือให้มีขนาดของ Guardband เพียง พอต่อการป้องกันการรบกวนระหว่างทรานส์พอนเดอร์) ดังนั้นเมื่อรวมความถี่ทั้งหมด 8 ทรานส์พอนเดอร์จะได้ความถี่ประมาณเท่ากับแบนด์วิดท์ในการใช้งานกับตัวดาวเทียมนั้นคือ 500 MHz นอกจากนี้แล้วก็จะมีเทคนิคที่เรียกว่า การใช้ความถี่ซ้ำ (Frequency reused) ซึ่งทำเสมือนกับการเพิ่มจำนวนทรานส์พอนเดอร์ของดาวเทียมโดยที่แบนด์วิดท์ของตัวดาวเทียมยังคงเท่าเดิม เทคนิคนี้จะใช้

การโพลาไรซ์ (Polarization) ในแบบ Orthogonal นั่นคือในทรานส์พอนเดอร์หนึ่งจะใช้การโพลาไรซ์ในแนวดิ่ง (Vertical Polarization) อีกทรานส์พอนเดอร์หนึ่งที่อยู่ติดกันจะใช้โพลาไรซ์ในแนวนอน (Horizontal Polarization) โดยการรบกวนระหว่างกันจะลดลงไปโดยธรรมชาติของการโพลาไรซ์ที่มีแนวสนามไฟฟ้าคนละแนว การแยกแยะสัญญาณรบกวนระหว่างกันสามารถทำได้ถึง 30 dB ดังรูปที่ 9 และอีกหลายวิธีการที่น่าสนใจ



รูปที่ 9 ตัวอย่างการใช้เทคนิคความถี่ซ้ำ (Frequency Reused)

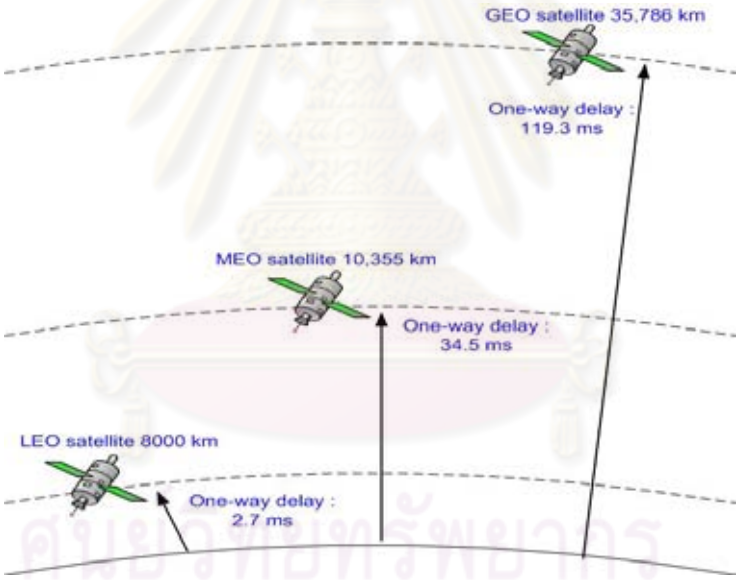
2.5.1 ประเภทของดาวเทียม

แบ่งตามลักษณะการโคจรจากพื้นโลกเป็นดาวเทียมวงโคจรต่ำ (Low-Earth Orbit, LEO), วงโคจรกลาง (Medium-Earth Orbit, MEO) และ ดาวเทียมค้างฟ้า (Geostationary-Earth Orbit, GEO) ซึ่งแต่ละประเภทจะโคจรห่างจากพื้นโลก ดังรูปที่ 10 มีรายละเอียดดังนี้

1. Low-Earth Orbit (LEO) โคจรห่างจากโลกประมาณ 700 - 2,000 กิโลเมตรและมีเวลาไปกลับ (*RTT*) ประมาณ 0.05 วินาที เมื่อเปรียบเทียบกับเวลาประวิงของการสื่อสารภาคพื้นดินที่ไม่ต่างกันมากนัก และคาบการโคจรรอบโลกประมาณ 100 – 127 นาที โดย LEO จะอยู่ใกล้พื้นผิวโลกขนาดจานสายอากาศและระดับกำลังส่งไม่สูงเมื่อเทียบกับ GEO แต่พื้นที่ครอบคลุมของดาวเทียมเล็กลงด้วย ในวงโคจรจำต้องประกอบด้วยดาวเทียมหลายดวงเพื่อให้ครอบคลุมพื้นที่ทั่วโลก เมื่อดาวเทียมเคลื่อนที่อย่างรวดเร็วผ่านพื้นผิวโลกไป ผู้ใช้อาจจำเป็นต้องทำการ handed off จากดาวเทียมดวงหนึ่งไปยังอีกดวงหนึ่ง หรืออาศัยจานสายอากาศแบบปรับทิศทางเพื่อติดตามรับสัญญาณแทน

2. Medium-Earth Orbit (MEO) โคจรห่างจากโลกประมาณ 10,000 กิโลเมตรและมีเวลาไปกลับ (*RTT*) ประมาณ 0.25 วินาที ลักษณะเช่นเดียวกับ LEO

3. Geostationary-Earth Orbit (GEO) โคจรห่างจากโลกประมาณ 36,000 กิโลเมตร เหนือเส้นศูนย์สูตร และมีเวลาไปกลับ (*RTT*) ประมาณ 0.55 วินาที คาบการโคจรรอบโลกประมาณ 24 ชั่วโมง ซึ่งจะมีความเร็วสัมพัทธ์เทียบกับโลกเป็นศูนย์คือมีความเร็วในการโคจรรอบโลกหนึ่งรอบเท่ากับที่โลกหมุนรอบตัวเองหนึ่งรอบ ที่ระดับความสูงของวงโคจรดาวเทียม GEO ทำให้สัญญาณสามารถครอบครุมพื้นที่ 1 ใน 3 ของพื้นผิวโลกได้ โดยพื้นที่ครอบคลุมที่ได้จะเรียกว่า footprint สามารถใช้ดาวเทียม 3 ดวงเพื่อครอบคลุมการสื่อสารทั่วทั้งโลก อย่างไรก็ตามค่าใช้จ่ายในการส่งดาวเทียมขึ้นสู่วงโคจรก็สูงเช่นเดียวกัน ระดับสัญญาณที่ถูกลดทอนไปด้วยระยะทาง ทำให้ต้องใช้ระบบจานสายอากาศและกำลังส่งของดาวเทียมที่มีประสิทธิภาพเพื่อรับส่งข้อมูลระหว่างเทอร์มินอลและตัวดาวเทียม



รูปที่ 10 เปรียบเทียบระดับความสูงของดาวเทียม 3 วงโคจร GEO, MEO และ LEO จากผิวโลก

2.5.2 โครงข่ายดาวเทียมอินเทอร์เน็ต

ที่ผ่านมาระบบดาวเทียมเป็นเสมือนท่อส่งข้อมูลของโครงข่ายขนาดใหญ่ที่ผลักดันระบบสื่อสารในหลายด้านโดยเฉพาะโทรศัพท์ผ่านทางไกล ระบบสื่อสารทางทหารและการถ่ายทอดสดโทรทัศน์ ในปัจจุบันระบบดาวเทียมนี้ได้กลายเป็นส่วนหนึ่งของโครงข่ายอินเทอร์เน็ตโปรโตคอล[22] ซึ่งกลายเป็นเทคโนโลยีทิศทางใหม่ในระบบสื่อสารที่ถูกนำมาใช้ทำการส่งผ่านทราฟฟิค อินเทอร์เน็ตที่

ครอบคลุม ด้วยการทำงานของชุดโปรโตคอล TCP/IP ซึ่งปัจจุบันมีอิทธิพลมากในการทำงานร่วมกับระบบโครงข่ายอื่นๆ โดยคุณสมบัติของโครงข่ายดาวเทียมเองที่สามารถครอบคลุมภูมิประเทศขนาดใหญ่ด้วยฟรุ๊ตปริ้นของจานสายอากาศ ความสามารถในการถ่ายทอดสัญญาณ พร้อมด้วยวิธีการมัลติเพล็กซ์ช่องสัญญาณความจุสูงด้วยสเปกตรัมของความถี่แบบ multiband

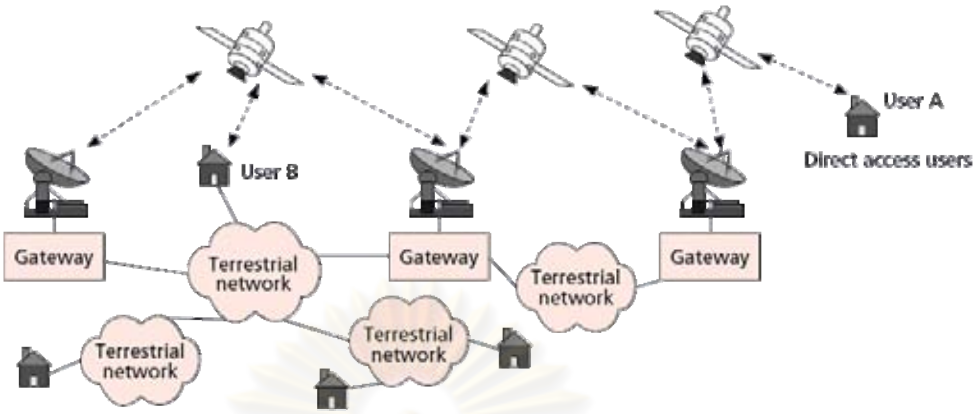
ด้วยความสามารถในการรองรับแบนด์วิดท์ขนาดใหญ่ จึงถูกคาดหวังว่าจะโครงข่ายนี้จะกลายเป็นระบบสื่อสารดาวเทียมในรูปแบบใหม่ที่ต่างจากเดิม ที่ประกอบด้วยโดยการรวมระบบการทำงานอัจฉริยะของอุปกรณ์บนบอร์ดวงจรของตัวดาวเทียม และใช้ ย่านความถี่ Ka และ V ร่วมกับเทคโนโลยีสปอตบีม และมีการทำงานของระบบประมวลผลบนบอร์ดทำหน้าที่เสมือนสวิตซ์ชุดข้อมูล packet โดยตรง และส่งลงไปให้กับผู้ใช้งานด้วยช่องสัญญาณขาลง รวมทั้งสามารถรวมข้อมูลของช่องสัญญาณขาขึ้นต่างๆ จากหลายสถานีเข้าด้วยกัน แล้วทำการป้อนข้อมูลรูปแบบมัลติคลาสไปยังสถานีผู้รับต่างๆ ซึ่งเป็นลักษณะการทำงานแบบมัลติคลาสที่มีประสิทธิภาพเหมือนในโครงข่ายภาคพื้นดิน รวมทั้งสามารถเป็นระบบโครงข่ายที่สามารถรองรับการประยุกต์ใช้งานได้หลากหลาย เช่น การให้บริการบรอดแบรนด์อินเทอร์เน็ต อินทราเน็ต VPN VOIP การประชุมในรูปแบบวิดีโอ ช่องสัญญาณ Leased line เป็นต้น

โครงสร้างของระบบดาวเทียมประกอบด้วยอุปกรณ์ภาคพื้นดินและบนอวกาศในส่วนภาคพื้นดินประกอบด้วย สถานีเกตย์เวย์ gateway station(GSs), ศูนย์กลางควบคุมโครงข่าย network control center(NCC) และ ศูนย์กลางควบคุมการทำงาน operation control centers (OCCs) โดย NCC และ OCCs ทำหน้าที่จัดการทรัพยากรของโครงข่ายทั้งหมด, การทำงานของดาวเทียมและควบคุมวงโคจร GSs จะทำหน้าที่เป็นส่วนเชื่อมต่อระหว่างโครงข่ายภายนอกอื่นๆกับโครงข่ายดาวเทียม จัดการโปรโตคอล ระบุที่อยู่ และการแปลงรูปแบบ ในส่วนของอวกาศประกอบด้วยตัวดาวเทียมซึ่งจะแบ่งไปตามรูปแบบของวงโคจร

การออกแบบระบบที่ขึ้นกับค่าใช้จ่ายที่ค่อนข้างสูงและเป็นไปได้ยากในการปรับปรุงและซ่อมแซม รวมทั้งสภาวะของอวกาศทำให้การออกแบบต้องเป็นรูปแบบการทำงานที่ไม่ซับซ้อนและคงทนโดยเฉพาะในระบบดาวเทียมแบบค้างฟ้า ที่ให้บริการเสมือนเป็นท่อส่งข้อมูลขนาดใหญ่ ซึ่งทำหน้าที่เป็นตัวถ่ายทอดสัญญาณระหว่างการสื่อสารสองตำแหน่งบนพื้นโลก โดยอาจจะมีระบบการประมวลผลบนบอร์ด On-Board Processing(OBP) ซึ่งรวมทั้งการดีมอดูเลทและการมอดูเลทซ้ำ การถอดรหัสและการเข้ารหัส การสวิตซ์ของทรานสปอร์เดอร์และบีม และการจัดหาเส้นทางเพื่อใช้ช่องสัญญาณอย่างมีประสิทธิภาพ หรือระบบ OBP ที่สามารถรองรับการเชื่อมต่อของช่องสัญญาณความจุสูงระหว่างดาวเทียมด้วยกัน Inter Satellite Links(ISLs) ซึ่งเชื่อมดาวเทียมเข้าด้วยกันแบบ line of sight

ช่วงความถี่ของดาวเทียม โดยทั่วไปช่วงความถี่ที่ใช้งานกันมากของดาวเทียมคือ C band (4-8GHz), KU band(10-18 GHz) และ Ka band(18-31GHz) ด้วยย่านความถี่ที่สูงขึ้นการตอบสนองความยาวคลื่นที่สั้นกว่า ทำให้จานสายอากาศขนาดเล็กสามารถใช้เพื่อรับสัญญาณได้เป็นอย่างดี ที่ Ka band

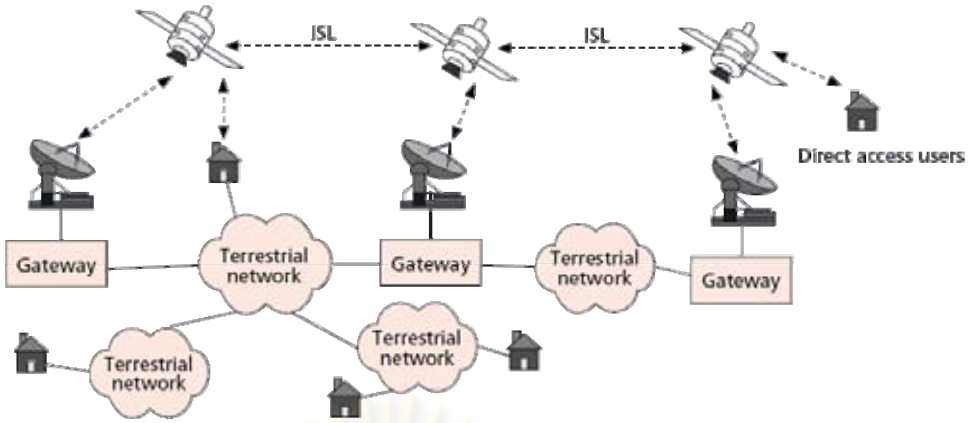
สามารถรองรับแบนด์วิดท์ได้มากกว่าย่าน KU และมีจานสายอากาศที่ขนาดเล็กว่า แต่จะได้รับผลกระทบจากสภาพแวดล้อม เช่น การจางหายของสัญญาณ และการลดทอนเนื่องจากฝน



รูปที่ 11 โครงข่ายดาวเทียมอินเทอร์เน็ตโครงสร้างแบบ bent-pipe

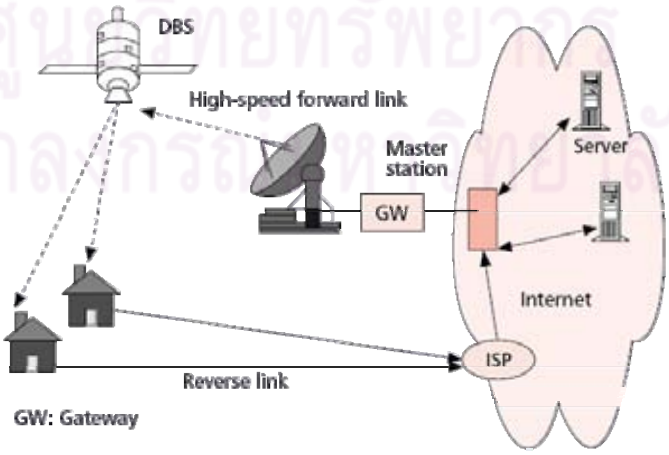
โครงสร้างการเชื่อมต่อของดาวเทียมอินเทอร์เน็ตมีโครงสร้างหลากหลายต่างกันไปขึ้นกับระบบของตัวดาวเทียม ชนิดของวงโคจร ลักษณะการใช้งาน โดยได้มีข้อเสนอที่จะทำการรวมประเภทของวงโคจรดาวเทียมต่างๆเข้าด้วยกัน ให้เป็นรูปแบบโครงข่ายแบบผสมเพื่อให้เกิดความคุ้มค่าในการใช้คุณลักษณะที่ดีสุดของวงโคจรแต่ละประเภท โครงข่ายของดาวเทียมในปัจจุบันสามารถรองรับการเชื่อมต่อแบคโบนอินเทอร์เน็ต และอินเทอร์เน็ตความเร็วสูง ลักษณะโครงข่ายดาวเทียมอินเทอร์เน็ตทั่วไปมีโครงสร้างแบบท่อส่งข้อมูลแบบ bent pipe แสดงได้ดังรูปที่ 11 โดยตัวดาวเทียมกำหนดให้อยู่ในวงโคจรได้ทั้ง GEO, MEO หรือ LEO ซึ่งทำหน้าที่ให้บริการอินเทอร์เน็ตเช่น เดียวกับการให้บริการข้อมูลแบบ data trunking โดยทำหน้าที่เป็นตัวเชื่อมต่อกับโครงสร้างของระบบอินเทอร์เน็ตภาคพื้นดินโดยผ่าน GSs ซึ่งอาจจะเป็นวิธีการเข้าใช้งานสำหรับผู้ใช้งาน(เช่น User A) เมื่อการสื่อสารแบบอื่นๆไม่สามารถดำเนินการได้ หรือใช้เป็นการเชื่อมต่อสำรองเพื่อแทนที่โครงข่ายภาคพื้นดิน(เช่น User B)

อย่างไรก็ตามโครงสร้างแบบ bent pipe ยังขาดทิศทางการสื่อสารในอวกาศที่มีผลให้การใช้ประโยชน์ของความถี่ไม่ดีพอและเกิดการประวิงเวลาของระบบสูง ดังนั้นระบบ OBP และ ISLs อาจนำมาใช้เพื่อช่วยในการเชื่อมโยงโครงข่ายบนฟ้า ดังรูปที่ 12 โครงสร้างนี้จะทำหน้าที่ในการเข้าถึงร่วมกันและสร้างช่องสัญญาณแบบ backbone ด้วยการเชื่อมต่อที่เพิ่มขึ้นในอวกาศทำให้มีความยืดหยุ่นเพิ่มขึ้น และมีความซับซ้อนของการจัดหาเส้นทางมากขึ้นด้วย



รูปที่ 12 โครงข่ายดาวเทียมอินเทอร์เน็ตโครงสร้างแบบ OBP และ ISL

ในโครงสร้างทั้งสองที่กล่าวมานั้น เทอร์มินอลของผู้ใช้งานจะสมมติให้เป็นอุปกรณ์ที่สามารถถ่ายโอนข้อมูลได้แบบสองทิศทาง ซึ่งหมายความว่าผู้ใช้สามารถทำการส่งข้อมูลขึ้นไปและรับจากดาวเทียมโดยตรง อย่างไรก็ตามแม้ว่าความก้าวหน้าในเทคโนโลยีดาวเทียมที่เป็นไปอย่างรวดเร็ว ด้วยการใช้เทอร์มินอลผู้ใช้งานขนาดเล็กเช่น ultra small aperture terminals(USAT) ด้วยจานสายอากาศเพียง 60 เซนติเมตร เทอร์มินอลข้อมูลแบบสองทิศทางก็ยังมีราคาสูงอยู่ เมื่อพิจารณาเข้าไปถึงการใช้งานข้อมูลทราฟฟิคอินเทอร์เน็ตพบว่าเป็นการใช้ทราฟฟิคแบบไม่สมมาตร(Asymmetry) ข้อมูลส่วนใหญ่จะถูกส่งจากเซอร์เวอร์ไปยังผู้ใช้งาน มากกว่าที่ผู้ใช้งานไปให้เซอร์เวอร์ เช่น การค้นหาข้อมูลบนเว็บ จึงมีอีกแนวทางดังรูปที่13 ในการเข้าถึงอินเทอร์เน็ตโดยผ่านดาวเทียมถ่ายทอดสัญญาณทิศทางเดียว direct broadcast satellite (DBSs) โดยมีลักษณะการทำงานเช่นการถ่ายทอดโทรทัศน์ ซึ่งแต่ละผู้ใช้งานจะมีจานดาวเทียมเพื่อทำหน้าที่รับข้อมูลจากช่องสัญญาณความเร็วสูงที่ถ่ายทอดมาจากดาวเทียมเพียงอย่างเดียว และจะทำการส่งข้อมูลกลับไปยังเซอร์เวอร์ด้วยโครงข่ายภาคพื้นดิน



รูปที่ 13 โครงข่ายดาวเทียมอินเทอร์เน็ตแบบ DBS

เทคนิคของการออกแบบโครงข่ายดาวเทียมอินเทอร์เน็ต

ในส่วนนี้จะกล่าวถึงเทคนิคในการออกแบบสำคัญที่จำเป็นต้องมีโดยเฉพาะเพื่อใช้งานโครงข่ายดาวเทียมอินเทอร์เน็ตที่แตกต่างจากโครงข่ายภาคพื้นดินทั่วไป เช่นการควบคุมการเข้าถึงแบบหลายผู้ใช้งาน (multiple access control), การจัดหาเส้นทางในระบบ และการนำพาข้อมูลโดยการปรับปรุงทรานสปอร์ตโปรโตคอล

การควบคุมการเข้าถึงแบบหลายผู้ใช้งาน (Multiple Access Control:MAC)

ในระบบดาวเทียมที่ถ่ายโอนข้อมูลแบบสองทิศทาง จำนวนเทอร์มินอลของผู้ใช้งานต่างๆจะกระจายอยู่โดยทั่วไปในพื้นที่ครอบคลุมของฟรุตปริ้น สำหรับช่องสัญญาณดาวเทียมแบบuplink MAC จะถูกกำหนดให้เป็นกฎเพื่อควบคุมการเข้าใช้งานในแชร์ช่องสัญญาณระหว่างผู้ใช้งาน ความสำคัญของกฎที่กำหนดจะมีผลต่อประสิทธิภาพและความเท่าเทียมกันในการจำกัดทรัพยากรระบบ MAC โปรโตคอลมีความสำคัญอย่างมากและจะมีผลต่อโปรโตคอลระดับชั้นที่สูงกว่ารวมทั้งคุณภาพการให้บริการของระบบด้วย MAC โปรโตคอลที่ดีต้องสามารถให้สิทธิทราฟฟิคแบบเวลาจริง(real-time traffic) ให้มีค่ามากกว่าทราฟฟิคแบบไม่เวลาจริงได้ ควรง่ายในการจัดการ มีความแน่นอนและยืดหยุ่น โดยการพิจารณาจะสามารถแบ่งประเภทเป็น 3 กลุ่ม คือแบบกำหนดตายตัว (fixed assignment), แบบเข้าถึงแบบสุ่ม(random access) และแบบกำหนดตามความต้องการ (demand assignment)

- แบบกำหนดตายตัว (fixed assignment)

กำหนดบนความถี่ช่วงเวลาหรือการเข้ารหัส เทคนิคหลักเช่น frequency-division multiple access (FDMA), time-division multiple access (TDMA), code-division multiple access(CDMA)

- แบบเข้าถึงแบบสุ่ม (random access)

เนื่องจากเทคโนโลยีที่ก้าวหน้าของเทอร์มินอลขนาดเล็ก เช่น VSATs และ USATs ที่สามารถเพิ่มจำนวนเป็นหลักพันต่อฟรุ๊ตปริ้นของดาวเทียม ทราฟฟิคที่ถูกสร้างโดยแต่ละผู้ใช้งานจะถูกกำหนดให้เป็นช่วงสั้นๆ(bursty) การกำหนดแบบตายตัวจึงถูกแทนที่ด้วยการเข้าถึงแบบสุ่มขึ้นกับการแข่งขัน เช่น Aloha ซึ่งจะทำการส่งข้อมูลโดยไม่คำนึงถึงการส่งของสถานีอื่น จะทำการส่งซ้ำหลังจากมีการชนกันเพิ่มขึ้นซึ่งจะเป็นสาเหตุให้มีแพ็คเกจล่าช้าเพิ่มขึ้นและอาจทำให้อัตราส่งผ่านลดลง

- แบบกำหนดตามความต้องการ (demand assignment)

ถึงแม้ว่าการเข้าถึงแบบสุ่มจะเหมาะที่จะรองรับบริการให้กับเทอร์มินอลจำนวนมากด้วยทราฟฟิคช่วงสั้นๆซึ่งไม่มีการรับประกัน QoS การเข้าถึงแบบหลายผู้ใช้งานตามความต้องการ Demand assignment multiple access (DAMA) จึงพยายามแก้ปัญหาโดยจัดสรรแบนด์วิดท์ให้ระบบแบบพลวัตเพื่อตอบสนองกับความต้องการของผู้ใช้งานขึ้น โดยทรัพยากรที่ได้รับการร้องขอต้องได้รับอนุญาติก่อนที่จะมีการส่งจริง เช่นการทำงานแบบ FODA (First-in first-out Ordered Demand Assignment)

การจัดหาเส้นทางในโครงข่าย

การจัดหาเส้นทางมีความสำคัญในระบบดาวเทียมอินเทอร์เน็ตที่วงโคจร LEO ที่มีระบบ OBP และ ISLs ซึ่งจะมีเวลาประวิงต่ำและมีการเชื่อมต่อแบบสมบูรณ์ ทำให้การจัดหาเส้นทางจำเป็นอย่างยิ่งเมื่อเส้นทางสื่อสารมีความซับซ้อนและขึ้นกับการเคลื่อนที่ของตัวดาวเทียม

- โครงสร้างแบบพลวัต (Dynamic Topology) เกิดจากความสัมพันธ์ของการเคลื่อนที่ระหว่างดาวเทียม LEO และโลก ซึ่งดาวเทียมดวงหนึ่งมีช่วงเวลาช่วงสั้นๆในการมองเห็นเทอร์มินอลผู้ใช้งานที่แทบไม่มีการเคลื่อนที่บนพื้นโลก ในช่วงเวลา 24 ชม.การกำหนดวงโคจรให้สามารถครอบคลุมผู้ใช้งานโดยอย่างน้อยที่สุดที่เวลาใดๆควรจะมีดาวเทียมอย่างน้อย 1 ดวงที่สามารถเชื่อมต่อแบบ line of sight กับผู้ใช้งานได้ และเมื่อพื้นที่ครอบคลุมดาวเทียมเดิมเคลื่อนย้ายไปไปจากผู้ใช้งาน ดวงอื่นจะต้องเคลื่อนเข้ามาแทนที่ ทำให้เกิดการแฮนด์โอเวอร์ขึ้น สำหรับดาวเทียมที่ประกอบด้วยจานสายอากาศแบบมัลติและทรานสปอร์นเดอร์ ฟรุ๊ตปริ้นจะถูกแบ่งไปเป็นจำนวนของสปอตบีม ซึ่งแต่ละจุดจะถูกครอบคลุมด้วยบีมสายอากาศ ดังนั้นช่วงของการแฮนด์โอเวอร์ระหว่างสปอตบีมจึงเกิดขึ้นภายในช่วงเวลาที่มองเห็นดาวเทียมเพียงหนึ่งดวง ISLs ในวงโคจรที่มีโครงสร้างแบบเมส ดาวเทียมแต่ละดวงสามารถเชื่อมโยงได้ถึง 4-8 ISLs ซึ่งแบ่งชนิดของ ISLs เป็นสองชนิดคือแบบ intraplane ที่เชื่อมดาวเทียมในวงโคจรเดียวกัน และ interplane ที่เชื่อมดาวเทียมเพื่อนบ้านที่วงโคจรข้างเคียง บาง interplane อาจปิดช่องเชื่อมต่อชั่วคราวเมื่อพบว่ามุมหรือระยะระหว่าง ดาวเทียมนั้นเปลี่ยนรวดเร็วเกินกว่าที่ระบบจานสายอากาศจะตามได้ ซึ่งอาจจะเกิดขึ้นระหว่างการหมุนตามเข็มรอบวงโคจรของทั้งคู่ หรือเมื่อดาวเทียมทั้งสองโคจรตัดกัน การหาเส้นทางควรทำงานในโทโปโลยีที่แตกต่างกันได้ ถึงแม้ว่าโทโปโลยีในการโคจรจะเปลี่ยนแปลงบ่อยครั้ง แต่ก็สามารถทำการคาดเดาได้โดยง่ายเพราะการจำกัดวงโคจรการเคลื่อนที่ของดาวเทียม กลไกลการหาเส้นทางที่นิยมในอินเทอร์เน็ต เช่น distance vector (DV) และ link state algorithm (LSA) ซึ่งไม่สามารถนำมาใช้ได้โดยตรงในการหาเส้นทางวงโคจร เนื่องจากการเปลี่ยนแปลงโทโปโลยีบ่อยครั้งเป็นสาเหตุให้เกิดโอเวอร์เฮดขนาดใหญ่และการออสซิเลตจึงมีการคิดค้นกลไกลแบบ discrete-time dynamic virtual topology routing(DT-DVTR) และ Virtual node(VN) ขึ้น ส่วนของการหาเส้นทางที่จำเป็นสำหรับดาวเทียมวงโคจรLEO ยังมีรายละเอียดที่ไม่ได้กล่าวถึงเนื่องจากเป็นส่วนที่นอกเหนือขอบเขตของวิทยานิพนธ์

ทรานสปอร์ตโปรโตคอลของโครงข่าย

TCP/IP และ UDP/IP เป็นชุดโปรโตคอลมาตรฐานของอินเทอร์เน็ต ซึ่งเกิดขึ้นเมื่อเริ่มคิดค้นระบบ และมีแนวโน้มว่าจะใช้งานต่อไปในอนาคตเนื่องจากแอพพลิเคชั่นต่างๆในปัจจุบันได้ถูกสร้างขึ้นให้ทำงานร่วมกับชุดโปรโตคอลเหล่านี้ ดังนั้นดาวเทียมอินเทอร์เน็ตจึงถูกออกแบบเพื่อให้สามารถรองรับการทำงานของ TCP และ UDP ด้วย อย่างไรก็ตามประสิทธิภาพในการทำงานของทั้งสองโปรโตคอลจะได้รับผลกระทบจากการประวิงเวลาและอัตราการผิดพลาดที่เป็นคุณลักษณะเฉพาะของ

ช่องสัญญาณดาวเทียม ผลกระทบจะเกิดกับTCP เป็นอันมาก การทำงานของ TCP ที่ใช้กลไกลการตอบกลับแบบบวกซึ่งเกี่ยวข้องกับการควบคุมอัตราและความเชื่อมั่นในการส่งข้อมูล การประวิงเวลาที่สูงของช่องสัญญาณดาวเทียมโดยเฉพาะวงโคจร GEO ทำให้เวลาการทำงานของ TCP จากปลายทางไปปลายทางและเวลาในการเดินทางของสัญญาณตอบกลับเพิ่มขึ้น การตอบกลับที่ล่าช้าเป็นจุดอ่อนของการควบคุมอัตราการส่งผ่านและความคับคั่ง ซึ่งส่งผลไปยังประสิทธิภาพการใช้งานของช่องสัญญาณด้วย นอกจากนี้การแปรปรวนที่มีค่ามากของรอบเวลาการทำงาน(RTT) เป็นเหตุให้เกิดการสิ้นสุดเวลาทำงาน(timeouts) และการส่งชุดข้อมูลซ้ำ(retransmission)อีกด้วย

ในขั้นตอนเริ่มต้นของกลไกล slow start ของการส่ง TCP ถึงแม้ว่าอัตราส่งจะเพิ่มขึ้นเป็นแบบ exponentially การทำงานก็ยังช้าอยู่สำหรับช่องสัญญาณดาวเทียมที่มีแบนด์วิดท์สูง ข้อเสนอหนึ่งคือให้ทำการเพิ่มค่าตั้งต้นของวินโดว์ในการส่งให้มากขึ้น โดย TCP ทั่วไปจะอนุญาติให้ค่าขนาดของวินโดว์เท่ากับ 64 กิโลไบต์ ซึ่งจำกัดอัตราการส่งผ่านสูงสุดแค่เพียง 64 กิโลไบต์/RTT ช่องสัญญาณดาวเทียมสามารถที่จะใช้งานได้อย่างคุ้มค่าโดยการปรับขนาดวินโดว์มีค่ามากเท่ากับผลคูณการประวิงแบนด์วิดท์ในช่องสัญญาณเพื่อเพิ่มอัตราการส่งผ่านที่สูงขึ้น ซึ่งการปรับขนาดของวินโดว์เพื่อปรับปรุง TCP ได้ถูกประกาศไว้ใน IETF Request for Comments(RFC) 1323

สัญญาณดาวเทียมมีข้อเสียหลายอย่างเช่น การกวน(interference), การจางหาย(fading) , การบดบัง(showdowing) และ การลดทอนเนื่องจากฝน (rain attenuation) จึงคาดการว่าอัตราการผิดพลาดบิต(bit error rate: BER) จะอยู่ในเกณฑ์สูงด้วย ถึงแม้ว่าได้มีการพัฒนาเทคนิคการมอดูเลท, การเข้ารหัสและตรวจสอบการผิดพลาด (forward error correction :FEC) เพื่อนำมาลด BER ในสภาพแวดล้อมที่ทำให้ BER มีค่าสูงมาก แต่ TCP ก็ไม่สามารถจำแนกระหว่างการเสียหายของข้อมูลที่เป็นเหตุจากกระบวนการส่งที่ผิดพลาด และการสูญหายของแพ็คเกจเนื่องจากความคับคั่งได้ ซึ่งทั้งคู่ไม่ส่งสัญญาณการตอบกลับและแสดงการเตือนให้ทราบว่าเกิดการคับคั่งขึ้น เมื่อมีการสูญหายของแพ็คเกจ TCP (มาตรฐาน) จะทำการลดขนาดของวินโดว์ลงครึ่งหนึ่งแม้ว่าไม่ได้เกิดการคับคั่งขึ้น อย่างไรก็ตามอัตราการผิดพลาดในการส่งข้อมูลบนช่องสัญญาณดาวเทียมจะเกิดขึ้นเป็นช่วงๆ ซึ่งถือเป็นคุณลักษณะเฉพาะเมื่ออยู่ภายใต้สภาพอากาศที่เปลี่ยนแปลง การผิดพลาดในช่วงสั้นๆ ใน 1 รอบRTT จะทำให้อัตราส่งผ่านลดลงอย่างรวดเร็ว ปัญหาอื่นที่เกิดขึ้นกับTCP อีก เช่น การทำงานร่วมกันระหว่างการเชื่อมต่อ TCP ต่างชนิดกันด้วย RTTs ที่ต่างกัน เมื่อ TCP ได้เข้ามาทำงานรวมกันในช่องสัญญาณแบบคอขวด (bottlenecked) โดย TCP ที่มี RTTs นานกว่าจะได้รับผลกระทบจากการจัดสรรแบนด์วิดท์อย่างไม่เป็นธรรม

องค์กร IETF ซึ่งดูแลส่วนงานโปรโตคอล TCP บนดาวเทียมจึงได้มีการปรับปรุง TCP จากรูปแบบที่ใช้ในโครงข่ายแบบมีสายเดิม เพื่อให้มีการยืดหยุ่นกับการใช้งานในสภาพแวดล้อมที่หลากหลายและมีประสิทธิภาพมากขึ้นโดยประกาศเป็น RFC ที่สำคัญมีดังนี้

- TCP selective acknowledgment(SACK): (RFC2018)เป็นส่วนเพิ่มเติมที่อนุญาติให้ผู้รับระบุเซกเมนต์ในการรับแบบสะสม ผู้ส่งจำเป็นต้องส่งซ้ำเฉพาะแพ็คเกจที่สูญหายเท่านั้น โดย TCP SACK สามารถแก้ปัญหาการสูญหายแบบหลายส่วนในวินโดว์ที่ส่งภายในรอบ RTT เดียว
- TCP สำหรับการจัดการ (transaction): (RFC1644) พยามลดการประวิงเวลาในการ hand-shakingจาก 2 RTTs ให้เหลือเพียง 1 RTT ซึ่งมีความสำคัญในการปรับปรุงการส่งผ่านข้อมูลระยะสั้น
- Persistent TCP: ซึ่งรองรับ HTTP1.1(RFC2068) ที่อนุญาติให้ข้อมูลส่วนเล็กๆหลายส่วนส่งผ่านเพื่อทำการดาวน์โหลดในการเชื่อมต่อ TCP แบบ Persistent ที่มีประสิทธิภาพสูง
- Maximum transfer unit(MTU): ทำหน้าที่สร้างกลไกลซึ่งอนุญาติให้ TCP สามารถใช้แพ็คเกจขนาดใหญ่ที่สุดเท่าที่ทำได้ โดยหลีกเลี่ยงการแบ่งเซกเมนต์ของ IP ซึ่งจะช่วยลดโอเวอร์เฮดและขบวนการแยกและรวมชุดข้อมูล

ส่วนขยาย TCP เหล่านี้สามารถแก้ปัญหาในการทำงานของมาตรฐาน TCP บนช่องสัญญาณดาวเทียมแต่ปัญหาอื่นๆ เช่นการประวิงเวลาระหว่างปลายทาง และความไม่สมมาตรของช่องสัญญาณยังไม่สามารถหลีกเลี่ยงได้ วิธีทางเดียวที่จะแก้ไขผลกระทบการประวิงเวลาระหว่างปลายทางคือ ทำการแบ่งการเชื่อมต่อของ TCP เป็น 2 ส่วน หรือมากกว่านั้นเมื่อเชื่อมกับ GS ในโครงข่ายดาวเทียมและภาคพื้นดิน โดยวิธีการหลักสรุปได้ดังนี้

- TCP spoofing: เป็นการแบ่งการเชื่อมต่อของ TCP ออกเป็นส่วนโดย GS จะทำการส่งสัญญาณการตอบรับปลอมกลับมาก่อนที่แพ็คเกจข้อมูลไปถึงยังปลายทางผู้รับโดยขึ้นกับการรับแพ็คเกจ ตำแหน่ง GS ที่จุดแบ่งจะรับผิดชอบการส่งซ้ำในชุดข้อมูลที่สูญหาย
- TCP splitting: มาแทนที่ spoofing โดยการเชื่อมต่อจะถูกแยกอย่างเต็มรูปแบบ ทรานสปอร์ตโปรโตคอลสามารถนำมาใช้ในโครงข่ายดาวเทียมโดยไม่มีการรบกวนกับ TCP มาตรฐานในโครงข่ายภาคพื้นดิน ซึ่งระบบค่อนข้างยืดหยุ่น แต่ตัวแปลงของโปรโตคอลบางชนิดที่จุดแบ่งต้องทำการพัฒนาให้เหมาะสม
- Web caching: จะมีความแตกต่างจากสองกระบวนการข้างต้น ซึ่งการเชื่อมต่อของ TCP จะถูกแบ่งแยกโดย web cache ในโครงข่าย โดยที่ผู้ใช้งานสามารถเชื่อมต่อเข้ามายัง web cache โดยไม่จำเป็นต้องสร้างการเชื่อมต่อในแบบของ TCP ถ้ามีการร้องขอข้อมูลในส่วนที่ถูกบรรจุไว้ใน cache ก่อนหน้า ทำให้มีประสิทธิภาพเพิ่มขึ้น ลดปัญหาการเชื่อมต่อที่ล่าช้าและการใช้แบนด์วิดท์ของช่องสัญญาณได้

เทคโนโลยีของระบบดาวเทียมอินเทอร์เน็ตในปัจจุบัน

ปัจจุบันเมื่อเทคโนโลยีก้าวหน้าขึ้นหลายบริษัทได้สร้างระบบดาวเทียมที่ให้บริการข้อมูลแบบบรอดแบรนด์ ด้วยความสามารถในการส่งผ่านข้อมูลแบบบรอดแบรนด์ด้วยเทอร์มินอลขนาดเล็กที่ความเร็วสูงโดยมีการพัฒนารูปแบบอุปกรณ์การทำงานระบบจากเดิมขึ้น โดยสรุปได้ดังนี้

1).ส่วนของตัวดาวเทียม

- ใช้เทคนิคจานสายอากาศแบบหลายลำคลื่น (multiple spot beams) ที่สามารถปรับแต่ละลำคลื่นไปในทิศทางและพื้นที่แตกต่างกัน เพื่อเพิ่มแบนด์วิดท์จากการใช้ความถี่ซ้ำ

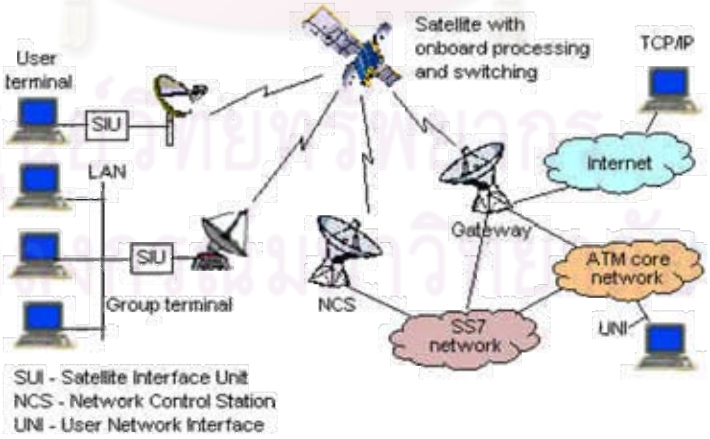
- ใช้ระบบการจัดการพลังงานแบบพลวัต (Dynamic Power Management) เพื่อจัดการกำลังส่งคลื่นขาลงของแต่ละบีมให้มีประสิทธิภาพดีที่สุดและเหมาะสมกับสภาวะอากาศ ทำให้ความสามารถการใช้งานในช่องสัญญาณในแต่ละบีมดียิ่งขึ้น

2).ส่วนภาคพื้นดิน

- ใช้เทคโนโลการเข้ารหัสและมอดูเลตสัญญาณที่มีประสิทธิภาพ เพื่อเพิ่มแบนด์วิดท์และลดความสิ้นเปลืองของการใช้พลังงาน

- มีการออกแบบโครงสร้างการส่งข้อมูลในช่องสัญญาณขาขึ้นและขาลงเพื่อการแบ่งการใช้งานของแบนด์วิดท์ระหว่างผู้ใช้งานที่มีจำนวนมาก

- ใช้ระบบการจัดสรรช่องสัญญาณแบบพลวัต (Dynamic Link Allocation) เพื่อปรับการมอดูเลท การเข้ารหัส และกำลังส่ง ของแต่ละสถานีที่มีสภาวะอากาศการลดทอนสัญญาณที่แตกต่างกัน



รูปที่ 14 โครงข่ายระบบดาวเทียมอินเทอร์เน็ตและการใช้งาน

2.6 ผลกระทบที่เกิดกับโปรโตคอล TCP บนโครงข่ายดาวเทียม

เดิมทีโปรโตคอล TCP ถูกคิดค้นขึ้นสำหรับการใช้งานในโครงข่ายแบบมีสาย หลังจากนั้นมีการประยุกต์มาใช้ในโครงข่ายไร้สายต่างๆรวมทั้งดาวเทียม เมื่อนำมาใช้บนโครงข่ายดาวเทียมมีหลายปัจจัยที่ส่งผลกระทบต่อการทำงานของโปรโตคอล เมื่อพิจารณาคุณลักษณะช่องสัญญาณในรูปแบบปลายทางไปยังปลายทางผลกระทบที่เกิดกับประสิทธิภาพการทำงานโปรโตคอล TCP ที่แตกต่างจากโครงข่ายมีสาย ทั้งการประวิงของเวลา แบนด์วิดท์ การสูญหายของแพ็คเกจเนื่องมาจากความคับคั่ง และอัตราผิดพลาดในการรับส่งข้อมูล สามารถสรุปได้ดังนี้

การประวิงของสัญญาณเนื่องจากการเดินทาง (Long Propagation Delay)

การประวิงเวลาในช่องสัญญาณมีองค์ประกอบหลัก 3 ส่วน คือ จากการเดินทางของสัญญาณ, การรับส่งข้อมูล และการจัดลำดับคิว ในกรณีของดาวเทียมแบบบรอดแบรนด์ส่วนที่มีนัยสำคัญมากที่สุดคือการประวิงของสัญญาณเนื่องจากการเดินทาง สำหรับการรับส่งข้อมูลในช่องสัญญาณของดาวเทียม GEO ที่ระดับความสูงเหนือพื้นผิวโลกประมาณ 36,000 km การประวิงของการเดินทางสัญญาณจากโลกไปยังดาวเทียมหรือจากดาวเทียมมายังโลกใช้เวลาประมาณ 125 ms แสดงได้ว่าการประวิงเวลาที่เกิดจากการเดินทางแบบ 1 ทิศทาง (one-way) โดยทั่วไปจะมีค่าอยู่ที่ 250 ms หรืออาจมากกว่านั้นซึ่งขึ้นอยู่กับช่วงเวลาในการแก้ไขอัตราผิดพลาดด้วย การแปรผันของการประวิงระยะรอบเวลาการเดินทาง (Round trip time) และของระบบสื่อสาร2 ทิศทาง (two-way system) มีประมาณ 500 ms รวมด้วยการประวิงเวลาจากโครงข่ายภาคพื้นดิน

เวลาที่ประวิงเนื่องจากการเดินทางสำหรับช่องสัญญาณของ GEO สามารถแก้ไขได้โดยใช้บัฟเฟอร์แบบ Doppler และจะทำให้ค่าการประวิงมีค่าประมาณ 300 ms ในกรณีของ LEO ที่วงโคจรอยู่ในระดับความสูงที่ต่ำกว่า เช่น 1000 km การประวิงของเวลามีค่า 20 ms ใน 1 ช่วง(hop) ซึ่งในแต่ละช่วงจะเพิ่มค่าประวิงขึ้นโดยขึ้นกับระยะทางระหว่างดาวเทียม และการเปลี่ยนแปลงเส้นทางเชื่อมต่อ สำหรับการเชื่อมต่อที่มีรอบเวลาการเดินทางนานนั้น TCP ใช้ช่วงระยะเวลายาวนานในขั้นตอนของ Slow Start ก่อนที่จะไปถึงค่าแบนด์วิทด์ที่เป็นไปได้ (available bandwidth) เวลาที่ใช้โดยช่วง Slow Start ของ TCP ไปถึงค่าแบนด์วิดท์ดาวเทียม (Sat BW) จะมีค่าประมาณ $RTT \times \log_2(\text{Sat BW} \times RTT)$

หากข้อมูลส่งได้รับการตอบกลับโดยไม่มีการผิดพลาด สำหรับการส่งผ่านข้อมูลที่มีขนาดสั้นมันจะสิ้นสุดกระบวนการในช่วง Slow Start ซึ่งเป็นการใช้แบนด์วิทด์อย่างไม่คุ้มค่า บางงานวิจัยได้นำเสนอการใช้วินโดว์ตั้งต้นที่มีขนาดใหญ่ (large initial window) [23] ซึ่งมีค่าถึง 4380 ไบต์แทน (หรือมากสุด 4 เซกเมนต์) ไฟล์ที่มีขนาดเล็กกว่า 4 กิโลไบต์จะสิ้นสุดการส่งผ่านข้อมูลเพียง 1รอบเวลาการเดินทาง

ผลการประวิงแบนด์วิดท์ขนาดใหญ่ (Large Bandwidth-Delay Product)

ในช่องสัญญาณของการสื่อสารดาวเทียมจะมีค่าผลการประวิงค์แบนด์วิทด์สูง ค่าผลการประวิงแบนด์วิดท์คือ ปริมาณข้อมูลสามารถที่ส่งเข้าไปยังช่องสัญญาณได้โดยไม่มี ACK ตอบกลังมายังผู้ส่ง หรือกล่าวได้ว่า หากมีท่อเชื่อมต่อจากผู้ส่งไปยังผู้รับ และผู้ส่งส่งข้อมูลผ่านท่อไป ผลการประวิงแบนด์วิดท์จะเป็นปริมาณข้อมูลทั้งหมดที่สามารถบรรจุเข้าไปในท่อได้ ค่าผลการประวิงแบนด์วิดท์หาได้จาก $(RTT \times BW)/2$ เมื่อ BW คือค่าแบนด์วิดท์ในช่องสัญญาณ ซึ่งในระบบการสื่อสารดาวเทียมจะมีค่า RTT และ BW สูง ทำให้ค่าผลการประวิงแบนด์วิดท์มีค่าสูงกว่าระบบโครงข่ายภาคพื้นดินมาก

สำหรับดาวเทียมวงโคจร GEO ที่ทำงานในย่านความถี่ k/ka ประมาณมีผลการประวิงแบนด์วิดท์ (ประมาณ 20 Mbps $\times$ 580 ms RTT) ในระบบขนาดใหญ่มากเมื่อเทียบกับโครงข่ายภาคพื้นดิน การที่จะใช้ประโยชน์จากช่องสัญญาณอย่างเต็มที่ต้องป้อนข้อมูลจำนวนมากเข้าไปในช่องสัญญาณ (เท่ากับผลการประวิงแบนด์วิดท์) ใน TCP จะส่งข้อมูลใหม่ก็ต่อเมื่อมีการตอบกลับจากข้อมูลชุดเก่า ซึ่งในส่วนโครงสร้างโปรโคตอล TCP พื้นฐาน วินโดว์ที่ได้มีการประกาศที่ด้านรับในส่วนหัว (header) ของ TCP มีค่าเพียงแค่ 2^{16} บิตหรือ 65,536 ไบต์ ซึ่งจำกัดอัตราการส่งผ่านของระบบไว้ที่ 1 เมกกะบิตต่อวินาทีในโครงข่าย การปรับขนาดวินโดว์ ได้ถูกนำเสนอขึ้นเพื่อทำการแก้ปัญหานี้ทำให้สามารถเพิ่มอัตราการส่งข้อมูลโดยการปรับแฟคเตอร์ที่อยู่ในส่วนของวินโดว์ อย่างไรก็ตามการปรับขนาดวินโดว์ไม่สามารถแก้ไขปัญหาได้ทุกส่วน เพราะมันใช้เวลายาวนานที่มากกว่าเมื่อเทียบกับโครงข่ายพื้นดินที่จะไปถึงค่าขนาดวินโดว์เป้าหมายเพราะว่าการประวิงสัญญาณเนื่องจากการเดินทางและอัลกอริทึม Slow Start ใน TCP และเทคนิคที่ใช้การทวีคูณวินโดว์ที่ทำให้วินโดว์มีขนาดใหญ่ขึ้นสามารถเกิดการคับคั่งได้โดยง่าย

อัตราการผิดพลาดบิตที่สูง (High Bit Error Rate)

ในช่องสัญญาณดาวเทียมอัตราการผิดพลาดบิต BER โดยทั่วไปอยู่ที่ 10^{-7} -10^{-4} การมอดูเลทและเทคนิคการเข้ารหัสสัญญาณ รวมทั้งการเพิ่มกำลังส่งให้สูงขึ้นจะช่วยทำให้อัตราการผิดพลาดบิตนั้นต่ำลงได้ถึง 10^{-10} ในระบบดาวเทียม GEO สำหรับระบบดาวเทียม LEO เส้นทางเชื่อมต่อแบบ multipath หรือ การถูกบดบังของสัญญาณอาจทำให้ค่า BER เปลี่ยนแปลงไปมาก ขณะที่ TCP ที่ออกแบบมาเพื่อป้องกันการสูญหายของข้อมูลเนื่องจากความคับคั่งในโครงข่าย หากได้รับผลกระทบจากอัตราการผิดพลาดบิตที่สูงอีก จะทำให้ TCP ลดระดับการคับคั่งของวินโดว์ (CWND) ลงทำให้ประสิทธิภาพการส่งผ่านข้อมูลลดต่ำลงได้

ความคับคั่งของข้อมูลในช่องสัญญาณ

ปัจจุบันระบบสื่อสารต่างๆ มีการส่งผ่านข้อมูลจำนวนมาก อาจจะนำพาข้อมูลเหล่านั้นโดยใช้ย่านความถี่สูงมาก ด้วยแบนด์วิดท์ขนาดใหญ่ของคลื่นความถี่วิทยุ หรือคลื่นความถี่แสง เมื่อระบบ

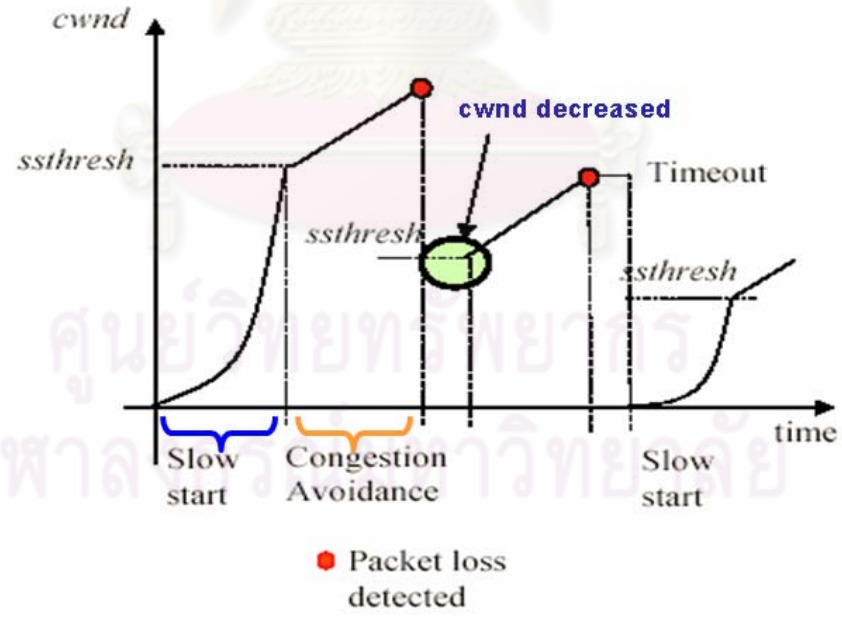
ดาวเทียมได้กลายเป็นส่วนหนึ่งของโครงข่ายสื่อสาร การเชื่อมต่อเพื่อรับส่งข้อมูลของช่องสัญญาณดาวเทียมเองเข้าร่วมกับระบบต่างๆจึงมีความสำคัญ โดยเฉพาะลักษณะช่องสัญญาณของดาวเทียมแม้มีแบนด์วิดท์ขนาดใหญ่แต่จำเป็นต้องมีการจัดสรรทรัพยากรเพื่อให้บริการแก่ผู้ใช้งานได้อย่างทั่วถึง ช่องสัญญาณดาวเทียมจึงมีลักษณะเสมือนเป็นช่องสัญญาณแบบคอขวดหนึ่ง (bottleneck) ที่ทำหน้าที่เชื่อมข้อมูลระหว่างตำแหน่งบนพื้นโลกที่ห่างไกล ด้วยองค์ประกอบในช่องสัญญาณที่ถูกจำกัดสเปคตรัมในการ uplink และ downlink รวมทั้งการเชื่อมต่อกับเข้ากับเกตเวย์และอุปกรณ์ในการจัดหาเส้นทางระหว่างระบบเครือข่ายย่อยต่างๆ ทำให้เกิดการคับคั่งขึ้นโดยง่าย ดังนั้นกระบวนการที่ทำหน้าที่ในการควบคุมการคับคั่งจึงตกมาอยู่ที่กระบวนการทำงานของตัวโปรโตคอลเป็นสำคัญ โดยโปรโตคอล TCP เองก็มีกระบวนการควบคุมความคับคั่งซึ่งกล่าวถึงไปข้างต้น เพื่อให้การทำงานของ TCP ร่วมกับช่องสัญญาณดาวเทียมมีประสิทธิภาพ จึงจำเป็นต้องมีการพัฒนาและปรับปรุงกระบวนการดังกล่าวให้สอดคล้องกับระบบโครงข่ายที่นำมาใช้งานยิ่งขึ้น

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

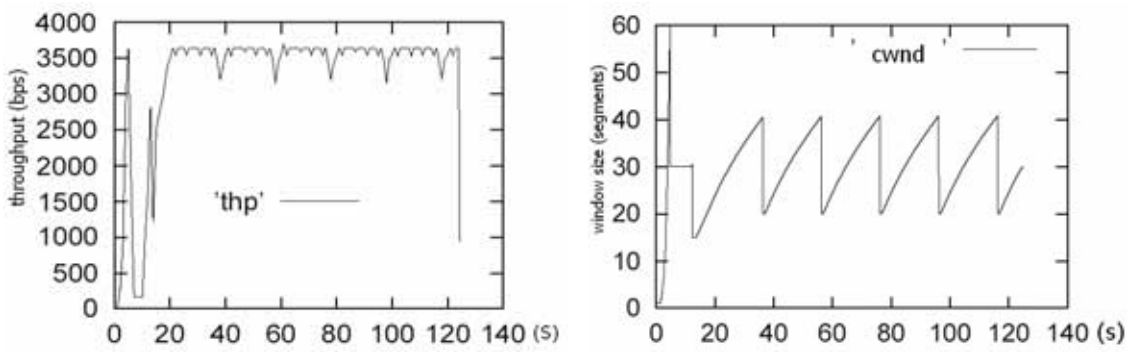
บทที่ 3

วิธีดำเนินการวิจัย

ในการควบคุมความคับคั่ง TCP ในโครงข่ายดาวเทียม เปรียบเสมือนการสร้างความน่าเชื่อถือให้กับข้อมูลและประสิทธิภาพของโครงข่าย โดยใช้กลไกลการควบคุมที่ช่วยให้ข้อมูลสามารถส่งผ่านช่องสัญญาณไปยังผู้รับได้อย่างคุ้มค่าและเกิดข้อผิดพลาดน้อยที่สุด ในมาตรฐานการควบคุมความคับคั่งของ TCP หรือ TCP-Reno และ NewReno [24] ที่นำมาใช้ในปัจจุบันอย่างแพร่หลาย จะใช้วิธีการตรวจสอบแบนด์วิดท์ที่เป็นไปได้บนช่องสัญญาณโดยการเพิ่มขนาดของวินโดว์จนกระทั่งโครงข่ายเกิดการคับคั่งและลดขนาดของวินโดว์ลงแบบทวีคูณดังรูปที่12 โดยอัลกอริทึม AIMD อย่างไรก็ตามวิธีการลดขนาดวินโดว์แบบนี้ไม่ค่อยจะสมเหตุสมผลในการใช้งานบนช่องสัญญาณดาวเทียมอินเทอร์เน็ต เช่น ใช้วิธีลดขนาด cwnd ลงครึ่งหนึ่งของค่า cwnd เดิมเพื่อส่งข้อมูลใหม่ เมื่อมีนัยบ่งชี้ว่ามีการคับคั่งในช่องสัญญาณ (ได้รับสัญญาณการตอบรับซ้ำ (dupACKs)) จากการจำลองโครงข่ายช่องสัญญาณที่ใช้มาตรฐานการควบคุมการคับคั่งนี้ในช่วงที่มีการลดขนาดของวินโดว์ลงอย่างรวดเร็วจะทำให้อัตราการส่งผ่านของข้อมูลในช่องสัญญาณลดลงไปด้วย ดังแสดงดังรูปที่ 13



รูปที่ 15 กลไกลการควบคุมการคับคั่งมาตรฐานเมื่อเกิดการลดขนาด cwnd



(ก) (ข)

รูปที่ 16 (ก) อัตราการส่งผ่านข้อมูล (throughput) (ข) ขนาดความคับคั่งของวินโดว์ (cwnd) ของ TCP-Reno เมื่อเกิดการคับคั่งในช่องสัญญาณ

ในวิทยานพนธ์นี้จึงมีแนวคิดที่จะนำวิธีการประมาณแบนด์วิดท์มาใช้ทางด้านผู้ส่งโดยใช้พื้นฐานมาจากการวัดช่วงเวลาระหว่างการส่งกลับของสัญญาณตอบรับ (ACK) มาประมาณเป็นค่าความเป็นไปได้แบนด์วิดท์ ที่แสดงเป็นนัยถึงอัตราข้อมูลที่ได้ทำการส่งไปในช่องสัญญาณ หลังจากนั้นจะนำค่าที่ได้จากการประมาณแบนด์วิดท์มาช่วยในการปรับขนาดวินโดว์ที่ใช้ในการบรรจุข้อมูลและจุดอิ่มตัวของช่องสัญญาณ (cwnd และ slowstart threshold) เพื่อหลีกเลี่ยงการลดลงของอัตราการส่งผ่านข้อมูลที่ไม่สมเหตุผล

3.1 การวัดแบนด์วิดท์ระหว่างปลายทาง (end to end)

ในรูปแบบของโครงข่ายที่รวมทั้งแหล่งข้อมูลและผู้รับข้อมูล การสร้างความน่าเชื่อถือและคุณภาพการให้บริการข้อมูลเป็นสิ่งสำคัญ โดยเฉพาะช่องสัญญาณในโครงข่ายที่มีลักษณะทางกายภาพหลากหลายมีแหล่งรับส่งข้อมูลจำนวนมาก การพิจารณาการรับส่งข้อมูลของโครงข่ายในรูปแบบปลายทางไปยังปลายทางจึงจำเป็นต้องให้ความสำคัญกับแหล่งส่งและรับข้อมูลโดยตรงเพื่อพิจารณาคุณภาพการบริการที่สามารถกระทำได้ในช่องสัญญาณ

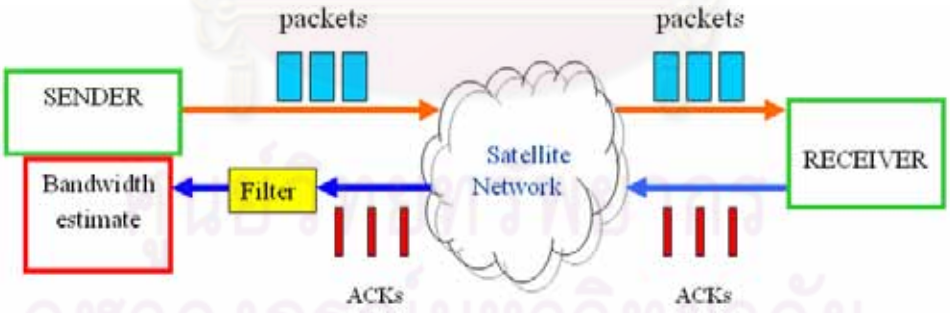
3.1.1 การวัดโดยใช้พื้นฐานของสัญญาณการตอบรับ (ACK-based)

พื้นฐานในการออกแบบอัลกอริทึมเพื่อแก้ปัญหาความคับคั่งของ TCP ที่อยู่ในรูปแบบปลายทางไปปลายทางนั้น โครงข่ายจะถูกพิจารณาเสมือนเป็นกล่องดำ ที่แหล่งส่งข้อมูล TCP ไม่สามารถรับทราบความคับคั่งของโครงข่ายได้อย่างชัดเจน ดังนั้นแหล่งส่งข้อมูลต้องกำหนดอัตราส่งข้อมูลที่มันสามารถส่งไปได้ ต้องมีการตรวจสอบข้อมูลโดยการเดินหน้าเพิ่มอินพุทโหลดให้กับช่องสัญญาณ (โดยกระทำผ่านกระบวนการ Slow Start และ Congestion Avoidance) จนกระทั่งมีการตอบรับที่แสดงเป็นนัย เช่นสิ้นสุดคาบเวลาการรอสัญญาณตอบรับหรือเกิดสัญญาณตอบรับที่ซ้ำกัน แสดงให้เห็นว่าสัญญาณที่ส่งไปได้ไปถึงค่าความจุที่สามารถรองรับได้

วิทยานิพนธ์นี้จึงนำเสนอแนวทางในการประมาณโดยใช้ประโยชน์จากสัญญาณการตอบรับ (ACK) ของ TCP [25] มาช่วยในการหาค่าที่เหมาะสมของการส่งข้อมูลในช่องสัญญาณ ด้วยวิธีการประมาณความเป็นไปได้ของแบนด์วิดท์ในการเชื่อมต่อของ TCP โดยใช้การวัดและการเฉลี่ยอัตราการส่งกลับของสัญญาณการตอบรับ

เมื่อ ACK ตอบกลับมายังผู้ส่ง มันจะเป็นตัวบ่งชี้ที่แสดงให้ทราบถึงอัตราปริมาณของข้อมูลที่ตอบสนองในการส่งผ่านสำเร็จและได้รับโดยปลายทาง ในกรณีที่กระบวนการส่งไม่ได้รับผลกระทบจากการสูญหายของข้อมูล สามารถทำการเฉลี่ยข้อมูลที่ส่งสำเร็จได้โดยพิจารณาสัญญาณตอบรับบนแกนเวลา เพื่อประมาณเป็นค่าแบนด์วิดท์ของช่องสัญญาณในขณะนั้น และเมื่อสัญญาณตอบรับซ้ำมาถึงยังผู้ส่งบ่งบอกให้ทราบว่ามีลำดับข้อมูลบางส่วนตกหล่นไปจะใช้การนับของสัญญาณตอบรับอันดับถัดไป โดยประมาณแบนด์วิดท์ด้วยค่าใหม่ที่ได้จากการคำนวณหลังการรับนั้นหลังจากเกิดการคับคั่ง (เช่นผู้ส่งรับสัญญาณการตอบรับซ้ำๆกัน 3 ชุดหรือหมดคาบเวลาACK) เมื่อได้ค่าประมาณแบนด์วิดท์แล้ว จะนำมาใช้ในกลไกลควบคุมการส่งข้อมูลในช่องสัญญาณทางด้านผู้ส่งเพื่อปรับขนาดวินโดว์ [26]ที่ใช้ในการบรรจุข้อมูลและจุดอิ่มตัวของช่องสัญญาณ (cwnd และ slowstart threshold)

อย่างไรก็ตามในกรณีมีข้อมูลตกหล่นผู้ส่งไม่รู้ว่าเซกเมนต์ไหนถูกกระตุ้นให้มีการส่งซ้ำโดยสัญญาณการตอบรับ และไม่สามารถที่จะอัพเดทข้อมูลการนับโดยค่าขนาดของเซกเมนต์ได้ จึงใช้วิธีการเฉลี่ยของขนาดเซกเมนต์ที่ถูกส่งไปอย่างต่อเนื่องในการเชื่อมต่อ โดยใช้วิธีการสะสมสัญญาณการตอบรับเพื่อใช้ในการปรับตัวของการประมาณโดยสมมติให้ทุก TCP เซกเมนต์มีขนาดเท่ากัน และกำหนดให้ลำดับเลขชุดข้อมูลเพิ่มขึ้น 1 ลำดับต่อชุดข้อมูลที่ส่งไป

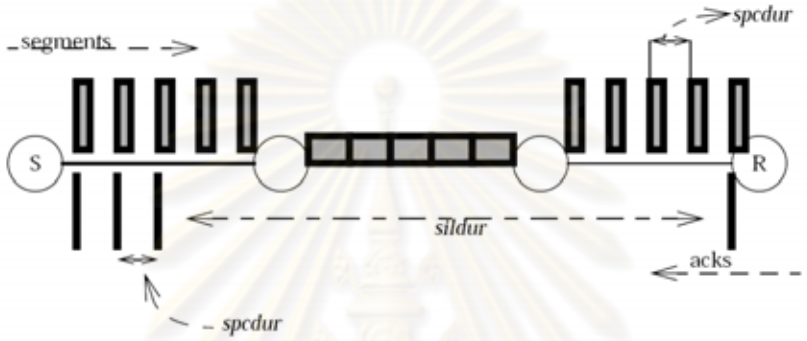


รูปที่ 17 ระบบการทำงานของการวัดแบนด์วิดท์โดยใช้พื้นฐานสัญญาณการตอบ

ส่วนสำคัญคือก่อนที่เกิดการคับคั่งในช่องสัญญาณจะเกิดการอิ่มตัวของอัตราการส่งข้อมูล จะพบว่าค่าประมาณของแบนด์วิดท์ในขณะนั้นจะมีค่ามากสุดที่เป็นไปได้ (แบนด์วิดท์ในสภาวะอิ่มตัว) จากนั้นหลังสภาวะอิ่มตัวจึงเกิดแพ็คเกจตกหล่นซึ่งทางด้านผู้ส่งสามารถรับรู้ได้จากการรับสัญญาณตอบรับซ้ำชุดเดิม ทำให้สามารถนำค่าการประมาณแบนด์วิดท์ที่ได้ก่อนหน้านั้นมาเป็นเกณฑ์ในการตั้งจุดอิ่มตัวในการส่งข้อมูลในช่องสัญญาณลำดับถัดไป

3.1.2 การกรองอัตราการรับ ACK

ภายในกลไกลการทำงานของ TCP เองมีวิธีการสร้างความน่าเชื่อถือของการส่งผ่านข้อมูลโดยใช้สัญญาณการตอบรับ (ACK) ที่แสดงให้แหล่งส่งข้อมูลทราบว่าข้อมูลชุดนั้นได้ไปถึงยังปลายทางแล้ว และเมื่อ ACK ตอบกลับมายังผู้ส่ง จะเป็นแสดงตัวเป็นนัยบ่งชี้ที่แสดงให้ทราบถึงอัตราปริมาณของข้อมูลที่ตอบสนองในการส่งผ่านสำเร็จและได้รับโดยปลายทาง ในช่วงเวลาก่อนที่กระบวนการส่งไม่ได้รับผลกระทบจากการสูญหายของข้อมูล สามารถทำการเฉลี่ยข้อมูลที่ส่งสำเร็จได้โดยพิจารณาสัญญาณตอบรับบนแกนเวลา เพื่อประมาณเป็นค่าความจุหรือแบนด์วิดท์ของช่องสัญญาณในขณะนั้น



รูปที่ 18 ชุด ACKs ในช่องสัญญาณกับปริมาณข้อมูลที่ส่งผ่านสำเร็จ

วิธีการคือใช้การพิจารณาสัญญาณ ACKs ที่ตอบกลับมายังผู้ส่งบนแกนเวลาโดยเสมือนเป็น trigger เพื่อสุ่มค่าปริมาณข้อมูลที่ส่งผ่านสำเร็จในช่องสัญญาณ

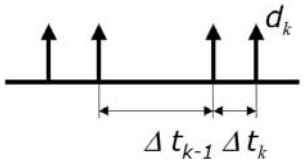
ถ้า ACK หนึ่งที่รับได้ทางผู้ส่ง ณ เวลา t_k บอกเป็นนัยได้ว่ามีปริมาณการตอบสนองของข้อมูล d_k ที่ฝั่งรับ โดยสามารถทำการสุ่มแบนด์วิดท์ที่ใช้ในการเชื่อมต่อนี้เป็น

$$b_k = \frac{d_k}{t_k - t_{k-1}} \tag{3.1}$$

เมื่อ d_k คือปริมาณของข้อมูลที่ได้รับการตอบกลับโดย ACK

t_{k-1} คือเวลาที่รับสัญญาณตอบกลับก่อนหน้า

เมื่อเกิดการคับคั่งขึ้นอัตราอินพุทของความถี่ต่ำเกินความจุของช่องสัญญาณ [27] จะใช้ตัวกรองความถี่ต่ำผ่านเพื่อเฉลี่ยผลจากการวัดสุ่มและรับเอาองค์ประกอบความถี่ต่ำของแบนด์วิดท์ที่สามารถเป็นไปได้



รูปที่ 19 การสุ่มปริมาณข้อมูลเมื่อเทียบกับแกนเวลา

ในการเลือกตัวกรอง การเฉลี่ยแบบเอ็กโพเนนเชียลพื้นฐานที่นำมาใช้ในการประมาณรอบเวลาการเดินทาง(RTT)ใน TCP นั้นไม่มีค่อยเหมาะสมในการกรององค์ประกอบของความถี่สูงออก จึงใช้ตัวกรองที่ไม่ต่อเนื่องทางเวลาซึ่งประกอบด้วยความไม่ต่อเนื่องของตัวกรองความถี่ต่ำผ่านต่อเนื่องที่ใช้การประมาณ Tustin [28]ได้ว่า

$$\hat{b}_k = \frac{\frac{2\tau}{t_k - t_{k-1}} - 1}{\frac{2\tau}{t_k - t_{k-1}} + 1} \times \hat{b}_{k-1} + \frac{b_k + b_{k-1}}{\frac{2\tau}{t_k - t_{k-1}} + 1} \tag{3.2}$$

เมื่อ b_k, b_{k-1} คือ ค่าการสุ่มที่เวลา k และเวลา k-1 ตามลำดับ

$\hat{b}_k$ คือ ค่าความเป็นไปได้ของแบนด์วิดท์โดยตัวกรองที่เวลา t_k และ $\frac{1}{\tau}$ คือจุดตัดความถี่ (cutoff frequency) ของตัวกรอง โดยโครงสร้างพื้นฐานของตัวกรอง (3.1) มีหลักการทำงานโดยการพิจารณาค่าคงที่ระหว่างเวลา $t_k - t_{k-1} = \Delta_k$ จัดรูปให้มีสัมประสิทธิ์เป็นค่าคงที่

$$\hat{b}_k = a\hat{b}_{k-1} + \frac{(1-a)}{2}\left[b_k + b_{k-1}\right] \tag{3.3}$$

เมื่อ $a = \frac{2\tau}{\Delta_k} - 1 \Big/ \frac{2\tau}{\Delta_k} + 1$

สัมประสิทธิ์ a เป็นผลที่ขึ้นกับการเปลี่ยนแปลงระหว่างช่วงเวลาโดยจะขึ้นกับค่า $t_k - t_{k-1} = \Delta_k$ เมื่อช่วงคาบเวลามีค่ามากขึ้น ค่าสุดท้าย $\hat{b}_{k-1}$ ควรจะมีนัยสำคัญน้อย ในทางกลับกันช่วงคาบของเวลามีค่าน้อยลง $\hat{b}_{k-1}$ ควรมีนัยสำคัญเพิ่มมากขึ้น ค่าสัมประสิทธิ์จะลดลงเมื่อช่วงคาบเวลาเพิ่มขึ้น และทำให้ค่า b_{k-1} ที่เวลาก่อนหน้ามีนัยสำคัญน้อยลงด้วยโดยค่าการสุ่ม 2 ครั้งล่าสุดซึ่งถูกนำมาคิดค่าน้ำหนักโดยคูณกับ $\frac{(1-a)}{2}$

ตัวอย่างเช่น ให้ค่า $\Delta_k = \frac{\tau}{10}$ ทำให้สมการ (3.3) มีค่าเป็น $\hat{b}_k = \frac{19}{21}\hat{b}_{k-1} + \frac{2}{21}\left(\frac{b_k + b_{k-1}}{2}\right)$ ค่า $\hat{b}_k$ ใหม่คือค่าที่ได้จากการประมาณ 90 % ของค่าก่อนหน้า ($\hat{b}_{k-1}$) บวกกับการประมาณ 10 % ในการคำนวณเฉลี่ยของสองลำดับการสุ่มสุดท้าย ($b_k + b_{k-1}$)

เมื่อการกรองโดยใช้จุดตัดความถี่ที่มีค่าเท่ากับ $\frac{1}{\tau}$ ทุกองค์ประกอบความถี่ที่เกินค่านี้จะถูกกรองออก ตามทฤษฎีการสุ่มของไนควิสต์ ในการสุ่มสัญญาณระยะคาบของการสุ่มจะต้องน้อยกว่าหรือเท่ากับ $\frac{\tau}{2}$ ในแต่ละชุดของสัญญาณ ACK นั้นจะอยู่ในรูปแบบอซิงโครนัส ซึ่งความถี่ในการสุ่มนั้นไม่แน่นอน ถ้าจะให้เป็นไปตามทฤษฎีของไนควิสต์ต้องสร้างเงื่อนไขว่า ถ้าพ้นระยะเวลา $\tau/m\,(\mathrm{m} \geq 2)$ เมื่อ

นับจาก ACK สุดท้ายที่ได้รับโดยไม่มีการรับ ACK ใหม่ ตัวกรองจะทำการกำหนดให้การกรองเป็นการสุ่มเสมือน (virtual sample)นั้นมีค่า $b_k = 0$ โดยแสดงสถานการณ์ได้ดังรูปที่ 17

รูปที่ 20 ช่วงระยะเวลาสุ่มที่มีการสุ่มเสมือน (virtual sample) แทรกอยู่

ซึ่งจะพิจารณาว่าหลังจากช่วงเวลาที่กำหนดหากไม่มีการรับ ACKs (เช่นเพราะไม่มีข้อมูลชุดใหม่ส่งไป) ตัวกรองจะทำหน้าที่เป็นการสุ่มแบบเสมือน เมื่อ t_{k-1} เป็นเวลาที่ได้รับ ACK, $\hat{t}_{k+j}$ เป็นระยะเวลาที่มาถึงของการสุ่มเสมือน $\hat{t}_{k+j+1} - \hat{t}_{k+j} = \tau/m$ เมื่อ j = 0, n-1 $b_{k+j} = 0$ เมื่อ j = 0, n-1 คือการสุ่มเสมือน และ $b_{k+n} = \frac{d_{k+n}}{t_{k+n} - t_{k+n-1}}$ เป็นการสุ่มที่คำนวณได้เมื่อได้รับ ACK ที่เวลา t_{k+n}

3.1.3 ค่าการประมาณแบนด์วิดท์กับการปรับขนาดวินโดว์

เมื่อช่องสัญญาณเกิดการคับคั่งขึ้นซึ่งเกิดการอิ่มตัวของอัตราการส่งข้อมูล จะพบว่าค่าประมาณของแบนด์วิดท์ในขณะนั้นจะมีค่ามากสุดที่เป็นไปได้ (แบนด์วิดท์ในสภาวะอิ่มตัว) จากนั้นหลังสภาวะอิ่มตัวจึงเกิดแพ็คเกจตกหล่นซึ่งทางด้านผู้ส่งสามารถรับรู้ได้จากการรับสัญญาณตอบรับซ้ำชุดเดิม ทำให้สามารถนำค่าการประมาณแบนด์วิดท์ที่ได้ก่อนหน้านั้นมาเป็นเกณฑ์ในการตั้งจุดอิ่มตัวในการส่งข้อมูลในช่องสัญญาณลำดับถัดไป

โดยนำค่าที่ได้จากการประมาณแบนด์วิดท์มาช่วยในกลไกลควบคุมการปรับขนาดวินโดว์ที่ทำหน้าที่ในการบรรจุข้อมูลกำหนดจุดอิ่มตัวของช่องสัญญาณ (cwnd และ slowstart threshold) เพื่อที่จะหลีกเลี่ยงการลดลงของอัตราการส่งผ่านข้อมูลอันไม่สมเหตุผลเมื่อเกิดการคับคั่ง ซึ่งเกิดขึ้นในวิธีการควบคุมการคับคั่งมาตรฐาน โดยจะมีเงื่อนไขหลักที่นำมาพิจารณาคือการกำหนดจุดอิ่มตัวของช่องสัญญาณใหม่ (ssthresh) ซึ่งมีผลกับการทำงานของ TCP ดังนี้

1. ถ้าหากกำหนดค่า ssthresh ต่ำเกินไป เมื่อทำการเปรียบเทียบกับผลการประวิงแบนด์วิดท์ (bandwidth-delay product) ในขั้นตอนการทำงานของ TCP จะใช้เวลานานเมื่อทำงานอยู่ในขั้นตอน congestion avoidance ทำให้ประสิทธิภาพในการใช้แบนด์วิดท์ช่องสัญญาณไม่คุ้มค่า

2. ถ้าหากกำหนดค่า ssthresh สูงเกินไป การทำงาน TCP ขั้นตอน slow start ที่มีการเพิ่มตัวของ cwnd แบบ exponential อาจเป็นเหตุให้เกิดการสูญหายของแพ็คเกจขึ้นหลายชุดในช่องสัญญาณแบบคอขวด ตามด้วยการทำงานกลไกล recovery ที่ยาวนานเมื่อเกิด timeouts

ดังนั้นเมื่อมีการคับคั่งขึ้นการอิ่มตัวของช่องสัญญาณทำให้มีแพ็คเกจข้อมูลสูญหาย หมายถึงการใช้ความจุของช่องสัญญาณเต็มที่ ซึ่งทางด้านผู้ส่งสามารถรับรู้ได้จากการรับสัญญาณตอบรับซ้ำชุดเดิม (ได้รับ ACKs 3 ชุดซ้ำ) ดังนั้นจึงควรกำหนดให้ด้านผู้ส่งทำการปรับตั้งค่า cwnd และ ssthresh ใหม่โดยขึ้นกับค่าแบนด์วิดท์ที่ประมาณไว้ ($\hat{b}_k$) ให้เท่ากับค่าความเป็นไปได้ของขนาดความจุช่องสัญญาณซึ่งมีค่าเป็นไปตามสมการ (3.4)

$$ssthresh = \frac{\hat{b} \times RTT_{\min}}{seg_size} \tag{3.4}$$

เมื่อค่า $\hat{b}_k$ คือ ค่าแบนด์วิดท์ที่ได้จากการประมาณ

$RTT_{\min}$ คือ ค่ารอบเวลาการทำงานที่น้อยสุด

seg_size คือ ขนาดของเซกเมนต์ข้อมูล

สร้างเงื่อนไขขึ้นกรณีแรกโดยถ้าผู้ส่งได้กรณีได้รับ 3 ACKs

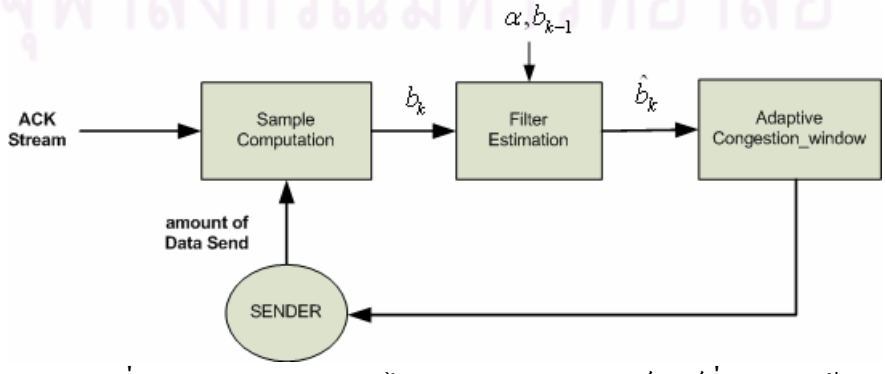
ถ้าค่าขนาดของ $cwnd > ssthresh$ ซึ่งอยู่ในขั้นตอนของ congestion avoidance ให้ทำการปรับค่า cwnd เป็นค่าใหม่ตามเงื่อนไข

$$\text{ถ้า } cwnd > ssthresh \rightarrow cwnd = ssthresh$$

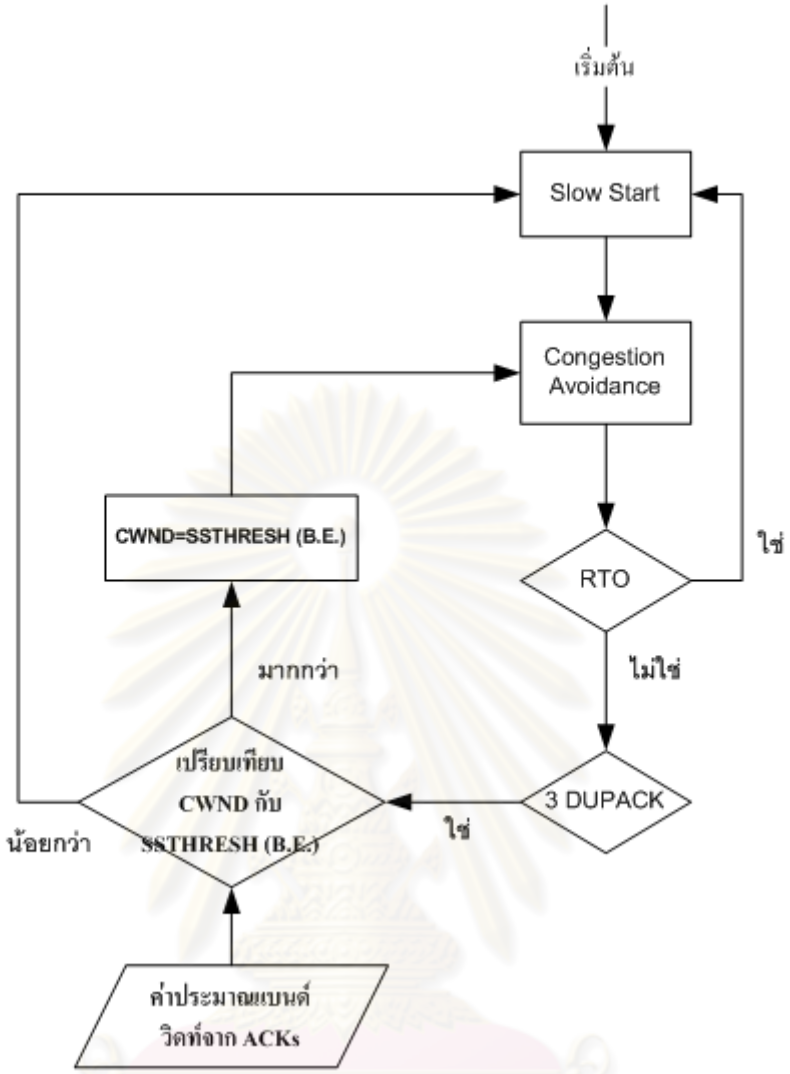
$$cwnd < ssthresh \rightarrow keep\ recent\ value$$

เพื่อปรับตั้งค่า cwnd ใหม่ให้เป็นไปตามค่าการประมาณ เพื่อลดค่า cwnd ที่เกินจริงในกลไกลมาตรฐานที่เป็นเหตุให้เกิดการการคับคั่งและลดขนาด cwnd ลงอย่างฉับพลัน cwnd จะถูกกำหนดให้มีค่าเท่ากับ ssthresh ในกลไกล congestion avoidance อีกครั้งเพื่อทำการตรวจสอบความเป็นไปได้จริงของแบนด์วิดท์ในช่องสัญญาณใหม่ โดยค่า $RTT_{\min}$ เป็นค่ารอบเวลาการทำงานที่น้อยสุดโดยการสังเกตุจากช่วงระยะเวลาในการเชื่อมต่อ ในการตั้งค่านี้จะทำให้ข้อมูลยังคงสามารถไหลเข้าสู่ช่องสัญญาณได้ และหาก cwnd < ssthresh ก็สามารถทำหน้าที่ตรวจสอบความจุช่องสัญญาณโดยกลไกล slow start เดิม

ในกรณีที่สองเมื่อเกิดการสิ้นสุดของคาบเวลา (timeout) ก็จะกำหนดลดค่าให้ cwnd มีค่าเริ่มต้นเป็น 1 ใหม่ โดยค่าของ ssthresh ยังคงเป็นไปตามสมการ (3.4) เพื่อให้เริ่มเข้าสู่กระบวนการ slow start และเพิ่มขนาด cwnd ในรูปแบบ exponential ใหม่



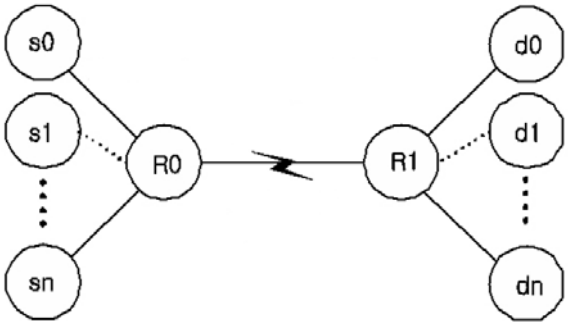
รูปที่ 21 การทำงานของกลไกลการประมาณแบนด์วิดท์ที่แหล่งส่งข้อมูล



รูปที่ 22 แผนภูมิการทำงานของกลไกลการประมาณแบนด์วิดท์เมื่อได้รับชุด ACKs ซ้ำ

3.2 แบบจำลองการทำงาน

การจำลองการทำงานในวิทยานิพนธ์นี้จะใช้โปรแกรมจำลองโครงข่าย Network Simulator (NS2) ด้วยโครงสร้างแบบจำลองแบบ Peer to Peer ซึ่งเป็นแบบจำลองที่มีลักษณะโครงสร้างเดียวกันกับการทำงานของโครงข่ายดาวเทียมอินเทอร์เน็ต ซึ่งรองรับการทำงานของโปรโตคอล TCP รูปแบบ end to end ซึ่งแบบจำลองนี้เป็นต้นแบบในการทดสอบหาประสิทธิภาพของโครงข่ายและความเท่าเทียมกันในการส่งข้อมูล โดยแบบจำลองจะประกอบไปด้วยแหล่งกำเนิดจำนวน N แหล่งกำเนิด ดังแสดงในรูป 23 ทำหน้าที่เป็นแหล่งกำเนิด TCP ทางด้านต้นทางปลายทาง



รูปที่ 23 โครงสร้างแบบจำลอง Peer to Peer

โดยมีข้อกำหนดของแบบจำลอง Peer to Peer ดังนี้

- ทำการสร้างจำนวนแหล่งกำเนิดที่จำเป็นเพื่อใช้แสดงสมรรถนะในการจำลอง
- ทุกแหล่งกำเนิดจะทำการส่งข้อมูลต่อเนื่องตลอดเวลา โดยโปรโตคอล TCP จะควบคุมการทำงานในการส่งข้อมูลให้เป็นไปตามกลไกลของการปรับขนาดวินโดว์ที่ได้กำหนดขึ้น
- ในช่องสัญญาณจะทำการกำหนดช่วงเวลาการประวิง(delay time) และความจุ (capacity) ของโครงข่ายให้ใกล้เคียงเสมือนโครงข่ายจริง
- ข้อมูลจะทำการส่งไปในทิศทางเดียว (unidirectional) โดยต้นทางจะทำหน้าที่ส่งข้อมูลส่วนด้านรับจะทำหน้าที่ส่งสัญญาณการตอบรับ
- ทุกแหล่งกำเนิดเริ่มต้นส่งข้อมูลพร้อมกัน

3.3 การหาสมรรถนะของ TCP

การประเมินหาการทำงานของโปรโตคอล TCP ในโครงข่ายในรูปแบบปลายทางไปปลายทาง (end-to-end) สามารถหาสมรรถนะในการทำงานได้โดยใช้พารามิเตอร์ goodput, efficiency, fairness และ friendliness [29] ซึ่งสามารถแสดงให้เห็นความสามารถในการทำงานของโปรโตคอลในด้านต่างๆ เพื่อพิจารณาให้เหมาะสมกับการใช้งาน

1). การหาอัตราการส่งผ่าน goodput

คือปริมาณข้อมูลที่แท้จริงที่ส่งเข้าสู่โครงข่ายแสดงไว้ในสมการที่ (3.5) ซึ่งแสดงถึงอัตราการส่งผ่านที่แท้จริงที่เข้าสู่โครงข่ายโดยไม่เกิดข้อผิดพลาดส่วนด้วยช่วงเวลาที่ใช้ในการส่งผ่านข้อมูล

$$Goodput = \frac{sent_data - retransmission_data}{transfer_time} \qquad (3.5)$$

2). การหาค่า efficiency

โดยกำหนดให้ N คือจำนวนคู่ของแหล่งกำเนิดกับปลายทางของ TCP ที่ส่งข้อมูลเข้าไปในโครงข่ายที่มี X_i คืออัตราการส่งผ่านของแหล่งกำเนิด i และ ให้ C คืออัตราการส่งผ่านของ TCP ที่มากสุดที่เป็นไปได้ (Maximum possible TCP throughput) ของช่องสัญญาณดังนั้นประสิทธิภาพของโครงข่ายคืออัตราส่วนของผลรวมของอัตราการส่งผ่าน TCP จริง (Sum of TCP throughput) ส่วนอัตราการส่งผ่านของ TCP มากสุดที่เป็นไปได้ สามารถเขียนเป็นสมการได้ดังนี้

$$Efficiency = \frac{\sum_{i=1}^{N} X_i}{C} \qquad (3.6)$$

3). การหาค่า Fairness

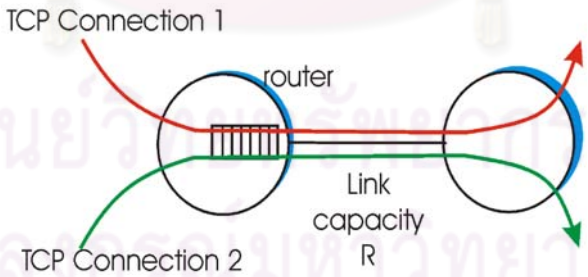
ค่า Fairness คือ ความสามารถในการทำงานร่วมกันของแบบแผน TCP เดียวกัน (intra protocol) ในการวัดสมรรถนะ Fairness จะใช้ Fairness Index Function ดังสมการ

$$\text{Fairness index} = \frac{\left[\sum_{i=1}^{N} X_i\right]^2}{N \times \sum_{i=1}^{N} X_i^{\,2}} \qquad (3.7)$$

X_i คือ อัตราการส่งผ่านของ TCP ลำดับที่ i

N คือจำนวนการเชื่อมต่อของ TCP

ค่า Fairness index จะมีค่าตั้งแต่ 1/N ถึง 1 หากมีการจัดสรรแบนด์วิดท์แบบสมบูรณ์ Fairness Index Function จะมีค่าเท่ากับ1



รูปที่ 24 โครงสร้างการทำงานร่วมกันของแบบแผน TCP เดียวกัน

4).การหาค่า Friendliness

Friendliness คือ ความสามารถในการทำงานร่วมกันของแบบแผน TCP ที่ต่างกัน (inter protocol) ในการวัดสมรรถนะ Friendliness จะทำโดยการวัดค่าอัตราการส่งผ่านของแต่ละแบบแผน TCP ที่อยู่บนโครงข่ายเดียวกัน (โดยใช้ช่องสัญญาณเดียวกัน)

บทที่ 4

ผลการจำลองและการวิเคราะห์

การจำลองผลในวิทยานิพนธ์นี้ทำการจำลองเพื่อหาสมรรถนะของการประมาณแบนด์วิดท์แบบปรับตัวได้ในช่องสัญญาณโดยใช้โปรแกรมในการจำลองการทำงาน NS2 (Network Simulator 2)[30] โดยได้จัดแบ่งหัวข้อการจำลองในการประเมินประสิทธิภาพดังนี้

1. สมรรถนะของกลไกลการประมาณแบนด์วิดท์ในช่องสัญญาณ
2. การปรับขนาดของ CWND, SSTHRESH และค่า goodput ของ TCP บนโครงข่ายดาวเทียม
3. อัตราการส่งผ่านข้อมูลในช่องสัญญาณเมื่อเทียบอัตราการผิดพลาดบิตและช่วงเวลาเดินทาง
4. ความสามารถในการทำงานร่วมกับแบบแผนทีซีพีที่ต่างกัน

4.1 สมรรถนะของกลไกลการประมาณแบนด์วิดท์ในช่องสัญญาณ

ในส่วนนี้จะทำการจำลองเพื่อประเมินหาสมรรถนะของกลไกลการประมาณแบนด์วิดท์ที่นำเสนอโดยทำการพิจารณา TCP 1 คู่เชื่อมต่อ ในช่องสัญญาณแบบคอขวดโดยทำงานร่วมกับโปรโตคอล UDP 2 คู่เชื่อมต่อ ที่มีรูปแบบแหล่งจ่ายแบบปิด-เปิดเพื่อให้มีการแบ่งแบนด์วิดท์ในช่องสัญญาณของ TCP ไป โดยทำการสร้างคู่เชื่อมต่อของ TCP เพียงหนึ่งการเชื่อมต่อเพื่อให้แสดงสมรรถนะของการประมาณโดยแท้จริง โดยพารามิเตอร์ที่ให้ความสนใจคือ ค่า τ ซึ่งเป็นคาบเวลาของตัวกรองซึ่งสัมพันธ์กับจุดตัดความถี่ของกลไกลการประมาณ รูปแบบการจำลองจะทำการกำหนดพารามิเตอร์ต่างๆ เป็นดังนี้

1). กำหนดช่องสัญญาณแบบคอขวด1 ช่องสัญญาณมีความจุ 5 Mb/s และค่าประวิงเวลา 30 ms

2). โหนดตัวกลางมีค่าความจุบัฟเฟอร์เท่ากันกับค่าผลการประวิงแบนด์วิดท์ในช่องสัญญาณ

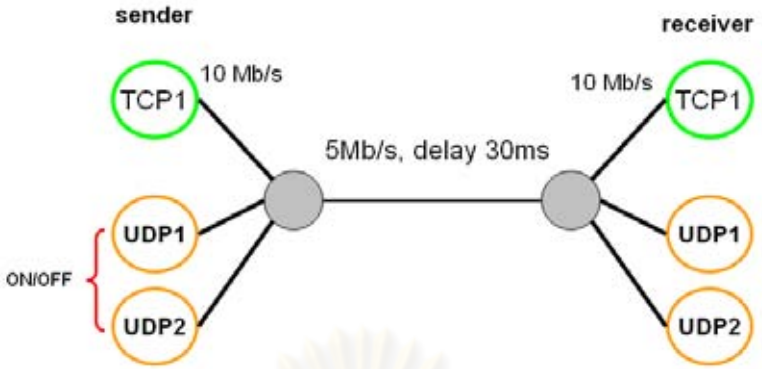
3). คู่การเชื่อมต่อของ TCP ที่ใช้กลไกลการประมาณโดยทำการส่งข้อมูลแบบต่อเนื่อง คู่การเชื่อมต่อของ UDP 2 คู่ที่เป็นแหล่งจ่ายแบบปิดเปิด

4). อัตราการส่งข้อมูลของ UDP มีค่าบิตเรทคงที่เท่ากับ 1 Mb/s ขณะเปิดแหล่งจ่าย

5). กำหนดให้มีการเปิดแหล่งจ่าย UDP คู่แรกที่ช่วงเวลา 50s – 200s , UDP คู่ที่สองที่เวลา 50s-75s และ 125s-175s

6). กำหนดขนาดชุดข้อมูล (packet size) ทั้งหมดมีค่า 400 byte

7). ทำการปรับค่าคาบเวลาของตัวกรอง τ โดยพิจารณาจากค่า RTT ในแต่ละการจำลอง

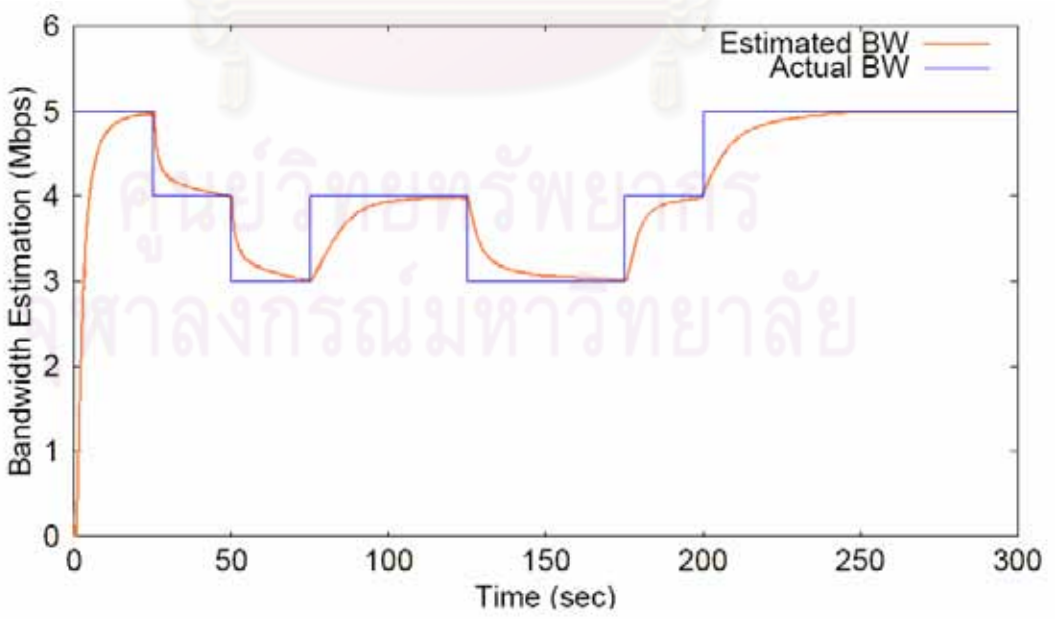


รูปที่ 25 โครงข่ายแบบการจำลองสมรรถนะการประมาณแบนด์วิดท์

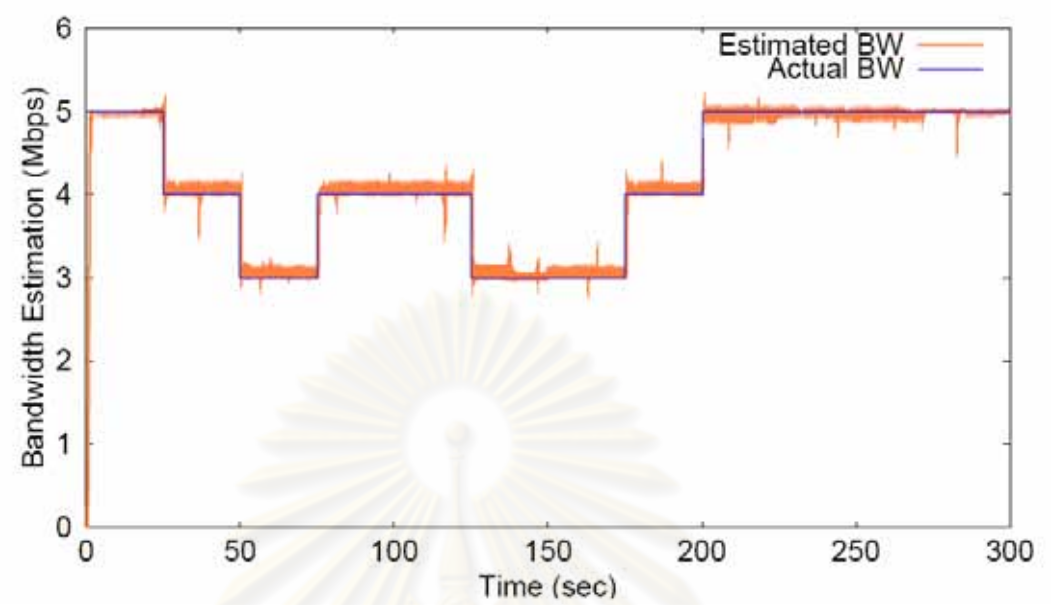
ตารางที่ 1 พารามิเตอร์จำลองสมรรถนะการประมาณแบนด์วิดท์

Connections:	1TCP , 2 UDP(ON/OFF)
UDP bit rate:	1 Mb/s (ON)
Time ON UDP:	UDP1: 50s-200s,UDP2: 50s-75s and 125s-175s
Bottleneck link:	BW 5 Mb/s delay 30 ms
buffer size:	bandwidth delay product
Packet size:	400 bytes
Adjust period:	cut-off frequency of filter

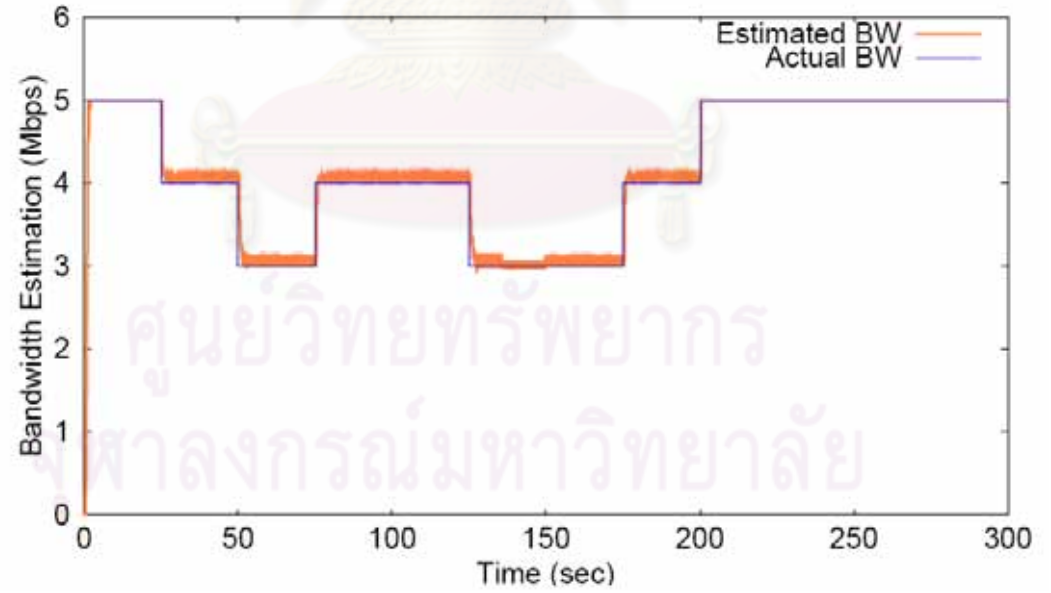
ผลการจำลอง



รูปที่ 26 เปรียบเทียบสมรรถนะการประมาณแบนด์วิดท์กับค่าแบนด์วิดท์จริงในช่องสัญญาณ ($\tau = 500$ ms)



รูปที่ 27 เปรียบเทียบสมรรถนะการประมาณแบนด์วิดท์กับค่าแบนด์วิดท์จริงในช่องสัญญาณ ($\tau = 10$ ms)



รูปที่ 28 เปรียบเทียบสมรรถนะการประมาณแบนด์วิดท์กับค่าแบนด์วิดท์จริงในช่องสัญญาณ ($\tau = 120$ ms)

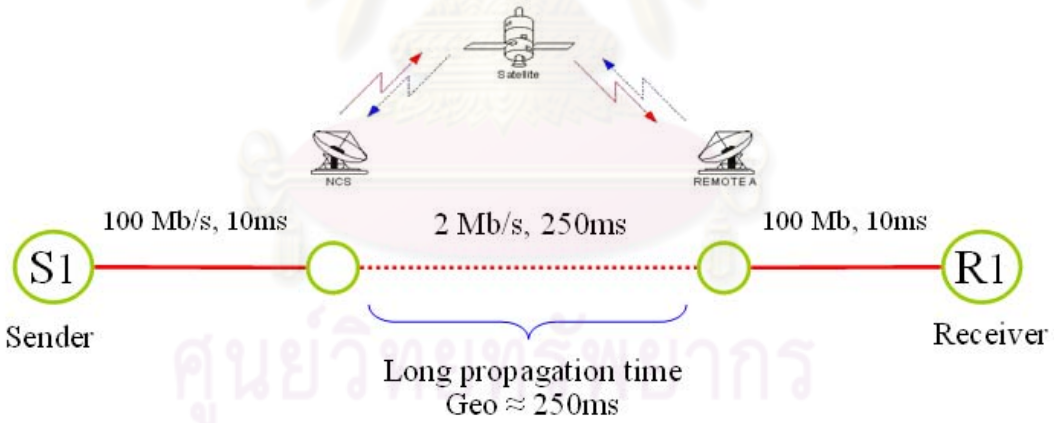
4.2. การปรับขนาดของ CWND, SSTHRESH และ GOODPUT ของ TCP บนโครงข่ายดาวเทียม

การจำลองค่าการปรับขนาดของ cwnd, ssthesh และค่าgoodput ในช่องสัญญาณโดยกลไกลการประมาณแบนด์วิดท์ ด้วยวิธีการนำค่าประมาณที่ได้ในช่องสัญญาณของโครงข่ายมาเป็นเกณฑ์ ในการปรับขนาดความคับคั่งของวินโดว์ (cwnd) ที่ด้านผู้ส่ง เพื่อใช้ในการบรรจุข้อมูลไปยังด้านรับและตั้งจุดอิ่มตัว (ssthresh) ของช่องสัญญาณเพื่อใช้ส่งข้อมูลในลำดับถัดไป โดยในการจำลองนี้จะตั้งค่าพารามิเตอร์โครงข่าย ให้ TCP ทำงานในโครงข่ายจำลองของช่องสัญญาณดาวเทียมและจะนำผลการจำลองมาเปรียบเทียบกับค่าของวิธีการควบคุมการคับคั่งแบบมาตรฐาน โดยทำการตั้งค่าพารามิเตอร์ในการจำลองดังนี้

1). กำหนดคู่การเชื่อมต่อของ TCP ที่ใช้กลไกลการประมาณโดยทำการส่งข้อมูลแบบต่อเนื่อง 1คู่เชื่อมต่อ เปรียบเทียบกับ TCP ที่ใช้วิธีการควบคุมความคับคั่งมาตรฐาน

2).ระหว่างโหนดที่เป็นช่องสัญญาณดาวเทียมมีค่ารอบเวลาการทำงาน (RTT) 500 ms มีความจุช่องสัญญาณการเชื่อมต่อ 2 Mb/s

3).ขนาดของ TCP เซกเมนต์มีค่า 1448 เซกเมนต์ ขนาดบัฟเฟอร์ที่โหนดเชื่อมต่อมีค่าเท่ากับค่าผลการประวิงแบนด์วิดท์ของช่องสัญญาณ

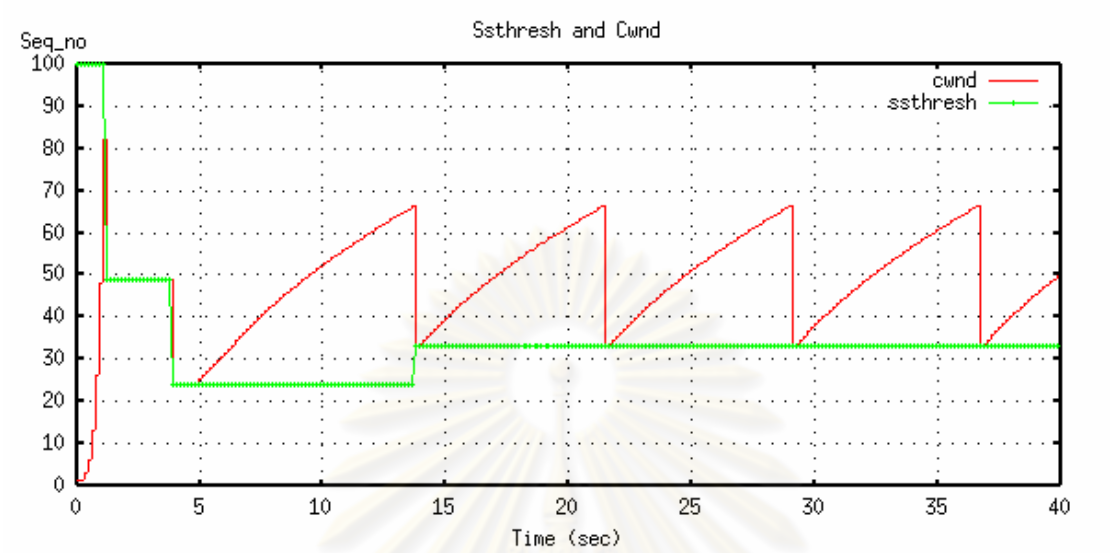


รูปที่ 29 โครงข่ายแบบการจำลองการปรับขนาด cwnd และ ssthresh ในช่องสัญญาณ

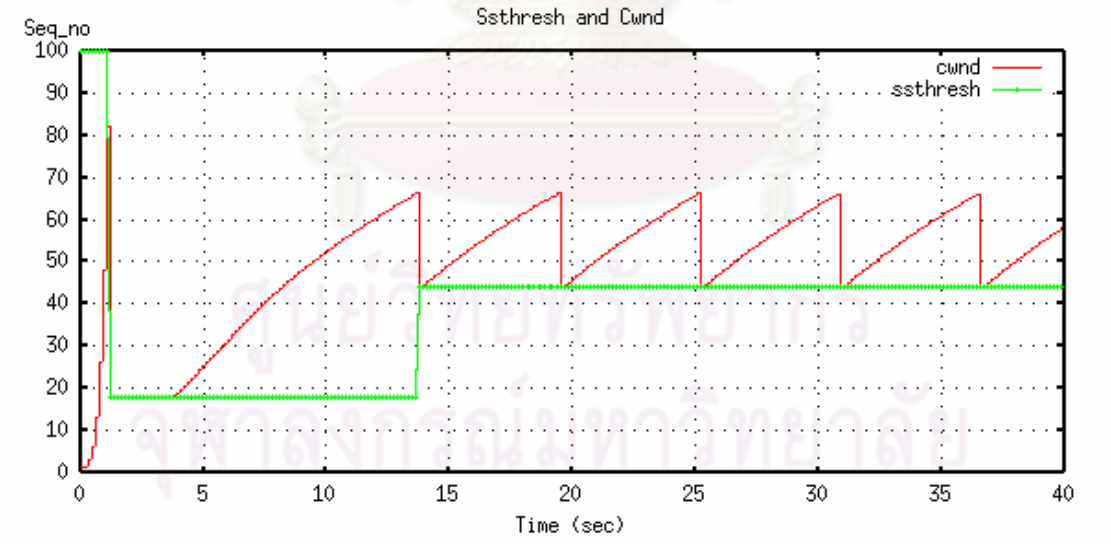
ตารางที่ 2 พารามิเตอร์จำลองการปรับขนาด cwnd และ ssthresh ในช่องสัญญาณ

Connections :	1 TCP (BE compare with standard)
Bottleneck link:	2 Mb/s delay: 250 ms RTT: 500 ms
Buffer size:	bandwidth delay product
Queuing :	First In First Out (FIFO)
TCP segment size:	1,448 bytes
Adjust:	Err. rate (Makov model in wirless link)
Compare:	cwnd, ssthresh and goodput of TCP BE with new-RENO

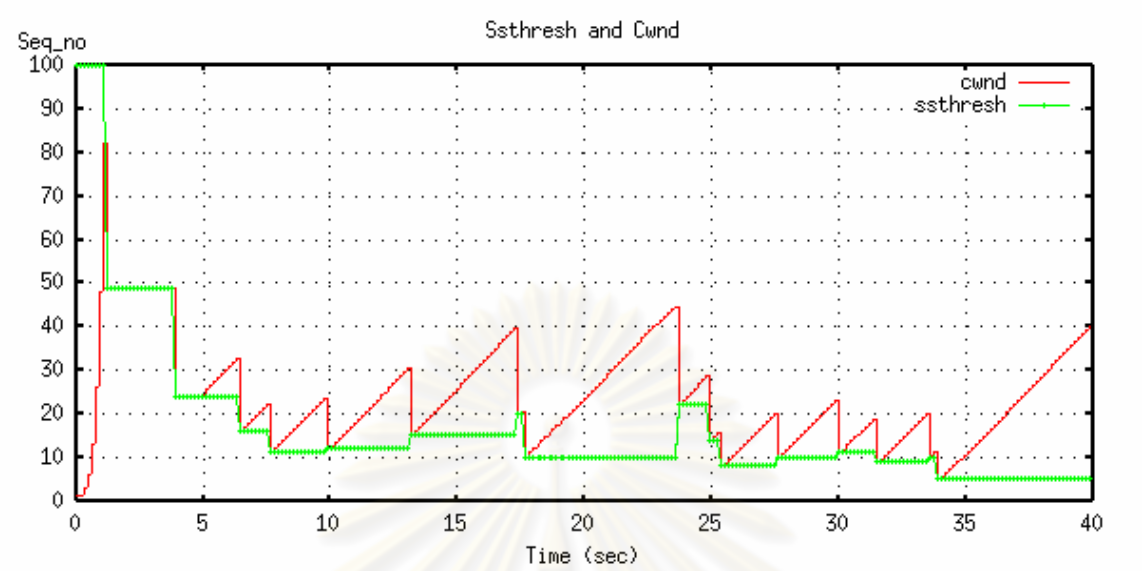
ผลการจำลอง



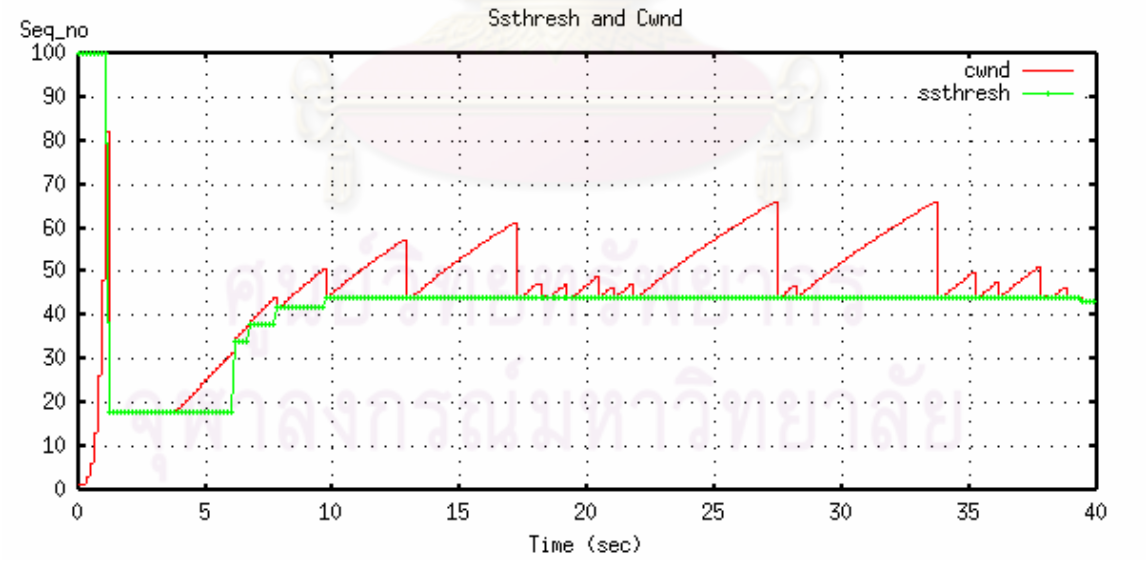
รูปที่ 30 การปรับขนาดของ cwnd และค่า ssthesh ในช่องสัญญาณเมื่อใช้การควบคุมการคับคั่งมาตรฐาน (ช่องสัญญาณไม่มีอัตราบิตผิดพลาด)



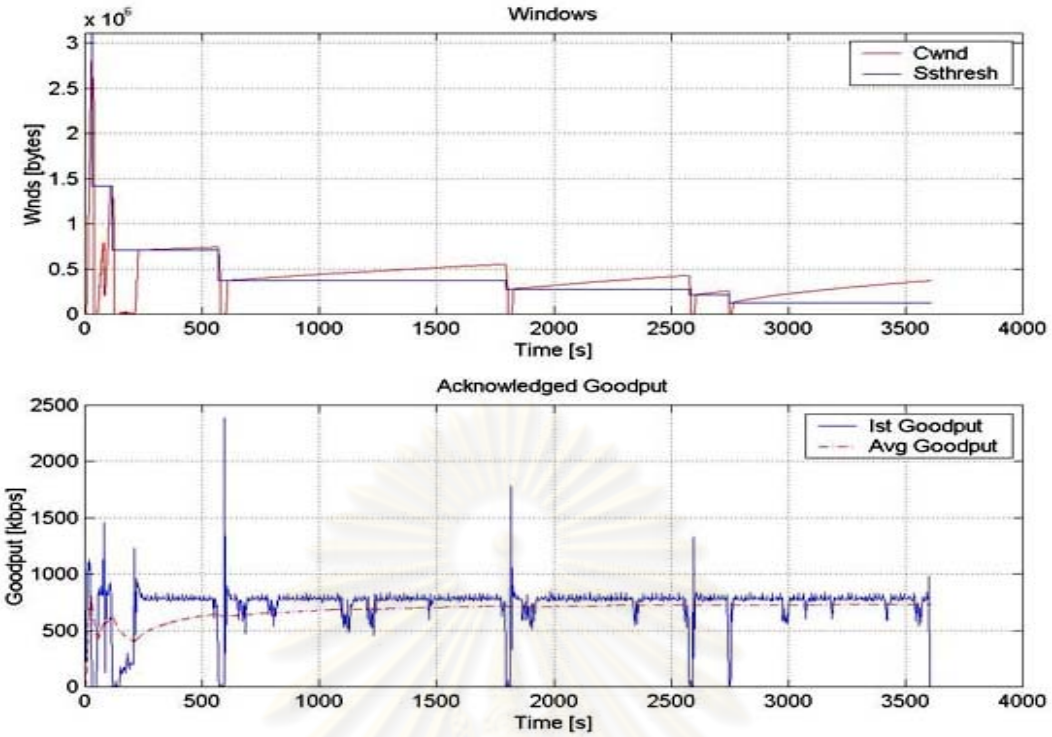
รูปที่ 31 การปรับขนาดของ cwnd และค่า ssthesh ในช่องสัญญาณเมื่อใช้กลไกลการประมาณ (ช่องสัญญาณไม่มีอัตราบิตผิดพลาด)



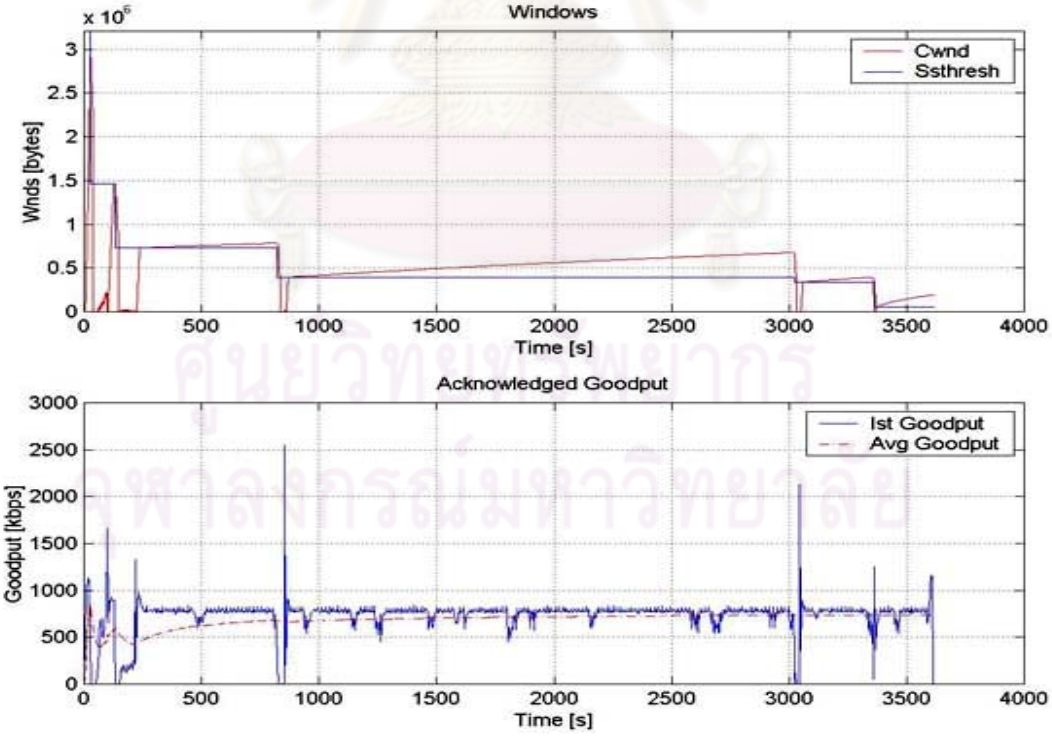
รูปที่ 32 การปรับขนาดของ cwnd และค่า ssthesh ในช่องสัญญาณเมื่อใช้การควบคุมการคับคั่งมาตรฐาน (ช่องสัญญาณอัตราบิตผิดพลาด 0.1%)



รูปที่ 33 การปรับขนาดของ cwnd และค่า ssthesh ในช่องสัญญาณเมื่อใช้กลไกลการประมาณ (ช่องสัญญาณอัตราบิตผิดพลาด 0.1%)



รูปที่ 34 การปรับขนาดของ cwnd, ssthesh และค่า goodput ในช่องสัญญาณเมื่อใช้การควบคุมการคับคั่งมาตรฐาน (ช่องสัญญาณมีอัตราบิตผิดพลาด 0.5%)



รูปที่ 35 การปรับขนาดของ cwnd, ssthesh และค่า goodput ในช่องสัญญาณเมื่อใช้กลไกลการประมาณ (ช่องสัญญาณมีอัตราบิตผิดพลาด0.5%)

4.3. อัตราการส่งผ่านข้อมูลในช่องสัญญาณเมื่อเทียบอัตราผิดพลาดบิตและช่วงเวลาเดินทาง

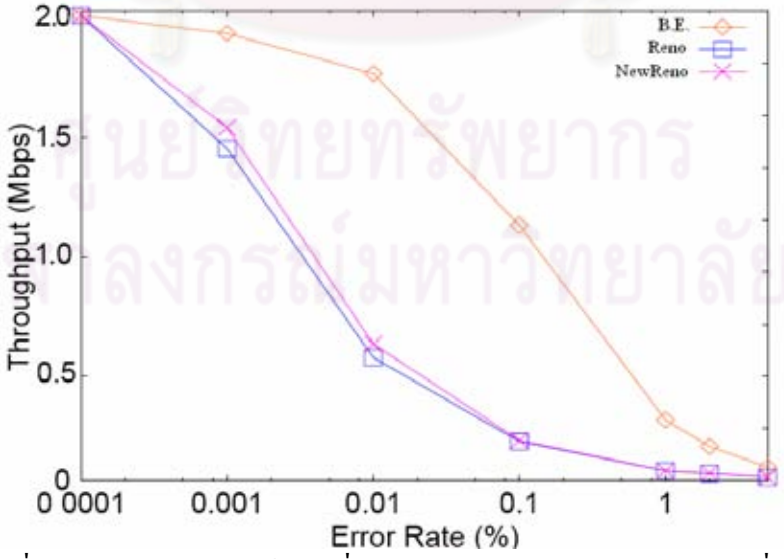
การจำลองการเปรียบเทียบอัตราการส่งผ่านข้อมูลของกลไกลการประมาณกับการควบคุมความคับคั่งมาตรฐานในช่องสัญญาณเมื่อทำการเปลี่ยนแปลงอัตราผิดพลาดบิตและช่วงเวลาเดินทาง โดยโครงข่ายแบบจำลองยังคงใช้เช่นเดียวกับการจำลองแรกหลังจากนั้นทำการปรับค่าอัตราผิดพลาดบิตและช่วงเวลาเดินทางสัญญาณตามลำดับ

ตารางที่ 3 พารามิเตอร์จำลองอัตราการส่งผ่านเมื่อเทียบอัตราผิดพลาดบิตและช่วงเวลาเดินทาง

Connections :	1 TCP (BE compare with standard Reno and NewReno)
Bottleneck link:	2 Mb/s delay: 250 ms RTT: 500 ms
Buffer size:	bandwidth delay product
Queuing :	First In First Out (FIFO)
TCP segment size:	1,448 bytes
Adjust:	1) Err. rate (Makov model in wirless link) 2) Propagation time
Compare:	cwnd, ssthresh and goodput of TCP BE, RENO and new-RENO

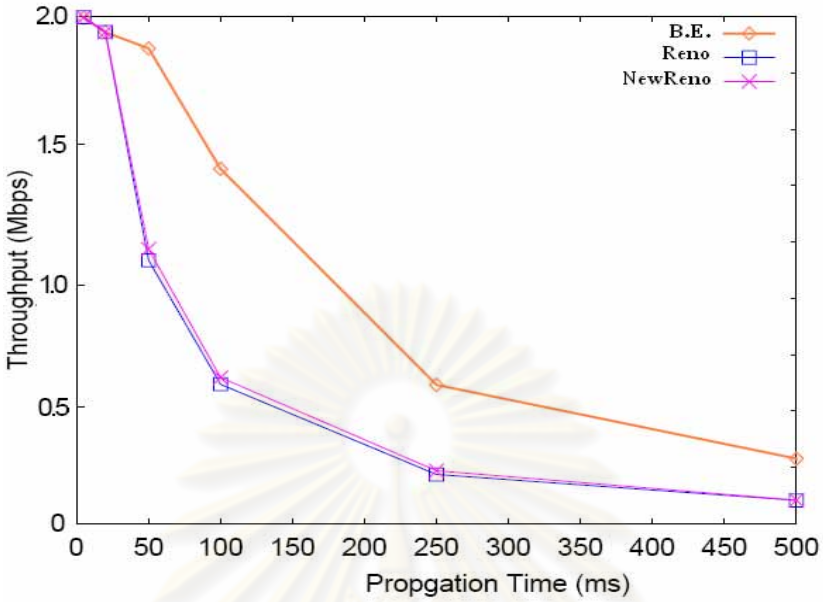
ผลการจำลอง

1). เปรียบเทียบอัตราการส่งผ่านข้อมูลของกลไกลการประมาณกับการควบคุมความคับคั่งเมื่อทำการเพิ่มอัตราผิดพลาดบิต



รูปที่ 36 อัตราการส่งผ่านข้อมูลเมื่ออัตราผิดพลาดบิตในช่องสัญญาณเพิ่มขึ้น

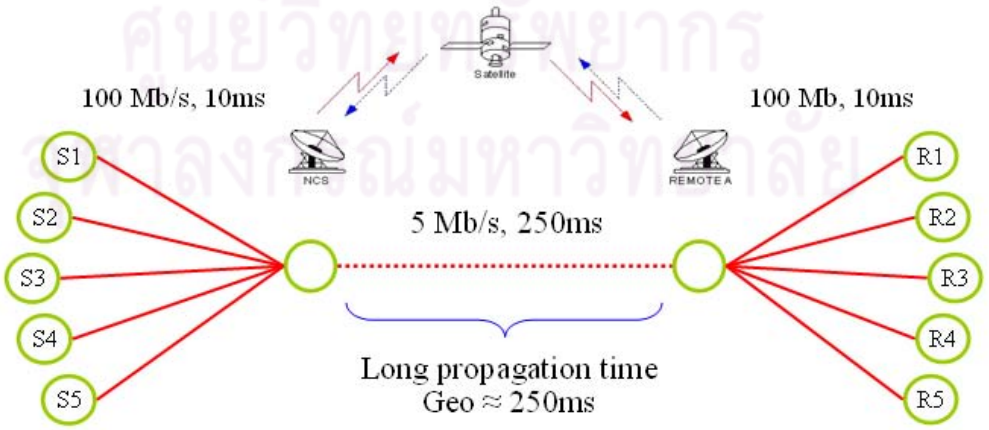
2). เปรียบเทียบอัตราการส่งผ่านข้อมูลของกลไกลการประมาณกับการควบคุมความคับคั่งเมื่อทำการเพิ่มระยะเวลาการเดินทางของสัญญาณ



รูปที่ 37 อัตราการส่งผ่านข้อมูลเมื่อช่วงเวลาการเดินทางในช่องสัญญาณเพิ่มขึ้น

4.4 ความสามารถในการทำงานร่วมกับแบบแผนTCPที่ต่างกัน

การจำลองนี้เป็นการวัดสมรรถนะความสามารถในการทำงานของกลไกลการประมาณแบนด์วิดท์ร่วมกับการควบคุมการคับคั่งมาตรฐานโดยสร้างการเชื่อมต่อของ TCP ทั้งหมด 5 การเชื่อมต่อกำหนดให้เริ่มแรกเป็นการใช้กลไกลการประมาณทั้งหมด จากนั้นเพิ่มจำนวนการเชื่อมต่อของการควบคุมการคับคั่งมาตรฐานเข้ามาแทนที่ทีละการเชื่อมต่อ เป็น 4:1, 3:2, 2:3 และ 1:4 ตามลำดับแล้วทำการเปรียบเทียบอัตราการส่งผ่านข้อมูลโดยเฉลี่ย

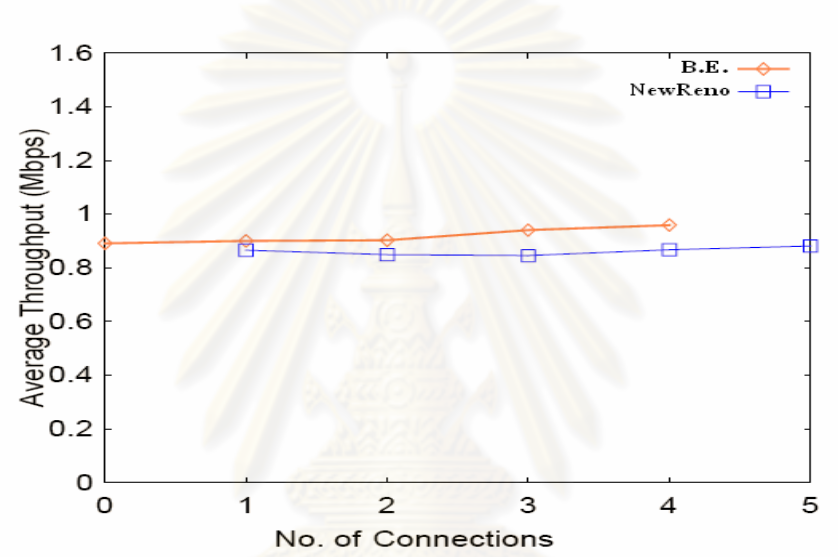


รูปที่ 38 โครงข่ายแบบการจำลองความสามารถในการทำงานร่วมของทีซีพีแบบแผนต่างกัน

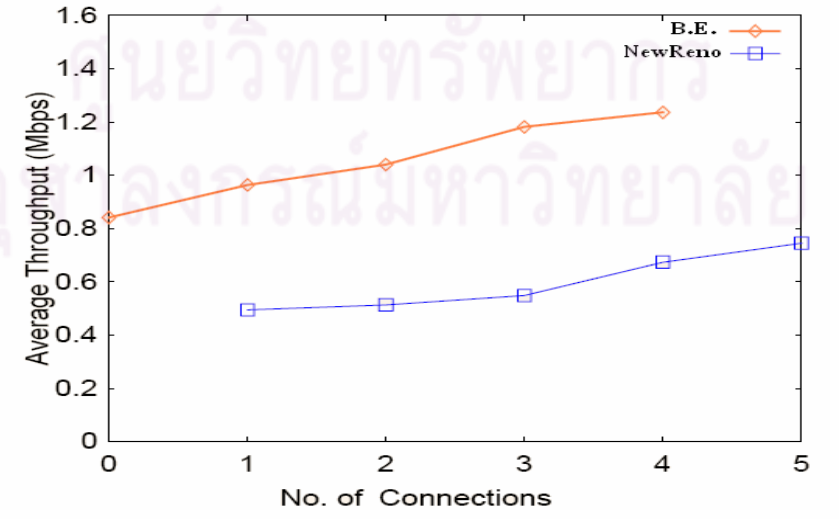
ตารางที่ 4 พารามิเตอร์จำลองความสามารถในการทำงานร่วมของทีซีพี

Connections :	5 TCP (BE join standard)
Bottleneck link:	5 Mb/s delay: 250 ms RTT: 500 ms
Buffer size:	bandwidth delay product
Queuing :	First In First Out (FIFO)
TCP segment size:	1,448 bytes
Adjust	Err. rate (Makov model in wirless link)
Compare	Average throughput (Mbps)

ผลการจำลอง



รูปที่ 39 ค่าเฉลี่ยอัตราการส่งผ่านข้อมูลเมื่อจำนวนการเชื่อมต่อการควบคุมการคับคั่งมาตรฐานในช่องสัญญาณเพิ่มขึ้น (ไม่มีอัตราบิตผิดพลาดในช่องสัญญาณ)



รูปที่ 40 ค่าเฉลี่ยอัตราการส่งผ่านข้อมูลเมื่อจำนวนการเชื่อมต่อการควบคุมการคับคั่งมาตรฐานในช่องสัญญาณเพิ่มขึ้น (ช่องสัญญาณมีอัตราบิตผิดพลาด 1%)

4.5 ผลการวิเคราะห์

ตารางที่ 5 ผลการวิเคราะห์ค่าคาบเวลาของตัวกรอง

ค่าคาบเวลาตัวกรอง (τ)	ค่าประมาณเมื่อเทียบกับค่าแบนด์วิดท์จริง
$\tau >> 3(\text{RTT})$	ใช้ช่วงเวลานานในการปรับค่าเข้าสู่แบนด์วิดท์จริง
$2(\text{RTT}) \le \tau \le 3(\text{RTT})$	มีสมรรถนะใกล้เคียงค่าแบนด์วิดท์จริง,ไม่เกิดออสซิเลท
$\tau < 2(\text{RTT})$	ใกล้เคียงกับค่าแบนด์วิดท์จริงแต่เกิดการค่าออสซิเลทบ่อยครั้ง

จากผลการจำลอง 4.1 ทำการเปรียบเทียบสมรรถนะการประมาณแบนด์วิดท์กับค่าแบนด์วิดท์จริงในช่องสัญญาณ โดยทำการปรับค่าคาบเวลาของตัวกรอง ดังรูปที่ 26, 27 และ28 ตามลำดับแล้วเปรียบเทียบค่าการประมาณแบนด์วิดท์ที่ได้พบว่าค่าคาบเวลาของตัวกรองที่ทำให้สมรรถนะในการประมาณใกล้เคียงกับค่าแบนด์วิดท์จริงจะมีค่าแปรผันกับค่ารอบเวลาการส่งข้อมูลในช่องสัญญาณ (RTT) พบว่าจากรูปที่ 26 เมื่อกำหนดให้ τ มีค่ามากกว่าค่า RTT มากจะทำให้การปรับตัวของค่าการประมาณในช่องสัญญาณที่เปลี่ยนแปลงแบนด์วิทด์เป็นไปได้ช้าต้องใช้เวลานานเพื่อปรับตัวเข้าสู่ค่าแบนด์วิดท์จริง จากรูปที่ 27 เมื่อกำหนดให้ τ มีค่าน้อยกว่าค่า RTT การปรับตัวของค่าการประมาณสามารถปรับตัวเข้าสู่ค่าแบนด์วิทด์จริงอย่างรวดเร็ว แต่จะพบว่ามีการออสซิเลทของค่าการประมาณบ่อยครั้งขณะที่แบนด์วิดท์ในช่องสัญญาณมีค่าคงที่ จากรูปที่ 28 เมื่อกำหนดให้ค่า τ มีค่าเป็น 2 เท่าของรอบเวลาการเดินทางในช่องสัญญาณ จะพบว่าที่คาบเวลานี้ทำให้ค่าการประมาณมีสมรรถนะที่ดีใกล้เคียงค่าแบนด์วิดท์จริงและไม่มีการออสซิเลทเกิดขึ้น และจากการทดลองที่ได้พบว่าค่าคาบเวลา (τ) ที่เหมาะสมมีค่าอยู่ในช่วง $2(RTT) \le \tau \le 3(RTT)$

ตารางที่ 6 ผลการวิเคราะห์และเปรียบเทียบค่า ssthresh และ cwnd ในช่องสัญญาณ

ssthresh และ cwnd	การควบคุมการคับคั่งมาตรฐาน	กลไกลการประมาณแบนด์วิดท์
	ไม่มีอัตราผิดพลาดบิต	
ssthresh	31	42
cwnd	Peak= 66	Peak= 66
	อัตราบิตผิดพลาด 1%	
ssthresh	เปลี่ยนแปลงค่าบ่อยครั้ง ค่าอยู่ระหว่าง 10-20	ปรับขึ้นจนถึง 41 และคงที่
cwnd	Peak= 42, Avr= 21	Peak= 65, Avr= 48

จากผลการจำลอง 4.2 รูปที่ 30 และ 31 การปรับขนาดของ cwnd และ ssthresh ในช่องสัญญาณที่ไม่มีอัตราบิตผิดพลาด ในวิธีการควบคุมการคับคั่งมาตรฐานพบว่าเมื่อเกิดการคับคั่งช่องสัญญาณขึ้น (โดยได้รับสัญญาณตอบรับซ้ำ) กลไกลจะทำการตั้งค่า ssthresh ใหม่เป็น ครึ่งหนึ่งของค่า cwnd เดิมก่อนที่จะเกิดการคับคั่ง จากนั้นทำการปรับ cwnd ใหม่ให้มีค่าเท่ากับ ssthresh เพื่อส่งข้อมูลใหม่ (เสมือนทำการลดค่า cwnd เพื่อส่งข้อมูลใหม่เหลือเพียงครึ่งของค่าเดิม) เมื่อทำเปรียบเทียบกับการใช้วิธีการประมาณที่ใช้การประมาณมาเป็นเกณฑ์ในการปรับตั้งค่า cwnd และ ssthresh พบว่าเมื่อใช้วิธีการประมาณทำให้การปรับตัวของ cwnd และ ssthresh จะมีประสิทธิภาพเพิ่มขึ้น โดยได้ให้ค่า sstresh หรือจุดอิ่มตัวของช่องสัญญาณที่ดีกว่า เช่นกันเมื่อเปรียบเทียบใน รูปที่ 32 และ33 มีอัตราผิดพลาดบิตในช่องสัญญาณเพิ่มเป็น 0.1% ในวิธีการควบคุมการคับคั่งมาตรฐานจะลดค่า cwnd เพื่อส่งข้อมูลใหม่บ่อยครั้ง หากเทียบกับวิธีการประมาณที่ใช้วิธีการกรองอัตราการรับสัญญาณตอบรับ และสะสมค่าประมาณทำให้สามารถกำหนดจุด ssthresh ในช่องสัญญาณและตั้งค่า cwnd เพื่อส่งข้อมูลที่มีประสิทธิภาพมากขึ้น

ตารางที่ 7 เปรียบเทียบอัตราส่งผ่านข้อมูลเมื่อเปลี่ยนแปลงอัตราผิดพลาดบิต

ERR(%)	Troughput(Mbps)		
	Reno	New-Reno	กลไกลประมาณ(B.E.)
0.0001	2.0	2.0	2.0
0.001	1.42	1.57	1.8
0.01	0.61	0.7	1.6
0.1	0.3	0.33	1.24
1	0.07	0.09	0.3

จากผลการจำลอง 4.3 รูปที่ 36 เปรียบเทียบอัตราการส่งผ่านข้อมูลเมื่อเพิ่มอัตราผิดพลาดบิต พบว่าที่อัตราบิตผิดพลาดต่ำ 0.0001% อัตราการส่งผ่านข้อมูลของวิธีการควบคุมการคับคั่งมาตรฐานและวิธีการประมาณยังไม่แตกต่างกันมากนัก เมื่อเพิ่มอัตราผิดพลาดบิตมากขึ้นซึ่งเป็นผลให้ช่องสัญญาณแสดงข้อบ่งชี้ได้ว่าน่าจะมีการคับคั่งเกิดขึ้น อัตราการส่งผ่านของการควบคุมการคับคั่งมาตรฐานได้ลดลงอย่างรวดเร็วซึ่งเป็นผลจากการปรับตัวของ cwnd และ ssthresh ที่กล่าวมา ส่วนในวิธีการประมาณนั้นพบว่า อัตราการส่งผ่านข้อมูลยังมีสมรรถนะที่ดีอยู่ในช่วง 0.0001% - 0.01% หลังจากนั้นก็มีการปรับตัวลดลงอย่างต่อเนื่องแต่ยังอยู่ในเกณฑ์ที่ดีกว่าการควบคุมการคับคั่งมาตรฐาน กรณีนี้แสดงเป็นนัยว่าเมื่อเปรียบเทียบพบว่าการนำวิธีการประมาณมาใช้กับช่องสัญญาณดาวเทียมที่มีอัตราผิดพลาดบิตสูงซึ่งเกิดผลกระทบจากสภาวะอากาศ การถูกรบกวนและการลดทอนของคลื่น

แม่เหล็กไฟฟ้าทำให้ได้อัตราการส่งผ่านที่แปรผันตามอัตราผิดพลาดบิตที่ดีกว่าวิธีการควบคุมการคับคั่งมาตรฐาน ในรูปที่ 37 เมื่อทำการเปลี่ยนแปลงค่าช่วงเวลาการเดินทางในช่องสัญญาณแบบคอขวดระหว่างโหนดพบว่ากลไกลที่แก้ปัญหาการคับคั่งข้อมูลมาตรฐานนั้นทำการปรับลดค่า cwnd ในช่องสัญญาณอย่างรวดเร็วเกินไปโดยที่ไม่มีเกณฑ์อ้างอิงกับความจุของช่องสัญญาณ ขณะที่เพิ่มช่วงเวลาเดินทางสัญญาณเพิ่มขึ้นทำให้เกิดผลกระทบกับอัตราส่งผ่านข้อมูลในช่องสัญญาณทำให้มีค่าลดลงรวดเร็วมากกว่าเมื่อเทียบกับการใช้วิธีการประมาณ

ตารางที่ 8 เปรียบเทียบอัตราส่งผ่านข้อมูลเมื่อเปลี่ยนแปลงค่าเวลาเดินทาง

Propagation time (ms)	Troughput(Mbps)		
	Reno	New-Reno	กลไกลประมาณ(B.E.)
50	1.2	1.3	1.7
150	0.48	0.5	1.24
250	0.23	0.25	0.6
350	0.19	0.20	0.5
500	0.12	0.14	0.27

ตารางที่ 9 เปรียบเทียบสมรรถนะความสามารถในการทำงานร่วมกับแบบแผนTCP ต่างกัน

ไม่มีอัตราผิดพลาดบิต		อัตราผิดพลาดบิต 1%	
Connections TCP NR : B.E.	Avr. Throughput Mbps	Connections TCP NR : B.E.	Avr. Throughput Mbps
0:5	0.87	0:5	0.82
1:4	0.85 : 0.91	1:4	0.50 : 0.94
2:3	0.83 : 0.92	2:3	0.52 : 1.03
3:2	0.82 : 0.94	3:2	0.55 : 1.19
4:1	0.85 : 0.97	4:1	0.63 : 1.22
5:0	0.82	5:0	0.82

จากผลการจำลอง 4.4 เป็นการวัดสมรรถนะความสามารถในการทำงานร่วมกับแบบแผนTCP ที่ต่างกันของกลไกลการประมาณแบนด์วิดท์เมื่อทำงานร่วมกับการควบคุมการคับคั่งมาตรฐาน เนื่องจากสมรรถนะของการทำงาน TCP ที่ดีนั้นจำเป็นต้องมีความสามารถในการทำงานร่วมกับแบบ

แผน TCP อื่นๆด้วยในโครงข่ายรวมที่มี TCP หลากหลายรูปแบบจึงนำการจำลองนี้มาหาสมรรถนะของการทำงานของกลไกลการประมาณแบนด์วิดท์โดยให้มีการทำงานร่วมกันกับวิธีการควบคุมการคับคั่งมาตรฐาน โดยเพิ่มจำนวนของการเชื่อมต่อวิธีการควบคุมการคับคั่งมาตรฐานเข้ามาแทนที่เพื่อเปรียบเทียบอัตราส่งผ่านเฉลี่ยของการเชื่อมต่อ รูปที่ 39 พบว่าในกรณีที่ไม่มีอัตราผิดพลาดบิต การทำงานทีซีพีที่ใช้ของกลไกลการประมาณกับการควบคุมการคับคั่งมาตรฐานเป็นไปได้ดี โดยดูจากอัตราการส่งผ่านเฉลี่ยในการเชื่อมต่อซึ่งได้ค่าไม่ต่างกันมากนัก รูปที่ 40 ในช่องสัญญาณที่มีอัตราบิตผิดพลาดสูง แสดงการทำงานร่วมกันของกลไกล TCP ทั้งสองพบว่าอัตราการส่งข้อมูลเฉลี่ยของทั้งคู่มีค่าแตกต่างกัน จึงไม่เหมาะนักหากจะนำวิธีการประมาณไปทำงานร่วมกับทีซีพีที่ใช้การควบคุมการคับคั่งมาตรฐาน เนื่องจากคุณลักษณะของการประมาณที่จะทำการปรับค่าจุดอิ่มตัว(ssthresh) ให้ได้ค่าที่สูงสุดจากการกรองสัญญาณตอบรับ ทำให้ไปครอบครองความจุของช่องสัญญาณโดยส่วนใหญ่เอาไว้ จึงทำให้การทำงานของการควบคุมการคับคั่งมาตรฐานมีอัตราการส่งผ่านที่ไม่ดีเท่าที่ควร จากผลจำลองจึงไม่ควรสร้างการเชื่อมต่อของ TCP ทั้งสองแบบหลายคู่เชื่อมต่อให้ทำงานร่วมในช่องสัญญาณเดียวกัน หากจำเป็นต้องใช้การจำกัดแบนด์วิดท์ของแต่ละการเชื่อมต่อให้เหมาะสม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

สรุปผลการวิจัย และข้อเสนอแนะ

5.1 สรุปผลการวิจัย

วิทยานิพนธ์ฉบับนี้นำเสนอวิธีการประมาณแบนด์วิดท์แบบปรับตัวได้ของTCPเพื่อควบคุมความคับคั่งข้อมูลในช่องสัญญาณอินเทอร์เน็ตผ่านดาวเทียม โดยได้นำแนวทางกลไกลประมาณโดยใช้ประโยชน์จากสัญญาณการตอบรับ(ACK) ที่เป็นตัวบ่งชี้อัตราปริมาณข้อมูลซึ่งส่งผ่านสำเร็จและได้รับโดยปลายทาง มาทำการวัดเปรียบเทียบค่าเฉลี่ยร่วมกับข้อมูลที่ส่งไปปลายทางและประมาณให้เป็นเกณฑ์ความจุช่องสัญญาณ เมื่อเกิดการคับคั่งในช่องสัญญาณขึ้น จะนำค่าเกณฑ์ประมาณที่ได้มาทำการปรับตั้งขนาดความคับคั่งวินโดว์และจุดอิ่มตัวช่องสัญญาณเพื่อส่งข้อมูลใหม่ จากผลการจำลองเมื่อกำหนดคุณลักษณะให้ช่องสัญญาณเกิดความคับคั่ง แล้วนำผลการทำงานมาเปรียบเทียบสมรรถนะกับวิธีการควบคุมความคับคั่งมาตรฐาน จากผลการจำลองแสดงให้เห็นว่าการใช้ค่าประมาณมาเป็นเกณฑ์ในการปรับตั้งค่า cwnd และ ssthresh เพื่อส่งชุดข้อมูลใหม่ ทำให้อัตราการส่งผ่านข้อมูลอยู่ในเกณฑ์ดีกว่าวิธีมาตรฐาน และช่วยแก้ปัญหาการลดขนาดวินโดว์ของผู้ส่งใช้ในการส่งข้อมูลและจุดอิ่มตัวลงอย่างรวดเร็ว ในการหัวข้อจำลองการวัดสมรรถนะการทำงานร่วมกับแบบแผน TCP ที่ต่างกัน โดยได้กำหนดการเชื่อมต่อทีซีพีที่ใช้กลไกลการประมาณแบนด์วิดท์ให้มีการใช้งานช่องสัญญาณร่วมกับกลไกลควบคุมมาตรฐาน จากผลการจำลองพบว่ายังไม่เหมาะนักหากจะนำวิธีการประมาณไปทำงานร่วมกับทีซีพีมาตรฐานในช่องสัญญาณที่มีหลายการเชื่อมต่อร่วมกัน เนื่องจากคุณลักษณะของกลไกลที่จะปรับค่าจุดอิ่มตัว(ssthresh) ตามเกณฑ์ประมาณที่ได้ เมื่อเพิ่มคู่การเชื่อมต่อมากขึ้นกลไกลประมาณจะไปครอบครองความจุของช่องสัญญาณโดยส่วนใหญ่ไว้ ทำให้การรักษาดุลยภาพของข้อมูลระหว่างแบบแผนทีซีพีจึงมีค่าไม่ดีเท่าที่ควร อย่างไรก็ตามหากช่องสัญญาณมีการทำงานของ TCP เพียงแบบแผนเดียว การใช้กลไกลการประมาณนี้เป็นแนวทางหนึ่งที่ช่วยในการปรับปรุงสมรรถนะของ TCP ในรูปแบบ end-to-end ที่ช่วยแก้ปัญหาความคับคั่งของวินโดว์ การลดจุดอิ่มตัวลงอย่างรวดเร็วเนื่องจากคุณลักษณะของโครงข่ายได้เป็นอย่างดี

5.2 ข้อเสนอแนะ

วิธีการประมาณแบนด์วิดท์แบบปรับตัวได้ของ TCP ที่นำเสนอวิทยานิพนธ์นี้จัดเป็นแนวทางนึงในการควบคุมความคับคั่งข้อมูลเนื่องจากคุณลักษณะช่องสัญญาณดาวเทียม ที่ระดับชั้นทรานสปอร์ท (transport layer) ในรูปแบบปลายทางไปยังปลายทาง (end-to-end) โดยทำการจำลองโครงข่ายด้วยรูปแบบพื้นฐานของระบบดาวเทียมอินเทอร์เน็ต หากจะทำการพิจารณาประสิทธิภาพการทำงานให้

เหมาะสมกับลักษณะการใช้งานในโครงข่ายจริงมากขึ้น จำเป็นต้องนำปัจจัยของการทำงานระดับชั้น (layer)รองลงมาเข้ามาพิจารณาในการจำลองด้วย เช่น ระบบประมวลผล การจัดลำดับคิว การเข้ารหัส การมอดูเลทสัญญาณ เพื่อให้เกิดประสิทธิภาพในการทำงานร่วมกับโครงข่ายอย่างสมบูรณ์

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

[1] D.E. Comer. Internetworking with TCP/IP. Vol. 1.Principles, Protocols and Architecture. 2nd ed. Prentice Hall,1991.

[2] V. Jacobson. Congestion avoidance and control. Proceedings ACM SIGCOMM, (Aug 1988): 314-329.

[3] D. Chiu and R. Jain. Analysis of the increase/decrease algorithms for congestion avoidance in computer networks. J. Computer Networks 17,1 (June 1989) : 1-14.

[4] G. Nasir and D. Sudhir. TCP/IP Enhancements for Satellite Networks. IEEE Communication Magazine (July 1999) : 64-72.

[5] M. West and S. McCann. Improving TCP Performance over long-delay and error-prone links. IEEE seminar on Satellite Services and the Internet (2000) : 1-9.

[6] C. Patridge and T. Shepard. TCP Performance over Satellite Links. IEEE Networks 11,5 (1997) :44-49.

[7] K. Ratnam and I. Matta. WTCP: An Efficient Mechanism for Improving TCP Performance over Wireless Links. IEEE Symposium on Computer and Communications(ISCC). (1998) : 74-78

[8] I. Akyidiz, G. Morabito, and S. Palazo. TCP-Peach A New Congestion Control Scheme for Satellite IP Networks. IEEE/ACM Transactions of Networking 9,3 (2001) : 307-321

[9] R. Jain, K. K. Ramakrishnan, and D.M. Chiu. Congestion Avoidance in Computer Networks with a Connectionless Network Layer. Technical Report DEC-TR-506 August 1987.

[10] R. Jain. A delay-based approach for congestion avoidance in interconnected heterogeneous computer network. Computer Communications Review (November 1989) : 56-71.

[11] R. Jain. A timeout-based congestion control scheme for window flow-controlled networks. IEEE Journal on Selected Areas in Communications (August 1988) : 314-322.

[12] W. Stevens. TCP/IP Illustrasted,Volume 1. Addison Wesley,1994.

[13] M. Allman, V.Paxson and W. Stevens. TCP congestion control. IETF RFC 2581 (April 1999).

[14] V. Jacobson. Berkeley tcp evolution from 4.3-tahoe to 4.3-reno. In Proceedings of the Eighteenth Internet Engineering Task Force(University of British Columbia,Vancouver) (September 1990) : 365.

[15] M. Mathis, J. Mahdavi, S. Floyd and A. Romanow. TCP Selective Acknowledgement Options. IETF RFC 2018 (October 1996).

[16] L.S. Brakmo, S.W. O'Malley and L.L. Peterson. TCP Vegas:new technique for congestion detection and avoidance. In Proceedings of ACM SIGCOMM, London, UK (October 1994) : 24-35

[17] S. Horan and R. Wang. Internet-type protocol testing in a simulated small satellite environment. In Proceeding of IEEE Aerospace Conference (2001) : 977-989

[18] I.F. Akyildiz, G. Morabito, and S.Palazzo. TCP-Peach: A new congestion control scheme for satellite IP network. IEEE/ACM Transactions Networking 9, (June 2001) : 321-337

[19] E.W. Fulp and D.S. Reeves. Dynamic Bandwidth Allocation Techniques. Center for Advanced Computing and Communications August 1997

[20] Z. Zhang and Z. Qui. A Novel Approach For Real-time Equivalent Bandwidth Estimation. International Conference on Communication Technology Proceeding 1 (2000): 1470-1474

[21] รศ.ดร.ประสิทธิ์ ทีฆพุฒิ. การสื่อสารดาวเทียม. พิมพ์ครั้งที่ 4. วิศวกรรมสถานแห่งประเทศไทยในพระบรมราชูปถัมภ์. กรุงเทพมหานคร : โรงพิมพ์จุฬาลงกรณ์มหาวิทยาลัย, 2540.

[22] A. Jamalipour, M. Marchese, S. Cruickshank, J.Neale and N. Verma. Guest Editorial Broadband IP Networks via Satellites-Part I. IEEE Journal on selected areas in communications 22 (February 2004) : 213-217.

[23] M. Allman, S. Floyd and C.Partridge. Increasing TCP's Initial Window. Network Working Group RFC 2414 1998

[24] W.stevens. TCP slow start, congestion avoidance, fast retransmit and fast recovery algorithms. IETF RFC 2001 1997

[25] M. Gerla, BKF. Ng, MY.Sanadidi, M. Valla and R. Wang. TCP Westwood with adaptive bandwidth estimation to improve efficiency/friendliness tradeoffs. Journal of Computer Communication 2004

[26] L. Kalampoukas, A Varma and KK. Ramakrishnan. Explicit Window Adaptation: A method Enhance TCP Performance. IEEE/ACM Transaction on Networking 10,3 (June 2002) : 338-350.

[27] S. Q. Li and C. Hwang. Link Capacity Allocation and Network Control by Filtered Input Rate in High speed Networks. IEEE/ACM Transaction on Networking 3,1 (February 1995) :10-25.

[28] K. B. Janiszowski. A Modification and the Tustin Approximation. IEEE Transactions on Automatic Control 38,8 (August 1993): 1313-1316.

[29] R. Goyal, R. Jain, S. Kalyanaraman, and S. Fahmy. UBR+:Improving performance of TCP over ATM-UBR service by fair buffer management. Proceedings of the IEEE International Communications Conferrence 1997.

[30] Network Simulator (NS). http://www.isi.edu/nsnam/ns/ns-documentation.html. UC Berkeley, LBL, USC/ISI and Xerox PARC : The VINT Project

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก

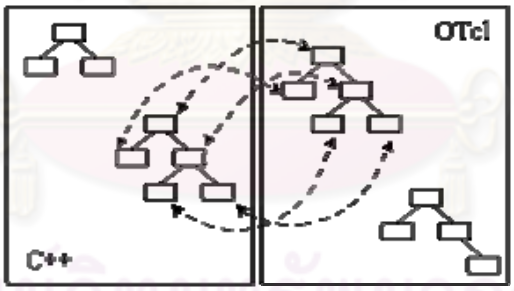
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

โปรแกรมในการจำลองโครงข่าย

1. โปรแกรม NS

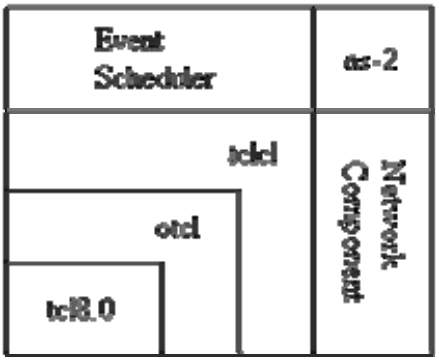
โปรแกรม NS (Network Simulator) เป็นโปรแกรมที่ใช้สำหรับการจำลองระบบเครือข่าย โดยตัวโปรแกรมสามารถทำงานได้บนหลายแพลตฟอร์ม ทั้งในระบบ Unix (FreeBSD, Linux, SunOS, Solaris) และบนระบบปฏิบัติการ Windows เป็นที่นิยมใช้กันอย่างมากในกลุ่มผู้พัฒนาระบบเครือข่าย โดยจะนำโปรแกรมนี้มาจำลองเครือข่ายในลักษณะต่างๆ แล้ววิเคราะห์ผล เพื่อนำไปปรับปรุงและพัฒนาต่อไป โปรแกรม NS จะสนับสนุนการจำลองในหลายรูปแบบ เช่น โปรโตคอล TCP, UDP การใช้ในลักษณะงานประยุกต์ เช่น FTP, Telnet, Web ระบบการจัดคิว เช่น DropTail, RED, CBQ รวมทั้งการใช้งานในแบบมัลติคาสต์ ทั้งบนระบบเครือข่ายทั่วไป (Wired Network) และในระบบเครือข่ายไร้สาย (Wireless Network)

โปรแกรม NS ถูกพัฒนาขึ้นโดยใช้ภาษา C++ และ ภาษา OTcl (เป็นภาษา Tcl ที่มีลักษณะการเขียนโปรแกรมเป็นแบบเชิงวัตถุ) โดยจะใช้ภาษา C++ ในการจัดการกับตัวข้อมูล ส่วน OTcl ใช้ในการควบคุมกระบวนการ โดยสาเหตุที่ใช้ 2 ภาษา ก็เพราะว่า ภาษา OTcl สามารถจัดการกับการควบคุมได้ง่ายและรวดเร็วกว่า ส่วนภาษา C++ สามารถจัดการกับตัวข้อมูลได้ดีกว่า ดังนั้นในการพัฒนาโปรแกรมจึงใช้การเชื่อมโยงระหว่าง 2 โปรแกรม ซึ่งจะใช้ฟังก์ชันการทำงานที่เรียกว่า OTcl linkage ดังรูปที่ 1



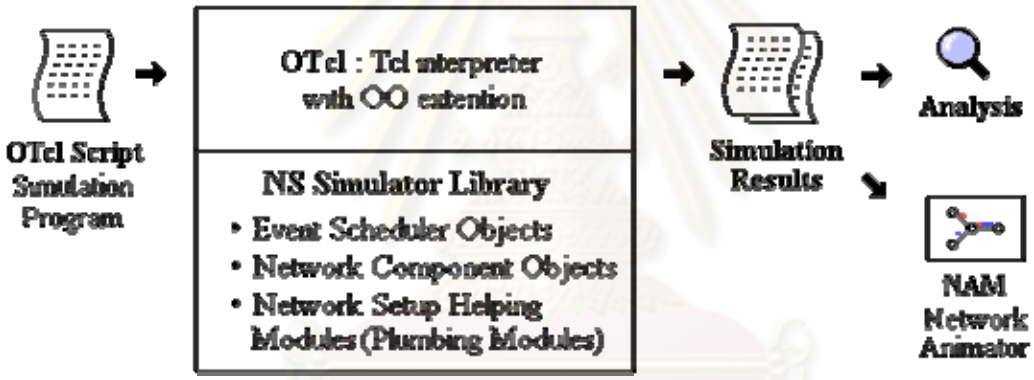
รูปที่ 1 ความสัมพันธ์ระหว่างภาษา C++ กับ OTcl

โดยผู้ที่จะทำการพัฒนารูปแบบการจำลองในรูปแบบใหม่ ก็จะต้องทำการพัฒนาในส่วนนี้ แต่สำหรับการใช้โปรแกรม NS เพื่อจำลองระบบเครือข่ายนั้น จะใช้ภาษา Tcl ในการเรียกฟังก์ชันต่างๆ ที่ได้มีผู้พัฒนามาแล้ว ก็คือเรียกใช้องค์ประกอบในส่วนต่างๆ (ซึ่งเรียกใช้การทำงานของคลาสต่างๆอีกทีหนึ่ง) นำมาทำงานร่วมกัน เพื่อให้เกิดเป็นระบบเครือข่ายที่ต้องการ ซึ่งจะเรียกรูปแบบการทำงานนี้ว่า Event Scheduler สำหรับโครงสร้างภายในตัว NS จะแสดงได้ดังรูปที่ 2



รูปที่ 2 โครงสร้างภายในของโปรแกรม NS

ในการใช้งานโปรแกรม NS เพื่อวิเคราะห์การทำงานนั้น จะมีลำดับขั้นตอนการทำงานดังรูปที่ 3 โดยเมื่อเราทำการเขียนคำสั่งกำหนดรูปแบบเครือข่ายเรียบร้อยแล้ว ก็ทำการเรียกใช้โปรแกรม จากนั้นเมื่อได้ผลการทำงานออกมาแล้ว ก็นำผลที่ได้ไปวิเคราะห์ โดยอาจจะใช้โปรแกรมจำลองเส้นทางการเดินทางของข้อมูล ที่ชื่อว่า NAM เพื่อใช้ช่วยในการวิเคราะห์การทำงานได้ดีขึ้น



รูปที่ 3 ลำดับขั้นตอนการทำงานในการวิเคราะห์ระบบเครือข่าย

สำหรับรูปแบบคำสั่งที่ใช้ในโปรแกรม NS จะมีรูปแบบทั่วไป ดังนี้

ns < tcl-script >

ในการใช้งานโปรแกรม NS ในการจำลองการทำงานต่างๆ นั้น จะใช้คำสั่ง Tcl ในการกำหนดค่าต่างๆ โดยจะเรียกใช้รูปแบบการทำงานที่ได้กำหนดไว้แล้ว ซึ่งในรูปแบบต่างๆ เหล่านี้จะถูกเรียกใช้จากคลาส Simulator ซึ่งเป็นคลาสหลักของโปรแกรม NS

รูปแบบคำสั่งที่ใช้งานเป็นส่วนหลักของโปรแกรม

- สร้างตัวตารางงาน

set ns [new Simulator]

- ลำดับเหตุการณ์

$ns at <time> <event>

โดย <time> เป็นเวลาที่กำหนดในลำดับเหตุการณ์

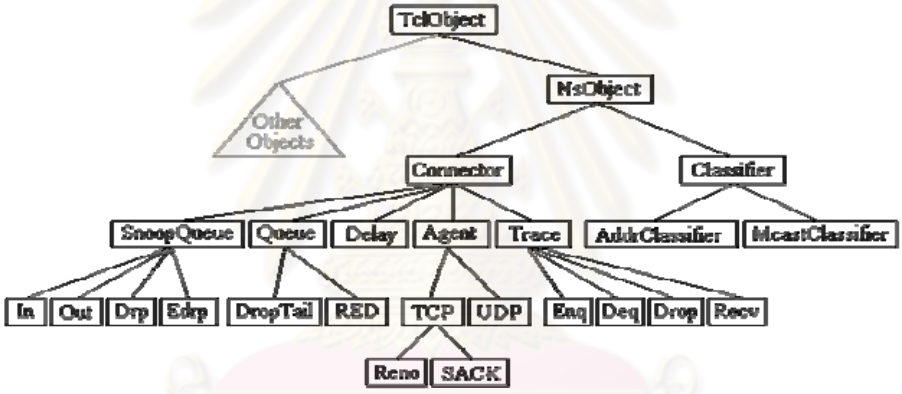
<event> เป็นเหตุการณ์ที่จะกระทำ

- เริ่มต้นตัวตารางงาน

$ns run

1.1 องค์ประกอบของเครือข่ายใน NS

ในเครือข่ายหนึ่งๆ จะประกอบไปด้วยส่วนต่างๆ ที่เชื่อมต่อกันในการทำงาน ดังเช่น จุดเชื่อมต่อ (node) ของจุดหนึ่ง จะถูกเชื่อมต่อด้วยเส้นทางการเชื่อมต่อ (link) ไปยังจุดเชื่อมต่ออีกจุดหนึ่ง ซึ่งในแต่ละส่วนจะประกอบไปด้วยองค์ประกอบอีกมากมาย ที่ถูกกำหนดให้ทำงานเป็นส่วนๆ ดังนั้น ในการพิจารณาเครือข่ายอย่างละเอียด จึงควรพิจารณาถึงองค์ประกอบในของแต่ละส่วน โดยหัวข้อนี้แสดงองค์ประกอบของเครือข่ายใน NS โดยจะแบ่งเป็นส่วนๆ ดังรูปที่ 4



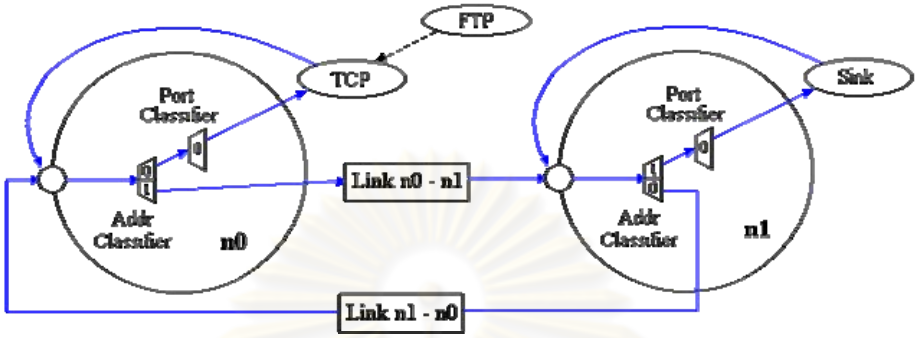
รูปที่ 4 ระดับชั้นขององค์ประกอบในโปรแกรม NS

ในรูปแสดงนี้เป็นเพียงส่วนหนึ่ง โดยจะพิจารณาในส่วนที่เกี่ยวข้องนี้ โดยจะมี TclObject เป็นคลาสแม่ของคลาสทั้งหมด โดยมี NSObject เป็นคลาสลูก ซึ่งจะเป็นคลาสแม่ขององค์ประกอบเครือข่ายทั้งหมด ถ้าแบ่งองค์ประกอบต่างๆ นี้ ตามจำนวนของเส้นทางของผลลัพธ์ที่เป็นไปได้ จะสามารถแบ่งออกเป็น 2 ประเภท คือ แบบ Connector และแบบ Classifier ก็คือองค์ประกอบพื้นฐานต่างๆ ที่มีเส้นทางของผลลัพธ์ในทางเดียว เช่น Enq, Deq, Drop, Recv ที่ใช้ในการตามข้อมูล (trace) จะจัดอยู่ในประเภท Connector ส่วนประเภท Classifier จะเป็นองค์ประกอบประเภทที่ใช้ในการเลือกเส้นทาง (switching) เมื่อนำทั้งสองประเภทมาประกอบรวมกัน จะได้เป็นองค์ประกอบต่างๆ

1.1.1 จุดเชื่อมต่อ (Node)

จุดเชื่อมต่อจะประกอบไปด้วย จุดรับข้อมูล (node entry) และองค์ประกอบประเภท Classifier โดยโหนดสามารถแบ่งได้เป็น 2 ประเภท คือ Unicast node และ Multicast node

Unicast node ในส่วนของ Classifier จะประกอบด้วย Addr Classifier ซึ่งใช้สำหรับการค้นหาเส้นทางในแบบ unicast (Unicast Routing) ซึ่งจะเชื่อมต่อกับโหนดอื่นๆ ต่อโดยเชื่อมต่อผ่าน link และอีกส่วนหนึ่งเรียกว่า Port Classifier ซึ่งใช้เชื่อมต่อกับการทำงานในระดับสูง ดังเช่น เชื่อมต่อกับทีซีพี เป็นต้น ซึ่งจะเรียกว่า ตัวแทน (Agent) ดังแสดงได้ดังรูปที่ 5



รูปที่ 5 โครงสร้างของ Unicast Node และ Multicast Node

Multicast node ในส่วนของ Classifier จะแตกต่างกับแบบ Unicast node โดยจะประกอบไปด้วย Classifier แบบUnicast node และ Multicast classifier ซึ่งทำให้สามารถทำงานได้ทั้งสองแบบ โดยจะมีตัวที่ทำหน้าที่แยกแพ็คเกจว่าเป็นแบบใด จะเรียกส่วนนี้ว่า switch สำหรับ Multicast Classifier จะใช้สำหรับการค้นหาเส้นทางแบบ multicast routing โดยจะมีตัวทำหน้าที่กระจายข้อมูลว่าไปให้โหนดต่างๆ ที่เชื่อมต่อกันอยู่ จะเรียกส่วนนี้ว่า Replicator สำหรับโครงสร้างแสดงได้ดังรูปที่ 5

สำหรับการใช้งาน Multicast Routing นั้น จะพิจารณาบิตสูงสุดของ Address ซึ่งเป็นตัวบอกว่าเป็นแบบ Unicast หรือแบบ Multicast โดยถ้ามีค่าเป็น 0 จะเป็น Unicast Address ถ้าเป็น 1 จะเป็น Multicast Address

ในการกำหนดค่านั้น โปรแกรม NS จะกำหนดค่าเริ่มต้นแบบ Unicast node ดังนั้นถ้าจะใช้แบบ Multicast node จะต้องกำหนดค่าเพิ่มเติม เพื่อบอกให้โปรแกรมรู้ก่อน โดยสรุปรูปแบบคำสั่งในการกำหนดรูปแบบ ดังนี้

- สร้างจุดเชื่อมต่อ (node)

  $ns node

- การเปลี่ยนแปลงค่าพารามิเตอร์ของจุดเชื่อมต่อ

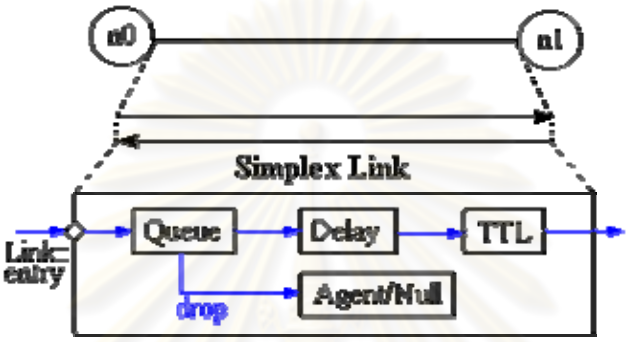
  $ns node-config - <config-parameter> <optional>

1.1.2 เส้นทางการเชื่อมต่อ (Link)

เส้นทางการเชื่อมต่อ หรือ link เป็นส่วนประกอบสำคัญส่วนหนึ่งในเครือข่าย เป็นส่วนที่เชื่อมจุดต่างๆ ในเครือข่ายเข้าด้วยกัน ดังที่เห็นกันได้ก็คือ เป็นส่วนเชื่อมต่อระหว่างโหนด เพื่อทำหน้าที่ส่งผ่านแพ็คเกจข้อมูลระหว่างกัน

สำหรับใน NS นั้น ได้สนับสนุนการใช้งานหลายรูปแบบ เช่น แบบ multi-access LAN รวมไปถึงการใช้งาน Wireless หรือ broadcast media โดยเราจะพิจารณาในรูปแบบ จุดต่อจุด (point-to-point) ซึ่งถือได้ว่าเป็นลักษณะพื้นฐานของรูปแบบต่างๆ

ในลักษณะการเชื่อมต่อทั่วไปนั้น จะเป็นในลักษณะการเชื่อมต่อแบบ 2 ทิศทาง (duplex-link) เมื่อพิจารณาโครงสร้างแล้ว จะมีลักษณะเหมือนกัน ดังนั้นแล้วเราจะพิจารณาในรูปแบบทิศทางเดียว ดังรูปที่ 6



รูปที่ 6 เส้นทางการเชื่อมต่อ (link)

จากรูปภายในนั้นจะประกอบไปด้วยองค์ประกอบ 5 ส่วน คือ

1. Link entry เป็นจุดเข้าของ link ซึ่งเป็นส่วนที่อยู่ด้านหน้าสุดของ link

2. Queue เป็นส่วนหลัก link มีหน้าที่จัดแพ็คเกจข้อมูล โดยทั่วไปจะมีเพียง 1 ส่วน เท่านั้นใน 1 link

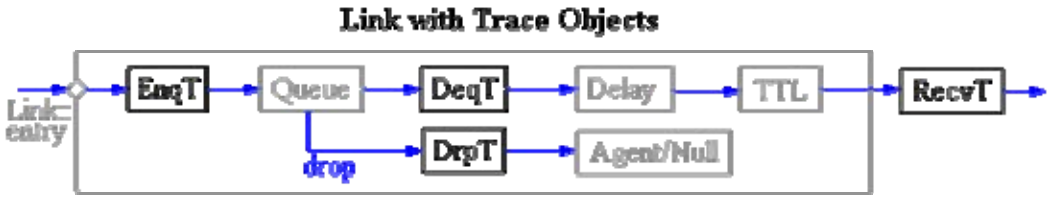
3. Delay หรือ link delay เป็นส่วนที่ทำหน้าที่หน่วงเวลาในการส่งผ่าน

4. TTL (Time To Live) เป็นค่าพารามิเตอร์ของเวลาในการส่งผ่านแพ็คเกจ ซึ่งจะมีการเปลี่ยนแปลงเมื่อแพ็คเกจส่งผ่านส่วนต่างๆไป

5. Agent / Null หรือบางทีจะใช้คำว่า drophead เป็นส่วนที่ใช้เก็บแพ็คเกจที่ไม่ได้ใช้งาน

การติดตามการเดินทางของแพ็คเกจ (Tracing)

ในการทดสอบประสิทธิภาพโดยทั่วไป จะมีการติดตามการไหลของแพ็คเกจในส่วนต่างๆ ว่ามีลักษณะเช่นไร เมื่อมีการใช้งานก็ทำให้ต้องเพิ่มองค์ประกอบเข้าไป แทรกในจุดต่างๆในองค์ประกอบนั้นๆ ในส่วนนี้ก็ทำให้โครงสร้างของ link มีลักษณะดังรูปที่ 7



รูปที่ 7 องค์ประกอบของ Trace ใน link

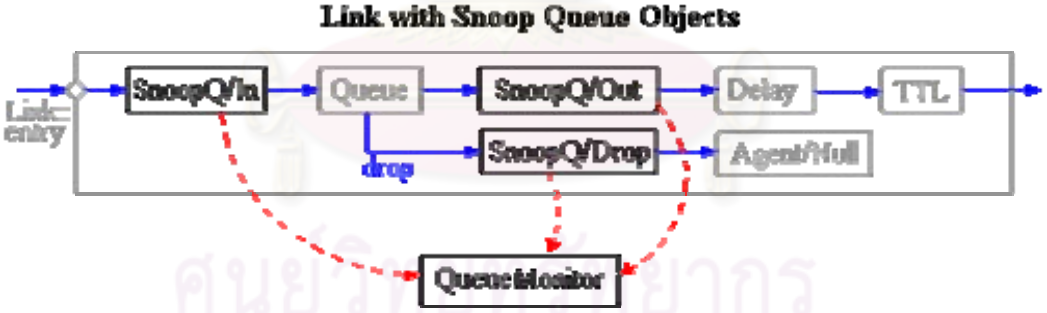
ส่วนที่เพิ่มเข้าไปจะประกอบด้วย

1. EnqT ทำหน้าที่เก็บผลของแพ็คเกจ เมื่อก่อนเข้า queue
2. DeqT ทำหน้าที่เก็บผลของแพ็คเกจ เมื่อออกจาก queue
3. DrpT ทำหน้าที่เก็บผลของแพ็คเกจ เมื่อแพ็คเกจ dropจาก queue
4. RecvT ทำหน้าที่เก็บผลของแพ็คเกจ เมื่อแพ็คเกจจะถูกส่งไปโหนดต่อไป

รายระเอียดเกี่ยวกับการติดตามการไหลของแพ็คเกจ (Tracing) จะอธิบายเพิ่มในส่วนต่อไป

การตรวจวัดการทำงานของการจัดคิว (Queue Monitor)

ในบางกรณีนั้น เราต้องการข้อมูลของการทำงานของระบบการจัดคิว (Queue) เพื่อไปวิเคราะห์ ไม่ว่าเป็นการจัดคิวในรูปแบบใด จะมีรูปแบบการทำงานในรูปแบบเดียวกัน ดังรูปที่ 8



รูปที่ 8 องค์ประกอบของการตรวจวัดการทำงานของการจัดคิว

โดยจะทำการเพิ่มตัวตรวจจับการทำงาน ที่เรียกว่า SnoopQ ไปคั่นระหว่างจุดต่างๆ ซึ่งจะประกอบเป็นระบบการทำงานของการจัดคิว

สำหรับรูปแบบคำสั่งที่ใช้งาน มีดังนี้

- การสร้างการเชื่อมต่อ แบบทิศทางเดียว (Simplex-link)

$ns simplex-link <n1> <n2> <bw> <delay> <queue_type> <args>

- การกำหนดค่าการเชื่อมต่อแบบทิศทางเดียว (สำหรับโปรแกรม NAM)

$ns simplex-link-op <n1> <n2> <option> <args>

- การสร้างการเชื่อมต่อ แบบสองทิศทาง (Duplex-link)

$ns duplex-link <n1> <n2> <bw> <delay> <queue_type> <args>

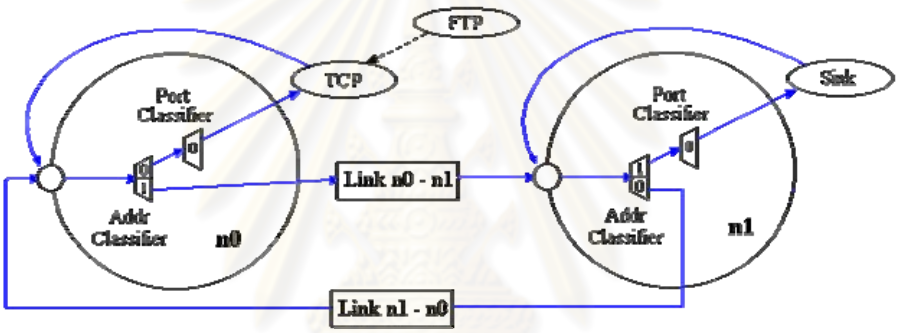
- การกำหนดค่า การเชื่อมต่อแบบสองทิศทาง (สำหรับโปรแกรม NAM)

$ns duplex-link-op <n1> <n2> <option> <args>

โดย <queue_type> เป็นประเภทของการจัดคิว เช่น DropTail, RED, CBQ, FQ, SFQ, DRR

1.1.3 ตัวแทนรูปแบบการทำงานในระดับสูง (Agent)

ตัวแทนรูปแบบการทำงานในระดับสูง หรือที่เรียกว่า Agent ก็คือรูปแบบต่างๆ ของการทำงานในระดับที่สูงขึ้น ที่ทำหน้าที่ต่อจากการทำงานในระดับล่างๆ จะเชื่อมต่อกัน (node) โดยจะแบ่งการทำงานออกเป็นส่วนๆ มีลักษณะเป็นลำดับขั้นเรื่อยๆ จนถึงรูปแบบการทำงานในระดับโปรแกรมประยุกต์ ดังเช่น TCP Agent, UDP Agent เป็นต้น เมื่อนำไปใช้งานในระบบแล้ว จะมีลักษณะดังรูปที่ 9



รูปที่ 9 ตัวแทนรูปแบบการทำงานในระดับสูง (Agent)

NS ได้สนับสนุน Agents ในหลายรูปแบบที่มีการใช้งาน ดังตัวอย่างข้างล่างซึ่งเป็นเพียงส่วนหนึ่ง

TCP เป็นโปรโตคอล TCP แบบ Tahoe ทางด้านส่ง

TCP/Reno เป็นโปรโตคอล TCP แบบ Reno ทางด้านส่ง

TCP/Newreno เป็นโปรโตคอล TCP แบบ Reno ที่มีการพัฒนาทางด้านส่ง

TCP/Sack1 เป็นโปรโตคอล TCP แบบ forward SACK ทางด้านส่ง

ในรูปแบบต่างๆ นี้เป็นรูปแบบการทำงานที่เรียกว่าเป็นคลาสลูกของ Agent Class ซึ่งจะมีรูปแบบที่เป็นพื้นฐานเหมือนๆ กัน ดังนั้นในการกำหนดการทำงานต่างๆ เหล่านี้จะมีลักษณะเหมือนกัน ในส่วนนี้จะสรุปคำสั่งพื้นฐานที่ใช้กำหนดรูปแบบ ซึ่งเป็นดังนี้

- สร้างรูปแบบการทำงานของ Agent

set <Agent_name> [new Agent/<AgentType>]

โดย <Agent_name> เป็นชื่อตัวแปรที่ใช้แทน Agent

<AgentType> เป็นชนิดของ Agent เช่น TCP, TCP/Reno, UDP เป็นต้น

- การเชื่อมต่อ Agent กับโหนด

$ns connect <node> <agent>

- การเชื่อมต่อระหว่างรูปแบบการทำงานของ Agent

$ns connect <source_agent> <destination_agent>

- การกำหนดการเชื่อมต่อระหว่าง Agent แบบสมบูรณ์

set <Agent_name> [$ns create-connection <scr_type> <scr> <dst_type> <dst> <pktclass>]

โดย <pktclass> คือ เป็นการกำหนดคลาสให้การเชื่อมต่อ

- การกำหนดในรายละเอียดของการทำงานของ Agent

$<agent> set <argument>

1.1.4. ส่วนประยุกต์ใช้งาน (Applications)

ส่วนประยุกต์ใช้งานหรือ Applications เป็นส่วนที่เชื่อมต่อระหว่างการทำงานในระดับชั้นทรานสปอร์ต (transport) ที่เราเรียกว่า Agent กับการใช้งานในระดับโปรแกรมประยุกต์ ซึ่งเป็นรูปแบบระดับการทำงานในระดับบน

ในโปรแกรม NS นั้นจะแบ่งในส่วนประยุกต์ใช้งานนี้เป็น 2 ประเภท คือ ตัวกำเนิดทราฟฟิก (Traffic generators) และการจำลองโปรแกรมประยุกต์ (Simulated applications)

สำหรับการเชื่อมต่อระหว่างระดับชั้นทรานสปอร์ตกับระดับชั้นโปรแกรมประยุกต์นั้น ก็จะใช้รูปแบบการรองรับการทำงานที่เรียกว่า Application Programming Interface (API) หรือที่รู้จักกันในชื่อว่า ซ็อคเก็ต (Sockets)

1.1.4.1 ตัวกำเนิดทราฟฟิก (Traffic generators)

ในโปรแกรม NS จะสนับสนุนอยู่ 4 ประเภท คือ

1. EXPOO Traffic

เป็นตัวกำเนิดทราฟฟิก ที่มีรูปแบบการทำงานแบบ Exponential On/Off โดยอัตราการส่งแพ็คเกจจะคงที่ ในช่วงเวลาทำการ (on periods) และจะไม่มีการส่งแพ็คเกจในช่วงเวลาหยุด (off periods) ขนาดแพ็คเกจมีขนาดคงที่

2. POO Traffic

เป็นตัวกำเนิดทราฟฟิก ที่มีรูปแบบการทำงานแบบ Exponential On/off เช่นกัน แต่ค่าช่วงเวลาทำการ และช่วงเวลาหยุดนี้ จะใช้รูปแบบของ Pareto

3. CBR Traffic

เป็นตัวกำเนิดทราฟฟิก ที่มีอัตราการส่งแพ็คเกจที่คงที่ ขนาดของแพ็คเกจคงที่ โดยสามารถกำหนดว่าให้มีการรบกวนเกิดในระหว่างการส่งได้

4. Traffic Trace

เป็นตัวกำเนิดทราฟฟิก ที่มีรูปแบบการทำงานตามค่าในไฟล์ตามเส้นทางที่แพ็คเกจผ่าน

สำหรับรูปแบบคำสั่งที่ใช้กำหนดการทำงาน จะมีรูปแบบดังนี้

set <Traffic_name> [new Application/Traffic/<Traffic_type>]

$<Traffic_name> attach-agent $<Agent_name>

หรือจะใช้คำสั่งว่า

Set <Traffic_name> [$<Agent_name> attach-app <Traffic_type>]

1.1.4.2 การจำลองโปรแกรมประยุกต์ (Simulated Application)

ในโปรแกรม NS จะสนับสนุนอยู่ 2 ประเภท คือ

1. FTP

เป็นรูปแบบการทำงานที่ใช้สำหรับการถ่ายโอนไฟล์ข้อมูล (file transfer)

2. Telnet

เป็นรูปแบบการทำงานที่ใช้สำหรับการติดต่อระหว่างเครื่องลูกข่ายกับเครื่องแม่ โดยจะมีรูปแบบการทำงานอยู่ 2 รูปแบบ คือ ถ้าค่า interval_ ไม่เป็นศูนย์ นั้นจำนวนแพ็คเกจที่ส่งจะมีรูปแบบ exponential แต่ถ้าค่า interval_ มีค่าไม่เป็นศูนย์ นั้นจำนวนแพ็คเกจที่ส่งจะมีรูปแบบของทีซีพี

1.2 กระบวนการหาเส้นทาง (Routing)

กระบวนการหาเส้นทาง เป็นกระบวนการที่จะกำหนดรูปแบบลำดับการเดินทางของแพ็คเกจข้อมูลที่มีการสื่อสารกันในเครือข่าย สามารถแบ่งออกเป็น 2 ประเภท คือ ยูนิแคส (Unicast Routing) และมัลติแคส (Multicast Routing)

1.2.1 กระบวนการหาเส้นทางแบบยูนิแคส (Unicast Routing)

รูปแบบกระบวนการหาเส้นทางแบบยูนิแคส ที่โปรแกรม NS สนับสนุน มี 4 แบบ คือ

1.Static Routing

เป็นรูปแบบที่ใช้เป็นค่าเริ่มต้นในการกำหนดกระบวนการหาเส้นทาง กระบวนการใช้แนวคิดแบบ Dijkstra's all-pairs SPF จะมีการหาเส้นทางเพียงครั้งเดียวเมื่อเริ่มต้นในการจำลองการทำงาน

2. Session Routing

เป็นกระบวนการที่ใช้แนวคิดแบบ Dijkstra's all-pair SPF เช่นกัน แต่การหาเส้นทางนั้นนอกจากจะเกิดขึ้นเมื่อตอนเริ่มต้นแล้ว ก็ยังมีการกระทำอีกครั้ง เมื่อมีการเปลี่ยนแปลงรูปแบบของเครือข่าย

3. DV Routing

เป็นกระบวนการที่ใช้รูปแบบ Distributed Bellman-Ford (หรือ Distance Vector) โดยวิธีนี้จะมีการปรับปรุงค่าเป็นระยะๆ

4. Manual Routing

เป็นกระบวนการที่ไม่ได้ใช้การคำนวณแต่จะใช้การควบคุมจากผู้ใช้งานเอง

สำหรับคำสั่งที่ใช้กำหนดรูปแบบของการหาเส้นทางจะเป็นรูปแบบดังนี้

$ns rtproto <routing-protocol> <args>

โดย <routing-protocol> คือ Static, Session, DV, Manual

1.2.2 กระบวนการหาเส้นทางแบบมัลติแคส (Multicast Routing)

รูปแบบกระบวนการหาเส้นทางแบบมัลติแคส ที่โปรแกรม NS สนับสนุนมี 4 แบบ คือ

1. Centralized Multicast หรือ CtrMcast

เป็นกระบวนการที่มีลักษณะคล้ายแบบ PIM-SM

2. Dense Mode หรือ DM

เป็นกระบวนการที่ใช้รูปแบบโปรโตคอลแบบ dense-made-like

3. Shared Tree Mode หรือ ST

เป็นรูปแบบของ sparese mode ซึ่งจะใช้โปรโตคอล shared-tree-multicast protocol

4. Bi-directional Shared Tree Mode หรือ BST

เป็นรูปแบบที่อยู่ในระหว่างการพัฒนา

สำหรับคำสั่งที่ใช้กำหนดรูปแบบกระบวนการหาเส้นทาง จะมีรูปแบบดังนี้

- กำหนดการใช้งานในรูปแบบมัลติแคส

set ns [new Simulator –multicast on]

หรือใช้คำสั่งว่า

set ns [new Simulator]

$ns multicast

- การกำหนดรูปแบบของมัลติแคส

$ns mrtproto <type>

โดย <type> คือ CtrMcast, DM, ST, BST

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

1.3 การจัดการบริการรอ (Queue Management)

การจัดการบริการรอ หรือการจัดคิวให้แพ็คเกจที่เข้ามาในส่วนต่างๆ หรือการจัดลำดับความสำคัญของแพ็คเกจต่างๆ ที่เข้ามารับบริการนั้นว่าแพ็คเกจตัวใดจะได้รับการบริการก่อนหรือหลัง ถือได้ว่าการจัดคิวเป็นส่วนสำคัญส่วนหนึ่งของระบบเครือข่าย ที่จะเป็นการกำหนดคุณภาพของการให้บริการ (Quality of Service:QOS) ในระบบเครือข่าย

NS ได้สนับสนุนการจัดคิวในหลายวิธี ดังเช่น

1.3.1 Drop-Tail Queuing (FIFO)

เป็นกลไกลการทำงานพื้นฐานที่สุดในการจัดคิว การทำงานจะจัดการกับแพ็คเกจที่เข้ามาอย่างต่อเนื่อง โดยลักษณะที่ตัวใดมาก่อนก็จะได้รับการจัดการก่อน เป็นวิธีการดั้งเดิมที่ใช้กันอย่างแพร่หลายในอุปกรณ์เครือข่ายอย่างง่าย ซึ่งลักษณะการจัดการนั้นถือเป็นการทำงานที่ไม่ดี ในการที่จะให้บริการระบบที่มีปริมาณความหนาแน่นทราฟฟิกสูงๆ

1.3.2 Fair Queuing (FQ)

เป็นกลไกลที่มีรูปแบบการทำงานแบบ Round Robin โดยจะทำการจัดเก็บแพ็คเกจที่เข้ามาลงไว้ในคิวที่ต่างกัน แล้วจึงให้บริการส่งทีละแพ็คเกจต่อคิว แล้วจึงให้บริการในคิวต่อไป

1.3.3 Stochastic Fairness Queuing (SFQ)

เป็นกลไกลที่มีรูปแบบการทำงานแบบ Round Robin เช่นกัน แต่ทำการเปลี่ยนแปลงการจัดเก็บแพ็คเกจที่เข้ามาเอาไว้รวมกันในคิวเดียว โดยใช้ Hashing Function ในการจัดการ

1.3.4 Deficit Round Robin Queuing (DRR)

เป็นกลไกลที่ใช้การกระจายแพ็คเกจลงในคิวแบบ SFQ แต่กลไกลการทำงานจะมีรูปแบบที่แตกต่างกันกับ Round Robin คือ จะมีการกำหนดขนาดของแต่ละคิวไว้ (Quantrum size) ก็คือถ้าคิวนี้ มีแพ็คเกจที่ใหญ่เกินไป ในการส่งขนาดแพ็คเกจที่เหลือที่ยังไม่ได้ส่งจะถูกนำไปรวมในครั้งต่อไป

1.3.5 Random Early Detection Queuing (RED)

เป็นกลไกลที่มีการจัดคิวต่างจากลักษณะ FIFO โดยวิธีการนี้จะพิจารณาความหนาแน่นของเครือข่ายเบื้องต้น เพื่อใช้การพิจารณาการควบคุมปริมาณข้อมูลที่จะเข้ามาในระบบ จะใช้การเลือกสุ่มการเชื่อมต่อหรือการไหลของแพ็คเกจที่ถูกทำเครื่องหมายไว้ ซึ่งจะเกี่ยวกับค่าความยาวเฉลี่ยของคิว (Average queue size) ถ้าค่าความยาวเฉลี่ยของคิวนี้มีค่าสูงขึ้นเกินกว่าระดับจุดจำกัดที่กำหนดไว้แล้ว ก็จะทำการหยุดแพ็คเกจบางส่วนไว้เพื่อให้ความหนาแน่นในเครือข่ายลดลง

1.3.6 Class-Based Queuing (CBQ)

เป็นกลไกลที่มีการกำหนดค่า Priority ให้กับแพ็คเกจโดยจะให้บริการกับแพ็คเกจที่มีค่า Priority สูงก่อน และจะมีการควบคุมการทำงานของจุดใช้งานร่วม (link-sharing)

สำหรับรูปแบบคำสั่งที่ใช้ควบคุมการทำงานในแต่ละแบบจะไม่ขอกล่าวในที่นี้ โดยจะขอกล่าวถึงคำสั่งที่ใช้ควบคุมโดยทั่วไป การกำหนดขนาดของบัฟเฟอร์ที่ใช้ในการจัดคิว จะมีรูปแบบคือ

$ns queue-limit <n1> <n2> <limit>

1.4 การติดตามเส้นทางการเดินทางของแพ็คเกจ (Trace and Monitering)

วิธีการที่ใช้ในการติดตามเส้นทางการเดินทางของแพ็คเกจนั้น สามารถทำได้ในหลายวิธี โดยทั่วไปแล้วจะมีการเก็บบันทึกรายละเอียดการติดตามเส้นทางการเดินทางของแพ็คเกจ ในขณะที่ทำการจำลองรูปแบบ หรือทำการบันทึกลงไปในไฟล์ เมื่อผ่านกระบวนการไปแล้ว ในการติดตามเส้นทางการเดินทางของแพ็คเกจนี้จะแบ่งออกเป็น 2 รูปแบบ คือ

1.4.1 การติดตามเส้นทางการเดินทางแพ็คเกจ หรือที่เรียกว่า Trace

จะเป็นการบันทึกข้อมูลของแพ็คเกจแยกเป็นแต่ละแพ็คเกจ โดยจะทำการบันทึกในแต่ละลำดับขั้นในการทำงานซึ่งผ่านในส่วนต่างๆ เช่น แพ็คเกจที่มาถึงยังโหนด, แพ็คเกจที่พร้อมที่จะออกจากโหนด เป็นต้น

ในการทำการติดตามเส้นทางการเดินทางของแพ็คเกจนั้น จะมีการเพิ่ม EnqT, DeqT, DrpT, RecvT เข้าไปในระหว่างเส้นทางการเชื่อมต่อ ดังที่กล่าวไว้แล้วหัวข้อการเชื่อมต่อ ซึ่งในแต่ละส่วนจะทำหน้าที่เก็บข้อมูลการทำงานเอาไว้ โดยสามารถกำหนดส่วนที่เพิ่มเข้าไปได้ โดยใช้คำสั่งว่า

$ns create-trace <type> <file> <source> <destination>

สำหรับรูปแบบคำสั่งที่ใช้ในการกำหนดการทำงาน จะมีรูปแบบดังนี้

- การทำการติดตามเส้นทางการเดินทางของแพ็คเกจ

  $ns trace-all <trace_file>

- การทำการติดตามเส้นทางการเดินทางของแพ็คเกจ (สำหรับ NAM)

  $ns namtrace-all <namtrace_file>

- การทำการติดตามเส้นทางการเดินทางของแพ็คเกจ โดยระบุเฉพาะ

  $ns trace-queue <n1> <n2>

- การทำการติดตามเส้นทางการเดินทางของแพ็คเกจ โดยระบุเฉพาะ (สำหรับ NAM)

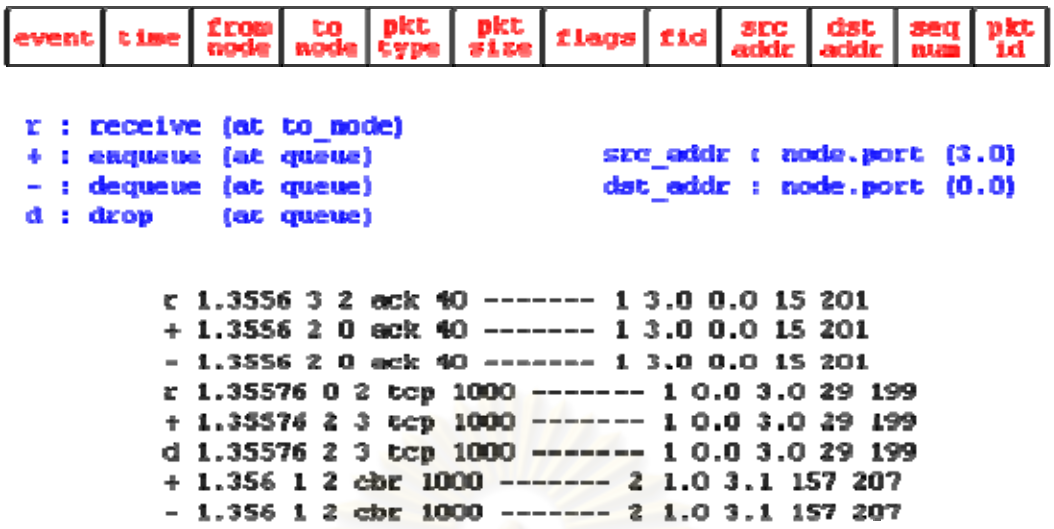
  $ns namtrace-queue <n1> <n2>

- คำสั่งที่ให้บันทึกข้อมูลการทำงาน (เป็นคำสั่งที่ใช้ก่อนที่จะจบการทำงาน)

  $ns flush-trace

สำหรับรูปแบบของไฟล์การติดตามเส้นทางการเดินทางของแพ็คเกจ (Trace file)

จะมีลักษณะชุดลำดับของค่าต่างๆ ดังรูปที่ 10



รูปที่ 10 รูปแบบของไฟล์ติดตามเส้นทางการเดินทางของแพ็คเกจ

ในแต่ละบรรทัด จะเป็นการทำงานในหนึ่งหน้าที่ โดยแต่ละบรรทัดก็จะแบ่งออกเป็น ส่วนๆ ทั้ง 12 ส่วน ดังนี้

1. event เป็นรูปแบบการทำงาน จะมีอยู่ด้วยกัน 4 ประเภท คือ

   r :receive คือ แพ็คเกจมาถึงโหนดแล้ว

   +:enqueue คือ แพ็คเกจอยู่ในระหว่างการรอคิว

   - :dequeue คือ แพ็คเกจที่ได้รับการจัดคิว

   d:drop คือ แพ็คเกจไม่ได้รับการจัดคิว

2. time เป็นเวลาที่กระทำงาน
3. from node เป็นส่วนที่ใช้บอกว่าแพ็คเกจถูกส่งมาจากที่ใด
4. to node เป็นส่วนที่ใช้บอกว่าแพ็คเกจจะถูกส่งไปที่ใด
5. packet type เป็นชนิดของแพ็คเกจ
6. packet size เป็นขนาดของแพ็คเกจ
7. flags ไม่มีการใช้งาน (มีลักษณะเป็น - - - - - -)
8. fid เป็นส่วนที่ใช้กำหนด ลำดับ IP ใน IPv6 ซึ่งยังไม่ได้ใช้งาน แต่จะใช้กำหนดสีให้กับกลุ่มแพ็คเกจในโปรแกรม NAM
9. source Address เป็นที่อยู่ของด้านส่ง จะมีรูปแบบเป็น node.port
10. destination Address เป็นที่อยู่ของด้านรับ จะมีรูปแบบเป็น node.port
11. sequence Number เป็นลำดับหมายเลขของแพ็คเกจ
12. packet ID เป็นการกำหนดค่าของแพ็คเกจ

1.4.2 การเฝ้าดูการเดินทางของแพ็คเกจ (Moniter)

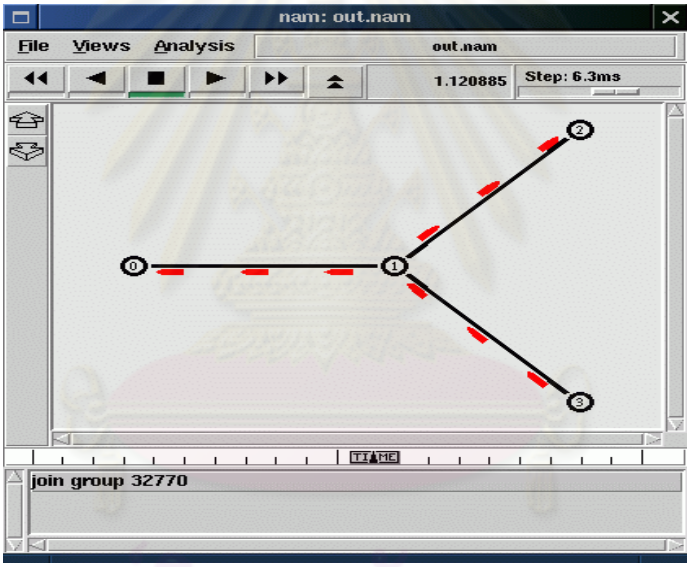
เป็นการบันทึกข้อมูลที่เจาะจงลงไปในส่วนที่สนใจเช่น ในลักษณะของการสนใจปริมาณแพ็คเกจที่ได้รับ หรือจำนวนไบต์ที่ได้รับ เป็นต้น ซึ่งในแบบนี้ก็สามารถทำการเฝ้าดูโดยรวมทั้งหมดได้ หรือ ในลักษณะที่เรียกว่า per-flow ซึ่งเป็นรูปแบบของ flow moniter

สำหรับรูปแบบคำสั่งที่ใช้ในการกำหนดการทำงาน จะมีรูปแบบดังนี้

$ns monitor-queue <n1> <n2> <qtrace> <optional:sampleinterval>

2. โปรแกรม NAM

โปรแกรม NAM หรือ Network Animator เป็นโปรแกรมที่ใช้ในการแสดงผลที่ออกมาสำหรับ NS โดยเฉพาะ เพื่อช่วยในการวิเคราะห์การทำงาน ซึ่งจะทำให้มีความเข้าใจในการทำงานได้ดีขึ้น ลักษณะโปรแกรม NAM แสดงได้ดังรูปที่ 11



รูปที่ 11 โปรแกรม NAM

สำหรับคำสั่งที่เรียกใช้งานโปรแกรม จะอยู่ในรูปแบบภาษา Tcl/Tk จะมีรูปแบบ ดังนี้

- เริ่มต้นการทำงานของโปรแกรม NAM

nam <option> <trace_file>

- การกำหนดให้โปรแกรม NS ทำการบันทึกไฟล์

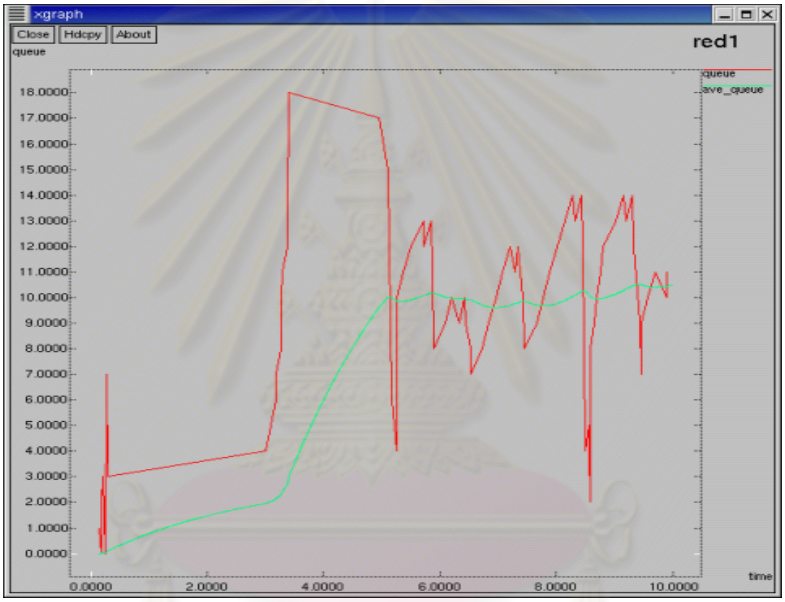
$ns namtrace-all <trace_file>

- การกำหนดให้โปรแกรมทำการเขียนลงไฟล์

$ns flush-trace

3. โปรแกรม Xgraph

โปรแกรม Xgraph เป็นโปรแกรมที่ใช้ในการเขียนกราฟ (plotting program) ในลักษณะกราฟ 2 มิติ หรือ กราฟ x-y ซึ่งเป็นที่นิยมใช้กันมากในกลุ่มที่ทำการวิเคราะห์เกี่ยวกับเครือข่าย โดยเฉพาะในกลุ่มที่ใช้งานโปรแกรมวิเคราะห์ NS ซึ่งจะมีฟังก์ชันที่ช่วยในการทำงาน ทั้งการย่อ ขยายกราฟ หรือการเลือกดูเฉพาะส่วน จะช่วยให้การวิเคราะห์ผลที่สะดวกขึ้น รวมทั้งจะมีฟังก์ชันสำหรับการพิมพ์กราฟ โดยในโปรแกรมนี้สามารถกำหนดสีให้กับกราฟต่างๆ ได้ในกรณีที่เขียนกราฟหลายๆ กราฟ ลักษณะโปรแกรม Xgraph มีลักษณะดังรูปที่ 12



รูปที่ 12 โปรแกรม Xgraph

สำหรับข้อมูลที่จะนำมาเขียนกราฟ จะเป็นลักษณะข้อมูลในรูปแบบ x-y โดยข้อมูลจะถูกแบ่งโดยช่องว่าง ไม่ว่าจะเป็นช่องว่างเดียวหรือการใช้การเว้นระยะ tab หรือ ถูกแบ่งโดยเครื่องหมาย ,; หรือ :

รูปแบบคำสั่งสำหรับการเรียกใช้งานโปรแกรม Xgraph จะมีรูปแบบดังนี้

xgraph <parameter> <filename>

ในการใช้งานโปรแกรม xgraph เพื่อเขียนกราฟนั้น โดยส่วนมากเราจะนำเอาข้อมูลที่ได้จากการติดตามเส้นทางการเดินทางของข้อมูล (trace file) มาใช้ในการเขียนกราฟ ซึ่งข้อมูลที่ได้มานี้ ก็จะมีรูปแบบตามที่ได้กล่าวมาแล้ว โดยจะเป็นข้อมูลในหลายคอลัมน์ แต่ข้อมูลที่ต้องการนำมาใช้ในการ

เขียนกราฟ จะใช้เพียงบางคอลัมน์เท่านั้น ดังนั้นจะต้องมีกระบวนการที่จะนำข้อมูลที่ต้องการออกมา ซึ่งสามารถทำได้ในหลายวิธีด้วยกัน โดยส่วนมากแล้วจะใช้วิธีเขียนโปรแกรมย่อย หรือที่เรียกว่า สคริปส์ (script) แทรกไว้ในการกำหนดการทำงานของ NS สำหรับภาษาที่นิยมใช้ในการเขียนสคริปส์นี้ จะใช้ภาษา Awk หรือ Perl Script เหตุที่นิยมใช้ 2 ภาษานี้ในการเขียนสคริปส์ ก็เพราะว่าทั้ง 2 ภาษาเป็นภาษาที่มีคำสั่งสำหรับจัดการกับข้อความต่างๆเป็นอย่างดี และมีรูปแบบคำสั่งที่ใช่งานไม่ยาก

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียนวิทยานิพนธ์

นายศราวุธ ไชยชนะ เกิดวันที่ 20 กรกฎาคม พ.ศ. 2526 ที่อำเภอเมือง จังหวัดภูเก็ต สำเร็จการศึกษาระดับปริญญาวิศวกรรมศาสตรบัณฑิต ภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2547 หลังจากนั้น ในปีพ.ศ. 2548 ได้เข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมไฟฟ้า (ระบบโทรคมนาคม) ภาควิชาวิศวกรรมไฟฟ้า จุฬาลงกรณ์มหาวิทยาลัย ปัจจุบัน ทำงานในตำแหน่ง วิศวกร กองวิศวกรรมข่ายสื่อสารการบิน บริษัท วิทยุการบินแห่งประเทศไทย จำกัด

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย