

กระบวนการจัดการอย่างมีประสิทธิภาพ  
สำหรับการใช้บัฟเฟอร์ของเกตเวย์ระหว่างเครือข่าย

EFFICACIOUS MANAGEMENT MECHANISM  
FOR BUFFER UTILIZATION OF INTERNETWORK GATEWAY

สุเมธ ประภาวัต  
SUMET PRABHAVAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชาวิศวกรรมไฟฟ้า  
บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2546

ISBN 974-324-332-1

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

กระบวนการจัดการอย่างมีประสิทธิภาพ  
สำหรับการใช้บัฟเฟอร์ของเกตเวย์ระหว่างเครือข่าย

EFFICACIOUS MANAGEMENT MECHANISM  
FOR BUFFER UTILIZATION OF INTERNETWORK GATEWAY



สุเมธ ประภาวัต

SUMET PRABHAVAT

เลขหน้.....  
เลขทะเบียน..... 47651  
วัน, เดือน, ปี..... 21 ส.ค. 2546

.b.....  
.i.....

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2546

ISBN 974-324-332-1

บัณฑิตวิทยาลัย  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ กระบวนการจัดการอย่างมีประสิทธิภาพสำหรับการใช้บัฟเฟอร์ของเกตเวย์  
ระหว่างเครือข่าย  
EFFICACIOUS MANAGEMENT MECHANISM FOR BUFFER  
UTILIZATION OF INTERNETWORK GATEWAY

ชื่อนักศึกษา นายสุเมธ ประภาวัต

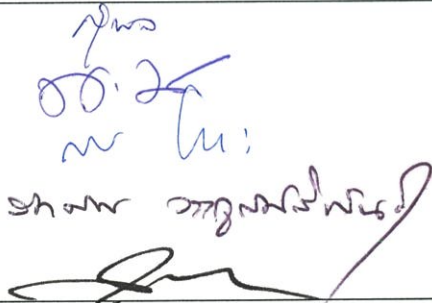
รหัสประจำตัว 41061169

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมไฟฟ้า

อาจารย์ผู้ควบคุมวิทยานิพนธ์ ผศ.ดร.สุทธิชัย นพนาศิพงษ์

อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม รศ.ดร.รัตติกร วรากุลศิริพันธุ์

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
รศ.ดร.สุวิพล	สิทธิชีวกาศ	
ดร.พิเชษฐ	ม่วงนวล	
รศ.สมยศ	จตุณณะปิยะ	
รศ.ดร.รัตติกร	วรากุลศิริพันธุ์	
ผศ.ดร.สุทธิชัย	นพนาศิพงษ์	

วัน/เดือน/ปี ที่สอบ 22 เมษายน 2546 เวลา 10.30-12.30 น.

สถานที่สอบ ณ อาคาร 12 ชั้น ชั้น 4 (ห้อง E12-404)



วันที่.....๒๒.....เดือน.....พฤษภาคม.....พ.ศ.....๒๕๔๖..

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย .....	I
บทคัดย่อ .....	II
กิตติกรรมประกาศ .....	III
สารบัญ .....	IV
สารบัญตาราง .....	VII
สารบัญรูป .....	VIII
บทที่ 1 บทนำ .....	1
1.1 ความเป็นมาและความสำคัญของปัญหา .....	1
1.2 วัตถุประสงค์ของวิทยานิพนธ์ .....	2
1.3 ทฤษฎีหรือแนวความคิด .....	2
1.4 ขอบเขตของวิทยานิพนธ์ .....	3
1.5 เนื้อหาภายในวิทยานิพนธ์ .....	3
บทที่ 2 บัฟเฟอร์ของเกตเวย์และทฤษฎีคิวที่เกี่ยวข้อง .....	5
2.1 นิยามของเกตเวย์ .....	5
2.2 หลักการทำงานของบัฟเฟอร์ของเกตเวย์ .....	6
2.3 ผลกระทบต่อประสิทธิภาพของเครือข่ายเมื่อเกิดปัญหาบัฟเฟอร์ล้น .....	6
2.4 การใช้ทฤษฎีคิววิเคราะห์ประสิทธิภาพการใช้บัฟเฟอร์ .....	9
บทที่ 3 กลไกการจัดการคิวในบัฟเฟอร์ .....	16
3.1 กลไกจัดการคิวแบบ Tail Drop .....	16
3.1.1 ลักษณะการทำงาน .....	16
3.1.2 แบบจำลองในการวิเคราะห์ประสิทธิภาพ .....	17
3.2 กลไกจัดการคิวแบบ Random Early Detection (RED) .....	20
3.2.1 วัตถุประสงค์หลักในการออกแบบกลไกแบบ RED .....	20
3.2.2 ลักษณะการทำงานของกลไกแบบ RED .....	23
3.2.3 แบบจำลองในการวิเคราะห์ประสิทธิภาพ .....	27

## สารบัญ (ต่อ)

	หน้า
บทที่ 4 กลไกการจัดการคิวในบัฟเฟอร์แบบ ExRED	30
4.1 พื้นฐานของกลไกจัดการคิวแบบ Extended Drop Slope RED (ExRED)	30
4.2 การหาค่าสัมประสิทธิ์ของสมการฟังก์ชันการดรอป	33
4.3 เงื่อนไขในการกำหนดพารามิเตอร์ของกลไก ExRED	34
4.4 แบบจำลองแสดงประสิทธิภาพของกลไกแบบ ExRED	37
บทที่ 5 การจำลองเหตุการณ์	39
5.1 หลักการสร้างแบบจำลองเหตุการณ์	39
5.2 การสร้างแบบจำลองเหตุการณ์ในงานวิจัย	42
5.3 อัลกอริทึมในการจำลองเหตุการณ์	44
5.4 การเก็บค่าทางสถิติ	52
5.5 ตัวอย่างผลการจำลองเหตุการณ์ในระบบคิวแบบ M/M/1/K	52
บทที่ 6 วิเคราะห์ และสรุปผล	56
6.1 การกำหนดพารามิเตอร์	56
6.2 การวิเคราะห์ทรูพุตและการสูญเสียจากการดรอป	58
6.3 การวิเคราะห์ค่าประวิงเวลาจากการรอคอยในคิว	61
6.4 การวิเคราะห์การเกิดการดรอปติดต่อกัน	63
6.5 ตัวอย่างผลการจำลองเมื่อมีการปรับค่าพารามิเตอร์ที่ควบคุมการใช้บัฟเฟอร์	68
6.6 สรุปผล	83
6.7 แนวทางการนำไปใช้และการปรับปรุงจากกลไกเดิม	84
เอกสารอ้างอิง	85
ภาคผนวก	86
ภาคผนวก ก. ตัวอย่างการหาการกระจายของที่น่าจะเป็นของขนาดคิว	87
ภาคผนวก ข. ผลงานวิจัยและบทความทางวิชาการที่ได้รับการตีพิมพ์	92

## สารบัญ (ต่อ)

ประวัติผู้เขียน.....	หน้า 99
----------------------	------------

# สารบัญตาราง

ตารางที่	หน้า
6.1 แสดงสมการเปรียบเทียบโอกาสเกิดการดื้อของกลไกทั้ง 3 แบบ .....	58

# สารบัญรูป

รูปที่	หน้า
2.1 แสดงตำแหน่งการเชื่อมต่อของเกตเวย์บนระบบเครือข่าย .....	5
2.2 ลักษณะการส่งข้อมูลภายใต้การควบคุมกลไกวินโดว์ที่ชี้ที่ .....	7
2.3 ความสัมพันธ์ในการเปลี่ยนเฟสของกลไกป้องกันและควบคุมความคับคั่ง .....	8
2.4 แบบจำลองคิวของแพ็คเกจในบัฟเฟอร์ .....	9
2.5 แบบจำลองของมาร์คอฟแสดงการเปลี่ยนสถานะของคิวในบัฟเฟอร์ .....	9
2.6 แบบจำลองคิวของแพ็คเกจในบัฟเฟอร์ที่จะใช้ในวิทยานิพนธ์ .....	13
3.1 อัลกอริทึมการทำงานของกลไกแบบ Tail Drop .....	17
3.2 อัลกอริทึมการทำงานของกลไกแบบ RED .....	22
3.3 แสดงฟังก์ชันการดรอป ( $p_d(\hat{q})$ ) ของกลไกแบบ RED .....	26
3.4 แบบจำลองขนาดคิวเฉลี่ยในบัฟเฟอร์ .....	27
4.1 อัลกอริทึมการทำงานของกลไกแบบ ExRED .....	31
4.2 ฟังก์ชันการดรอปของกลไกแบบ ExRED .....	32
4.3 ฟังก์ชันการดรอปเมื่ออัตราส่วนระหว่างค่ากำหนดขั้นสูงสุดต่อ ค่ากำหนดขั้นต่ำมีค่าต่างๆ .....	35
4.4 ฟังก์ชันการดรอป เมื่อขนาดบัฟเฟอร์ $K = 100$ , $k_l = 10$ , $k_h = 30$ และ $\max_p = 0.1$ .....	36
4.5 ฟังก์ชันการดรอป เมื่อขนาดบัฟเฟอร์ $K = 100$ , $k_l = 10$ , $k_h = 50$ และ $\max_p = 0.1$ .....	36
5.1 การแบ่งชั้นในการสร้างแบบจำลองเหตุการณ์ .....	39
5.2 อัลกอริทึมหลักในการจำลองเหตุการณ์ .....	44
5.3 อัลกอริทึมในการกำหนดค่าสุ่ม .....	45
5.4 อัลกอริทึมในการสร้างรายการลำดับเหตุการณ์ที่มีแพ็คเกจเข้ามาในคิวใน- บัฟเฟอร์ .....	45
5.5 อัลกอริทึมในการสร้างรายการลำดับเหตุการณ์ที่มีแพ็คเกจออกจากคิวใน- บัฟเฟอร์ .....	46
5.6 อัลกอริทึมในการคำนวณขนาดคิวในบัฟเฟอร์ .....	47
5.7 อัลกอริทึมในการดำเนินการตามกลไกแบบ Tail Drop .....	48

# สารบัญรูป (ต่อ)

รูปที่	หน้า
5.8 (ก) อัลกอริทึมในการดำเนินการตามกลไกแบบ RED ในส่วนการคำนวณหาขนาดคิวเฉลี่ย.....	48
5.8 (ข) อัลกอริทึมในการดำเนินการตามกลไกแบบ RED ในส่วนการตัดสินใจว่าจะครี้อปแพคเก็ตหรือไม่ (Packet Marking/Drop Decision).....	49
5.9 อัลกอริทึมในการดำเนินการตามกลไกแบบ ExRED ในส่วนการตัดสินใจว่าจะครี้อปแพคเก็ตหรือไม่ (Packet Marking/Drop Decision).....	50
5.10 อัลกอริทึมแก้ไขรายการลำดับเหตุการณ์.....	51
5.11 ตัวอย่างผลการหาค่าเฉลี่ยขนาดคิวในระบบคิว M/M/1/K.....	53
5.12 ตัวอย่างผลการหาค่าความน่าจะเป็นที่จะเกิดการครี้อปในระบบคิวแบบ M/M/1/K.....	54
5.13 ตัวอย่างผลการหาค่าเฉลี่ยจำนวนแพคเก็ตครี้อปติดต่อกันในระบบคิวแบบ M/M/1/K.....	54
5.14 ตัวอย่างผลค่าประวิงเวลาจากการรอคอยในคิวในบัฟเฟอร์ของระบบคิวแบบ M/M/1/K.....	55
6.1 กราฟเปรียบเทียบขนาดคิวเฉลี่ยที่คำนวณได้ กับขนาดคิวที่เกิดขึ้น.....	57
6.2 กราฟการกระจายขนาดคิวที่สถานะระบบคงตัว (Stationary Probability - Distribution).....	57
6.3 แสดงทรูพุทเปรียบเทียบกันระหว่างกลไกทั้ง 3 แบบ จากการจำลองทางคณิตศาสตร์.....	59
6.4 แสดงทรูพุทเปรียบเทียบกันระหว่างกลไกทั้ง 3 แบบ จากการจำลองเหตุการณ์.....	59
6.5 แสดงค่าประวิงเวลาเปรียบเทียบกันระหว่างกลไกทั้ง 3 แบบ.....	62
6.6 แสดงความน่าจะเป็นที่จะเกิดแพคเก็ตครี้อปติดต่อกัน $n$ แพคเก็ต ( $\frac{\lambda}{\mu} = 0.9$ ).....	64
6.7 แสดงความน่าจะเป็นที่จะเกิดแพคเก็ตครี้อปติดต่อกัน $n$ แพคเก็ต ( $\frac{\lambda}{\mu} = 1.2$ ).....	64
6.8 แสดงความน่าจะเป็นที่จะเกิดแพคเก็ตครี้อปติดต่อกัน $n$ แพคเก็ต ( $\frac{\lambda}{\mu} = 1.5$ ).....	65
6.9 แสดงความน่าจะเป็นที่จะเกิดแพคเก็ตครี้อปติดต่อกัน $n$ แพคเก็ต ( $\frac{\lambda}{\mu} = 2.0$ ).....	65

## สารบัญรูป (ต่อ)

รูปที่	หน้า
6.10 แสดงความน่าจะเป็นที่จะเกิดแพกเก็ตหรือปติดต่อกัน $n$ แพกเก็ต ( $n = 1$ ถึง $4$ )	66
6.11 แสดงความน่าจะเป็นที่จะเกิดแพกเก็ตหรือปติดต่อกัน $n$ แพกเก็ต ( $n = 5$ ถึง $8$ )	67
6.12 ค่าเฉลี่ยของจำนวนแพกเก็ตที่จะถูกหรือปติดต่อกัน $E(N)$	67
6.13 ค่าเฉลี่ยขนาดคิว $E(Q)$ เมื่อ $k_l = 10$ แพกเก็ต และ $k_h = 30$ แพกเก็ต	69
6.14 การกระจายของขนาดคิวในการทดลองแบบที่ 1 โดย $k_l = 5$ และ $k_h = 20$ แพกเก็ต	71
6.15 ค่าเฉลี่ยขนาดคิว $E(Q)$ ในการทดลองแบบที่ 1 โดย $k_l = 5$ และ $k_h = 20$ แพกเก็ต	71
6.16 การกระจายของขนาดคิวในการทดลองแบบที่ 2 โดย $k_l = 15$ และ $k_h = 50$ แพกเก็ต	72
6.17 ค่าเฉลี่ยขนาดคิว $E(Q)$ ในการทดลองแบบที่ 2 โดย $k_l = 15$ และ $k_h = 50$ แพกเก็ต	72
6.18 การกระจายของขนาดคิวในการทดลองแบบที่ 3 โดย $k_l = 10$ และ $k_h = 50$ แพกเก็ต	73
6.19 ค่าเฉลี่ยขนาดคิว $E(Q)$ ในการทดลองแบบที่ 3 โดย $k_l = 10$ และ $k_h = 50$ แพกเก็ต	73
6.20 การกระจายของขนาดคิวในการทดลองแบบที่ 4 โดย $\max_p = 0.15$	74
6.21 ค่าเฉลี่ยขนาดคิว $E(Q)$ ในการทดลองแบบที่ 4 โดยกำหนด $\max_p = 0.15$	74
6.22 ทฤษฎีในการทดลองแบบที่ 1 (ยอมให้ใช้บัฟเฟอร์น้อยลง)	76
6.23 ทฤษฎีในการทดลองแบบที่ 2 (ยอมให้ใช้บัฟเฟอร์มากขึ้น)	76
6.24 ทฤษฎีในการทดลองแบบที่ 3 (เมื่อเปลี่ยนแปลง $k_h / k_l$ )	77
6.25 ทฤษฎีในการทดลองแบบที่ 4 (เมื่อเปลี่ยนแปลง $\max_p$ )	77
6.26 ค่าประวิงเวลาจากการรอคอยในคิวในการทดลองแบบที่ 1 (ยอมให้ใช้บัฟเฟอร์น้อยลง)	78
6.27 ค่าประวิงเวลาจากการรอคอยในคิวในการทดลองแบบที่ 2 (ยอมให้ใช้บัฟเฟอร์มากขึ้น)	78

## สารบัญรูป (ต่อ)

รูปที่	หน้า
6.28 ค่าประวิงเวลาจากการรอคอยในคิวในการทดลองแบบที่ 3 (เมื่อเปลี่ยนแปลง $k_h / k_l$ ) .....	79
6.29 ค่าประวิงเวลาจากการรอคอยในคิวในการทดลองแบบที่ 4 (เมื่อเปลี่ยนแปลง $\max_p$ ) .....	79
6.30 การเกิดการครี้อปติคต่อกันในการทดลองแบบที่ 1 (ยอมให้ใช้บัฟเฟอร์น้อยลง) .....	81
6.31 การเกิดการครี้อปติคต่อกันในการทดลองแบบที่ 2 (ยอมให้ใช้บัฟเฟอร์มากขึ้น) .....	81
6.32 การเกิดการครี้อปติคต่อกันในการทดลองแบบที่ 3 (เมื่อเปลี่ยนแปลง $k_h / k_l$ ) .....	82
6.33 การเกิดการครี้อปติคต่อกันในการทดลองแบบที่ 4 (เมื่อเปลี่ยนแปลง $\max_p$ ) .....	82
ก.1 ความน่าจะเป็นในการเปลี่ยนสถานะของขนาดคิวที่เวลา $t$ ใดๆ .....	90

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ในระบบเครือข่ายสื่อสารที่มีการส่งข้อมูลแบบแพ็คเกจซึ่งมีความหนาแน่นของการจราจรสูง ปัญหาใหญ่ที่พบคือ การมีแพ็คเกจสูญหาย และเวลาที่ใช้ในการรับส่งแพ็คเกจผ่านเครือข่ายนาน ปัญหาแพ็คเกจสูญหายเกิดจากการที่มีความคับคั่งสูงของแพ็คเกจที่ต้องการผ่านเกตเวย์ซึ่งเป็นจุดเชื่อมต่อระหว่างเครือข่าย เมื่อเกตเวย์ให้บริการส่งแพ็คเกจข้ามเครือข่ายไม่ทันกับอัตราการมาขอใช้บริการ ทำให้เกิดแพ็คเกจสูญหาย

บัฟเฟอร์จึงได้ถูกนำมาใช้ที่เกตเวย์เพื่อแก้ปัญหาแพ็คเกจสูญหาย แพ็คเกจที่มาขอใช้บริการผ่านเกตเวย์ในขณะที่แพ็คเกจก่อนหน้ายังไม่ได้รับบริการและยังไม่ถูกส่งออกไปจะถูกเก็บไว้ที่บัฟเฟอร์ และแพ็คเกจใหม่ที่เข้ามาขอใช้บริการต่อมา ถ้าเกตเวย์ยังไม่พร้อมจะให้บริการอีกแพ็คเกจนั้นก็จะต้องควักไว้ที่บัฟเฟอร์อีก จนกว่าแพ็คเกจที่กำลังรับบริการอยู่จะเสร็จสิ้นและถูกส่งออกไป ในการให้บริการแบบ First-In-First-Out (FIFO) แพ็คเกจที่ต่อคิวอยู่อันดับหนึ่งก็จะได้รับบริการเป็นอันดับต่อไป และแพ็คเกจลำดับถัดไปจะถูกเลื่อนขึ้นมาตามลำดับ

กลไกจัดการคิวในบัฟเฟอร์ที่ใช้กันอยู่คือ Tail Drop เมื่อแพ็คเกจมีอัตราการขอใช้บริการสูงจนเกตเวย์ให้บริการไม่ทัน จนกระทั่งบัฟเฟอร์เต็ม แพ็คเกจที่เข้ามาขอใช้บริการต่อจากนั้นจะถูกครีโปก์ซึ่งเรียกว่าเกิดบัฟเฟอร์ล้น (Buffer Overflow) จนกว่าแพ็คเกจที่ต่อคิวในบัฟเฟอร์จะได้รับบริการและถูกส่งออกไปเพื่อให้เนื้อที่ในบัฟเฟอร์ว่างลง ซึ่งปัญหาหลักของระบบจัดการคิวแบบ Tail Drop ตามที่ระบุไว้ในงานวิจัยอื่นและ RFC 2309 [2] ที่นำเสนอโดยหน่วยงานกลางที่มีชื่อว่า Internet Engineering Task Force (IETF) คือ การเกิดการยึดครองใช้บัฟเฟอร์โดยผู้ขอใช้บริการ (Flow) เพียงรายเดียวหรือกลุ่มเล็กๆ ที่เรียกว่าล็อกเอาท์ (Lock-Out) และปัญหาการถูกครีโปก์อย่างต่อเนื่องของแพ็คเกจที่มาเป็นชุด (Burst Packet) จากการเกิดบัฟเฟอร์เต็ม เมื่อถึงภาวะบัฟเฟอร์เต็มและแพ็คเกจถูกครีโปก์อย่างต่อเนื่อง โดยเฉพาะอย่างยิ่งในระบบเครือข่าย TCP/IP จะเกิดปัญหาการซิงโครไนซ์บนเครือข่าย (Network Synchronization) เกิดการหดตัวของการขอใช้บริการลงพร้อมกัน ซึ่งทำให้เกิดการใช้งานเครือข่ายที่ไม่เต็มประสิทธิภาพ (Network Under Utilization) จากการที่มีอัตราการส่งข้อมูลในเครือข่ายโดยรวมลดลง

นอกจากนี้ปัญหาอีกอย่างหนึ่งคือ เวลาที่ใช้ในการรับส่งข้อมูลนานขึ้นจากการมีส่วนร่วมที่เกิดจากการรอคอยต่อคิวในบัฟเฟอร์ (Queue Waiting Time) เพิ่มขึ้นมา ถ้ามีแพคเกจต่อคิวในบัฟเฟอร์มากก็จะทำให้เวลารอคอยในส่วนนี้นานมากขึ้นไปด้วย

เพื่อแก้ปัญหาดังกล่าวของกลไกแบบ Tail Drop อันได้แก่ ปัญหาบัฟเฟอร์ล้นและลือกเอาท์หน่วยงาน IETF จึงได้เสนอ Random Early Detection (RED) เป็นกลไกจัดการคิวในบัฟเฟอร์แบบแอคทีฟ (Active Queue Management; AQM) ที่จะใช้ในเกตเวย์เพื่อใช้ป้องกันปัญหาดังที่กล่าวมาแต่อย่างไรก็ตามปัญหาใหญ่ของกลไกแบบ RED คืออัตราการครีโปกที่เกิดจากการทำงานของตัวมันเองที่ทำให้ทรูพุกของเกตเวย์ต่ำ แต่เดิมนั้นกลไกแบบ RED ได้ถูกนำเสนอเพื่อแก้ปัญหาล้นและการเกิดการครีโปกติดต่อกัน แม้ว่ากลไกแบบ RED จะสามารถช่วยแก้ปัญหาดังกล่าวได้ในระดับหนึ่ง คือมีการครีโปกติดต่อกันน้อยกว่ากลไกแบบ Tail Drop แต่ถ้าการจราจรคับคั่งจนเกินค่าหนึ่งซึ่งเป็นระดับที่ตัวกลไกรับได้ อัตราเฉลี่ยการครีโปกต่อเนื่องก็มีแนวโน้มสูงขึ้นอย่างมากเมื่อการจราจรข้อมูลมีความคับคั่งมากขึ้น

## 1.2 วัตถุประสงค์ของวิทยานิพนธ์

- 1) เพื่อศึกษาวิเคราะห์ผลกระทบด้านประสิทธิภาพของการให้บริการของเกตเวย์ที่เกิดจากการต่อคิวรอคอยในบัฟเฟอร์ของแพคเกจเมื่อใช้กลไกจัดการคิวแบบดั้งเดิม คือ กลไกแบบ Tail Drop และกลไกแบบ RED ที่ถูกนำเสนอโดย [1] ซึ่งถูกสนับสนุนให้นำมาใช้จัดการคิวในบัฟเฟอร์ของเกตเวย์ที่เป็นอุปกรณ์หาเส้นทาง (Router) โดย [2][3] และถูกนำมาศึกษาวิจัยเพื่อใช้จัดการคิวในบัฟเฟอร์ของเกตเวย์ที่เป็นเว็บพร็อกซี่ (WEB Proxy) โดย [11]
- 2) เพื่อนำเสนอกฎในการจัดการคิวแบบใหม่ คือ กลไกแบบ Extended Drop Slope RED (ExRED) ที่พัฒนาต่อจากกลไกแบบ RED และทำการวิเคราะห์เปรียบเทียบกับกลไกทั้งสองแบบที่ได้กล่าวมาแล้วข้างต้น

## 1.3 ทฤษฎีหรือแนวความคิด

งานวิจัยในวิทยานิพนธ์นี้ จะใช้ทฤษฎีคิวในการวิเคราะห์ผลกระทบด้านประสิทธิภาพของการให้บริการของเกตเวย์ ในส่วนที่เกิดจากการต่อคิวรอคอยในบัฟเฟอร์ของแพคเกจ โดยใช้แบบจำลองของมาร์คอฟในการสร้างแบบจำลองในการวิเคราะห์คิวที่เกิดในบัฟเฟอร์ และอีกส่วนที่เกี่ยวกับการ

นำเสนออัลกอริทึมของกลไกแบบ ExRED จะกำหนดฟังก์ชันการครีโปปให้มีลักษณะการเพิ่มขึ้นเป็นแบบฟังก์ชันโพลิโนเมียล เพื่อให้สอดคล้องกับวัตถุประสงค์หลักที่ต้องการลดการครีโปปทิ้งที่ไม่จำเป็นลง แต่ต้องสามารถเพิ่มอัตราการครีโปปทิ้งได้เมื่อที่ว่างในบัฟเฟอร์ใกล้หมด

## 1.4 ขอบเขตของวิทยานิพนธ์

งานวิจัยนี้ต้องการศึกษาผลกระทบของกลไกจัดการคิวแบบ Tail Drop และแบบ RED ในเรื่องทราฟฟิค (Throughput) การประวิงเวลาจากการรอคอยในคิวในบัฟเฟอร์ (Queuing Delay) และการเกิดการครีโปปของแพคเกจอย่างต่อเนื่อง (Consecutive Drops)

นอกจากนี้จะนำเสนอกลไกแบบ ExRED เพื่อแก้ไขจุดบกพร่องของกลไกแบบ RED ดังที่มีการศึกษาและสรุปไว้ในงานวิจัยอื่นก่อนหน้านี้อันได้แก่ ปัญหาการครีโปปมากเกินไป และปัญหาทราฟฟิคต่ำ ซึ่งจะมีการวิเคราะห์ผลกระทบของกลไกแบบ ExRED เช่นเดียวกับกลไกแบบ Tail Drop และแบบ RED โดยผลที่ได้จะมาจากผลการจำลองทางคณิตศาสตร์ (Mathematical Results) และผลการจำลองเหตุการณ์ (Simulation Results)

## 1.5 เนื้อหาภายในวิทยานิพนธ์

ในวิทยานิพนธ์ฉบับนี้เป็นการเสนอกฎการจัดการคิวแบบ ExRED เพื่อแก้ปัญหาของกลไกแบบ RED ที่ถูกนำเสนอเพื่อแก้ปัญหาที่เกิดจากกลไกแบบ Tail Drop อีกที ซึ่งกลไกทั้งสามแบบเป็นกลไกในการจัดการคิวในบัฟเฟอร์ของเกตเวย์ รวมทั้งแสดงผลการวิเคราะห์เปรียบเทียบระหว่างกลไกทั้งสามแบบดังกล่าว ดังนั้นเนื้อหาต่างๆในวิทยานิพนธ์นี้ จะถูกแยกไว้เป็นแต่ละบทดังนี้

บทที่ 2 อธิบายนิยามของเกตเวย์ หลักการทำงานของบัฟเฟอร์ในเกตเวย์ ผลกระทบจากการเกิดบัฟเฟอร์ล้นอันเนื่องมาจากการจัดการคิวที่ไม่มีประสิทธิภาพ และหลักการวิเคราะห์ประสิทธิภาพการใช้บัฟเฟอร์ด้วยทฤษฎีคิวและแบบจำลองของมาร์คอฟ

บทที่ 3 อธิบายหลักการทำงานของกลไกจัดการคิวในบัฟเฟอร์แบบ Tail Drop และแบบ RED รวมทั้งแสดงแบบจำลองทางคณิตศาสตร์ของสมการทราฟฟิค ค่าประวิงเวลาที่เกิดจากการรอคอยในคิวในบัฟเฟอร์ และความน่าจะเป็นในการเกิดแพคเกจครีโปปติดต่อกันอย่างต่อเนื่อง

บทที่ 4 นำเสนอกฎไกจัดการคิวแบบ ExRED อธิบายหลักการทำงานพื้นฐาน แสดงแบบจำลองทางคณิตศาสตร์ของสมการของทรูพูท ค่าประวิงเวลาอันเกิดจากการรอคอยในคิวในบัฟเฟอร์ และความน่าจะเป็นในการเกิดแพคเกจที่รื้อปัดต่อกันอย่างต่อเนื่อง รวมทั้งแสดงข้อแตกต่างจากกฎไกแบบ RED ที่เป็นกฎไกคั่นแบบ

บทที่ 5 อธิบายหลักการสร้างแบบจำลองเหตุการณ์ในการเกิดการรอคอยในคิวในบัฟเฟอร์ และแสดงตัวอย่างเปรียบเทียบกับแบบจำลองทางคณิตศาสตร์

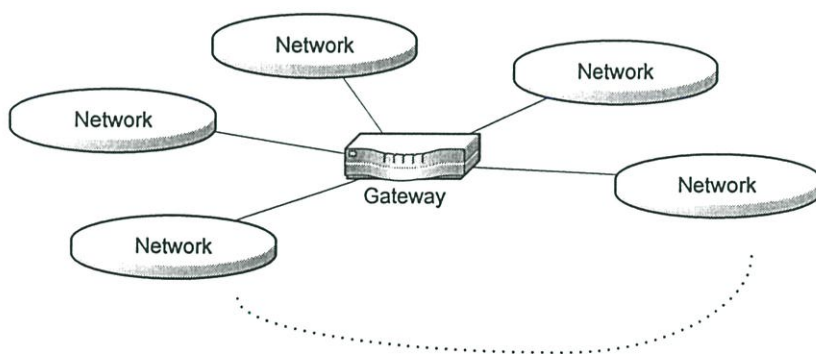
บทที่ 6 เปรียบเทียบประสิทธิภาพของกฎไกทั้งสามแบบ และสรุปผล โดยการวิเคราะห์จากผลทางคณิตศาสตร์ (Mathematical Results) และผลจากการจำลองเหตุการณ์ (Simulation Results)

## บทที่ 2

# บัพเฟอร์ของเกตเวย์และทฤษฎีคิวที่เกี่ยวข้อง

### 2.1 นิยามของเกตเวย์

เกตเวย์เป็นส่วนที่มีบทบาทมากในระบบเครือข่ายสื่อสารข้อมูล โดยนิยามหมายถึง โหนดบนระบบเครือข่ายที่ทำหน้าที่เป็นอุปกรณ์เชื่อมต่อจากระบบเครือข่ายหนึ่งไปยังอีกระบบเครือข่ายหนึ่ง ดังรูปที่ 2.1 ซึ่งการทำงานของเกตเวย์นอกจากจะเป็นอุปกรณ์เชื่อมต่อเพื่อให้ข้อมูลถูกส่งไปยังปลายทางแล้ว ยังอาจต้องแปลงรูปแบบข้อมูลจากรูปแบบที่เหมาะสมกับเครือข่ายด้านหนึ่ง ให้เหมาะสมกับระบบเครือข่ายอีกด้านที่ข้อมูลนั้นจะต้องถูกส่งออกไป จากทฤษฎีการสื่อสารข้อมูลของเครือข่ายแบบแพ็คเกจเกตเวย์มีการทำงานได้สูงสุดถึงระดับเจ็ดตามการอ้างอิงกับแบบจำลอง OSI หรือระดับชั้นประยุกต์ (Application Layer) เช่น ตัวเชื่อมต่อเครือข่าย ATM (Asynchronous Transfer Mode Network Gateway) ตัวหาเส้นทางบนเครือข่ายทีซีพีไอพี (TCP/IP Network Router) หรือ ตัวให้บริการเว็บพร็อกซี่ (WEB Proxy Server) เป็นต้น



รูปที่ 2.1 แสดงตำแหน่งการเชื่อมต่อของเกตเวย์บนระบบเครือข่าย

วิทยานิพนธ์นี้จะอ้างอิงเกตเวย์เป็นอุปกรณ์ที่ทำหน้าที่หาเส้นทางบนเครือข่ายสื่อสารทีซีพีไอพี (TCP/IP) เป็นหลัก ซึ่งระบบเครือข่ายจะประกอบด้วยโหนดสองแบบ คือ โหนดที่เป็นจุดปลาย และโหนดที่เป็นทางผ่าน โดยโหนดจุดปลาย คือ โหนดที่เป็นผู้ขอใช้และผู้ให้บริการบนเครือข่าย ส่วนโหนดที่เป็นทางผ่าน คือ โหนดที่ทำหน้าที่เชื่อมต่อระหว่างเครือข่าย หรือเรียกว่าเกตเวย์ ซึ่งโหนดที่เป็นตัวหาเส้นทาง (Router) ถือเป็นเกตเวย์ของเครือข่ายที่จะออกสู่เครือข่ายอื่น กรณีนี้ถือว่าเกตเวย์มีการทำงานถึงระดับสามตามการอ้างอิงกับแบบจำลอง OSI หรือระดับชั้นเครือข่าย (Network Layer)

โดยมีการรับส่งข้อมูลเป็นแพ็คเกจ กล่าวสรุปคือ โหนดต้นทางจากเครือข่ายหนึ่งจะส่งแพ็คเกจไปปลายทางที่อยู่อีกเครือข่าย ผ่านยังโหนดที่เป็นเกตเวย์ โดยเกตเวย์จะให้บริการโดยการทำหน้าที่เป็นตัวหาเส้นทางเพื่อส่งต่อไปยังปลายทาง

## 2.2 หลักการทำงานของบัฟเฟอร์ของเกตเวย์

เมื่อมีแพ็คเกจมาขอใช้บริการผ่านเกตเวย์ ถ้าเกตเวย์ยังไม่สามารถให้บริการได้ แพ็คเกจนั้นจะถูกครีโปกทิ้ง และสิ่งนี้จะเป็นปัญหามากโดยเฉพาะในสภาวะที่เครือข่ายมีการจราจรหนาแน่นสูง มีแพ็คเกจมาขอผ่านเกตเวย์ในอัตราสูงกว่าที่เกตเวย์จะสามารถให้บริการได้ ก็จะเกิดเหตุการณ์ที่แพ็คเกจถูกครีโปกทิ้งเป็นจำนวนมาก เกิดความสูญเสียมาก ดังนั้นบัฟเฟอร์จึงได้ถูกนำมาเป็นกลไกในการลดความสูญเสียดังกล่าว เมื่อมีแพ็คเกจที่มาถึงในขณะที่เกตเวย์ยังไม่สามารถให้บริการได้ แพ็คเกจนั้นจะถูกเก็บพักไว้ในคิว จนกว่าเกตเวย์จะสามารถให้บริการต่อไปได้ แพ็คเกจที่ถูกพักไว้ในคิวจึงจะได้รับบริการในการผ่านออกไปยังอีกเครือข่ายเพื่อไปสู่โหนดปลายทาง

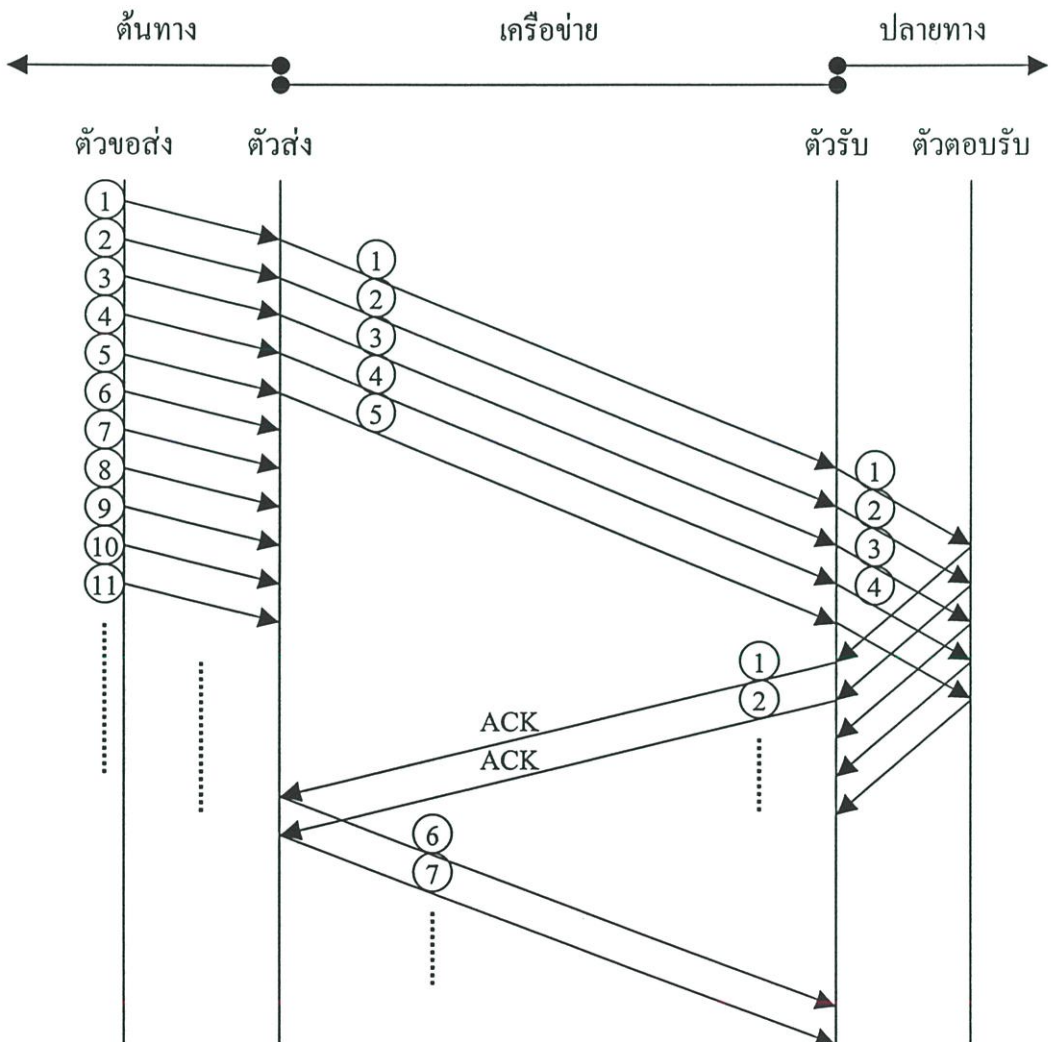
รูปแบบการให้บริการแพ็คเกจที่รอคิวอยู่ในบัฟเฟอร์ของเกตเวย์ที่จะใช้เป็นสมมติฐานในที่นี้คือ แบบ First Come First Served (FCFS) คือ แพ็คเกจที่มาถึงเกตเวย์ก่อนจะได้รับบริการก่อน ถ้ามองในรูปแบบของการเข้าออกจากคิวในบัฟเฟอร์ก็จะเป็นแบบ First In First Out (FIFO) ดังนั้นแพ็คเกจที่อยู่ลำดับแรกในคิวของบัฟเฟอร์จะได้รับบริการก่อน และลำดับของแพ็คเกจที่เหลือในคิวก็นจะถูกเลื่อนชั้นถัดขึ้นมาตามลำดับ

แต่อย่างไรก็ตามเมื่อแพ็คเกจสะสมในบัฟเฟอร์เกินขนาดบัฟเฟอร์ แพ็คเกจที่เกินมาจะถูกบดล็อกและถูกครีโปกทิ้ง เรียกว่าบัฟเฟอร์ล้น (Buffer Overflow) เป็นความสูญเสียจากการด้อยประสิทธิภาพของการใช้บัฟเฟอร์ ซึ่งมีผลต่อประสิทธิภาพการให้บริการของเกตเวย์และระบบเครือข่าย

## 2.3 ผลกระทบต่อประสิทธิภาพของเครือข่ายเมื่อเกิดปัญหาบัฟเฟอร์ล้น

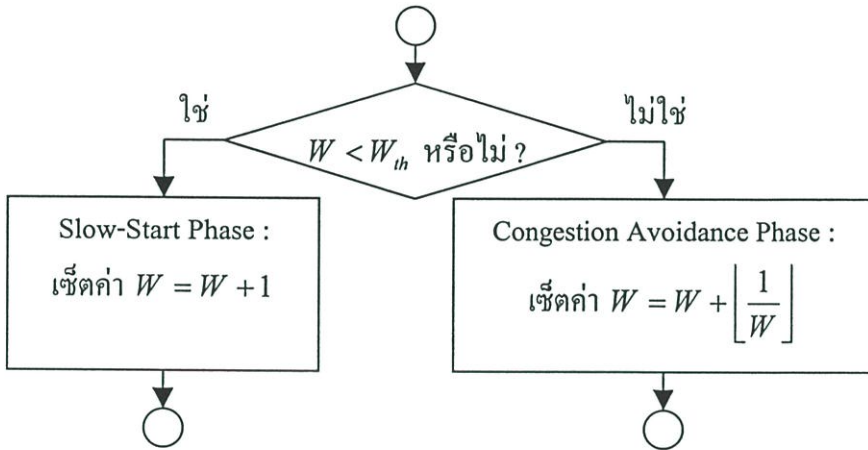
เมื่อบัฟเฟอร์เต็มจะเกิดความสูญเสียจากการที่แพ็คเกจถูกบดล็อกและครีโปก ถ้าเป็นการส่งข้อมูลจากโหนดต้นทางที่ไม่สนองตอบต่อสภาวะของเครือข่าย (Non-Responsive Source) เช่นจากโหนดต้นทางที่ส่งแพ็คเกจ UDP/IP ก็จะเกิดการสูญเสียที่เกิดจากการครีโปก และถ้าเป็นการส่งจากโหนดที่มีการสนองตอบต่อสภาวะของเครือข่าย (Responsive Source) เช่นจากโหนดต้นทางที่ส่งแพ็คเกจ TCP/IP ซึ่งในกรณีของโหนดที่มีการส่งแบบ TCP/IP นอกจากการสูญเสียจากการครีโปกแล้ว ยังมีเรื่องการชิงโครโนซ์ในการลดอัตราการส่งข้อมูลของโหนดต้นทางในระบบเครือข่าย ทำให้การใช้ระบบเครือข่ายโดยรวมลดลงมาก ซึ่งจะนำไปสู่การใช้ระบบเครือข่ายที่ไม่เต็มประสิทธิภาพ

การลดอัตราการส่งข้อมูลของโหนดต้นทาง มีสาเหตุมาจากการทำงานของกลไกป้องกันและควบคุมความคับคั่ง (Congestion Avoidance and Control Mechanism) [9][13] ที่จะลดขนาดวินโดว์ควบคุม (Control Window) ในกลไกวินโดว์ที่ซีพี (TCP Window Mechanism) เมื่อข้อมูลที่ถูกส่งไปมีการสูญหาย หรือตรวจพบการต้องส่งซ้ำในระบบเครือข่าย ซึ่งจุดประสงค์หลักของกลไกป้องกันและควบคุมความคับคั่งนั้นถูกสร้างขึ้นเพื่อป้องกันและจัดการกับความคับคั่งที่อาจเกิดขึ้นในระบบเครือข่าย โดยใช้การกำหนดโควตา (Quota) ในการส่งของแต่ละโหนดที่โหนดเหล่านั้นสามารถจะทำการส่งได้ทันทีโดยไม่ต้องรอการตอบรับยืนยันความสำเร็จในการส่ง (Acknowledgement) จากโหนดปลายทาง ในที่นี้จำนวนโควตาก็คือขนาดวินโดว์ควบคุมซึ่งมีหน่วยเป็นแพคเกจ เช่น ถ้ากำหนดขนาดวินโดว์ควบคุมไว้ที่ 5 แพคเกจ จะมีการส่งเป็นลักษณะดังรูปที่ 2.2



รูปที่ 2.2 ลักษณะการส่งข้อมูลภายใต้การควบคุมกลไกวินโดว์ที่ซีพี

กลไกป้องกันและควบคุมความคับคั่ง มีการทำงานแบ่งออกได้เป็น 2 เฟส คือ เฟสสโลว์สตาร์ท (Slow-Start Phase) และเฟสป้องกันการเกิดความคับคั่ง (Congestion Avoidance Phase) ถ้านิยาม  $W$  เป็นขนาดวินโดว์ควบคุม และ  $W_{th}$  เป็นขีดจำกัดในการเปลี่ยนเฟสการทำงาน การเปลี่ยนแปลงของวินโดว์ควบคุมจะเกิดทุกครั้งที่มีการตอบรับกลับการส่งสำเร็จจากโหนดปลายทางด้วยแพ็คเกจชนิด ACK ที่ไม่ซ้ำค่ากัน (Non-Repeated ACK Packet) ดังนั้นการเปลี่ยนแปลงของวินโดว์ควบคุมในกลไกป้องกันและควบคุมความคับคั่ง แสดงได้ดังรูปที่ 2.3



รูปที่ 2.3 ความสัมพันธ์ในการเปลี่ยนเฟสของกลไกป้องกันและควบคุมความคับคั่ง

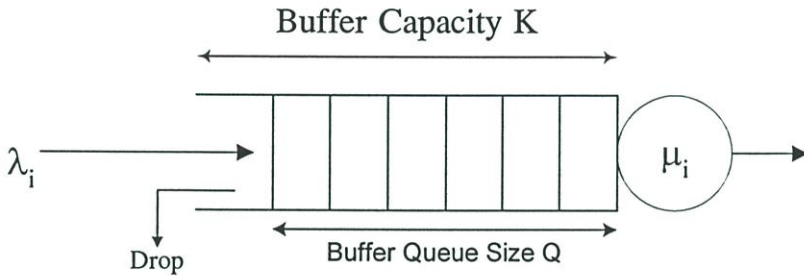
รายละเอียดของกลไกป้องกันและควบคุมความคับคั่งมีอธิบายในหนังสืออ้างอิงทั่วไป เช่น [10] ซึ่งล้วนมีลักษณะไปในแนวเดียวกันแต่จะต่างกันตรงรายละเอียดเล็กน้อย ในที่นี้จะยกกลไกควบคุมที่ใช้กันในปัจจุบันที่มีชื่อว่ากลไกแบร์โน (TCP-Reno Mechanism) กลไกจะการทำงานภายใต้ 2 เงื่อนไข คือ

- เมื่อเกิดการสูญหายของแพ็คเกจที่ส่งไป ซึ่งกลไกจะรู้ได้จากการที่ไม่มีการยืนยันการส่งสำเร็จจากโหนดปลายทางภายในเวลาที่กำหนด จะทำให้เกิดการเซตค่า  $W_{th} = \frac{W}{2}$  และ  $W = 1$  จากนั้นจะเปลี่ยนเข้าสู่เฟสสโลว์สตาร์ท (Slow-Start Phase)
- เมื่อเกิดจำนวนครั้งการส่งซ้ำจนถึงค่าที่กำหนดขอบเขตไว้ การส่งซ้ำดังกล่าวจะเกิดจากที่แพ็คเกจที่ถูกส่งไปแล้วเกิดการสูญหาย ซึ่งกลไกจะรู้ได้จากการมีแพ็คเกจแบบ ACK ซ้ำกัน (Repeated ACK Packet) ที่ดูได้จากเลขลำดับการส่งแพ็คเกจ (Packet Sequence Number) ในกรณีนี้จะทำให้เกิดการเซตค่า  $W_{th} = \frac{W}{2}$  และ  $W = W_{th}$  จากนั้นจะเปลี่ยนเข้าสู่เฟสป้องกันการเกิดความคับคั่ง (Congestion Avoidance Phase)

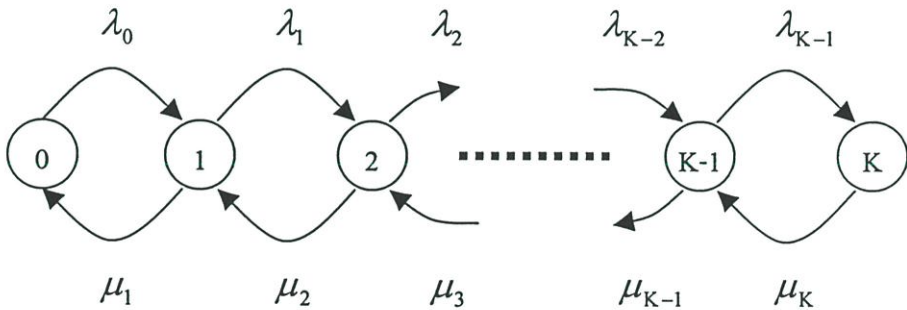
ซึ่งจะเห็นได้ว่าเมื่อเกิดการครี๊อปของแพคเกจ จะทำให้อัตราการส่งแพคเกจของโหนดต้นทางลดลง ในกรณีเกิดบัฟเฟอร์ล้นโอกาสจะเกิดการครี๊อปติดต่อกันก็จะสูง (ซึ่งจะแสดงให้เห็นโดยการ ใช้แบบจำลองเหตุการณ์ในลำดับต่อไปของวิทยานิพนธ์นี้) และเมื่อมีการครี๊อปติดต่อกันสูง จะทำให้อัตราการส่งของโหนดในเครือข่ายลดลงเป็นอย่างมาก เมื่อเทียบกับการครี๊อปที่เกิดแบบกระจายออกไป ดังนั้นในบทต่อไปจะนำเสนอกลไกการจัดการคิวที่แก้ปัญหา

### 2.4 การใช้ทฤษฎีคิววิเคราะห์ประสิทธิภาพการใช้บัฟเฟอร์

แบบจำลองคิวของแพคเกจในบัฟเฟอร์เป็นดังรูปที่ 2.4 ซึ่งมีอัตราการมาถึงเกตเวย์เท่ากับ  $\lambda_i$  มีอัตราการให้บริการเท่ากับ  $\mu_i$  มีบัฟเฟอร์ขนาดจำกัดเท่ากับ K แพคเกจ



รูปที่ 2.4 แบบจำลองคิวของแพคเกจในบัฟเฟอร์



รูปที่ 2.5 แบบจำลองของมาร์คอฟแสดงการเปลี่ยนสถานะของคิวในบัฟเฟอร์

โดยที่

$\lambda_i$  : อัตราการมาถึงของแพคเกจที่เกตเวย์ในขณะที่มี  $i$  แพคเกจอยู่ในบัฟเฟอร์ของเกตเวย์ เมื่อ  $i = 0, 1, 2, \dots, K - 1$

$\mu_i$  : อัตราการให้บริการของเกตเวย์ในขณะที่มี  $i$  แพคเกจอยู่ในบัฟเฟอร์ของเกตเวย์ เมื่อ  $i = 1, 2, 3, \dots, K$

จากแบบจำลองคิวในรูปแบบที่ 2.4 สามารถแสดงการเปลี่ยนสถานะของคิวด้วยแบบจำลองมาร์คอฟ ดังรูปที่ 2.5 ซึ่งเป็นห่วงโซ่มาร์คอฟแบบเวลาต่อเนื่อง (Continuous-time Markov Chain; CTMC) และเขียนเมตริกซ์เจเนอเรเตอร์ (Generator Matrix) [8] [12] [15] [16] ได้เป็น

$$\Theta = \begin{pmatrix} -\lambda_0 & \lambda_0 & 0 & \dots & \dots & \dots & \dots & \dots \\ \mu_1 & -(\lambda_1 + \mu_1) & \lambda_1 & 0 & \dots & \dots & \dots & \dots \\ 0 & \mu_2 & -(\lambda_2 + \mu_2) & \lambda_2 & 0 & \dots & \dots & \dots \\ 0 & 0 & \mu_3 & -(\lambda_3 + \mu_3) & \lambda_3 & 0 & \dots & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 & \mu_{K-1} & -(\lambda_{K-1} + \mu_{K-1}) & \lambda_{K-1} \\ 0 & 0 & 0 & \dots & 0 & 0 & \mu_K & -\mu_K \end{pmatrix} \quad (2.1)$$

ถ้าพิจารณาให้สถานะของคิว คือ ขนาดคิวในบัฟเฟอร์ เพราะฉะนั้นจะนิยาม  $Q$  เป็นตัวแปรสุ่มของขนาดคิวในบัฟเฟอร์ของเกตเวย์  $Q$ , เป็นตัวแปรสุ่มของขนาดคิวที่เวลา  $t$  และนิยาม  $p_{ij}(t)$  คือความน่าจะเป็นในการเปลี่ยนสถานะ (State-Transition Probability) ที่จะเปลี่ยนจากสถานะ  $i$  ณ เวลา  $s$  ไปสถานะ  $j$  ณ เวลา  $s+t$  และสถานะ  $i$  หมายถึง ณ เวลานั้นมีจำนวน  $i$  แพคเกจอยู่ในบัฟเฟอร์ของเกตเวย์ซึ่งแสดงได้ดังนี้

$$p_{ij}(t) = \Pr(Q_{s+t} = j | Q_s = i) \quad (2.2)$$

นิยาม  $P(t)$  เป็นเมตริกซ์ของความน่าจะเป็นของขนาดคิวในการเปลี่ยนสถานะดังนี้

$$P(t) = \begin{pmatrix} p_{00}(t) & p_{01}(t) & \dots & p_{0K}(t) \\ p_{10}(t) & p_{11}(t) & \dots & p_{1K}(t) \\ \vdots & \vdots & \ddots & \vdots \\ p_{K0}(t) & p_{K1}(t) & \dots & p_{KK}(t) \end{pmatrix} \quad (2.3)$$

จากสมการอนุพันธ์แบบก้าวหน้าของโคลโมโกรอฟ (Kolmogorov's forward differential equation) [15] [16] อัตราการเปลี่ยนแปลงของ  $P(t)$  สามารถแสดงได้ดังสมการที่ (2.4)

$$\frac{d}{dt}P(t) = P(t)\Theta \quad (2.4)$$

จากการแก้สมการที่ (2.4) จะได้  $P(t)$  ดังสมการที่ (2.5) โดยมีเงื่อนไขตั้งต้นดังสมการที่ (2.6) ซึ่งจะได้  $P(t)$  กระจายในรูปอนุกรมได้ดังสมการที่ (2.7)

$$P(t) = P(0)e^{t\Theta} \quad (2.5)$$

$$P(0) = I \quad (2.6)$$

$$P(t) = \sum_{k=0}^{\infty} \frac{(t\Theta)^k}{k!} \quad (2.7)$$

ถ้า  $\hat{\Theta}$  เป็นเมตริกซ์ค่าเฉพาะ (Eigen Value Matrix) และเมตริกซ์คล้าย (Similar Matrix) ของเมตริกซ์  $\Theta$  โดยที่  $\{x_0, x_1, \dots, x_K\}$  เป็นเซตของค่าเฉพาะซึ่งปกติ  $\forall i = 0, 1, 2, \dots, K$  ,  $x_i \leq 0$  และมี  $X$  เป็นเมตริกซ์เวกเตอร์เฉพาะ (Eigen Vector Matrix) ที่สอดคล้องกับ  $\hat{\Theta}$  จะแสดงความสัมพันธ์ได้ดังสมการที่ (2.8)

$$\Theta = X\hat{\Theta}X^{-1} \quad (2.8)$$

โดยที่

$$\hat{\Theta} = \begin{pmatrix} -|x_0| & 0 & 0 & 0 \\ 0 & -|x_1| & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & -|x_K| \end{pmatrix} \quad (2.9)$$

และ

$$X = [\underline{X}_0 \quad \underline{X}_1 \quad \dots \quad \underline{X}_K] \quad (2.10)$$

ดังนั้นจากสมการที่ (2.7) ถึงสมการที่ (2.10) จะเขียน  $P(t)$  ได้เป็นดังสมการที่ (2.11) และ (2.12)

$$P(t) = X \left( \sum_{k=0}^{\infty} \frac{(t\hat{\Theta})^k}{k!} \right) X^{-1} \quad (2.11)$$

จะได้

$$P(t) = X \begin{pmatrix} e^{-|x_0|t} & 0 & 0 & 0 \\ 0 & e^{-|x_1|t} & 0 & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \dots & e^{-|x_K|t} \end{pmatrix} X^{-1} \quad (2.12)$$

เมื่อระบบเข้าสู่สถานะคงตัว (Steady-State) ถ้า  $P$  เป็นการกระจายของความน่าจะเป็นของขนาดคิวของระบบที่สถานะคงตัว (Stationary Probability Distribution) ที่หาได้จากสมการที่ (2.13) ซึ่งเป็นการหาความน่าจะเป็นในระบบที่สถานะคงตัวจากความน่าจะเป็นในการเปลี่ยนสถานะ

$$P = \lim_{t \rightarrow \infty} P(t) \quad (2.13)$$

เนื่องจากการวิเคราะห์เกี่ยวกับคิวและผลกระทบต่อประสิทธิภาพบนเครือข่ายโดยทั่วไปจะเน้นวิเคราะห์ช่วงที่ระบบเข้าสู่สถานะคงตัว เพื่อวิเคราะห์การทำงานของระบบในระยะยาว (Long-Run) ดังนั้นการหา  $P$  โดยตรงก็สามารถทำได้ เมื่อเวลาผ่านไปนานๆ และระบบเข้าสู่สถานะคงตัว ความน่าจะเป็นของขนาดคิวจะถือได้ว่าคงที่ ซึ่งแสดงได้ดังสมการที่ (2.14)

$$\lim_{t \rightarrow \infty} \frac{d}{dt} P(t) = 0 \quad (2.14)$$

จากสมการที่ (2.4) และ (2.14) จะได้

$$P\Theta = 0 \quad (2.15)$$

นำเมตริกซ์จากสมการ  $P(t)$  ในสมการที่ (2.3) เขียนใหม่ได้เป็น

$$P = \begin{pmatrix} P_{00} & P_{01} & \cdots & P_{0K} \\ P_{10} & P_{11} & \cdots & P_{1K} \\ \vdots & \vdots & \ddots & \vdots \\ P_{K0} & P_{K1} & \cdots & P_{KK} \end{pmatrix} \quad (2.16)$$

ซึ่ง  $P$  ได้จากการแก้สมการที่ (2.15) และสมการนอร์มอลไลซ์ (Normalization Equation) ที่ (2.17)

$$\sum_{j=0}^K p_{ij} = 1 \quad (2.17)$$

ถ้านิยาม  $\pi(q)$  เป็นค่าความน่าจะเป็นของขนาดคิว  $q$  หรือความน่าจะเป็นที่จะมี  $q$  แพคเกจอยู่ในบัฟเฟอร์ของเกตเวย์เมื่อระบบอยู่ในสถานะคงตัว โดยกำหนดให้คิวว่างในขณะเริ่มต้นที่เวลา  $t = 0$  ดังสมการที่ (2.18) จะได้  $\pi(q)$  ดังสมการที่ (2.19)

$$\pi(q) = p_{0q} := \Pr(Q_{t \rightarrow \infty} = q | Q_{t=0} = 0) \quad (2.18)$$

$$\pi(q) = \frac{\prod_{i=0}^{q-1} \lambda_i}{1 + \sum_{j=1}^K \prod_{i=0}^{j-1} \lambda_i \mu_{i+1}} \quad (2.19)$$

โดยที่

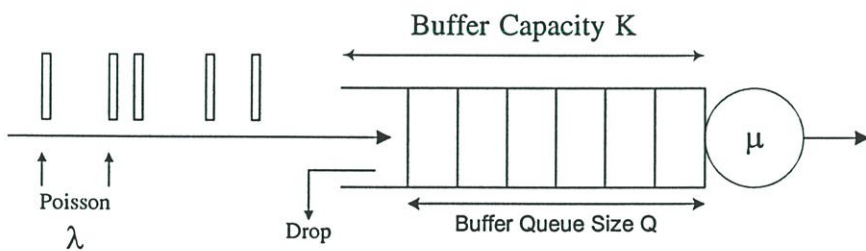
$\lambda_i$  : อัตราการมาถึงของแพ็คเกจที่เกตเวย์ในขณะที่มี  $i$  แพ็คเกจอยู่ในบัฟเฟอร์ของเกตเวย์ เมื่อ  $i = 0, 1, 2, \dots, K-1$

$\mu_i$  : อัตราการให้บริการของเกตเวย์ในขณะที่มี  $i$  แพ็คเกจอยู่ในบัฟเฟอร์ของเกตเวย์ เมื่อ  $i = 1, 2, 3, \dots, K$

$K$  : ขนาดบัฟเฟอร์ในหน่วยแพ็คเกจ

เพื่อพิสูจน์สมการที่ (2.19) จะแสดงตัวอย่างการคำนวณความน่าจะเป็นในการเปลี่ยนสถานะของขนาดคิว  $P(t)$  และความน่าจะเป็นของขนาดคิวที่สถานะคงตัว  $\pi(q)$  ในภาคผนวก ก.

งานวิจัยในวิทยานิพนธ์นี้จะวิเคราะห์ค่าเฉลี่ยของระบบที่สถานะคงที่ ซึ่งสมมติให้มีอัตราการมาถึงของแพ็คเกจที่เกตเวย์เป็นไปตามกระบวนการปัวร์ซอง (Poisson Process) มีอัตราเฉลี่ยเท่ากับ  $\lambda$  แพ็คเกจต่อวินาที และเวลาที่ใช้ในการให้บริการเป็นค่าสุ่มที่มีการกระจายแบบเอ็กซ์โปเนนเชียล (Exponential Distribution) โดยมีอัตราเฉลี่ยการให้บริการเท่ากับ  $\mu$  แพ็คเกจต่อวินาที จากรูปที่ 2.4 สามารถแสดงแบบจำลองคิวที่ใช้ในวิทยานิพนธ์นี้ได้ดังรูปที่ 2.6 และจากแบบมาร์คอฟในรูปที่ 2.5 จะได้ว่า  $\lambda_i = \lambda$  และ  $\mu_{i+1} = \mu$  ที่  $i = 0, 1, 2, \dots, K-1$



รูปที่ 2.6 แบบจำลองคิวของแพ็คเกจในบัฟเฟอร์ที่จะใช้ในวิทยานิพนธ์

จะคำนวณความน่าจะเป็นที่จะเกิดการครีอ์ปที่เกตเวย์ได้จากสมการที่ (2.20)

$$P_{drop} = \sum_{q=0}^K \pi(q) p_d(q) \quad (2.20)$$

โดยที่

$P_{drop}$	: ความน่าจะเป็นที่จะมีการครีโปกเกิดขึ้นที่เกตเวย์
$q$	: ขนาดคิวปัจจุบันในบัฟเฟอร์ของเกตเวย์ (แพคเกจ)
$p_d(q)$	: ความน่าจะเป็นที่แพคเกจจะถูกครีโปก เมื่อมีขนาดคิวเท่ากับ $q$
$K$	: ขนาดบัฟเฟอร์ (แพคเกจ)

ถ้านิยามให้ทรูพุทในวิทยานิพนธ์นี้ คืออัตราการยอมให้แพคเกจผ่านออกจากเกตเวย์ เทียบกับการมีแพคเกจมาขอใช้บริการผ่านเกตเวย์ โดยไม่คิดการส่งใหม่ (Retransmission) จะสามารถคำนวณทรูพุท ( $S$ ) ได้จากความน่าจะเป็นที่จะเกิดการครีโปก ( $P_{drop}$ ) ดังสมการที่ (2.21) โดยที่  $\lambda$  คือ อัตราเฉลี่ยการมีแพคเกจมาขอผ่านเกตเวย์

$$S = (1 - P_{drop})\lambda \quad (2.21)$$

ในการพิจารณาประสิทธิภาพการให้บริการเครือข่ายของเกตเวย์ นอกจากการวิเคราะห์ทรูพุทแล้ว ยังมีค่าประวิงเวลาจากการรอคอยในบัฟเฟอร์ที่เกตเวย์ที่ต้องพิจารณาคด้วย ซึ่งค่าประวิงเวลาสามารถคำนวณได้จากการใช้ทฤษฎีของลิตเติล (Little's Theorem) ซึ่งแสดงได้ดังสมการที่ (2.22)

$$D = \frac{E(Q)}{(1 - P_{drop})\lambda} - \frac{1}{\mu} \quad (2.22)$$

โดยที่

$D$	: ค่าประวิงเวลาจากการเข้าคิวรอคอย (วินาที)
$Q$	: ตัวแปรสุ่มขนาดคิวในบัฟเฟอร์ของเกตเวย์ (แพคเกจ)
$E(Q)$	: ค่าเฉลี่ยทางคณิตศาสตร์ (Arithmetic Mean) ของขนาดคิวในบัฟเฟอร์
$P_{drop}$	: ความน่าจะเป็นที่จะมีการครีโปกเกิดขึ้นที่เกตเวย์
$\lambda$	: อัตราเฉลี่ยการมีแพคเกจมาขอผ่านเกตเวย์ (แพคเกจต่อวินาที)
$\mu$	: อัตราเฉลี่ยการให้บริการของเกตเวย์ (แพคเกจต่อวินาที)

ซึ่งจะสามารถหา  $E(Q)$  ได้จากสมการที่ (2.23)

$$E(Q) = \sum_{q=0}^K (q \times \pi(q)) \quad (2.23)$$

สำหรับความน่าจะเป็นที่จะเกิดแพคเกจติดหรือปัดต่อกัน (Consecutive Drops) ที่เกตเวย์ซึ่งเป็นประเด็นที่น่าสนใจ เนื่องจากสามารถชี้ถึงโอกาสการเกิดการติดหรือปัดต่อเนื่อง โดยเฉพาะอย่างยิ่งเมื่อเกิดความคับคั่งของการจราจรข้อมูล เป็นสาเหตุโดยนัยจากการเกิดบัฟเฟอร์ล้น (Buffer Overflow) ซึ่งจะมีผลเสียอย่างมากต่อประสิทธิภาพในการให้บริการของเกตเวย์ต่อระบบเครือข่าย ทำให้เกิดผลกระทบดังที่ได้กล่าวในหัวข้อ 2.3 และการเกิดซิงโครไนซ์บนเครือข่าย (Network Synchronization) ซึ่งความน่าจะเป็นที่จะเกิดแพคเกจติดหรือปัดต่อกันสามารถคำนวณได้จากการหาความน่าจะเป็นที่แพคเกจที่มาถึงในลำดับถัดไปจะถูกครีโปกหลังจากมีการครีโปกไปแล้วก่อนหน้านี้

การปรับปรุงประสิทธิภาพเพื่อลดการเกิดแพคเกจติดหรือปัดต่อกันเป็นประเด็นที่งานวิจัยที่นำเสนอในวิทยานิพนธ์นี้ให้ความสำคัญด้วยนอกเหนือจากทฤษฎี แต่การคำนวณจะแตกต่างกันไปสำหรับในแต่ละกลไกที่ใช้จัดการคิวในบัฟเฟอร์ดังกล่าวจะแสดงในบทที่ 3 และ 4 ต่อไป

## บทที่ 3

# กลไกการจัดการคิวในบัฟเฟอร์ของเกตเวย์

การจัดการคิวในบัฟเฟอร์ คือ การควบคุมจำนวนข้อมูลที่จะเก็บพักชั่วคราวในบัฟเฟอร์ให้เป็นไปตามรูปแบบที่ต้องการ มาตรการที่ใช้ในการควบคุมคือ การยอมหรือไม่ยอมให้แพ็คเกจที่มาขอผ่านเกตเวย์ (แต่เกตเวย์ยังไม่สามารถพร้อมให้บริการ) สามารถเข้าพักในบัฟเฟอร์ ถ้ากลไกการจัดการยอมให้ผ่าน แพ็คเกจดังกล่าวจะสามารถเข้าพักในบัฟเฟอร์ในลำดับถัดมา แต่ถ้าไม่ยอมให้ผ่าน กลไกจะทิ้งแพ็คเกจ (Discard) โดยการครีอแพ็คเกจดังกล่าว หรือรีเจ็ค (Reject) จากบัฟเฟอร์คิว

ในการกล่าวถึงต่อไป คำว่า “การบล็อก (Blocking)” หรือ “การไม่ยอมให้ผ่าน” หรือ “การทิ้ง (Discard)” จะใช้คำว่าครีอ (Drop) แทน ดังนั้นความน่าจะเป็นที่แพ็คเกจที่มาถึงจะถูกบล็อกหรือถูกทิ้ง จะกล่าวแทนภายใต้คำนิยามของคำว่า ความน่าจะเป็นที่แพ็คเกจจะถูกครีอ ซึ่งถูกกำหนดด้วยฟังก์ชันการครีอ (Drop Function) และฟังก์ชันการครีอจะมีลักษณะเฉพาะของแต่ละกลไกการจัดการคิวในบัฟเฟอร์ ซึ่งในบทนี้จะแสดงรายละเอียดกลไกการจัดการคิวในบัฟเฟอร์แบบดั้งเดิมคือ กลไกแบบ Tail Drop และกลไกแบบ RED ที่ถูกนำเสนอโดย IETF ใน RFC 2309 [2]

ในที่นี้คำว่า “คิว (Queue)” หมายถึงการเก็บพักแพ็คเกจตามลำดับการมาถึงไว้ในบัฟเฟอร์ของเกตเวย์ เพื่อรอรับการส่งออกหรือส่งต่อโดยโปรเซสเซอร์ (Processor) ของเกตเวย์ ดังนั้นคำว่า “ขนาดคิว (Queue Size)” หมายถึงจำนวนแพ็คเกจที่ยังถูกเก็บพักไว้ในบัฟเฟอร์

### 3.1 กลไกการจัดการคิวแบบ Tail Drop

#### 3.1.1 ลักษณะการทำงาน

กลไกแบบ Tail Drop เป็นกลไกที่ยอมให้แพ็คเกจเข้ามาพักในบัฟเฟอร์ได้จนกว่าขนาดคิวในบัฟเฟอร์จะสูงเกินกว่าที่บัฟเฟอร์จะรองรับได้ ทำให้แพ็คเกจที่มาขอใช้บริการผ่านเกตเวย์ในขณะที่บัฟเฟอร์เต็มถูกครีอทิ้งโดยไม่ยอมให้เข้าพักต่อคิวในบัฟเฟอร์ ซึ่งสามารถแสดงอัลกอริทึมการทำงานของกลไกได้ดังรูปที่ 3.1 และแสดงฟังก์ชันการครีอได้ดังสมการที่ (3.1)

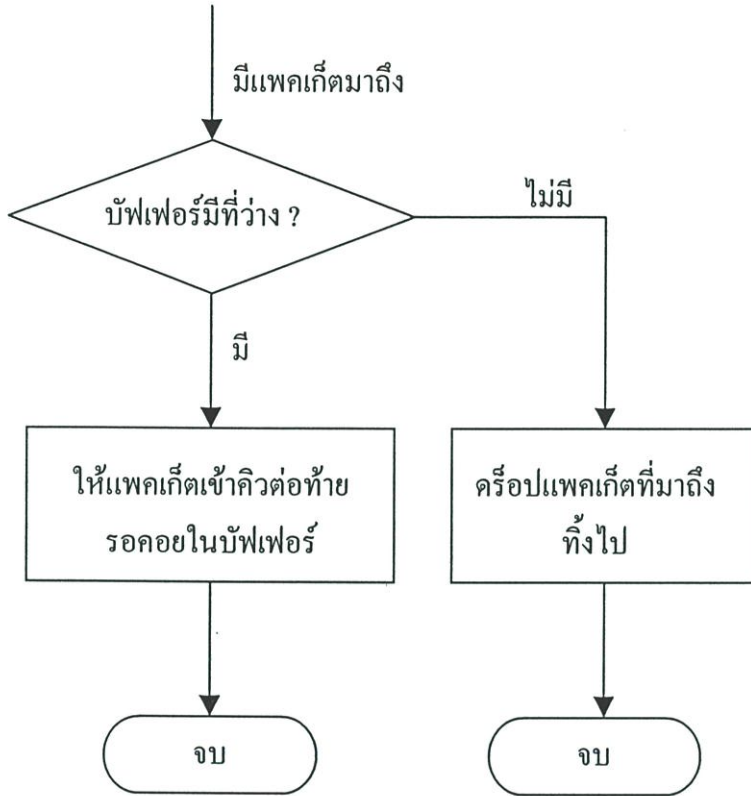
$$p_d(q) = \begin{cases} 0 & ; q < K \\ 1 & ; q = K \end{cases} \quad (3.1)$$

โดยที่

$q$  : ขนาดคิวปัจจุบันในบัฟเฟอร์ของเกตเวย์ (Instantaneous Queue Size) ในหน่วยแพ็คเกจ

$p_d(q)$  : ความน่าจะเป็นที่แพ็คเกจจะถูกครีโปก (Packet Drop Probability) เมื่อมีขนาดคิวเท่ากับ  $q$

$K$  : ขนาดบัฟเฟอร์ในหน่วยแพ็คเกจ (Buffer Size)



รูปที่ 3.1 อัลกอริทึมการทำงานของกลไกแบบ Tail Drop

### 3.1.2 แบบจำลองในการวิเคราะห์ประสิทธิภาพ

จากสมมติฐานในการวิเคราะห์ประสิทธิภาพของการใช้บัฟเฟอร์ในวิทยานิพนธ์นี้ กำหนดให้เกตเวย์มีบัฟเฟอร์ขนาดความจุ  $K$  แพ็คเกจ การมาขอใช้บริการผ่านเกตเวย์เป็นไปตามกระบวนการปัวร์ซอง (Poisson Process) มีอัตราเฉลี่ยการมาขอผ่านเท่ากับ  $\lambda$  แพ็คเกจต่อวินาที และเวลาที่ใช้ในการให้บริการเป็นค่าสุ่มที่มีการกระจายแบบเอ็กซ์โปเนนเชียล (Exponential Distribution) โดยมีอัตราเฉลี่ยการให้บริการเท่ากับ  $\mu$  แพ็คเกจต่อวินาที ทำให้สามารถเขียนสมการของการกระจายของขนาดคิวในบัฟเฟอร์ได้จากสมการที่ (2.19) เป็นดังสมการที่ (3.2)

$$\pi(q) = \frac{\left(1 - \frac{\lambda}{\mu}\right) \left(\frac{\lambda}{\mu}\right)^q}{1 - \left(\frac{\lambda}{\mu}\right)^{q+1}} \quad (3.2)$$

ค่าความน่าจะเป็นที่จะเกิดการครีโปกที่เกตเวย์สามารถหาได้จากสมการที่ (2.20) โดยใช้ความน่าจะเป็นที่แพคเกจจะถูกครีโปกจากสมการที่ (3.1) จะได้ความน่าจะเป็นที่จะเกิดการครีโปกที่เกตเวย์ที่ใช้กลไกการจัดการคิวแบบ Tail Drop ดังสมการที่ (3.3) และจากสมการ (2.21) ถ้า  $S$  คือ ทราฟฟิคที่ออกจากเกตเวย์จะคำนวณได้จากสมการที่ (3.4)

$$P_{drop} = \pi(K) \quad (3.3)$$

$$S = (1 - P_{drop})\lambda \quad (3.4)$$

ค่าประวิงเวลาจากการรอคอยในคิว (Queuing Delay) สามารถหาได้จากสมการที่ (3.5) ซึ่งเป็นการยกมาแสดงจากสมการที่ (2.22) โดยคำนวณค่าเฉลี่ยขนาดคิว  $E(Q)$  ได้จากสมการ (2.23) ซึ่งค่าเฉลี่ยขนาดคิวของกลไกแบบ Tail Drop แสดงในสมการที่ (3.6)

$$D = \frac{E(Q)}{(1 - P_{drop})\lambda} - \frac{1}{\mu} \quad (3.5)$$

$$E(Q) = \begin{cases} \frac{\frac{\lambda}{\mu} \left(1 - (K+1) \left(\frac{\lambda}{\mu}\right)^K + K \left(\frac{\lambda}{\mu}\right)^{K+1}\right)}{\left(1 - \left(\frac{\lambda}{\mu}\right)^{K+1}\right) \left(1 - \frac{\lambda}{\mu}\right)} & ; \lambda \neq \mu \\ \frac{K}{2} & \lambda = \mu \end{cases} \quad (3.6)$$

โดยที่

$D$  : ค่าประวิงเวลาจากการเข้าคิวรอคอย (Queuing Delay) ในหน่วย วินาที

$Q$  : ตัวแปรสุ่มขนาดคิวในบัฟเฟอร์ของเกตเวย์ (Queue Size) ในหน่วย แพคเกจ

$E(Q)$  : ค่าเฉลี่ยทางคณิตศาสตร์ของขนาดคิวในบัฟเฟอร์ (Mean Queue Size)

$P_{drop}$  : ความน่าจะเป็นที่จะมีการครีโปกเกิดขึ้นที่เกตเวย์ (Drop Probability)

- $\lambda$  : อัตราเฉลี่ยการมีแพคเกจเข้ามาขอผ่านเกตเวย์ (Arrival Rate) ในหน่วย แพคเกจต่อวินาที  
 $\mu$  : อัตราเฉลี่ยการให้บริการของเกตเวย์ (Service Rate) ในหน่วย แพคเกจต่อวินาที

การวิเคราะห์ในประเด็นเกี่ยวกับการเกิดแพคเกจติดครีอปติดต่อกัน (Consecutive Drops) ตามที่ได้เกริ่นไว้ถึงความจำเป็นที่ต้องพิจารณาในตอนท้ายของบทที่ 2 สามารถวิเคราะห์ได้ในรูปแบบความน่าจะเป็นที่จะมีการครีอปต่อเนื่องกัน และค่าเฉลี่ยของจำนวนแพคเกจที่จะถูกครีอปติดต่อกัน

สมมติให้การครีอปใดๆเกิด ณ เวลา  $t = 0$  ที่เกตเวย์แบบ Tail Drop และด้วยเหตุที่การกระจายแบบเอ็กซ์โปเนนเชียล (Exponential Distribution) เป็นการกระจายที่เหตุการณ์ปัจจุบันไม่ได้รับอิทธิพลจากเหตุการณ์ในอดีตและไม่มีผลต่อเหตุการณ์ในอนาคต แพคเกจที่มาถึงในลำดับถัดไปนั้นจะถูกครีอปก็ต่อเมื่อแพคเกจนั้นมาถึงเกตเวย์ในขณะที่บัฟเฟอร์เต็มอยู่และเกตเวย์กำลังให้บริการแพคเกจที่อยู่หัวคิวยังไม่เสร็จ ดังนั้นเมื่อแพคเกจหนึ่งถูกครีอป โอกาสที่แพคเกจที่มาถึงในลำดับถัดไปจะถูกครีอปต่อเนื่องเท่ากับ  $p$  โดยที่

$$p = \int_0^{\infty} \mu(1 - e^{-\lambda x})e^{-\mu x} dx = \frac{\lambda}{\lambda + \mu} \quad (3.7)$$

ถ้านิยามให้  $N$  เป็นตัวแปรสุ่มของจำนวนแพคเกจที่ถูกครีอปติดต่อกันอย่างต่อเนื่อง และ  $n$  เป็นค่าของ  $N$  เมื่อเกิดการครีอปขึ้นมาครั้งหนึ่ง ความน่าจะเป็นที่แพคเกจจะถูกครีอปติดต่อกันอย่างต่อเนื่องอีก  $n$  แพคเกจ สามารถแสดงได้ดังสมการ (3.8)

$$\forall n \geq 0, P_{drop}^{cons}(N > n) = p^n \quad (3.8)$$

จากสมการที่ (3.8) สามารถหาค่าเฉลี่ยทางคณิตศาสตร์ (Arithmetic Mean) ของจำนวนแพคเกจที่จะถูกครีอปติดต่อกันได้จากสมการที่ (3.9)

$$E(N) = 1 + \frac{\lambda}{\mu} \quad (3.9)$$

## 3.2 กลไกจัดการคิวแบบ Random Early Detection (RED)

ในระบบเครือข่ายแบบดั้งเดิมที่มีกลไกจัดการคิวในบัฟเฟอร์แบบ Tail Drop เมื่อเกิดความคับคั่งของการจราจรข้อมูลในระบบเครือข่ายบนบัฟเฟอร์ของเกตเวย์เต็ม เกตเวย์จะเริ่มครีโปกแพกเก็ตในระบบเครือข่ายแบบทีซีพี (TCP) จะมีการส่งสัญญาณไปโหนดต้นทางเพื่อเข้าสู่เฟสสโลว์สตาร์ท (Slow Start Phase) เพื่อลดภาระบนเครือข่าย และบรรเทาความคับคั่งลง ซึ่งในรูปแบบนี้จะมีควมยุ่งยาก 2 อย่าง คือ การต้องส่งแพกเก็ตที่สูญหายไปซ้ำ เป็นการเพิ่มภาระแก่ระบบเครือข่าย และเสียเวลาเพิ่มสำหรับโฟลว์แบบทีซีพีอื่นๆ (TCP Flows) และที่แย่ไปกว่านั้นคือการเกิดปรากฏการณ์ที่เรียกว่า โกลบอลซิงโครไนเซชัน (Global Synchronization) ได้แก่การสื่อสารข้อมูลแบบที่มีข้อมูลมาหนาแน่นเป็นระลอก (Burst Traffic) จนทำให้บัฟเฟอร์คิวล้น และเกิดแพกเก็ตครีโปกจำนวนมาก มีแนวโน้มที่การเชื่อมต่อแบบทีซีพีจำนวนมากที่ได้รับผลกระทบดังกล่าว จะเปลี่ยนการทำงานเข้าสู่เฟสสโลว์สตาร์ท เกิดการลดการใช้เครือข่ายชั่วคราว จนกระทั่งเกิดช่วงเวลาที่ไม่มีการใช้งาน และด้วยเหตุที่มีทีซีพีโฟลว์จำนวนมากเข้าและออกจากเฟสสโลว์สตาร์ทพร้อมๆกันจนไปมาเกิดการวนเป็นวัฏจักรของเหตุการณ์ที่ชั่วขณะก็มีการใช้งานเครือข่ายหนักหน่วง และอีกชั่วขณะต่อมาก็แทบจะไม่มีการใช้งานเครือข่าย

วิธีการแก้ปัญหาดังกล่าวโดยการเพิ่มขนาดบัฟเฟอร์ที่เกตเวย์เพื่อลดความน่าจะเป็นของการเกิดแพกเก็ตครีโปก จะทำให้เกิดค่าประวิงเวลาสูงเนื่องจากการรอคอยในบัฟเฟอร์ และเป็นการแก้ปัญหาที่ไม่รู้จบเนื่องจากไม่รู้ว่าจะต้องใช้บัฟเฟอร์ขนาดเท่าใดให้เพียงพอต่อการรองรับการมาถึงของแพกเก็ตที่มาต่อเนื่องกันแบบเป็นระลอก

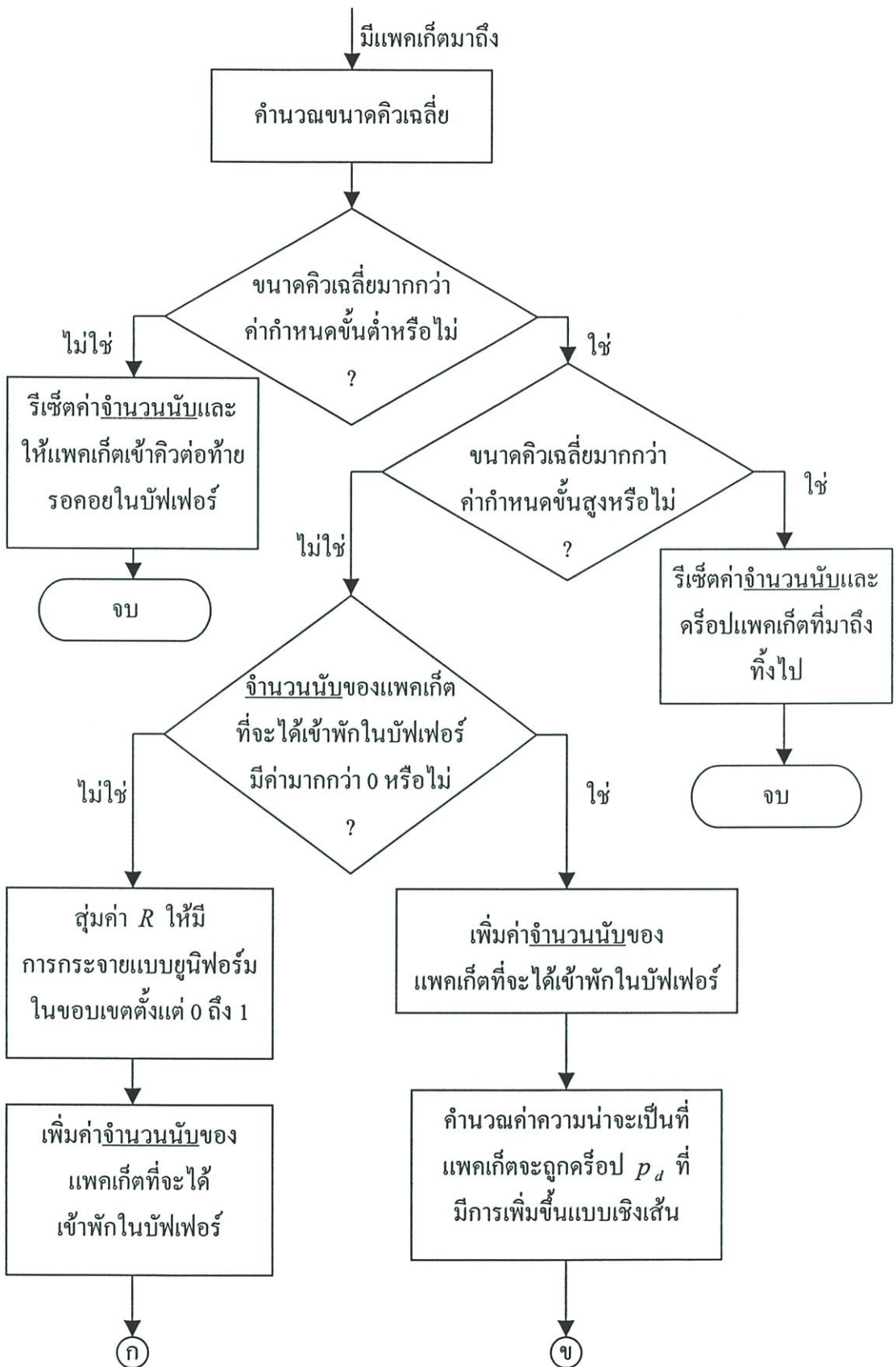
ทางแก้ที่ดีจึงควรเป็นการคาดการณ์พร้อมกับการดำเนินการล่วงหน้าก่อนที่จะเกิดความคับคั่งของการจราจรข้อมูลขึ้น และจัดการให้โหนดต้นทางตัวใดตัวหนึ่งลดอัตราการส่งลง จากนั้นวัดผลที่เกิดขึ้น ถ้าจำเป็นจึงค่อยจัดการให้โหนดต้นทางอื่นอีกหนึ่งโหนดลดอัตราการส่งลงด้วย ด้วยวิธีแก้แบบนี้เมื่อเริ่มเกิดความคับคั่งขึ้น การลดอัตราการใช้เครือข่ายจะค่อยๆเกิดขึ้นทีละน้อย ซึ่งจะเกิดผลกระทบต่อการติดต่อแบบทีซีพีเพียงเล็กน้อยเท่านั้น และไม่ทำให้เกิดการซิงโครไนเซชันบนเครือข่าย (Network Synchronization) วิธีดังกล่าวนี้เป็นเทคนิคของกลไกแบบ RED

### 3.2.1 วัตถุประสงค์หลักในการออกแบบกลไกแบบ RED

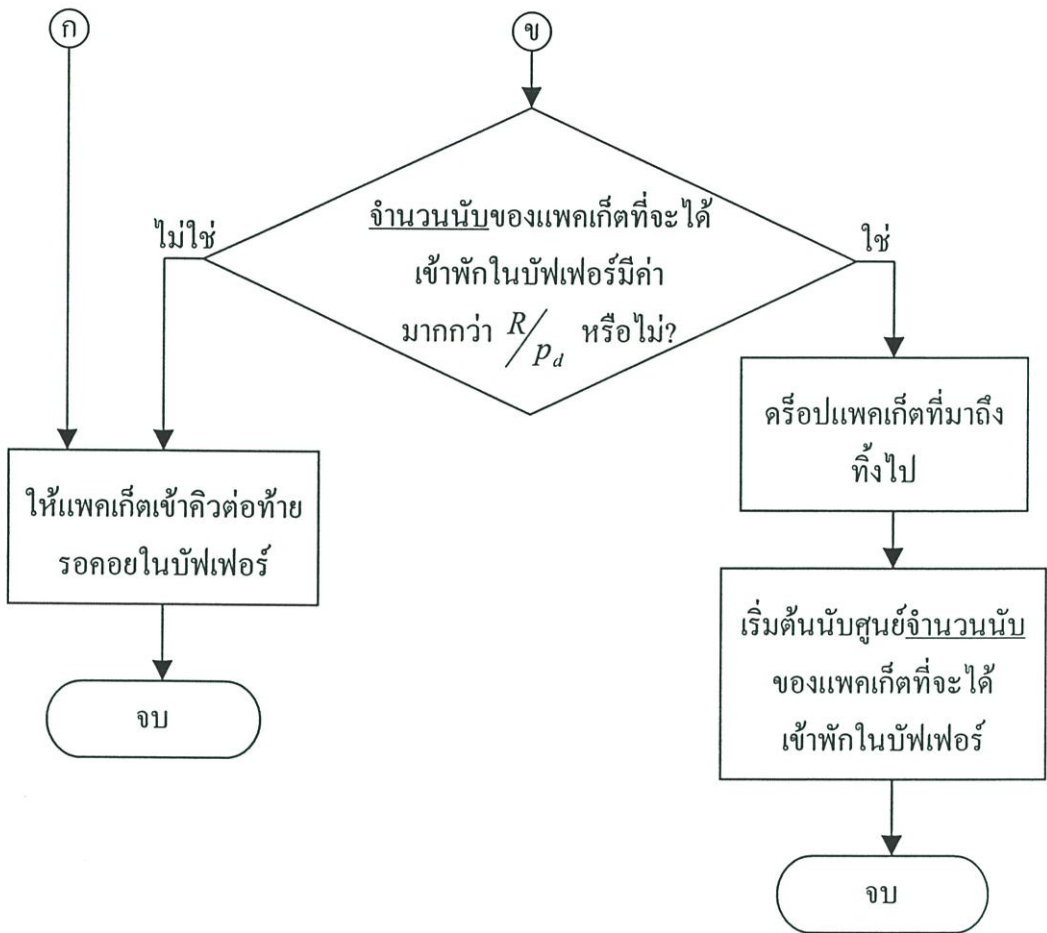
RED เป็นกลไกที่ถูกนำเสนอครั้งแรกโดย [1] และถูกนำเสนอโดย [2] ให้เป็นทางเลือกหนึ่งสำหรับการจัดการกับความคับคั่ง (Congestion) ในระบบเครือข่าย โดยการครีโปกแพกเก็ตล่วงหน้า

ก่อนที่ปัญหาจะเกิด และด้วยเทคนิคนี้ แพคเกจที่มาถึงจะถูกจัดเรียงล่วงหน้าก่อนที่บัฟเฟอร์จะเต็ม เพื่อที่จะปรับปรุงประสิทธิภาพในการให้บริการของเครือข่าย เทคนิคดังกล่าวนี้ถูกออกแบบเพื่อใช้กับคิวเดี่ยวแบบ FIFO (Single FIFO queue) ดังนั้นจึงใช้ได้กับสถาปัตยกรรมเครือข่ายได้ทุกแบบ ซึ่งในรายละเอียดของสถาปัตยกรรมรวมการให้บริการ (Integrated Services Architecture; ISA) นั้นการครอบแพคเกจล่วงหน้า สามารถกระทำกับคิวเพียงหนึ่งเดียวหรือหลายคิว ซึ่งใช้ได้สำหรับการจราจรข้อมูลที่อาจมีการเปลี่ยนแปลงได้ตลอดเวลา เพื่อปรับปรุงประสิทธิภาพของเครือข่ายในการรองรับการจราจรข้อมูลเหล่านั้น วัตถุประสงค์หลักตามที่นำเสนอโดย [1] ได้แก่

- 1) ป้องกันการเกิดการคับคั่งของข้อมูล (Congestion Avoidance) : กลไกแบบ RED ถูกออกแบบมาเพื่อป้องกันการเกิดการคับคั่งมากกว่าที่จะแก้ไข ดังนั้น RED ต้องตรวจจับอาการบ่งบอกว่าจะเกิดการคับคั่งให้ได้เพื่อป้องกันเครือข่ายจากการเกิดการประวิงเวลา และการดรอปสูง
- 2) ป้องกันการเกิดโกลบอลซิงโครไนเซชัน (Global Synchronization Avoidance) : เมื่อเริ่มมีอาการบ่งบอกว่าจะเกิดการคับคั่ง เกตเวย์ต้องตัดสินใจว่าโหนดใด หรือกลุ่มโหนดใด ควรถูกเตือนให้ลดอัตราการส่ง ในปัจจุบันจะใช้อัตราเตือนโดยนัย (Implicit) โดยการครอบแพคเกจ ซึ่งจะสามารถป้องกันการเกิดโกลบอลซิงโครไนเซชันได้โดยการตรวจจับการเกิดการคับคั่ง และเตือนล่วงหน้าเฉพาะกลุ่มโหนดเท่าที่จำเป็น
- 3) ป้องกันการเกิดผลกระทบเชิงลบจากการที่มีข้อมูลมาเป็นระลอก : อาการที่จะเกิดการคับคั่งมีแนวโน้มว่าจะเกิดจากกลุ่มแพคเกจที่มาถึงติดต่อกันเป็นระลอกต่อเนื่อง อาจมาจากโหนดเพียงหนึ่งเดียว หรือเป็นกลุ่ม การมาถึงแบบเป็นระลอกแบบนี้จะเพิ่มภาระหนักในการใช้บัฟเฟอร์ เมื่อเปรียบเทียบกับโหนดต้นทางที่กำเนิดแพคเกจแบบเกลี่ย (Smooth Traffic Source) ที่อัตราเฉลี่ยการสร้างทราฟฟิกเดียวกัน ถ้ามีเฉพาะกลุ่มที่มาถึงนี้เท่านั้นที่ถูกครอบ ก็จะกลายเป็นว่ากลไกนี้ก่อผลกระทบเชิงลบกับโหนดต้นทางของกลุ่มแพคเกจที่มาถึงติดต่อกันเป็นระลอกต่อเนื่อง (Burst Traffic Source)
- 4) เพื่อกำหนดขอบเขตของขนาดคิวเฉลี่ยได้ : กลไกแบบ RED จะต้องสามารถที่จะควบคุมการใช้คิวโดยเฉลี่ย ซึ่งจะควบคุมค่าประวิงเวลาเฉลี่ยได้ด้วย



รูปที่ 3.2 อัลกอริทึมการทำงานของกลไกแบบ RED



รูปที่ 3.2 (ต่อ)

### 3.2.2 ลักษณะการทำงานของกลไกแบบ RED

กลไกแบบ RED มีการทำงานที่แสดงได้ดังอัลกอริทึมในรูป 3.2 ซึ่งแตกต่างจากกลไกแบบ Tail Drop ตรงที่มีการตรวจสอบการใช้บัฟเฟอร์อยู่ตลอดเวลา และมีการตอบสนองล่วงหน้าเพื่อป้องกันการเกิดบัฟเฟอร์ล้น ลักษณะการตอบสนองมีสองรูปแบบ ได้แก่

- 1) การบังคับครีอปหรือไม่ยอมให้แพ็คเกจเข้าคิวในบัฟเฟอร์ เพื่อลดอัตราการมาถึงของแพ็คเกจที่เกิดแคว้น นอกจากนี้ยังทำให้โหนดต้นทางชนิดที่มีการตอบสนอง (Responsive Source Node) ลดอัตราการส่งข้อมูล ด้วยกลไกการป้อนกลับแบบอิมพลีซิที (Implicit Feedback) และการทำงานของกลไกควบคุมการส่ง อย่างเช่น กลไก TCP Window

- 2) การทำเครื่องหมาย (Mark) เป็นการตอบสนองในรูปแบบที่โหนดต้นทาง หรือโหนดที่เกี่ยวข้องต้องสนับสนุนได้ด้วย แพคเกจที่มาถึงจะได้รับอนุญาตให้เข้าพักรอในคิวในบัฟเฟอร์ได้เหมือนปกติ แต่จะมีการทำเครื่องหมายที่บิตส่วนหัวของแพคเกจข้อมูลเพื่อให้โหนดต้นทางและโหนดข้างเคียงที่เป็นทางผ่านทราบถึงภาวะที่บัฟเฟอร์ใกล้เต็ม ซึ่งกลไกย่อยที่ใช้สนับสนุนการนี้ ได้แก่ กลไกแบบ Explicit Congestion Notification (ECN) เป็นต้น หรืออีกวิธีหนึ่งคือ การใช้กลไกแบบแรงดันย้อนกลับ (Back Pressure) ซึ่งกลไกย่อยที่ใช้สนับสนุนการนี้ ได้แก่ การใช้ ICMP Source Quench ส่งสัญญาณไปเตือนโหนดต้นทางเพื่อให้ทำการลดอัตราการส่งข้อมูลลง เป็นต้น

แต่ในงานวิจัยในวิทยานิพนธ์นี้จะเลือกใช้การตอบสนองในลักษณะที่มีการบังคับครีอป ดังนั้นผลการวิเคราะห์จะเป็นผลจากการบังคับครีอปเท่านั้น

ข้อแตกต่างของกลไกแบบ Tail Drop และ RED อีกประการหนึ่งคือ วิธีการติดตามการใช้บัฟเฟอร์ กลไกแบบ Tail Drop จะพิจารณาจากขนาดคิวขณะใดๆ แต่กลไก RED จะพิจารณาจากขนาดคิวเฉลี่ย ซึ่งขนาดคิวเฉลี่ยจะถูกคำนวณทุกครั้งที่มีแพคเกจเข้ามา การคำนวณขนาดคิวเฉลี่ยจะเป็นไปตามสมการที่ (3.10)

$$\hat{q} = \begin{cases} (1-w)\hat{q} + wq & ; \quad q \neq 0 \\ (1-w)^\mu \hat{q} & ; \quad otherwise \end{cases} \quad (3.10)$$

โดยที่

- $\hat{q}$  : ขนาดคิวเฉลี่ยหรือจำนวนแพคเกจเฉลี่ยในบัฟเฟอร์ของเกตเวย์ (Average Queue Size)  
 $q$  : ขนาดคิวหรือจำนวนแพคเกจปัจจุบันในบัฟเฟอร์ของเกตเวย์ (Instantaneous Queue Size)  
 $w$  : น้ำหนักถ่วงในการคำนวณขนาดคิวเฉลี่ย (Queue Weight) โดยที่  $0 \leq w \leq 1$   
 $\lambda$  : อัตราเฉลี่ยการมีแพคเกจเข้ามาของเกตเวย์ (Arrival Rate) ในหน่วยแพคเกจต่อวินาที  
 $\mu$  : อัตราเฉลี่ยการให้บริการของเกตเวย์ (Service Rate) ในหน่วยแพคเกจต่อวินาที

ขนาดคิวเฉลี่ยจะมีผลต่อการตัดสินใจในการครีอปแพคเกจที่เพิ่งมาใหม่ โดยการใช้ค่าที่ได้จากฟังก์ชันการครีอปเป็นตัวกำหนดการเกิดความน่าจะเป็นที่แพคเกจจะถูกครีอป แบ่งได้ 3 กรณีดัง 3.2.2.1 ถึง 3.2.2.3 ดังนี้

### 3.2.2.1 ในกรณีที่ขนาดคิวเฉลี่ยยังไม่เกินค่ากำหนดขั้นต่ำ (Minimum Threshold)

ทุกแพ็คเกจที่มาถึงจะได้รับอนุญาตให้เข้าคิวพักในบัฟเฟอร์เพื่อรอให้บริการต่อไป ค่ากำหนดขั้นต่ำดังกล่าวสัมพันธ์กับการกำหนดน้ำหนักถ่วงในการคำนวณขนาดคิวเฉลี่ย ( $w$ ) ดังนี้

$$L + 1 + \frac{(1-w)^{L+1} - 1}{w} < k_l \quad (3.11)$$

โดยที่

$k_l$  : ค่ากำหนดขั้นต่ำ (Minimum Threshold) ของจำนวนแพ็คเกจที่เก็บพักในบัฟเฟอร์

$L$  : จำนวนแพ็คเกจที่มาถึงแบบเบิร์สต์ (Burst) โดยไม่ทำให้ขนาดคิวเฉลี่ยสูงเกินค่ากำหนดขั้นต่ำ โดยสมมติว่าไม่มีแพ็คเกจรอคอยในคิวในบัฟเฟอร์ขณะที่หัวขบวนของแพ็คเกจแบบเบิร์สต์มาถึง

### 3.2.2.2 ในกรณีขนาดคิวเฉลี่ยเกินค่ากำหนดขั้นต่ำ (Minimum Threshold) แต่ยังไม่เกินค่ากำหนดขั้นสูง (Maximum Threshold)

แพ็คเกจที่มาถึงจะถูกครอบแบบสุ่ม โดยที่การครอบจะเกิดขึ้นทุกๆครั้งที่จำนวนแพ็คเกจที่ผ่านมาเข้าคิวในบัฟเฟอร์ได้จำนวนหนึ่งซึ่งเป็นจำนวนสุ่มที่มีการกระจายแบบยูนิฟอร์ม (Uniform Distribution) ถ้า  $X$  เป็นจำนวนสุ่มดังกล่าว ความน่าจะเป็นที่จะมีจำนวนแพ็คเกจผ่านเข้าคิวเท่ากับ  $x$  แสดงได้ดังนี้

$$\Pr[X = x] = \begin{cases} \frac{p_d(\hat{q})}{1 - (x-1)p_d(\hat{q})} \prod_{i=0}^{x-2} \left(1 - \frac{p_d(\hat{q})}{1 - ip_d(\hat{q})}\right) & ; \quad 1 \leq x \leq \frac{1}{p_d(\hat{q})} \\ 0 & ; \quad x > \frac{1}{p_d(\hat{q})} \end{cases} \quad (3.12)$$

และมีค่าเฉลี่ยเท่ากับ

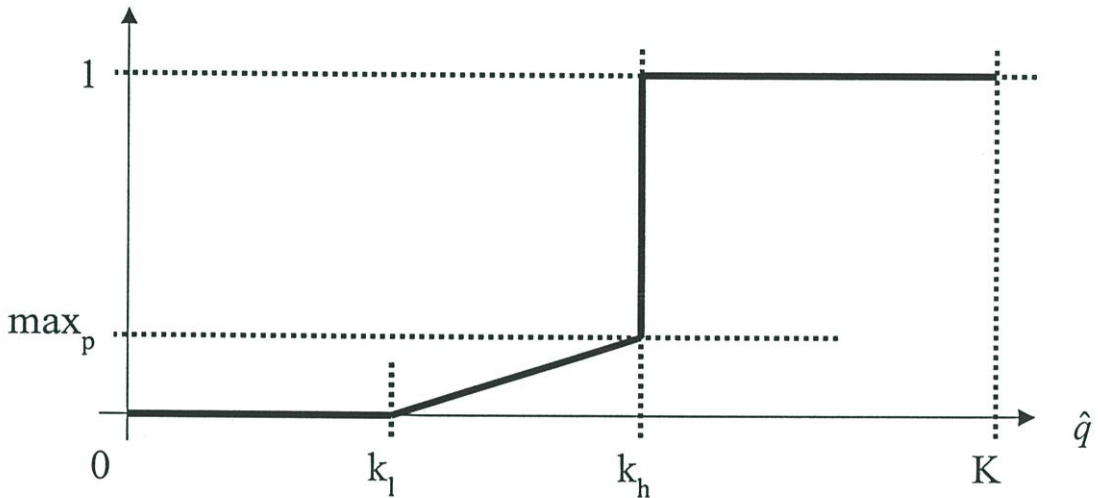
$$E[X] = \frac{1}{2p_d(\hat{q})} + \frac{1}{2} \quad (3.13)$$

โดยที่  $p_d(\hat{q})$  เป็นค่าของฟังก์ชันการครอบของกลไกแบบ RED โดยที่  $0 \leq p_d(\hat{q}) \leq 1$

### 3.2.2.3 ในกรณีขนาดคิวเฉลี่ยเกินค่ากำหนดขั้นสูง (Maximum Threshold)

ทุกแพคเกจที่มาถึงจะถูกครีโปกด้วยความเป็นเท่ากัป 1 ซึ่งหมายความว่าทุกๆ แพคเกจที่มาถึงจะถูกครีโปกหมด

ซึ่งฟังก์ชันการครีโปก ( $p_d(\hat{q})$ ) ของกลไกแบบ RED แสดงได้ดังรูป 3.3 และสมการ (3.14)



รูปที่ 3.3 แสดงฟังก์ชันการครีโปก ( $p_d(\hat{q})$ ) ของกลไกแบบ RED

$$p_d(\hat{q}) = \begin{cases} 0 & ; \hat{q} < k_l \\ \frac{\hat{q} - k_l}{k_h - k_l} \times \max_p & ; k_l \leq \hat{q} < k_h \\ 1 & ; \hat{q} \geq k_h \end{cases} \quad (3.14)$$

โดยที่

$\hat{q}$  : ขนาดคิวเฉลี่ยในบัฟเฟอร์ของเกตเวย์ (Average Queue Size)

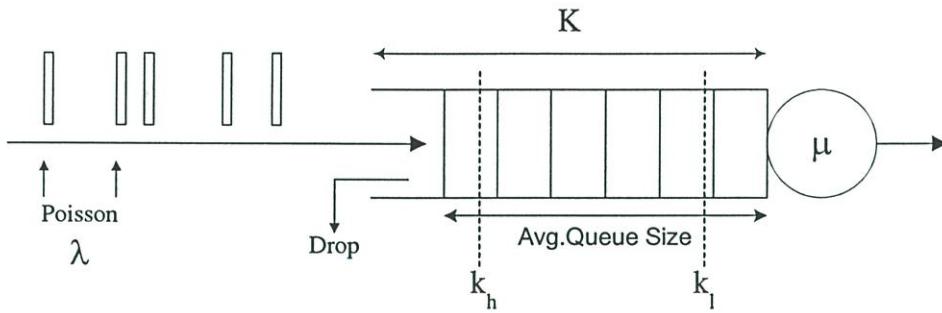
$p_d(\hat{q})$  : ความเป็นที่แพคเกจจะถูกครีโปก เมื่อขนาดคิวเฉลี่ยเท่ากับ  $\hat{q}$

$\max_p$  : ความเป็นสูงสุดที่แพคเกจจะถูกครีโปกโดยกลไก

$k_l$  : ค่ากำหนดขั้นต่ำ (Minimum Threshold) ของจำนวนแพคเกจที่เก็บพักในบัฟเฟอร์

$k_h$  : ค่ากำหนดขั้นสูง (Maximum Threshold) ของจำนวนแพคเกจที่เก็บพักในบัฟเฟอร์

$K$  : ขนาดบัฟเฟอร์



รูปที่ 3.4 แบบจำลองขนาดคิวเฉลี่ยในบัฟเฟอร์

### 3.2.3 แบบจำลองในการวิเคราะห์ประสิทธิภาพ

จากแบบจำลองคิวในรูปที่ 3.4 และสมมติฐานในการวิเคราะห์เช่นเดียวกับในการวิเคราะห์กลไกแบบ Tail Drop กำหนดให้เกตเวย์มีบัฟเฟอร์ขนาดความจุ  $K$  แพคเกจ การมาขอใช้บริการผ่านเกตเวย์เป็นไปตามกระบวนการปัวร์ซอง (Poisson Process) มีอัตราเฉลี่ยการมาขอผ่านเท่ากับ  $\lambda$  แพคเกจต่อวินาที และเวลาที่ใช้ในการให้บริการเป็นค่าสุ่มที่มีการกระจายแบบเอ็กซ์โปเนนเชียล (Exponential Distribution) โดยมีอัตราเฉลี่ยการให้บริการเท่ากับ  $\mu$  แพคเกจต่อวินาที ทำให้เขียนสมการของการกระจายของขนาดคิวเฉลี่ยในบัฟเฟอร์ได้จากสมการ (2.19) เป็นดังสมการ (3.15)

$$\pi(\hat{q}) = \frac{\left(\frac{\lambda}{\mu}\right)^{\hat{q}} \prod_{i=0}^{\hat{q}-1} (1 - p_d(i))}{1 + \sum_{j=1}^K \left(\frac{\lambda}{\mu}\right)^j \prod_{i=0}^{j-1} (1 - p_d(i))} \quad (3.15)$$

ค่าความน่าจะเป็นที่จะเกิดการครีอปที่เกตเวย์สามารถหาได้จากสมการ (2.20) โดยใช้ความน่าจะเป็นที่แพคเกจจะถูกครีอปจากสมการ (3.14) จะได้ความน่าจะเป็นที่จะเกิดการครีอปที่เกตเวย์ที่ใช้กลไกการจัดการคิวแบบ RED ดังสมการที่ (3.16) และจากสมการ (2.21) ถ้า  $S$  คือ ทราฟฟิคที่ออกจากเกตเวย์จะคำนวณได้จากสมการที่ (3.17)

$$P_{drop} = \sum_{\hat{q}=k_h}^K \pi(\hat{q}) + \sum_{\hat{q}=k_l}^{k_h-1} (p_d(\hat{q}) \times \pi(\hat{q})) \quad (3.16)$$

$$S = (1 - P_{drop})\lambda \quad (3.17)$$

ค่าประวิงเวลาจากการรอคอยในคิว (Queuing Delay) สามารถหาได้จากสมการ (3.18) ซึ่งเป็นยกมาแสดงซ้ำอีกทีจากสมการ (2.22) โดยคำนวณค่าเฉลี่ยทางคณิตศาสตร์ของขนาดคิวเฉลี่ย  $E(\hat{Q})$  หาได้จากสมการ (2.23) ซึ่งค่าเฉลี่ยของขนาดคิวเฉลี่ยของกลไก RED แสดงในสมการ (3.19)

$$D = \frac{E(\hat{Q})}{(1 - P_{drop})\lambda} - \frac{1}{\mu} \quad (3.18)$$

$$E(\hat{Q}) = \frac{\sum_{k=0}^K k \left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1 - p_d(\hat{q}))}{\sum_{k=0}^K \left(\left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1 - p_d(\hat{q}))\right)} \quad (3.19)$$

โดยที่

- $D$  : ค่าประวิงเวลาจากการเข้าคิวรอคอย (Queuing Delay) ในหน่วยวินาที
- $\hat{Q}$  : ตัวแปรสุ่มขนาดคิวเฉลี่ยในบัฟเฟอร์ของเกตเวย์ (Average Queue Size)
- $E(\hat{Q})$  : ค่าเฉลี่ยทางคณิตศาสตร์ของขนาดคิวเฉลี่ย (Mean of Average Queue Size)
- $P_{drop}$  : ความน่าจะเป็นที่จะมีการครี้อปเกิดขึ้นที่เกตเวย์ (Drop Probability)
- $\lambda$  : อัตราเฉลี่ยการมีแพคเกจเข้ามาของผ่านเกตเวย์ (Arrival Rate) ในหน่วยแพคเกจต่อวินาที
- $\mu$  : อัตราเฉลี่ยการให้บริการของเกตเวย์ (Service Rate) ในหน่วยแพคเกจต่อวินาที

การวิเคราะห์ในประเด็นเกี่ยวกับการเกิดแพคเกจครี้อปติดต่อกัน (Consecutive Drops) ตามที่ได้เกริ่นไว้ถึงความจำเป็นที่ต้องพิจารณาในตอนท้ายของบทที่ 2 สามารถวิเคราะห์ได้ในรูปแบบความน่าจะเป็นที่จะมีการครี้อปต่อเนื่องกัน และค่าเฉลี่ยของจำนวนแพคเกจที่จะถูกครี้อปติดต่อกัน

ถ้า  $p_d(\hat{q}_k)$  คือ ค่าความน่าจะเป็นที่แพคเกจที่มาถึงในลำดับที่  $k$  จะถูกครี้อป และ  $p_d(\hat{q}_{k+1})$  คือ ค่าความน่าจะเป็นที่แพคเกจที่มาถึงในลำดับที่  $k+1$  จะถูกครี้อป ส่วน  $\hat{q}_k$  และ  $\hat{q}_{k+1}$  เป็นขนาดคิวเฉลี่ยในขณะที่แพคเกจลำดับที่  $k$  และ  $k+1$  มาถึงตามลำดับ จะได้ว่า  $\hat{q}_{k+1} \geq \hat{q}_k$  แต่  $\hat{q}_{k+1} - \hat{q}_k$  จะมากน้อยเท่าไรขึ้นกับจำนวนแพคเกจที่ได้รับการบริการในระหว่างการมาถึงในลำดับถัดกันของคู่หนึ่งๆ เมื่อพิจารณาจากความเป็นจริงในการเกิดแพคเกจครี้อปติดต่อกันอย่างต่อเนื่อง แพคเกจที่มาถึงในลำดับถัดกันจะถูกครี้อปด้วยค่าความน่าจะเป็นที่แตกต่างกันน้อยมาก โดยเฉพาะอย่างยิ่งในช่วงที่การป้อนภาระ (Offered Load) ที่สูง เพราะช่วงเวลาในการมาถึงจะสั้นกว่าช่วงเวลาที่จะมีแพคเกจลำดับถัดไปได้รับการบริการ และการเพิ่มขึ้นที่เล็กน้อยของค่าของฟังก์ชันการครี้อป

ดังนั้นจะสามารถถือได้ว่าค่าความน่าจะเป็นที่จะถูกครีโปกของคู่ที่มีลำดับติดกันใดๆมีค่าเท่ากัน เพื่อสามารถนำไปประมาณค่าขอบเขตบน (Upper Bound) ของจำนวนแพกเก็ตที่จะถูกครีโปกติดต่อกันในลำดับต่อเนื่องกัน

ถ้า  $\pi(\cdot | drop)$  คือ การกระจายของขนาดคิวโดยมีเงื่อนไขต่อการเกิดการครีโปก และสมมติว่า  $\pi(K | drop)$  มีค่าน้อยมาก ความน่าจะเป็นที่จะเกิดการครีโปกต่อเนื่องแสดงได้ดังนี้

$$\forall n \geq 0, P_{drop}^{cons}(N > n) = \sum_{\hat{q}=0}^{K-1} \pi(\hat{q} | drop) p_d(\hat{q})^n \quad (3.20-a)$$

จากกฎของเบย์ (Bayes 's Formula) จะได้ว่า

$$\pi(\hat{q} | drop) = \frac{\pi(\hat{q})}{P_{drop}} p_d(\hat{q}) \quad (3.20-b)$$

ถ้านิยามให้  $N$  เป็นตัวแปรสุ่มของจำนวนแพกเก็ตที่ถูกครีโปกติดต่อกันอย่างต่อเนื่อง และ  $n$  เป็นค่าของ  $N$  เมื่อเกิดการครีโปกขึ้นมาครั้งหนึ่ง ความน่าจะเป็นที่แพกเก็ตจะถูกครีโปกติดต่อกันอย่างต่อเนื่องอีก  $n$  แพกเก็ต สามารถแสดงได้ดังสมการ (3.21)

$$\forall n \geq 0, P_{drop}^{cons}(N > n) = \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) (p_d(\hat{q}))^{n+1}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) p_d(\hat{q})} \quad (3.21)$$

จากสมการ (3.21) สามารถหาค่าเฉลี่ยทางคณิตศาสตร์ (Arithmetic Mean) ของจำนวนแพกเก็ตที่จะถูกครีโปกติดต่อกันได้จากสมการที่ (3.22)

$$E(N) = 1 + \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) \frac{(p_d(\hat{q}))^2}{1-p_d(\hat{q})}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) p_d(\hat{q})} \quad (3.22)$$

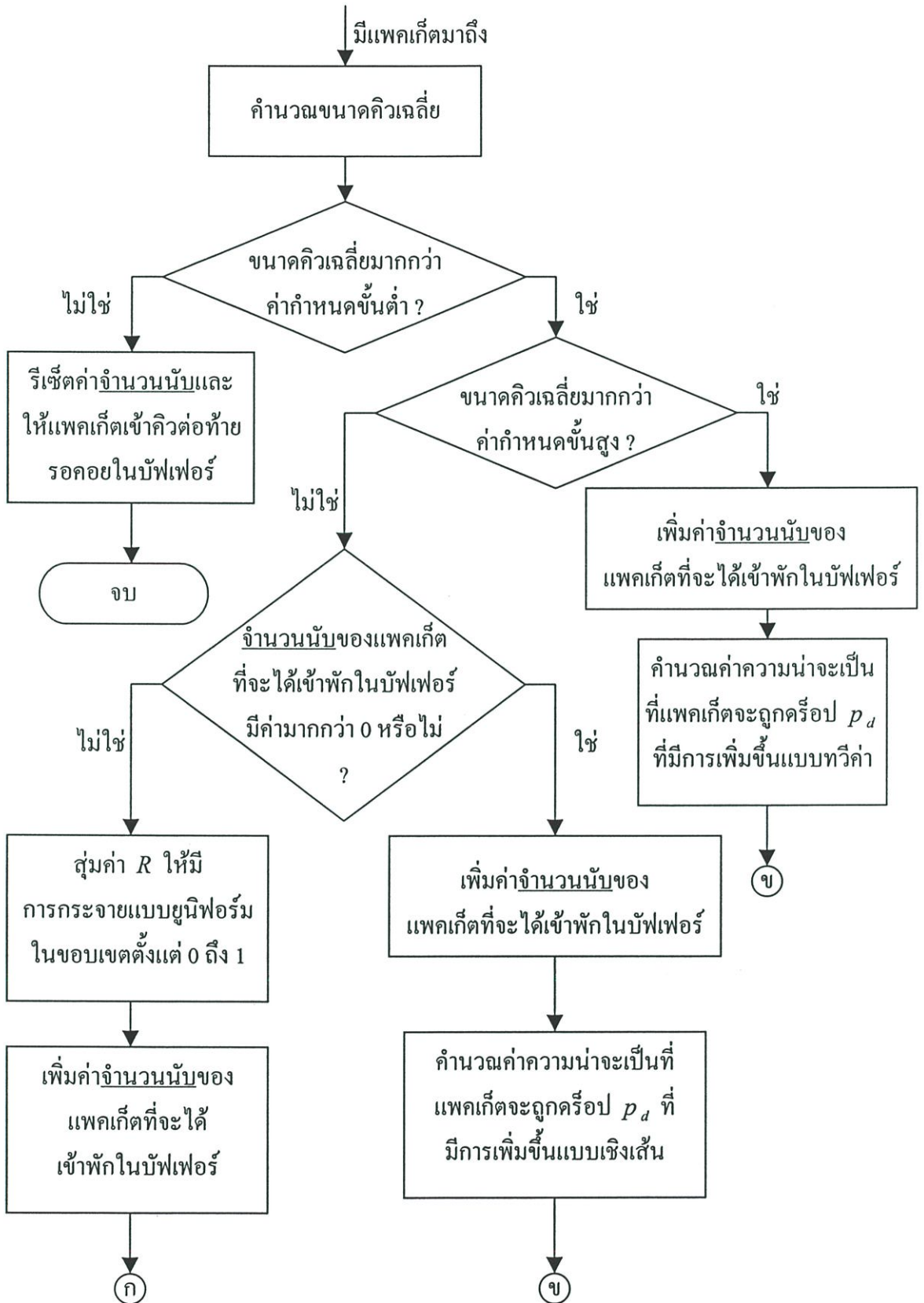
## กลไกการจัดการคิวในบัฟเฟอร์ของเกตเวย์แบบ ExRED

จากฟังก์ชันการครีอปของกลไกแบบ RED ในสมการที่ (3.14) เมื่อขนาดคิวเฉลี่ยสูงเกินค่ากำหนดขั้นสูง จะทำให้ทุกแพ็คเกจที่เข้ามาใหม่ถูกรีอปทั้งหมด แม้ว่าการครีอปทั้งหมดจะสามารถป้องกันการเกิดปัญหาบัฟเฟอร์แต่ก็ทำให้เกิดการครีอปแพ็คเกจมากเกินความจำเป็น ซึ่งทำให้กลไกแบบ RED มีทรูพุตต่ำดั่งที่กล่าวถึงในทงงานวิจัยก่อนหน้านี้ที่ได้มีการศึกษาเกี่ยวกับประสิทธิภาพของ RED มาแล้ว เช่น [4] เป็นต้น ดังนั้นการวิจัยของวิทยานิพนธ์นี้จึงได้เสนอกลไกการจัดการคิวในบัฟเฟอร์ของเกตเวย์แบบ ExRED [6] เพื่อแก้ไขปัญหาดังกล่าว

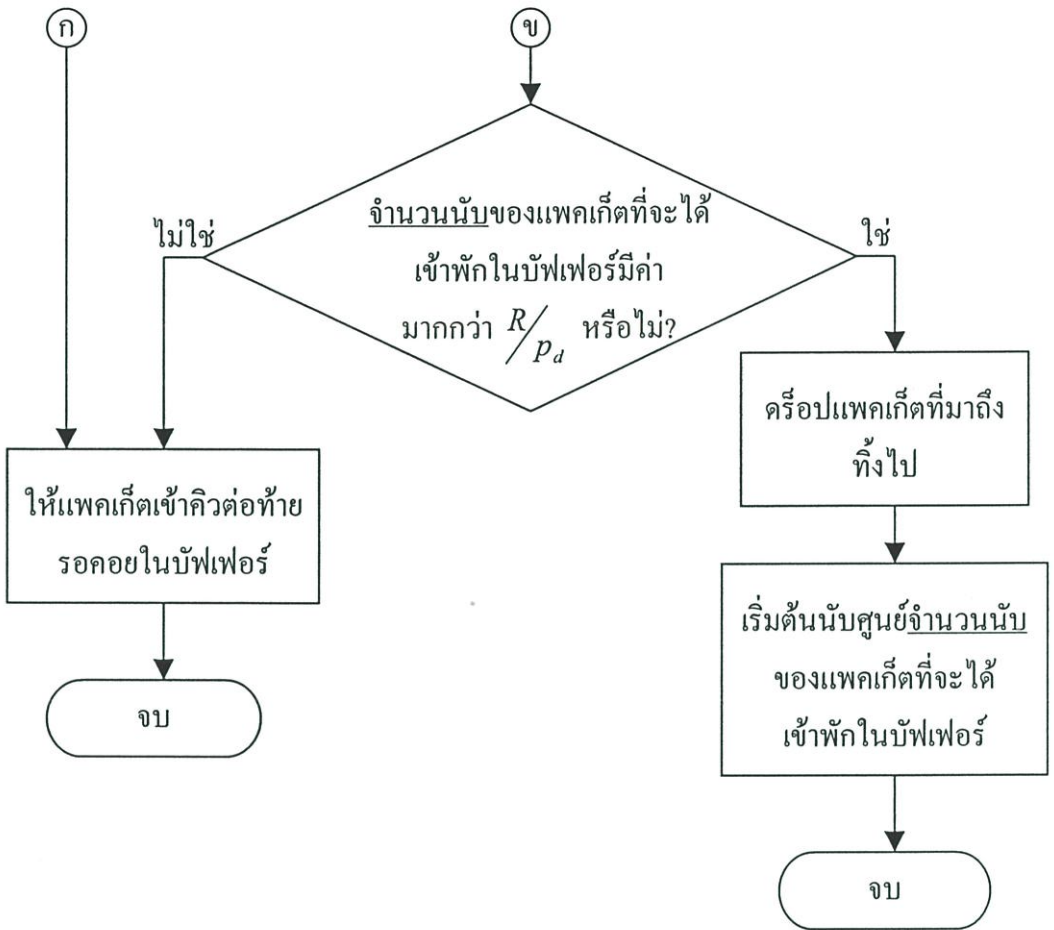
### 4.1 พื้นฐานของกลไกจัดการคิวแบบ Extended Drop Slope RED (ExRED)

ลักษณะการทำงานของกลไกแบบ ExRED เป็นดังรูป 4.1 โดยมีหลักการทำงานเบื้องต้นเหมือนกลไกแบบ RED คือ ตรวจสอบการใช้งานบัฟเฟอร์โดยการคำนวณขนาดคิวเฉลี่ย ซึ่งขนาดคิวเฉลี่ยจะถูกคำนวณมาจากค่าขนาดคิวปัจจุบันที่วัดได้ขณะมีแพ็คเกจเข้ามาขอใช้บริการผ่านเกตเวย์ ดังสมการที่ (3.10) แต่มีการปรับปรุงในส่วนที่ทำหน้าที่ตัดสินใจและการกำหนดค่าความน่าจะเป็นของการครีอปแพ็คเกจที่เพิ่งจะมาถึงในขณะนั้น สำหรับกลไกแบบ ExRED จะมีการเพิ่มเติมแก้ไขตรงที่ให้มีการเพิ่มขึ้นของความน่าจะเป็นที่แพ็คเกจจะถูกครีอป ต่อจากส่วนที่มีการเพิ่มขึ้นแบบเป็นเชิงเส้นออกไปอีก กล่าวคือ เมื่อขนาดคิวเฉลี่ยสูงกว่าค่ากำหนดขั้นสูง แทนที่จะให้มีการครีอปแพ็คเกจทั้งหมดในทีเดียวเหมือนกลไกแบบ RED แต่จะให้ความน่าจะเป็นที่แพ็คเกจจะถูกครีอปมีอัตราการเพิ่มขึ้นแปรตามการเพิ่มขึ้นของขนาดคิวเฉลี่ย และเพื่อให้สอดคล้องกับวัตถุประสงค์ในการป้องกันการเกิดบัฟเฟอร์ล้น ความน่าจะเป็นดังกล่าวจะเพิ่มขึ้นแบบทวีค่า โดยประยุกต์ฟังก์ชันโพลิโนเมียลกำลังสองกับฟังก์ชันการครีอป รูปที่ 4.2 แสดงกราฟของฟังก์ชันการครีอปดังกล่าว

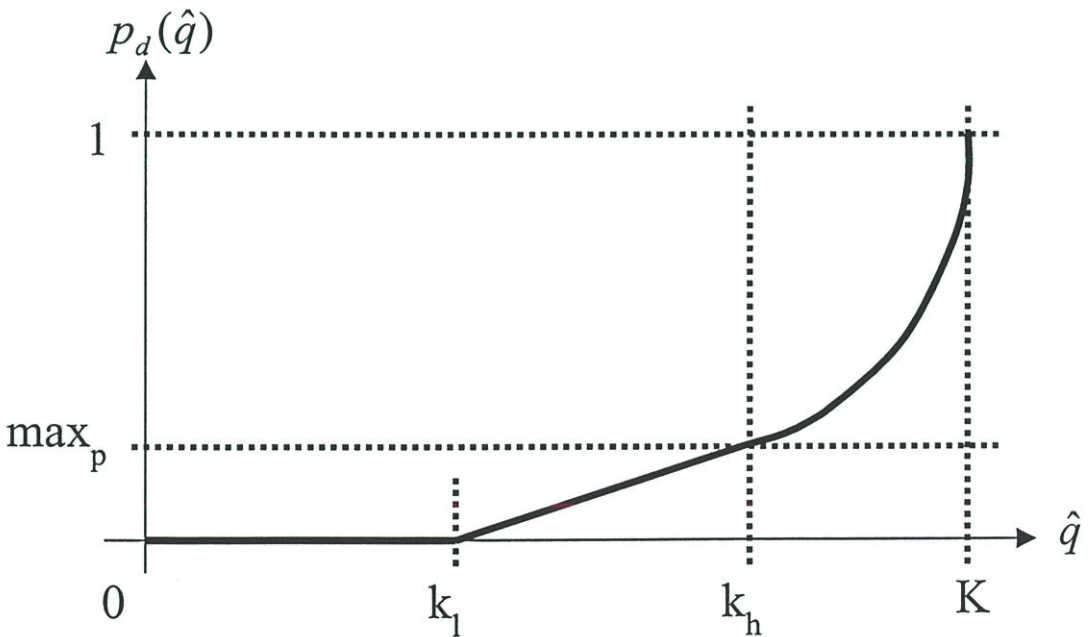
จากคุณสมบัติการทวีค่าของฟังก์ชัน โพลิโนเมียลกำลังสอง ทำให้เมื่อขนาดคิวเฉลี่ยสูงเกินค่ากำหนดขั้นสูง ความน่าจะเป็นที่แพ็คเกจที่เพิ่งมาใหม่จะถูกครีอปจะสูงขึ้นแบบทวีค่า ถ้าเนื้อที่ว่างในบัฟเฟอร์ยิ่งลดลง ความน่าจะเป็นที่จะถูกครีอปก็จะยิ่งเพิ่มด้วยอัตราเพิ่มที่สูงขึ้น และความน่าจะเป็นดังกล่าวจะสูงขึ้นจนเท่ากับ 1 นั่นคือทุกแพ็คเกจที่มาใหม่จะถูกครีอปทั้งหมดเมื่อที่บัฟเฟอร์ไม่มีที่ว่างที่จะรองรับแพ็คเกจที่มาใหม่ได้อีกต่อไป ซึ่งสมการ (4.1) แสดงรูปทั่วไปของฟังก์ชันการครีอปสัมประสิทธิ์  $a_2$ ,  $a_1$  และ  $a_0$  หาได้จากสมการเงื่อนไขขอบเขตดังกล่าวถึงในหัวข้อย่อยต่อไป



รูปที่ 4.1 อัลกอริทึมการทำงานของกลไกแบบ ExRED



รูปที่ 4.1 (ต่อ)



รูปที่ 4.2 ฟังก์ชันการครีอปของกลไกแบบ ExRED

$$p_d(\hat{q}) = \begin{cases} 0 & ; \hat{q} < k_l \\ \frac{\hat{q} - k_l}{k_h - k_l} \times \max_p & ; k_l \leq \hat{q} < k_h \\ a_2 \hat{q}^2 + a_1 \hat{q} + a_0 & ; \hat{q} \geq k_h \end{cases} \quad (4.1)$$

โดยที่

- $\hat{q}$  : ขนาดคิวเฉลี่ยหรือจำนวนแพคเกจเฉลี่ยในบัฟเฟอร์ของเกตเวย์
- $p_d(\hat{q})$  : ความน่าจะเป็นที่แพคเกจจะถูกครีโปก เมื่อขนาดคิวเฉลี่ยมีค่าเท่ากับ  $\hat{q}$
- $\max_p$  : ความน่าจะเป็นสูงสุดที่แพคเกจจะถูกครีโปกโดยกลไก
- $k_l$  : ค่ากำหนดขั้นต่ำของจำนวนแพคเกจที่ถูกเก็บพักในบัฟเฟอร์
- $k_h$  : ค่ากำหนดขั้นสูงของจำนวนแพคเกจที่ถูกเก็บพักในบัฟเฟอร์

## 4.2 การหาค่าสัมประสิทธิ์ของสมการฟังก์ชันการครีโปก

การหาเงื่อนไขขอบเขตเพื่อให้ได้ฟังก์ชันการครีโปกดังรูปที่ 4.2 สามารถหาได้จากการกำหนดเงื่อนไข 3 ข้อดังนี้

- 1) เมื่อขนาดคิวเฉลี่ยมีค่าเท่ากับค่ากำหนดขั้นสูง กำหนดให้แพคเกจที่มาถึงถูกครีโปกด้วยความน่าจะเป็นเท่ากับค่าความน่าจะเป็นสูงสุดที่แพคเกจจะถูกครีโปกโดยกลไก ( $\max_p$ )
- 2) เมื่อขนาดคิวเฉลี่ยมีค่าเกินค่ากำหนดขั้นสูง กำหนดให้แพคเกจที่มาถึงถูกครีโปกด้วยค่าความน่าจะเป็นที่มีอัตราการเพิ่มขึ้นแบบทวีคูณ และมีความต่อเนื่องทุกๆค่าจากจุดที่ขนาดคิวเฉลี่ยมีค่าเท่ากับค่ากำหนดขั้นสูง จนถึงจุดที่บัฟเฟอร์ไม่มีที่ว่างเหลืออยู่
- 3) เมื่อบัฟเฟอร์ไม่มีที่ว่างเหลืออยู่ กำหนดให้แพคเกจที่มาถึงถูกครีโปกด้วยค่าความน่าจะเป็นเท่ากับ 1 คือ ทุกแพคเกจที่มาถึงจะถูกครีโปกทั้งหมด

จากเงื่อนไขทั้งสามข้างต้นนี้ สามารถเขียนเป็นสมการเงื่อนไขขอบเขตได้ดังนี้ตามลำดับ

$$p_d(k_h) = \max_p \quad (4.2)$$

$$\left. \frac{d}{d\hat{q}} p_d(\hat{q}) \right|_{\hat{q}=k_h} = \frac{\max_p}{k_h - k_l} \quad (4.3)$$

$$p_d(K) = 1 \quad (4.4)$$

จากสมการเงื่อนไขขอบเขตทั้งสามข้างต้นนี้ จะเขียนเป็นสมการได้ดังนี้ตามลำดับ

$$k_h^2 a_2 + k_h a_1 + a_0 = \max_p \quad (4.5)$$

$$2k_h a_2 + a_1 = \frac{\max_p}{k_h - k_l} \quad (4.6)$$

$$K^2 a_2 + K a_1 + a_0 = 1 \quad (4.7)$$

เมื่อแก้สมการทั้งสามนี้ก็จะได้ค่าสัมประสิทธิ์ทั้งสามค่าออกมา ฟังก์ชันการตอบแยกจากสมการที่ (4.1) มาเขียนซ้ำได้เป็น

$$p_d(\hat{q}) = \begin{cases} 0 & ; \quad \hat{q} < k_l \\ \frac{\hat{q} - k_l}{k_h - k_l} \times \max_p & ; \quad k_l \leq \hat{q} < k_h \\ a_2 \hat{q}^2 + a_1 \hat{q} + a_0 & ; \quad \hat{q} \geq k_h \end{cases}$$

โดยที่

$$a_2 = \frac{(k_h - k_l) - (K - k_l) \max_p}{(k_h - k_l)(K - k_h)^2}$$

$$a_1 = \frac{(k_h^2 + K^2 - 2k_h k_l) \max_p - 2k_h^2 + 2k_h k_l}{(k_h - k_l)(K - k_h)^2}$$

$$a_0 = \frac{k_h^3 - k_h^2 k_l - (k_h^2 + 2k_h k_l + k_l K) K \max_p}{(k_h - k_l)(K - k_h)^2}$$

(4.8)

### 4.3 เงื่อนไขในการกำหนดพารามิเตอร์ของกลไก ExRED

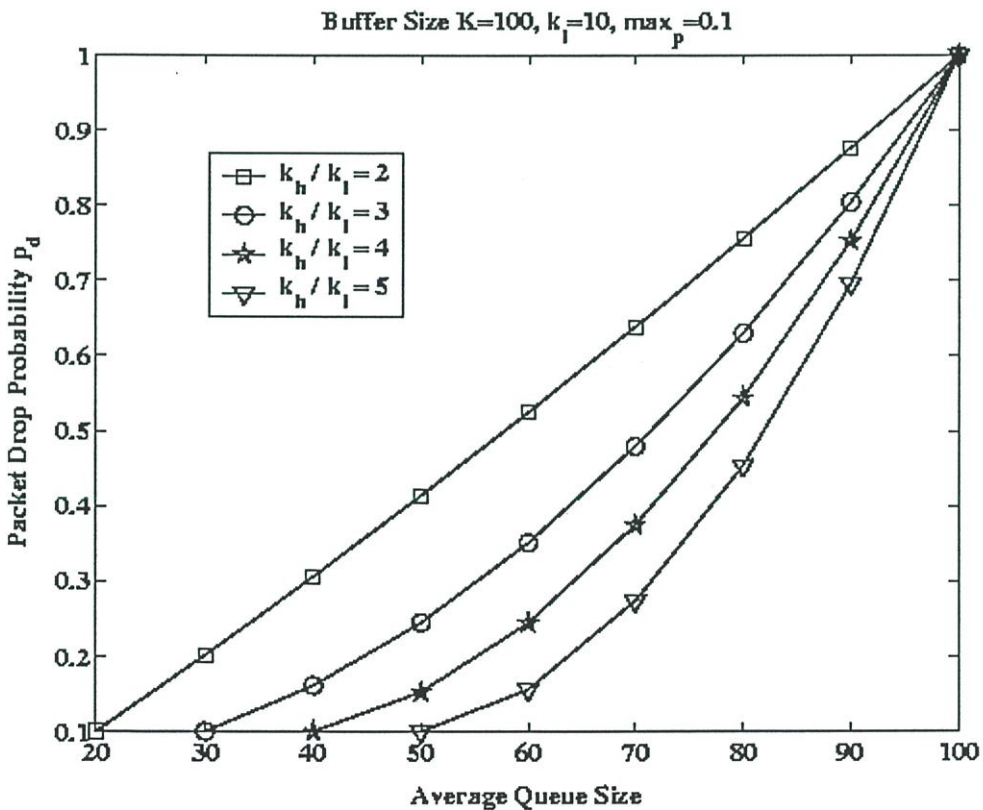
เมื่อค่าเฉลี่ยแพ็คเกจมีค่าสูงขึ้น หมายถึงที่ว่างในบัฟเฟอร์ที่จะสามารถรองรับแพ็คเกจที่จะเข้ามาพักลดลง ความน่าจะเป็นที่จะถูกครีอปของแพ็คเกจที่มาถึงควรมีค่าเพิ่มแบบทวีคูณอย่างต่อเนื่อง ดังนั้นอัตราการเพิ่มขึ้นของความน่าจะเป็นที่แพ็คเกจจะถูกครีอปต้องมีค่าเป็นบวก เงื่อนไขนี้เขียนแสดงได้ดังสมการที่ (4.9)

$$\forall \hat{q}, k_h \leq \hat{q} \leq K \quad ; \quad \frac{d^2}{d\hat{q}^2} p_d(\hat{q}) > 0 \quad (4.9)$$

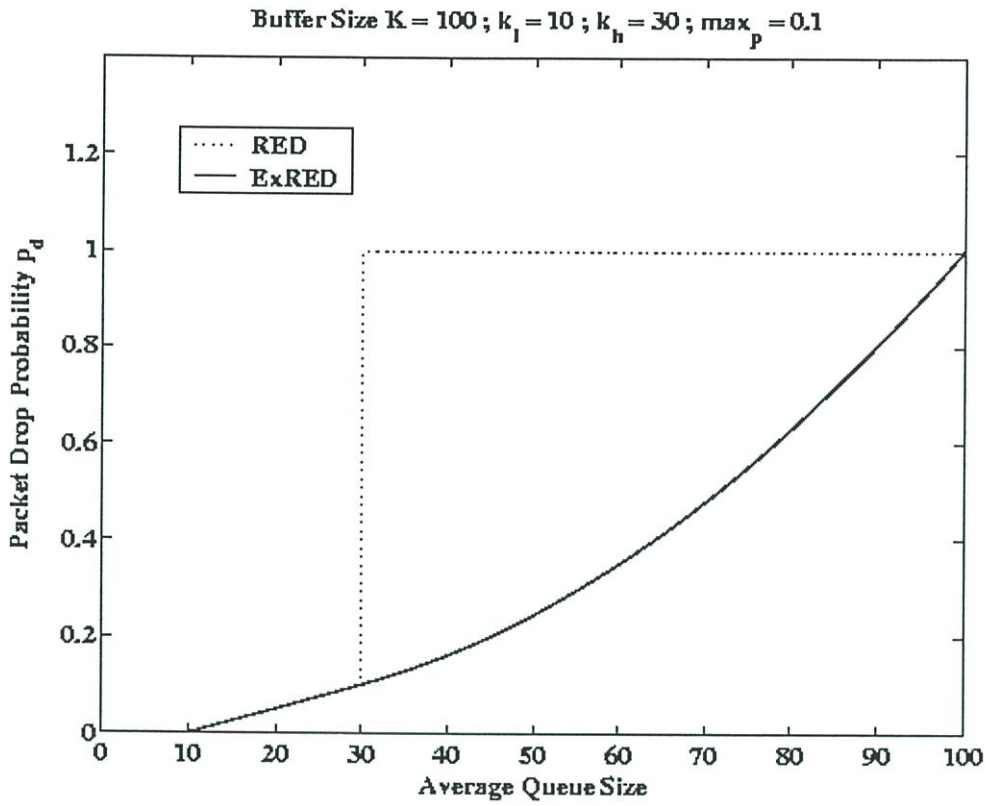
และนำไปสู่เงื่อนไขในการกำหนดพารามิเตอร์ดังสมการที่ (4.10)

$$\max_p < \frac{k_h - k_l}{K - k_l} \quad (4.10)$$

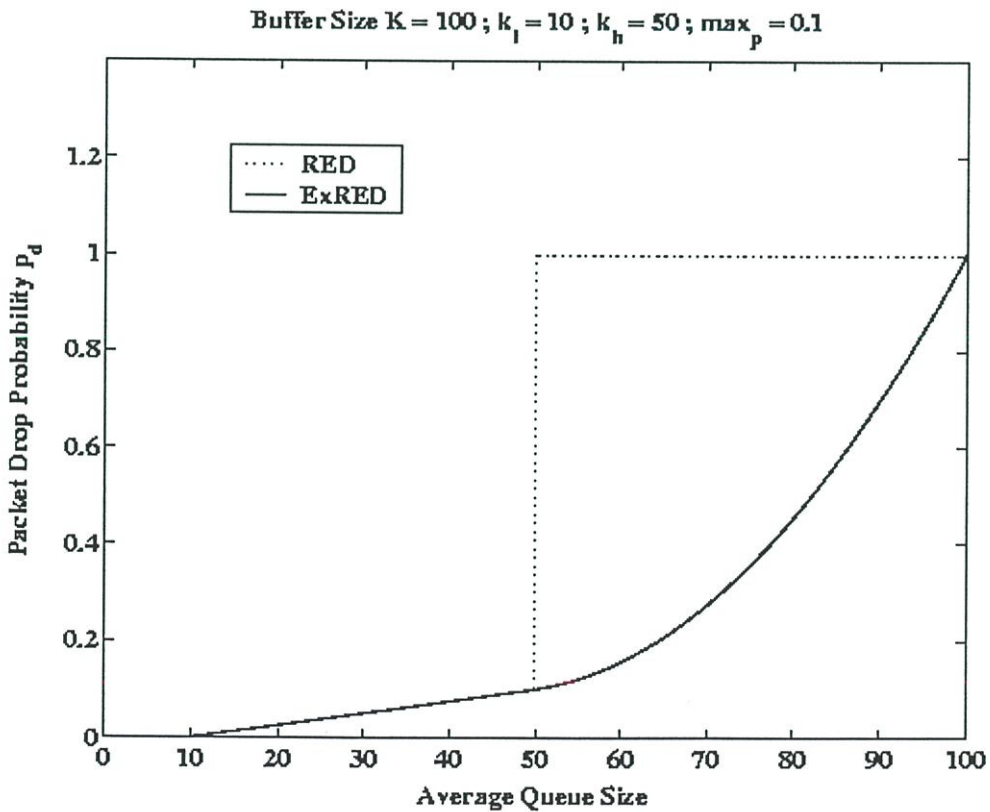
การกำหนดค่าพารามิเตอร์มีผลต่อฟังก์ชันการครอบงำรูปที่ 4.3, 4.4 และ 4.5 ซึ่งรูปที่ 4.3 แสดงให้เห็นว่าถ้ากำหนดให้อัตราส่วนระหว่างค่ากำหนดขั้นสูงต่อค่ากำหนดขั้นต่ำมีค่ามาก จะทำให้โอกาสที่แพคเกจจะถูกครอบงำลดลง แต่อัตราการเพิ่มขึ้นของโอกาสที่แพคเกจจะถูกครอบงำจะสูงกว่า โดยเฉพาะอย่างยิ่งเมื่อที่ว่างในบัฟเฟอร์ใกล้หมดลง ส่วนรูปที่ 4.4 และ 4.5 แสดงลักษณะฟังก์ชันการครอบงำของกลไกแบบ RED และ ExRED เมื่อกำหนดค่าพารามิเตอร์เป็นค่าต่างๆ



รูปที่ 4.3 ฟังก์ชันการครอบงำเมื่ออัตราส่วนระหว่างค่ากำหนดขั้นสูงต่อค่ากำหนดขั้นต่ำมีค่าต่างๆ



รูปที่ 4.4 ฟังก์ชันการครีอ เมื่อขนาดบัฟเฟอร์  $K = 100$ ,  $k_l = 10$ ,  $k_h = 30$  และ  $\max_p = 0.1$



รูปที่ 4.5 ฟังก์ชันการครีอ เมื่อขนาดบัฟเฟอร์  $K = 100$ ,  $k_l = 10$ ,  $k_h = 50$  และ  $\max_p = 0.1$

#### 4.4 แบบจำลองแสดงประสิทธิภาพของกลไกแบบ ExRED

จากสมมติฐานในการวิเคราะห์เช่นเดียวกับการวิเคราะห์กลไกแบบ Tail Drop และแบบ RED โดยกำหนดให้เกตเวย์มีบัฟเฟอร์ขนาดความจุ  $K$  แพคเกจ การมาขอใช้บริการผ่านเกตเวย์เป็นไปตามกระบวนการปัวร์ซอง (Poisson Process) มีอัตราเฉลี่ยการมาขอผ่านเท่ากับ  $\lambda$  แพคเกจต่อวินาที และเวลาที่ใช้ในการให้บริการเป็นค่าสุ่มที่มีการกระจายแบบเอ็กซ์โปเนนเชียล (Exponential Distribution) โดยมีอัตราเฉลี่ยการให้บริการเท่ากับ  $\mu$  แพคเกจต่อวินาที ทำให้เขียนสมการของการกระจายของขนาดคิวเฉลี่ยในบัฟเฟอร์ได้จากสมการที่ (2.19) เป็นดังสมการที่ (4.11)

$$\pi(\hat{q}) = \frac{\left(\frac{\lambda}{\mu}\right)^{\hat{q}} \prod_{i=0}^{\hat{q}-1} (1-p_d(i))}{1 + \sum_{j=1}^K \left(\frac{\lambda}{\mu}\right)^j \prod_{i=0}^{j-1} (1-p_d(i))} \quad (4.11)$$

ค่าความน่าจะเป็นที่จะเกิดการครีโปกที่เกตเวย์สามารถหาได้จากสมการที่ (2.20) โดยใช้ความน่าจะเป็นที่แพคเกจจะถูกครีโปกจากสมการที่ (4.8) จะได้ความน่าจะเป็นที่จะเกิดการครีโปกที่เกตเวย์ที่ใช้กลไกการจัดการคิวแบบ RED ดังสมการที่ (4.12) และจากสมการ (2.21) ถ้า  $S$  คือ ทરพุดที่ออกจากเกตเวย์จะคำนวณได้จากสมการที่ (4.13)

$$P_{drop} = \sum_{\hat{q}=k_h}^K \pi(\hat{q}) \times (a_2 \hat{q}^2 + a_1 \hat{q} + a_0) + \sum_{\hat{q}=k_l}^{k_h-1} p_d(\hat{q}) \times \pi(\hat{q}) \quad (4.12)$$

$$S = (1 - P_{drop}) \lambda \quad (4.13)$$

ค่าประวิงเวลาจากการรอคอยในคิว (Queuing Delay) สามารถหาได้จากสมการที่ (4.14) ซึ่งยกมาแสดงซ้ำอีกทีจากสมการ (2.22) โดยคำนวณค่าเฉลี่ยทางคณิตศาสตร์ของขนาดคิวเฉลี่ย  $E(\hat{Q})$  ได้จากสมการ (2.23) ซึ่งค่าเฉลี่ยของขนาดคิวเฉลี่ยของกลไกแบบ ExRED แสดงในสมการที่ (4.15)

$$D = \frac{E(\hat{Q})}{(1 - P_{drop}) \lambda} - \frac{1}{\mu} \quad (4.14)$$

$$E(\hat{Q}) = \frac{\sum_{k=0}^K \frac{k \left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1 - p_d(\hat{q}))}{\sum_{k=0}^K \left(\left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1 - p_d(\hat{q}))\right)}}{\quad} \quad (4.15)$$

โดยที่

- $D$  : ค่าประวิงเวลาจากการเข้าคิวรอคอย ในหน่วยวินาที  
 $\hat{Q}$  : ตัวแปรสุ่มขนาดคิวเฉลี่ยในบัฟเฟอร์ของเกตเวย์  
 $E(\hat{Q})$  : ค่าเฉลี่ยทางคณิตศาสตร์ (Arithmetic Mean) ของขนาดคิวเฉลี่ยในบัฟเฟอร์  
 $P_{drop}$  : ความน่าจะเป็นที่จะมีการครี้อปเกิดขึ้นที่เกตเวย์  
 $\lambda$  : อัตราเฉลี่ยการมีแพคเกจเข้ามาขอผ่านเกตเวย์ ในหน่วยแพคเกจต่อวินาที  
 $\mu$  : อัตราเฉลี่ยการให้บริการของเกตเวย์ ในหน่วยแพคเกจต่อวินาที

การวิเคราะห์ในประเด็นการเกิดแพคเกจครี้อปติดต่อกัน (Consecutive Drops) ของกลไกแบบ ExRED สามารถใช้ที่มาของหลักการวิเคราะห์เช่นเดียวกับในหัวข้อ 3.2.2 จึงสามารถยกสมการที่ (3.21) และ (3.22) มาแสดงซ้ำได้ดังนี้

ถ้านิยามให้  $N$  เป็นตัวแปรสุ่มของจำนวนแพคเกจที่ถูกครี้อปติดต่อกันอย่างต่อเนื่อง และ  $n$  เป็นค่าของ  $N$  เมื่อเกิดการครี้อปขึ้นมารั้งหนึ่ง ความน่าจะเป็นที่แพคเกจจะถูกครี้อปติดต่อกันอย่างต่อเนื่องอีก  $n$  แพคเกจ สามารถแสดงได้ดังสมการ (4.16)

$$\forall n \geq 0, \quad P_{drop}^{cons}(N > n) = \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) (p_d(\hat{q}))^{n+1}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) p_d(\hat{q})} \quad (4.16)$$

จากสมการ (4.16) สามารถหาค่าเฉลี่ยทางคณิตศาสตร์ (Arithmetic Mean) ของจำนวนแพคเกจที่จะถูกครี้อปติดต่อกันได้จากสมการที่ (4.17)

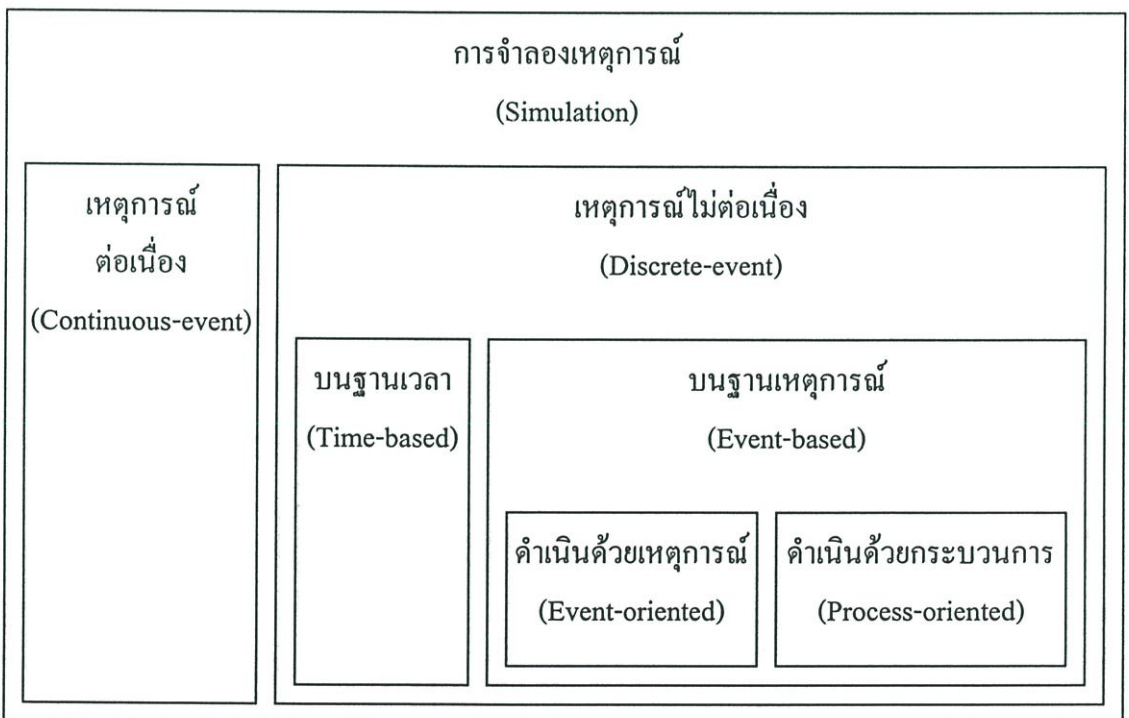
$$E(N) = 1 + \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) \frac{(p_d(\hat{q}))^2}{1 - p_d(\hat{q})}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) p_d(\hat{q})} \quad (4.17)$$

## บทที่ 5

### การจำลองเหตุการณ์

นอกจากการวิเคราะห์เชิงคณิตศาสตร์ที่ได้กล่าวถึงมาแล้ว การจำลองเหตุการณ์ [8][14] เพื่อสร้างภาวะที่ต้องการ และศึกษาผลตอบสนอง แล้วนำมาสรุปผล ก็เป็นส่วนสนับสนุนหนึ่งในการนำเสนองานวิจัย ดังนั้นในบทนี้จะนำเสนอการจำลองเหตุการณ์ในการเกิดแพคเกจที่สมมติว่าถูกส่งมาจากโหนดต้นทาง เพื่อขอใช้บริการผ่านเกตเวย์ไปยังโหนดปลายทางอีกเครือข่ายหนึ่ง ในการจำลองเหตุการณ์นี้อัตราการเกิดแพคเกจจะเป็นเงื่อนไขที่ถูกสร้างขึ้นเพื่อกระตุ้น โดยจะวัดผลด้วยการเก็บค่าสถิติและหาค่าเฉลี่ยของอัตราส่วนของแพคเกจที่สามารถผ่านเกตเวย์ออกไป เทียบกับจำนวนแพคเกจที่ขอผ่านเกตเวย์ ซึ่งในวิทยานิพนธ์ได้นิยามว่าเป็นทรูพุทของเกตเวย์ นอกจากนี้ยังหาค่าเฉลี่ยของเวลารอคอยในคิวในบัฟเฟอร์ และค่าเฉลี่ยจำนวนแพคเกจที่ถูกครอบครองเนื่องด้วย

#### 5.1 หลักการสร้างแบบจำลองเหตุการณ์



รูปที่ 5.1 การแบ่งชั้นในการสร้างแบบจำลองเหตุการณ์

การสร้างแบบจำลองเหตุการณ์มีรูปแบบที่แบ่งชั้นได้ดังรูปที่ 5.1 โดยการจำลองเหตุการณ์สามารถเลือกได้ว่าจะมีการจำลองในลักษณะเหตุการณ์ที่เกิดขึ้นแบบต่อเนื่อง (Continuous-event)

หรือแบบไม่ต่อเนื่อง (Discrete-event) เนื่องจากเหตุการณ์ที่จะจำลองในงานวิจัยนี้เป็นการวิเคราะห์เหตุการณ์ที่เกิดแบบไม่ต่อเนื่อง ดังนั้นจะกล่าวเฉพาะลักษณะการจำลองเหตุการณ์แบบไม่ต่อเนื่องเท่านั้น

ในการสร้างแบบจำลองเหตุการณ์แบบไม่ต่อเนื่อง การเกิดเหตุการณ์และการเก็บค่าเพื่อไปคำนวณค่าทางสถิติ มีฐานในการดำเนินไปอยู่ 2 แบบ คือ

#### 1) บนฐานเวลา (Time-based)

การจำลองแบบนี้มักเรียกว่าการจำลองแบบซิงโครนัส (Synchronous simulation) ซึ่งลูปควบคุมหลัก (Main Control Loop) ในการจำลอง จะควบคุมขั้นเวลาในการดำเนินการเป็นขั้นแบบคงที่ ซึ่งที่สถานะเริ่มต้นของลูปควบคุม เวลา  $t$  จะเพิ่มเป็นขั้นคงที่  $\Delta t$  ไปเป็นเวลา  $t + \Delta t$  ด้วยค่า  $\Delta t$  น้อยๆ จากนั้นจะทำการตรวจสอบสิ่งที่เกิดขึ้นในช่วงเวลา  $[t, t + \Delta t]$  และดำเนินการตามขั้นตอนต่อไปทั้งนี้ขึ้นกับสิ่งที่เกิดขึ้นในช่วงเวลาดังกล่าวแล้วจึงดำเนินการรอบใหม่ในการจำลองต่อไปเรื่อยๆ

การจำลองบนฐานเวลาสามารถทำได้ง่าย การสร้างแบบจำลองนี้ใกล้เคียงกับกระบวนการเชิงเลขในการหาคำตอบของสมการอนุพันธ์ แต่การจำลองแบบนี้ก็มีข้อเสียตรงที่สมมติฐานของการจำลอง คือ สมมติฐานที่ว่าลำดับของเหตุการณ์ที่เกิดขึ้นในช่วงเวลา  $[t, t + \Delta t]$  จะไม่มีความสำคัญ และสมมติฐานที่ว่าแต่ละเหตุการณ์ที่เกิดขึ้นเหล่านั้นจะเป็นอิสระต่อกันได้ก็ต่อเมื่อ  $\Delta t$  ต้องมีค่าน้อยมากจนพอที่จะทำให้ความน่าจะเป็นที่จะทำให้แต่ละเหตุการณ์ไม่อิสระต่อกันน้อยที่สุด ด้วยเหตุผลนี้จึงทำให้โดยปกติ  $\Delta t$  จะถูกกำหนดให้มีค่าน้อยๆ ซึ่งทำให้การจำลองขาดประสิทธิภาพไป เพราะมีหลายช่องของขั้นเวลาสั้นๆดังกล่าว ที่จะถูกดำเนินการและวิเคราะห์ทั้งที่ไม่มีเหตุการณ์ใดเกิดขึ้นในช่วงเวลาสั้นๆนั้นๆ ทำให้การจำลองบนฐานเวลาไม่ค่อยเป็นที่นิยม

#### 2) บนฐานเหตุการณ์ (Event-based)

การจำลองแบบนี้มักเรียกว่าการจำลองแบบอะซิงโครนัส (Asynchronous simulation) ซึ่งแตกต่างกับแบบฐานเวลาตรงที่ขั้นเวลาในการดำเนินการเป็นขั้นเวลาที่สามารถเปลี่ยนแปลงค่าได้ การจำลองถูกควบคุมโดยการเกิดเหตุการณ์ถัดไปโดยไม่คำนึงว่าจะจะเป็นช่วงเวลาห่างเท่าไร ซึ่งเป็นสิ่งที่ทำให้การจำลองแบบนี้มีประสิทธิภาพ ขั้นเวลาอาจยาวนานหรือสั้นก็ได้ เพื่อให้ได้ 1 เหตุการณ์ใน 1 ขั้นเวลา ทำให้ปัญหาการเกิดเหตุการณ์ที่ไม่อิสระต่อกันในช่วงหนึ่งขั้นเวลาหมดไป

เมื่อใดก็ตามที่มีเหตุการณ์หนึ่งๆเกิดขึ้น เหตุการณ์ที่เกิดขึ้นใหม่ๆก็จะเป็นเหตุการณ์ที่เกิดในอนาคต เมื่อพิจารณาช่วงเวลาใดๆ จะได้คิวของการเกิดเหตุการณ์ ซึ่งเหตุการณ์เหล่านี้จะทำให้สถานะระบบเปลี่ยนไป เมื่อเหตุการณ์ปัจจุบันทำให้สถานะเปลี่ยน และเหตุการณ์ถัดไปก็ถือว่าเป็นเหตุการณ์ที่รอการดำเนินการเพื่อให้สถานะระบบเปลี่ยนในอนาคต ซึ่งเหตุการณ์ในอนาคตเหล่านั้นมักถูกรวมเป็นรายการลำดับเหตุการณ์ (Ordered event list) เมื่อใดก็ตามที่เหตุการณ์แรกถูกจำลอง(หรือถูกดำเนินการ)ไปแล้ว เหตุการณ์ถัดไปที่จะถูกดำเนินการ จะถูกอ่านจากรายการดังกล่าว และเวลาที่จะถูกบันทึกในการจำลองก็จะถูกปรับค่าตาม ในการจำลองเหตุการณ์แบบนี้ เหตุการณ์ใหม่จะถูกสร้างขึ้น และแทรกในรายการลำดับเหตุการณ์ตรงตำแหน่งเวลาที่ถูกต้อง หลังจากนั้นเหตุการณ์ที่อยู่ลำดับต้นๆในรายการ จะถูกดำเนินการ

การจำลองเหตุการณ์แบบไม่ต่อเนื่องในการวิเคราะห์ประสิทธิภาพของคอมพิวเตอร์และการสื่อสารมักเป็นการจำลองแบบตั้งอยู่บนฐานเหตุการณ์ ซึ่งการจำลองแบบนี้มีข้อจำกัดเพียงแค่ว่าจะต้องคำนวณตำแหน่งเวลา ณ ช่วงของเหตุการณ์ที่จะเกิดในอนาคต ลำดับถัดไปเอง แต่ข้อจำกัดนี้อาจเป็นปัญหาในบางกรณี เช่น สำหรับในการจำลองที่กำหนดให้มีการกระจายแบบที่ซับซ้อนของค่าประวิงเวลา หรือระบบที่เป็นไดนามิกแบบปรับค่าได้ต่อเนื่อง (Continuous-variable dynamic system) ในกรณีที่เป็นปัญหาดังกล่าวนี้ การจำลองบนฐานเวลาอาจเหมาะสมกว่า

นอกจากที่กล่าวมาข้างต้น การจำลองเหตุการณ์ในการเกิดแพคเกจที่มาขอผ่านเกตเวย์เหมาะสมที่จะใช้การจำลองแบบไม่ต่อเนื่องบนฐานเหตุการณ์ รูปแบบในการทำการจำลองมี 2 แบบ คือ

#### 1) ดำเนินเรื่องด้วยเหตุการณ์ (Event-oriented)

ในการจำลองโดยให้เหตุการณ์เป็นตัวดำเนินเรื่อง นิยามการดำเนินการ  $P_i$  สำหรับเหตุการณ์ทุกประเภทของเหตุการณ์ที่  $i$  ที่สามารถเกิดขึ้นได้ ซึ่งรายการเหตุการณ์ (Event list) จะถูกนิยามขึ้นในการจำลองนั้นๆ เมื่อเริ่มต้นรายการเหตุการณ์รูปควบคุมหลักจะเริ่มทำงาน เหตุการณ์แรกที่จะเกิด จะถูกเอามาจากรายการเหตุการณ์ เวลาที่ดำเนินไปในการจำลองจะถูกเพิ่มขึ้นจนเท่ากับค่าเวลาที่เหตุการณ์ปัจจุบันเกิด จากนั้นถ้าถึงลำดับเหตุการณ์ที่  $i$  การดำเนินการ  $P_i$  จะทำงาน ซึ่งสถานะของระบบที่จำลองเหตุการณ์จะเปลี่ยนตามสิ่งที่เกิดจากเหตุการณ์  $i$  และเหตุการณ์ใหม่จะถูกสร้างขึ้น แล้วถูกแทรกเพิ่มไปในรายการลำดับเหตุการณ์ในตำแหน่งที่ถูกต้อง หลังจากการดำเนินการ  $P_i$  หยุดลง ค่าจะถูกเก็บเป็นสถิติ และรูปควบคุมหลักจะทำงานต่อไป โดยทั่วไปหลักเกณฑ์ในการหยุดทำงานจะรวมเวลาในการ

จำลอง จำนวนเหตุการณ์ที่ถูกประมวลผล จำนวนเวลาที่ใช้ในการประมวลผล หรือความกว้างของช่วงเวลาที่ยึดมั่นที่ถูกคำนวณได้ในการวัดค่าที่อยู่ในช่วงที่สนใจ [8]

## 2) ดำเนินเรื่องด้วยกระบวนการ (Process-oriented)

เมื่อเปรียบเทียบกันแล้ว การจำลองโดยให้เหตุการณ์เป็นตัวดำเนินเรื่อง จะมีการจัดการเหตุการณ์ที่ถูกมองภาพได้อย่างชัดเจน ในขณะที่การจำลองโดยให้กระบวนการเป็นตัวดำเนินเรื่อง จะมีกระบวนการนั้นแฝงตัวประกอบคิดในเหตุการณ์ทุกแบบ กระบวนการเหล่านี้จะแลกเปลี่ยนข้อมูลกันเพื่อสื่อสารการเปลี่ยนสถานะกันกับกระบวนการอื่นๆ โดยการให้การส่งข้อความ (Message) ระหว่างกัน หรือใช้ตัวแปรร่วม การปฏิบัติการของระบบการจำลองเหตุการณ์ สามารถพบเห็นได้เสมือนเส้นทางในการทำให้สำเร็จของกระบวนการ เหตุการณ์การติดต่อสื่อสารนั้นๆ ซึ่งตารางเวลาในของเหตุการณ์ในการจำลองจะถูกกระทำอย่างเป็นนัย (Implicitly)

## 5.2 การสร้างแบบจำลองเหตุการณ์ในงานวิจัย

จากหัวข้อ 5.1 การสร้างแบบจำลองเหตุการณ์ที่จะเลือกใช้ในการวิเคราะห์ประสิทธิภาพ จะเป็นการจำลองแบบโดยวิธีการจำลองเหตุการณ์แบบไม่ต่อเนื่อง (Discrete-event Simulation) ที่ดำเนินไปบนฐานเหตุการณ์ (Event-based) โดยมีเหตุการณ์เป็นตัวดำเนินเรื่อง (Event-oriented) ซึ่งเหตุการณ์ที่จะพิจารณา คือ เหตุการณ์ที่มีแพ็คเกจเข้ามาขอใช้บริการผ่านเกตเวย์ และการตอบสนองของระบบ ถ้าสมมติให้  $\delta(t)$  แสดงแทนการเกิดแพ็คเกจที่เข้ามาขอใช้บริการผ่านเกตเวย์ที่เวลาเริ่มต้น  $t = 0$  โดยแสดงในรูปฟังก์ชันได้เป็น

$$\delta(t) = \begin{cases} 1 & ; t = 0 \\ 0 & ; t \neq 0 \end{cases} \quad (5.1)$$

และ

$$\int_{-\infty}^{\infty} \delta(t) dt = 1 \quad (5.2)$$

จากสมการที่ (5.1) สามารถนิยามให้  $\delta(t - t_m^a)$  แสดงแทนแพ็คเกจที่มาถึงเกตเวย์เป็นลำดับที่  $m$  ที่เวลา  $t_m^a$  ในการกำหนดรายการลำดับเหตุการณ์แพ็คเกจที่มาถึงเกตเวย์ตั้งแต่  $m = 1, 2, 3, \dots, M$  สามารถแสดงรายการดังกล่าวได้เป็นดังสมการที่ (5.3)

$$f_{arrival} = \sum_{m=1}^M \delta(t - t_m^a) \quad (5.3)$$

ซึ่ง

$$t_m^a = \sum_{m=1}^M t_m^{ia} \quad (5.4)$$

โดยที่  $t_m^{ia}$  เป็นค่าของตัวแปรสุ่มที่มีการกระจายแบบเอ็กซ์โปเนนเชียลของช่วงเวลาดังกล่าว (InterArrival) ของแพคเกจที่มาถึง โดยมี  $m$  เป็นตัวบอกลำดับการมาถึง ส่วนรายการลำดับเหตุการณ์ของแพคเกจที่ออกจากเกตเวย์ไปตั้งแต่  $n = 1, 2, 3, \dots, N$  สามารถแสดงรายการดังกล่าวเป็น

$$f_{departure} = \sum_{n=1}^N \delta(t - t_n^a - t_n^w - t_n^s) \quad (5.5)$$

โดยที่  $t_n^w$  เป็นเวลาที่รอคอยในคิวในบัฟเฟอร์เป็นค่าที่ได้จากการวัดและคำนวณ ส่วน  $t_n^s$  เป็นค่าของตัวแปรสุ่มที่มีการกระจายแบบเอ็กซ์โปเนนเชียลของเวลาในการให้บริการแพคเกจ โดยมี  $n$  เป็นตัวบอกลำดับของแพคเกจที่ถูกให้บริการและส่งออก

จากสมการที่ (5.3) และ (5.5) จะได้รายการลำดับเหตุการณ์ของแพคเกจที่เข้าและออกจากบัฟเฟอร์ที่เขียนแสดงแทนได้ดังสมการที่ (5.6) โดยมีเงื่อนไขว่า  $M > N$  เนื่องจากการทำงานของกลไกจัดการคิวในบัฟเฟอร์จะตรวจสอบขนาดคิวทุกครั้งที่มีแพคเกจใหม่มาถึง ดังนั้นการหาขนาดคิวจึงเกิดที่เวลา  $t_m^a; \forall m \in I^+$  ซึ่งจะพิจารณาในที่แพคเกจที่  $M$  เข้ามา

$$f = \sum_{m=1}^M \delta(t - t_m^a) - \sum_{n=1}^N \delta(t - t_n^a - t_n^w - t_n^s) \quad (5.6)$$

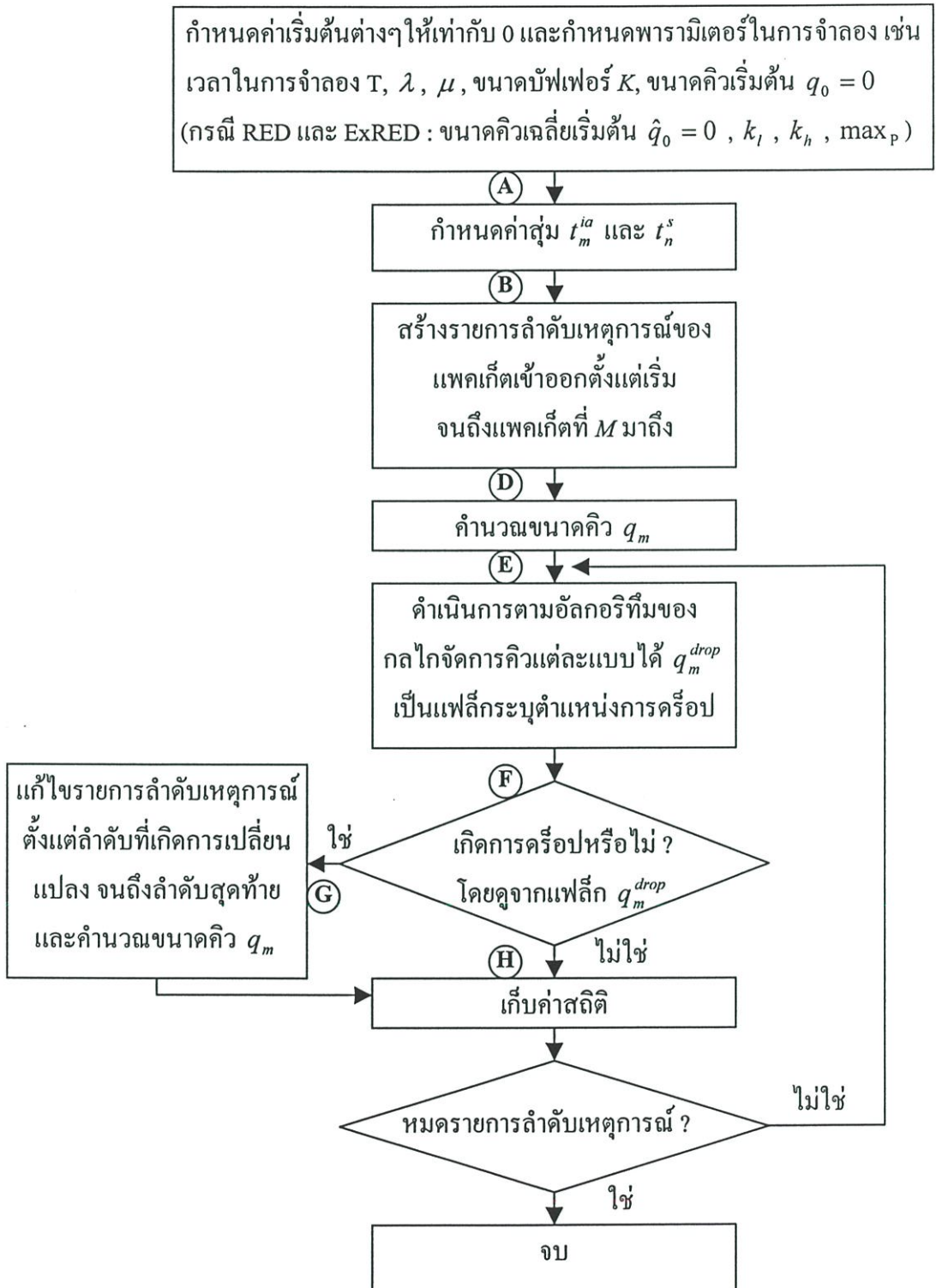
จากคุณสมบัติในสมการที่ (5.2) เมื่อนำมาใช้ในสมการที่ (5.6) จะได้ขนาดคิวในบัฟเฟอร์ที่เวลา  $t_m^a$  ดังสมการที่ (5.7)

$$q_m = \int_0^{t_m^a} f \cdot dt = M - N \quad (5.7)$$

การกำหนดขอบเขตสิ้นสุดการจำลอง จะมีการกำหนดเวลาในการจำลองเหตุการณ์เท่ากับ  $T$  และดำเนินการวนลูปจนกระทั่ง  $\sum_{m=1}^M t_m^a \geq T$  แล้วหยุดการจำลองไว้ที่การมาถึงของแพคเกจที่  $M$  เท่านั้น จากนั้นก็ขึ้นอยู่กับอัลกอริทึมในการจำลองเหตุการณ์ ที่คำนวณและเก็บเป็นสถิติไว้ เพื่อประมวลผลให้ได้ค่าต่างๆที่ต้องการ

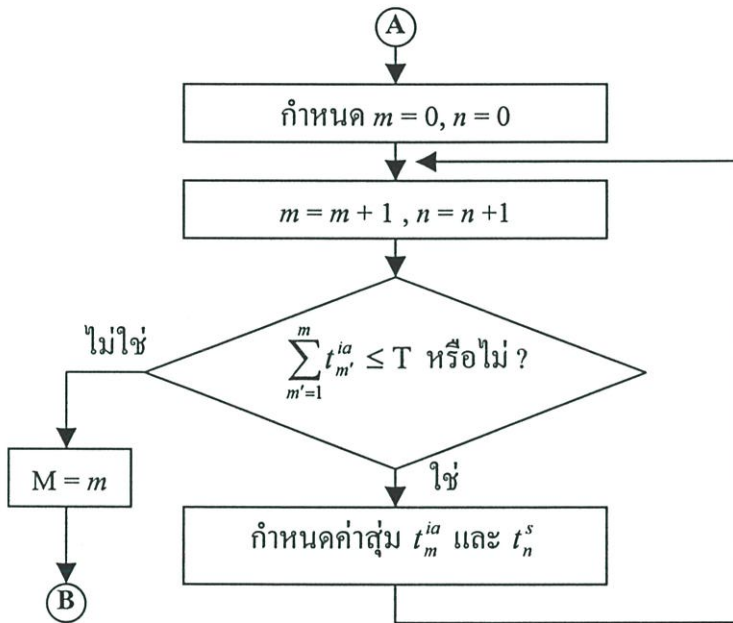
### 5.3 อัลกอริทึมในการจำลองเหตุการณ์

การจำลองเหตุการณ์ในงานวิจัยนี้ มีอัลกอริทึมหลักที่แสดงได้ดังในรูปที่ 5.2

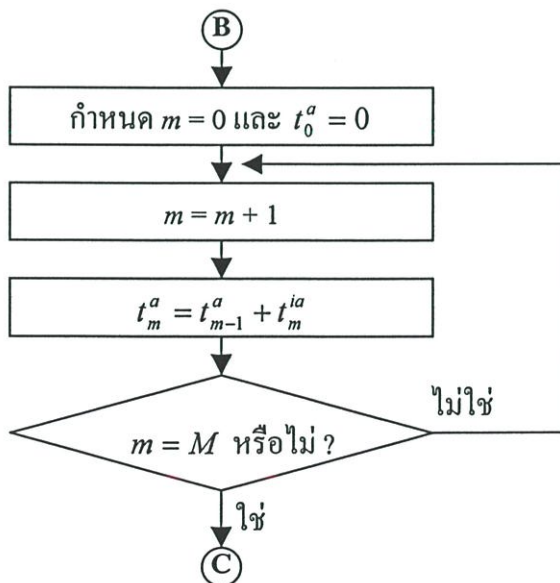


รูปที่ 5.2 อัลกอริทึมหลักในการจำลองเหตุการณ์

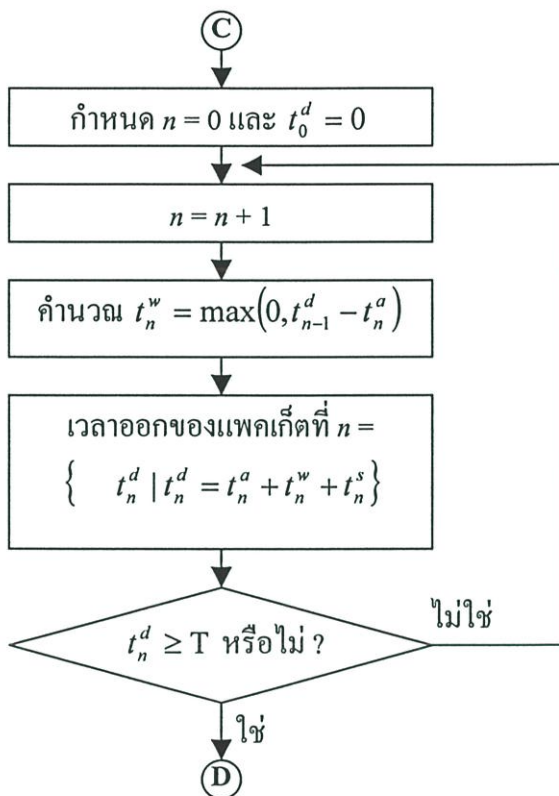
อัลกอริทึมในการกำหนดค่าสุ่มเป็นดังรูป 5.3 ส่วนอัลกอริทึมในการสร้างรายการลำดับเหตุการณ์ที่มีแพ็คเกจเข้ามาในบัฟเฟอร์เป็นดังรูปที่ 5.4 ซึ่งจำนวนเหตุการณ์ในรายการลำดับเหตุการณ์ของแพ็คเกจที่เข้ามาในบัฟเฟอร์จะขึ้นกับจำนวนค่าสุ่ม และจะเป็นตัวกำหนดขอบเขตจำนวนเหตุการณ์ในรายการลำดับเหตุการณ์ที่มีแพ็คเกจออก เนื่องจากในขั้นตอนการจำลองเหตุการณ์ต่อไป จำนวนแพ็คเกจที่เข้ามาเท่านั้นที่จะถูกดำเนินการต่อ(ซึ่งหมายถึงถูกให้บริการและส่งออก) และรูปที่ 5.5 รายการลำดับเหตุการณ์ที่มีแพ็คเกจได้รับบริการและส่งออกจากบัฟเฟอร์



รูปที่ 5.3 อัลกอริทึมในการกำหนดค่าสุ่ม

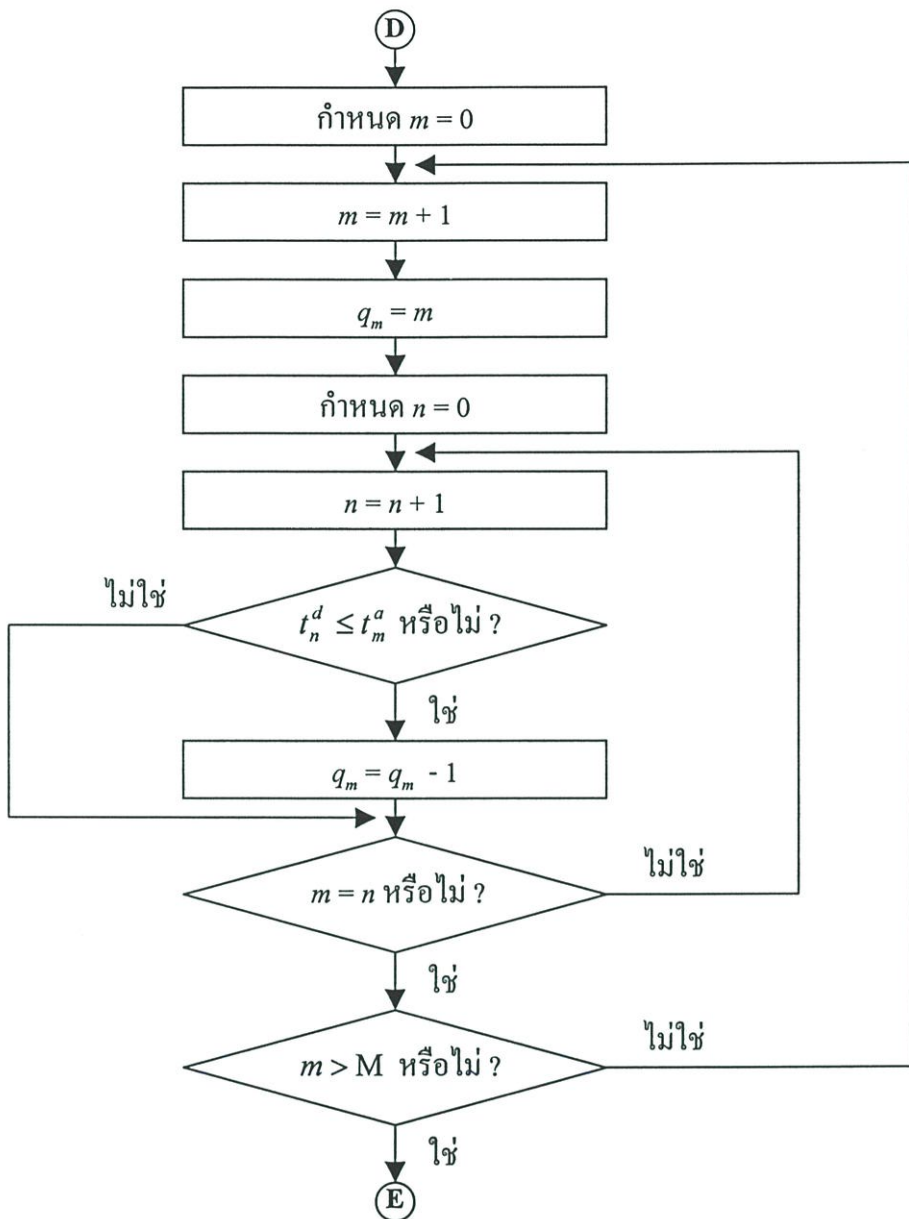


รูปที่ 5.4 อัลกอริทึมในการสร้างรายการลำดับเหตุการณ์ที่มีแพ็คเกจเข้ามาในคิวในบัฟเฟอร์



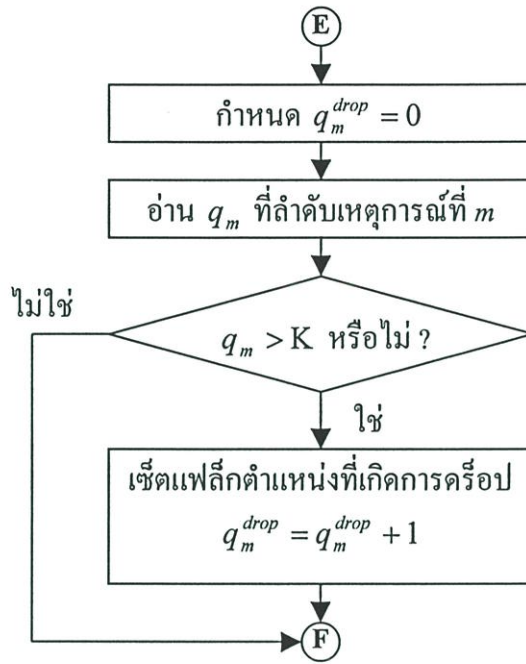
รูปที่ 5.5 อัลกอริทึมในการสร้างรายการลำดับเหตุการณ์ที่มีแพ็คเกจออกจากคิวในบัฟเฟอร์

หลังจากได้รายการลำดับเหตุการณ์แล้ว รายการดังกล่าวจะถูกนำมาดำเนินการต่อโดยการหาขนาดคิว โดยอัลกอริทึมในการหาขนาดคิวแสดงได้ดังรูปที่ 5.6

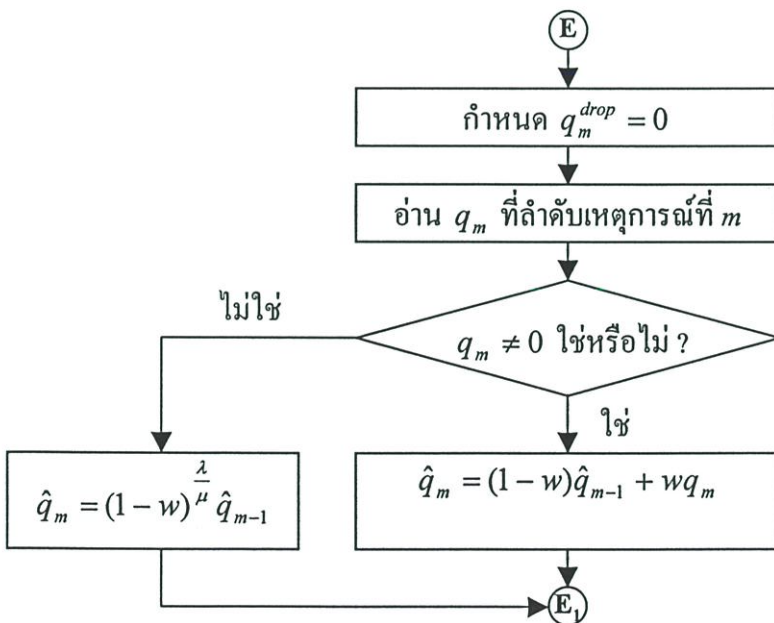


รูปที่ 5.6 อัลกอริทึมในการคำนวณขนาดคิวในบัฟเฟอร์

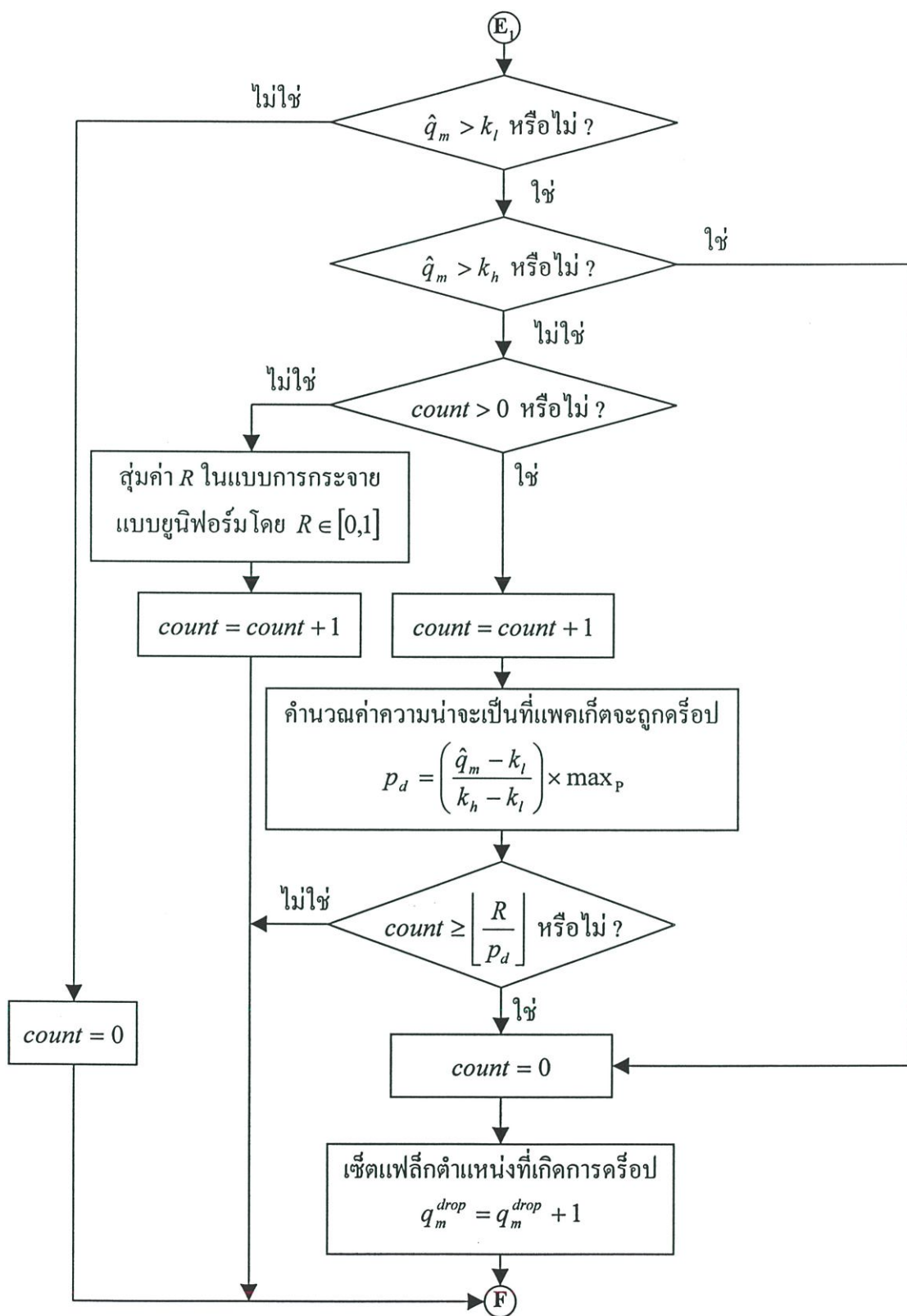
ขนาดคิวที่คำนวณได้จะถูกนำไปดำเนินการต่อในอัลกอริทึมถัดไป ซึ่งรูปแบบการดำเนินการขึ้นอยู่กับกลไกการจัดคิวที่ต้องการจำลอง รูปแบบอัลกอริทึมสามารถอ้างอิงได้ในบทที่ 3 และ 4 ซึ่งรูปที่ 5.7 ถึง 5.9 จะแสดงตัวอย่างของการดำเนินการตามแบบกลไกแบบ Tail Drop แบบ RED และแบบ ExRED ตามลำดับ รูปที่ 5.8 (ก) แสดงส่วนที่ทำหน้าที่คำนวณขนาดคิวเฉลี่ยสำหรับกลไกแบบ RED และ รูปที่ 5.8 (ข) แสดงส่วนที่ทำหน้าที่ว่าจะครีอปแพคเกิดขึ้นหรือไม่ สำหรับกลไกแบบ ExRED จะมีส่วนที่ทำหน้าที่คำนวณขนาดคิวเฉลี่ยสำหรับกลไกแบบ ExRED เหมือนของกลไกแบบ RED ดังรูป 5.8 (ก) ส่วนรูปที่ 5.9 แสดงส่วนที่ทำหน้าที่ว่าจะครีอปแพคเกิดขึ้นหรือไม่ของกลไกแบบ ExRED ซึ่งจะแตกต่างจากกลไกแบบ RED



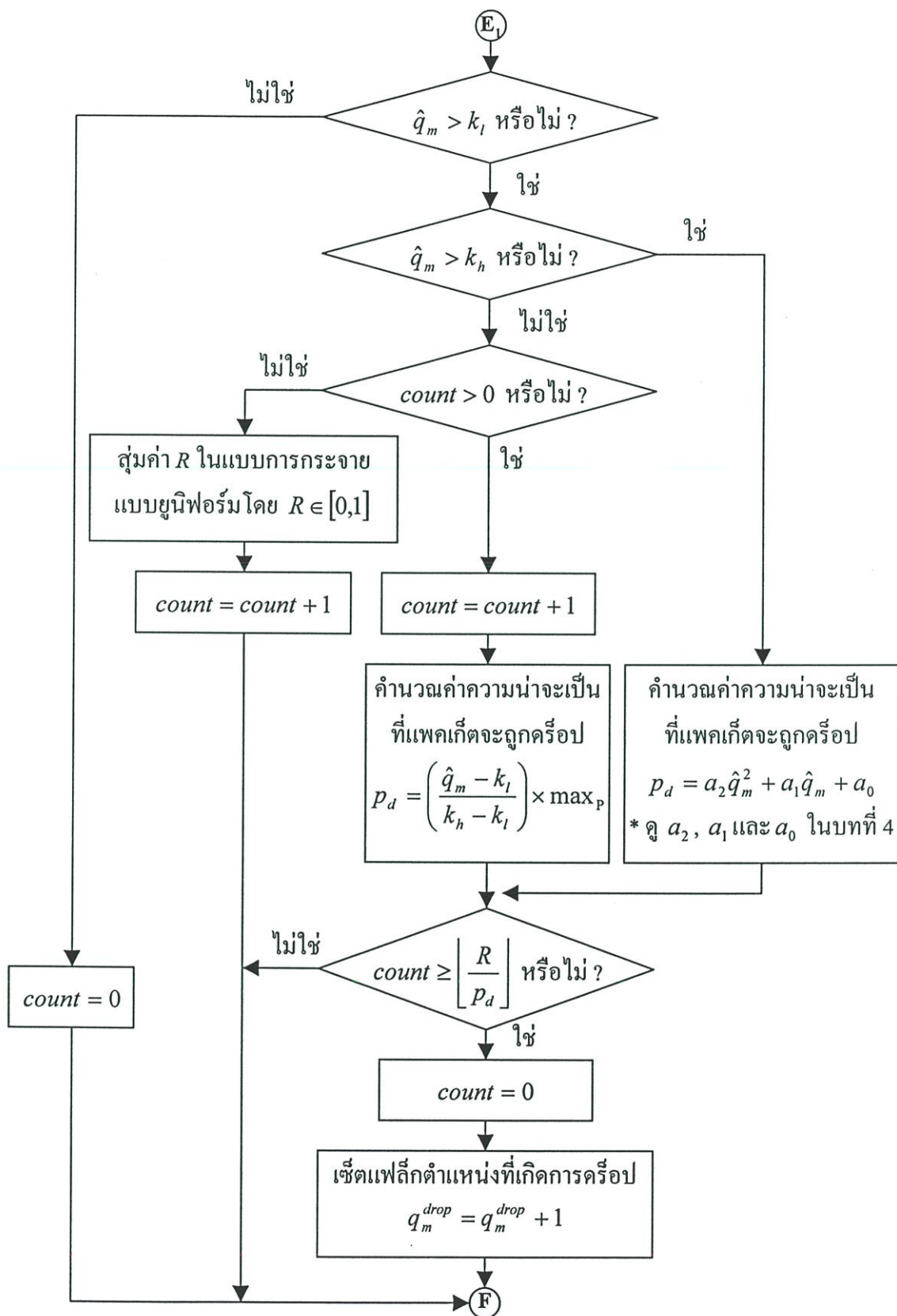
รูปที่ 5.7 อัลกอริทึมในการดำเนินการตามกลไกแบบ Tail Drop



รูปที่ 5.8 (ก) อัลกอริทึมในการดำเนินการตามกลไกแบบ RED ในส่วนการคำนวณหาขนาดคิวเฉลี่ย

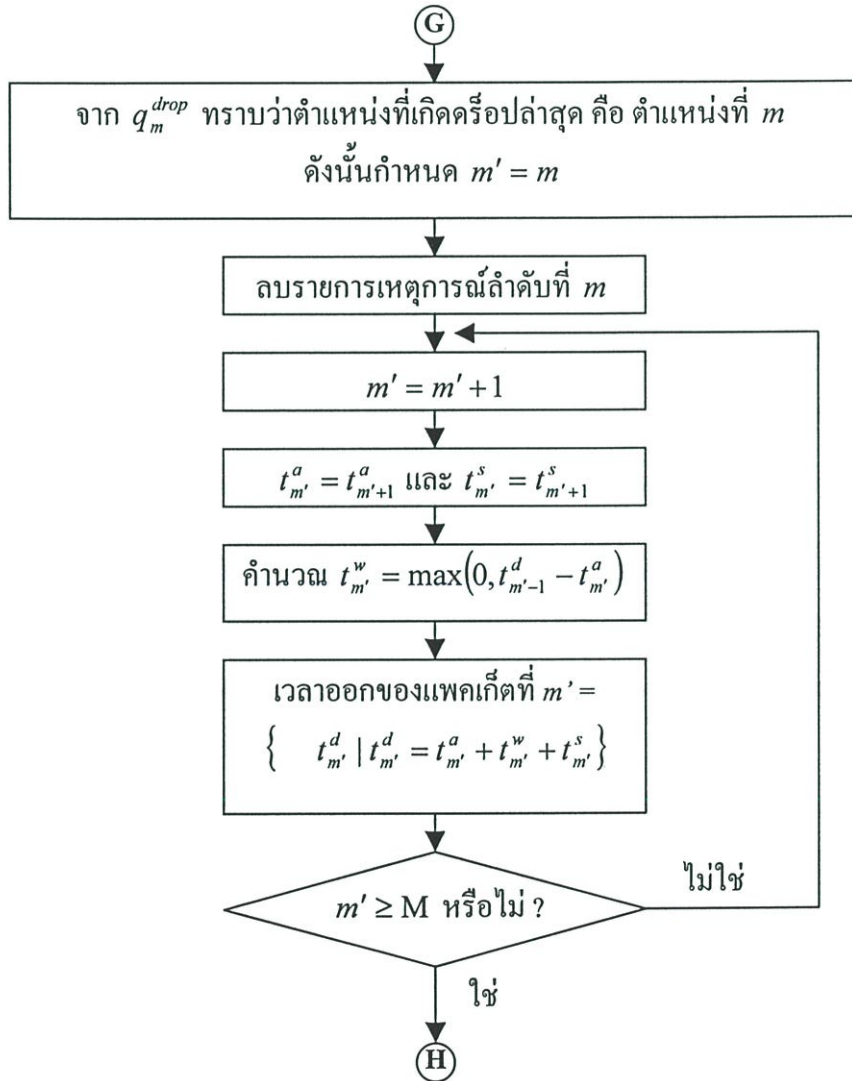


รูปที่ 5.8 (ข) อัลกอริทึมในการดำเนินการตามกลไกแบบ RED ในส่วนการตัดสินใจว่าจะครี้อปแพคเกจหรือไม่ (Packet Marking/Drop Decision)



รูปที่ 5.9 อัลกอริทึมในการดำเนินการตามกลไกแบบ ExRED ในส่วนการตัดสินใจว่าจะครีอแพ็คเกจหรือไม่ (Packet Marking/Drop Decision)

หลังจากจำลองเหตุการณ์ในช่วงที่มีการทำงานของกลไกจัดการคิวแต่ละแบบ จะทราบว่าเกิดการเปลี่ยนแปลงในคิวในบัฟเฟอร์หรือไม่ ด้วยค่าเฟล็ก  $q_m^{drop}$  จากอัลกอริทึมหลักในการจำลองเหตุการณ์ดังรูปที่ 5.2 ถ้า  $q_m^{drop}$  มีค่ามากกว่าศูนย์ แสดงว่ามีการครีโปกแพกเกิดขึ้นที่ตำแหน่งที่  $m$  ก็จะเป็นหน้าที่ของอัลกอริทึมในการแก้ไขลำดับเหตุการณ์ดังรูปที่ 5.10 เป็นลำดับต่อไป แต่ถ้า  $q_m^{drop}$  มีค่าเท่ากับศูนย์ แสดงว่าไม่มีการครีโปกเกิดขึ้น ดังนั้นค่าต่างๆที่เกิดขึ้นก็จะถูกเก็บเป็นสถิติเอาไว้และเข้าสู่การจำลองในรอบต่อไป จนกว่าจะหมดรายการลำดับเหตุการณ์ที่ตั้งเอาไว้



รูปที่ 5.10 อัลกอริทึมแก้ไขรายการลำดับเหตุการณ์

## 5.4 การเก็บค่าทางสถิติ

หลังจากจำลองเหตุการณ์แล้ว ค่าทางสถิติต่างจะถูกเก็บเพื่อนำไปประมวลผลและวิเคราะห์ ซึ่งการวิเคราะห์จะพิจารณาที่เมื่อระบบเข้าสู่สถานะคงตัว (Steady-State) ซึ่งค่าที่จะนำไปทำการวิเคราะห์ประสิทธิภาพเพื่อหาข้อสรุป ได้แก่ ค่าเฉลี่ยขนาดคิว (Mean Queue Size) ความน่าจะเป็นที่จะเกิดการครี้อปโดยเฉลี่ย (Average Drop Probability) ค่าเฉลี่ยของค่าประวิงเวลาที่เกิดจากการรอคอยในคิว (Mean Queuing Delay) และค่าเฉลี่ยของจำนวนแพคเกจที่ถูกรี้อปติดต่อกัน (Mean Number of Consecutive Dropped Packet หรือเรียกสั้นๆว่า Consecutive Drops)

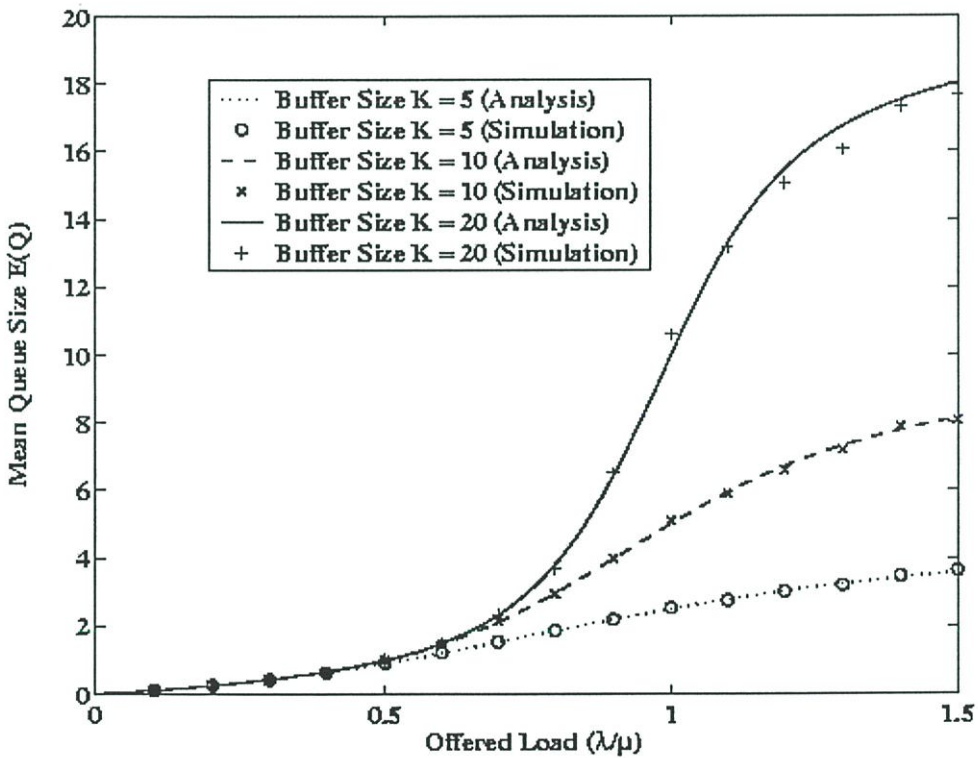
ถ้า  $Q$  เป็นขนาดคิวในบัฟเฟอร์ของเกตเวย์ และ  $q_m$  จากสมการที่ (5.7) เป็นค่าของ  $Q$  ในขณะทีแพคเกจที่  $m$  มาถึงเกตเวย์ โดยที่  $m = 1, 2, 3, \dots, M$  แล้ว จะได้ว่า  $E(Q)$  เป็นค่าเฉลี่ยขนาดคิว (Mean Queue Size) แสดงดังตัวอย่างในรูปที่ 5.11 และจากตัวแปร  $q_m^{drop}$  จะทำให้ทราบจำนวนแพคเกจทั้งหมดที่จะถูกรี้อปได้ ดังนั้น สมมติให้  $M'$  เป็นจำนวนแพคเกจที่ถูกรี้อป จะหาความน่าจะเป็นที่จะเกิดการครี้อปเฉลี่ยได้เท่ากับ  $\frac{M'}{M}$  ดังตัวอย่างในรูปที่ 5.12 นอกจากนี้  $q_m^{drop}$  ยังสามารถใช้ในการหาค่าจำนวนแพคเกจที่ถูกรี้อปติดต่อกันได้ ถ้า  $X$  เป็นจำนวนแพคเกจที่ถูกรี้อปติดต่อกัน และ  $x_n$  เป็นค่าของ  $X$  ซึ่งเป็นจำนวนครั้งที่มิแพคเกจถูกรี้อปติดต่อกัน  $n$  แพคเกจ และ  $\sum_{n=1}^{\infty} x_n$  เป็นจำนวนครั้งที่เกิดการครี้อปติดต่อกันทั้งหมด จะหาความน่าจะเป็นที่จะเกิดการครี้อปติดต่อกัน  $n$  แพคเกจได้เท่ากับ  $\frac{x_n}{\sum_{n=1}^{\infty} x_n}$  และค่าเฉลี่ยได้เท่ากับ  $E(X)$  ดังตัวอย่างในรูปที่ 5.13

ส่วนการหาค่าเฉลี่ยของค่าประวิงเวลาที่เกิดจากการรอคอยในคิวในบัฟเฟอร์ ถ้า  $T''$  เป็นค่าประวิงเวลาที่เกิดจากการรอคอยในคิวในบัฟเฟอร์ และ  $t_m''$  เป็นค่าของ  $T''$  ที่เป็นของแพคเกจที่มาถึงเกตเวย์ในลำดับที่  $m$  จะได้ว่า  $E(T'')$  เป็นค่าเฉลี่ยของค่าประวิงเวลาที่เกิดจากการรอคอยในคิวในบัฟเฟอร์ ดังตัวอย่างในรูปที่ 5.14

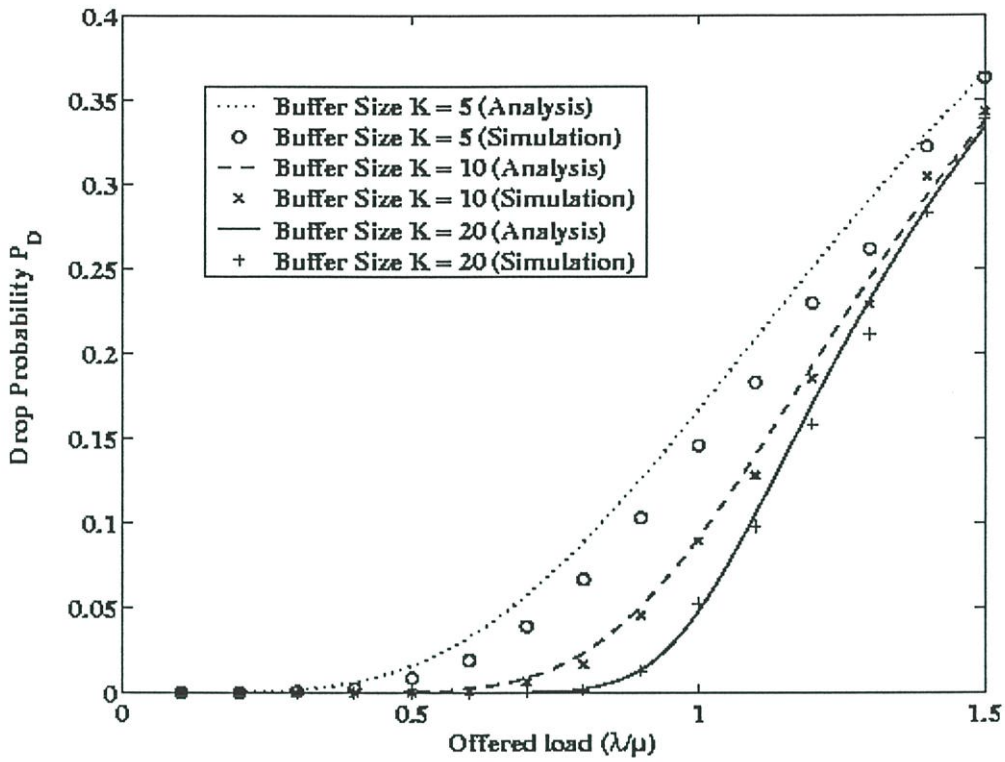
## 5.5 ตัวอย่างผลการจำลองเหตุการณ์ในระบบคิวแบบ M/M/1/K

จากงานวิจัยอื่นก่อนหน้าเช่น [4] มีการอ้างอิงว่าพฤติกรรมของการเข้าคิวที่มีการจัดการกลไกจัดการคิวบัฟเฟอร์แบบ Tail Drop มีความสอดคล้องกับระบบคิวแบบ M/M/1/K ซึ่งพิสูจน์ได้ด้วย

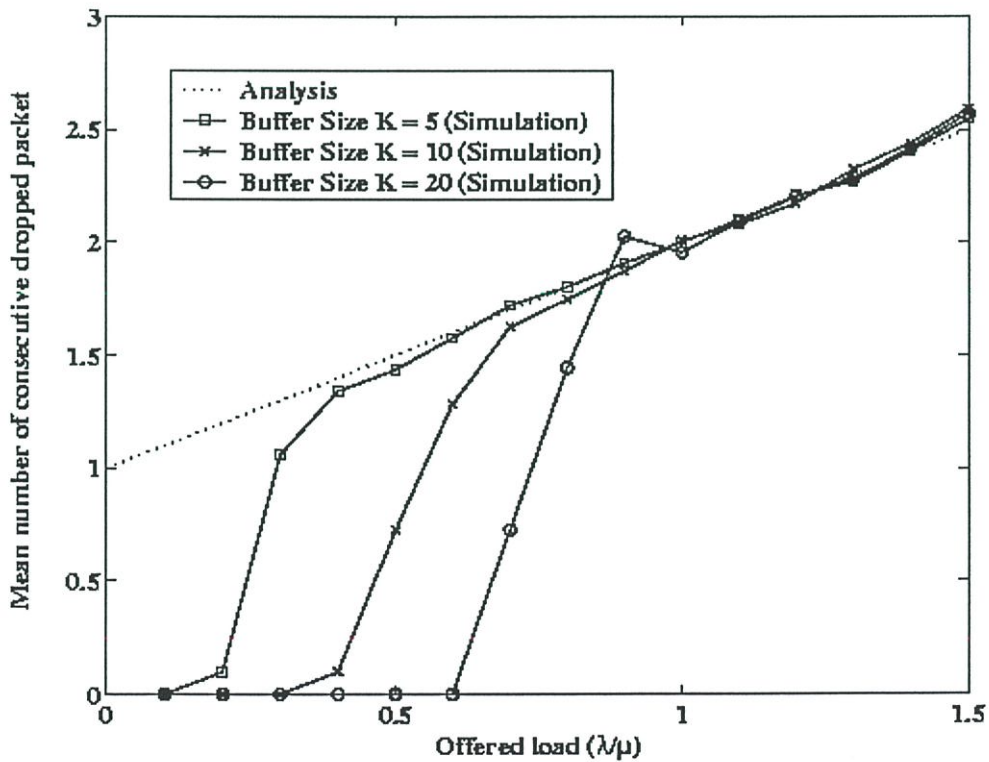
การใช้สมการที่ (3.1) แทนในสมการที่ (2.19) เมื่อจัดการรูปอนุกรมแล้วจะได้สมการของการกระจายของขนาดคิวในบัฟเฟอร์ดังสมการที่ (3.2) ซึ่งเป็นสมการในระบบคิวแบบ M/M/1/K ดังนั้นการพิสูจน์กระบวนการในการจำลองเหตุการณ์เพื่อเปรียบเทียบกับผลเชิงเลขที่ได้จากกระบวนการทางคณิตศาสตร์ เป็นการเพิ่มความน่าเชื่อถืออย่างหนึ่งในการนำกระบวนการจำลองดังกล่าวไปใช้ในการยืนยันผลอื่นๆต่อไป จาก รูปที่ 5.11 ถึง 5.14 เป็นตัวอย่างผลการจำลองระบบคิวในบัฟเฟอร์ขนาดความจุ  $K = 5, 10$  และ  $20$  แพคเกจ ซึ่งแสดงเปรียบเทียบผลของค่าเฉลี่ยขนาดคิว ความน่าจะเป็นที่จะเกิดการครี้อป จำนวนแพคเกจครี้อปติดต่อกัน และค่าประวิงเวลาตามลำดับ โดยผลดังกล่าวเกิดเมื่อมีการสร้างภาระให้ (Offered Load;  $\lambda/\mu$ ) ตั้งแต่ 0 ถึง 1.5



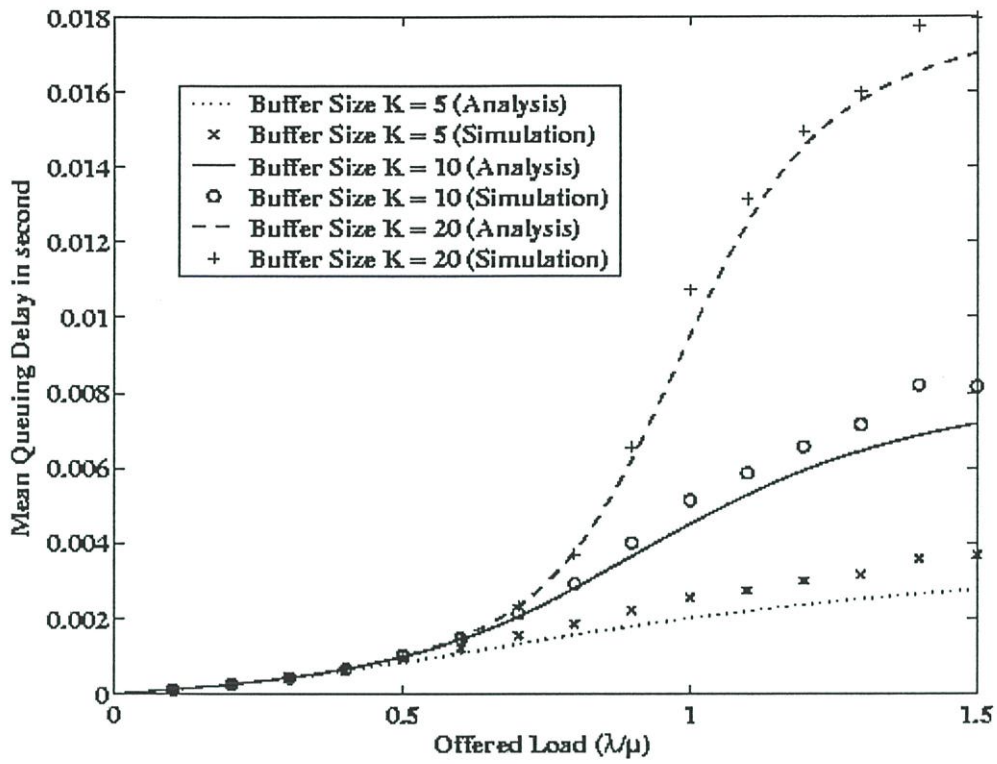
รูปที่ 5.11 ตัวอย่างผลการหาค่าเฉลี่ยขนาดคิวในระบบคิว M/M/1/K



รูปที่ 5.12 ตัวอย่างผลการหาค่าความน่าจะเป็นที่จะเกิดการครี้อปในระบบคิวแบบ M/M/1/K



รูปที่ 5.13 ตัวอย่างผลการหาค่าเฉลี่ยจำนวนแพคเกจที่ครี้อปติดต่อกันในระบบคิวแบบ M/M/1/K



รูปที่ 5.14 ตัวอย่างผลค่าประวิงเวลาจากการรอคอยในคิวในบัฟเฟอร์ของระบบคิวแบบ M/M/1/K

## บทที่ 6

### วิเคราะห์ และสรุปผล

#### 6.1 การกำหนดพารามิเตอร์

ก่อนทำการจำลองเพื่อวิเคราะห์ผล ต้องกำหนดพารามิเตอร์ในการจำลองสำหรับกลไกแต่ละแบบเพื่อทำการเปรียบเทียบผล นอกจากเงื่อนไขที่ได้จากสมการที่ (4.10) ที่เป็นเงื่อนไขในการทำงานของกลไกแบบ ExRED แล้ว พารามิเตอร์ดังกล่าวต้องสอดคล้องกับกลไกแบบอื่นๆ ที่ถูกใช้เปรียบเทียบด้วย สำหรับกลไกแบบ Tail Drop นั้นเป็นกลไกแบบง่ายๆ ที่ไม่มีเงื่อนไขใดในการกำหนดค่าพารามิเตอร์ แต่ในกลไกแบบ RED มีเงื่อนไขที่ถูกระบุใน [1] และ [7] ไว้ดังนี้

$$4k_l \leq K \quad (6.1)$$

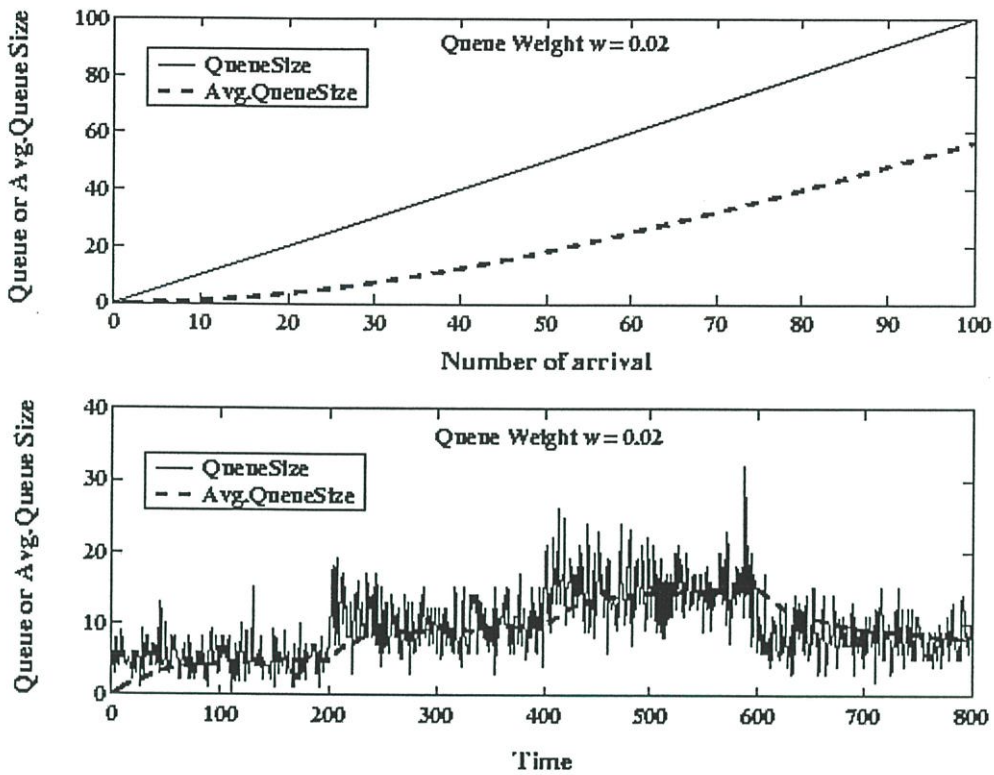
$$2k_l \leq k_h \ll K \quad (6.2)$$

$$\max_p < 0.2 \quad (6.3)$$

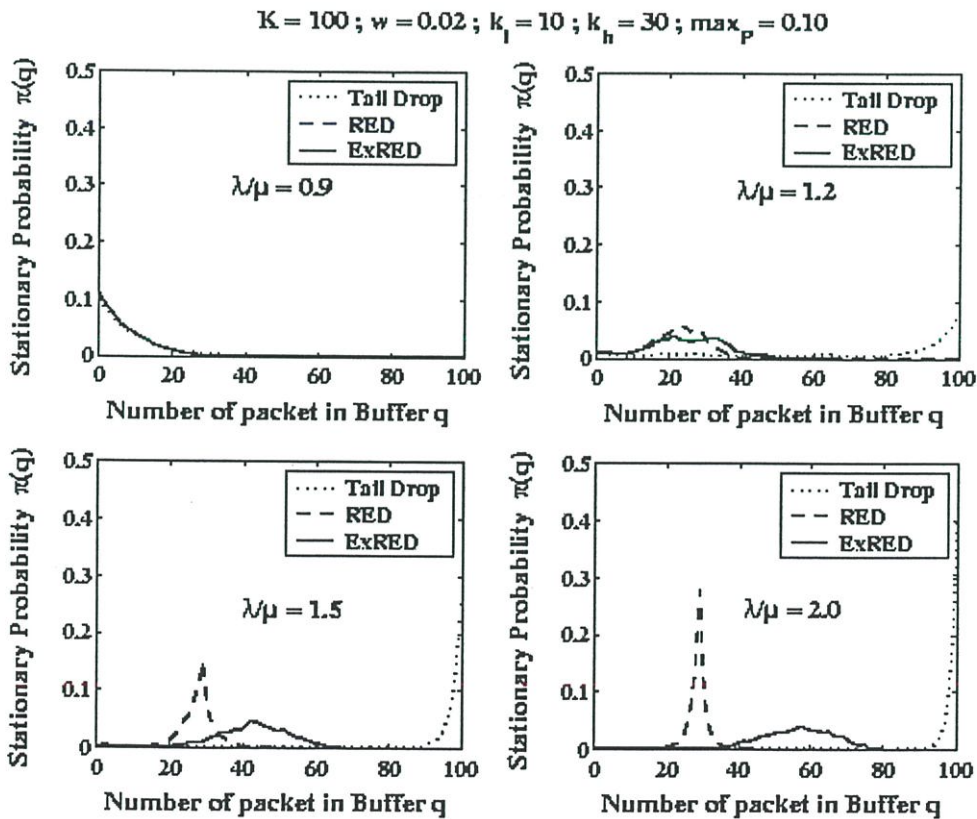
เพื่อให้การกำหนดค่าพารามิเตอร์ดังกล่าวสอดคล้องกับเงื่อนไขในสมการ (4.10), (6.1), (6.2) และ (6.3) ดังนั้นพารามิเตอร์ในงานวิจัยนี้จะกำหนดให้

- ขนาดบัฟเฟอร์  $K$  เท่ากับ 100 แพคเกจ
- อัตราการให้บริการ  $\mu$  เท่ากับ 1000 แพคเกจต่อวินาที
- ค่ากำหนดขั้นต่ำ  $k_l$  และขั้นสูง  $k_h$  เท่ากับ 10 แพคเกจและ 30 แพคเกจตามลำดับ
- ค่าความน่าจะเป็นสูงสุดที่แพคเกจจะถูกครีโปกโดยกลไก  $\max_p = 0.1$

สำหรับการกำหนดน้ำหนักถ่วงในการคำนวณขนาดคิวเฉลี่ย ( $w$ ) ถ้าสมมติให้ช่วงเวลาเริ่มต้นที่เวลา  $t = 0$  เกตเวย์มีบัฟเฟอร์คิวว่างเปล่า โดยมีค่าขนาดคิว  $q$  และขนาดคิวเฉลี่ย  $\hat{q}$  เป็นศูนย์ ทั้งกลไกแบบ RED และ ExRED ซึ่งเป็นกลไกแบบแอคทีฟจะเริ่มสุ่มบล็อกและครีโปกแพคเกจเมื่อมีแพคเกจแบบเบิร์สต์ (Burst) ขนาดเกินหนึ่งในสามของขนาดบัฟเฟอร์ทั้งหมดเข้ามาในคิว จากการคำนวณค่าน้ำหนักถ่วงเพื่อใช้คำนวณขนาดคิวเฉลี่ยจากสมการ (3.11) จะได้ขอบเขตของการกำหนดน้ำหนักถ่วงในการคำนวณขนาดคิวเฉลี่ยได้เป็น  $w \leq 0.021843$  ดังนั้นในวิทยานิพนธ์นี้จะเลือกใช้  $w = 0.02$  ซึ่งรูปที่ 6.1 แสดงตัวอย่างกราฟเปรียบเทียบระหว่างขนาดคิวเฉลี่ย (Avg. Queue Size) ที่คำนวณได้ กับขนาดคิวที่เกิดขึ้น (Queue Size) รูปที่ 6.1 (รูปบน) เมื่อมีแพคเกจแบบเบิร์สต์ขนาด 100 แพคเกจ ส่วนรูปที่ 6.1 (รูปล่าง) เมื่อขนาดคิวเกิดขึ้นโดยการสร้างเลียนแบบเหตุการณ์จริง



รูปที่ 6.1 กราฟเปรียบเทียบขนาดคิวเฉลี่ยที่คำนวณได้ กับขนาดคิวที่เกิดขึ้น



รูปที่ 6.2 กราฟการกระจายขนาดคิวที่สถานะระบบคงตัว (Stationary Probability Distribution)

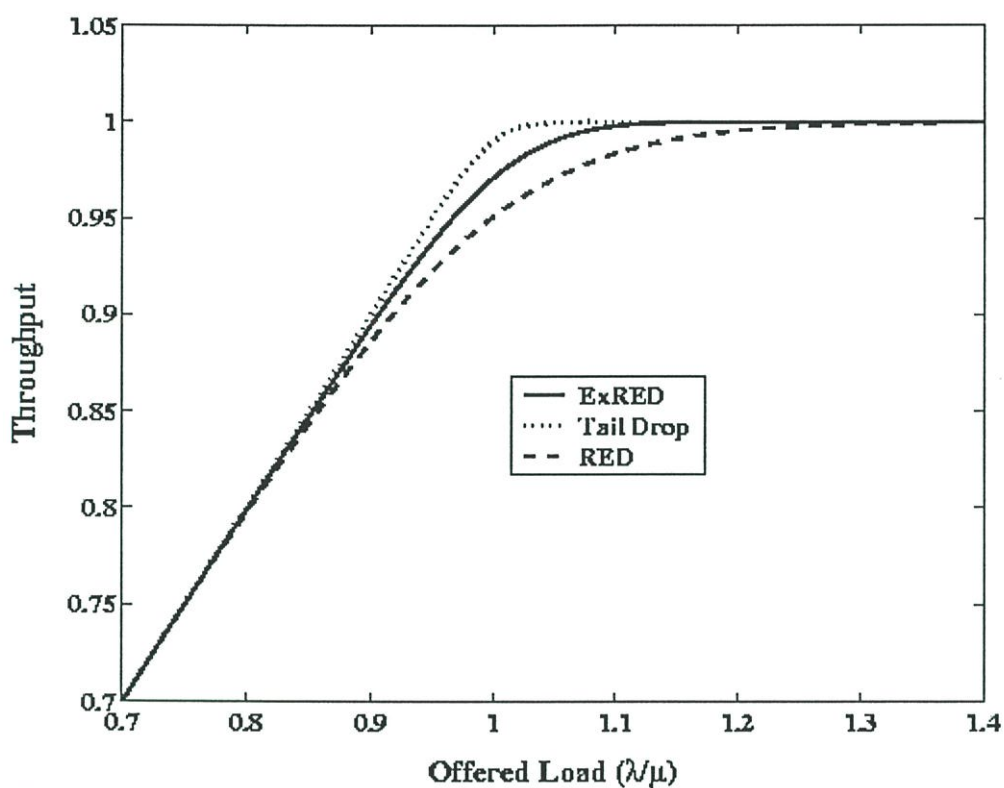
สำหรับตัวอย่างการกำหนดพารามิเตอร์แบบอื่นๆจะแสดงผลการจำลองในหัวข้อที่ 6.5 ซึ่งจะกำหนดให้พารามิเตอร์ค่าน้ำหนักถ่วงเพื่อคำนวณขนาดคิวเฉลี่ย ( $w$ ) และค่าความน่าจะเป็นสูงสุดที่แพ็คเกจจะถูกครี้อปโดยกลไก ( $\max_p$ ) คงที่ แต่จะเปลี่ยนค่ากำหนดขั้นต่ำ ( $k_l$ ) และขั้นสูง ( $k_h$ )

## 6.2 การวิเคราะห์ทรูพุทและการสูญเสียจากการครี้อป

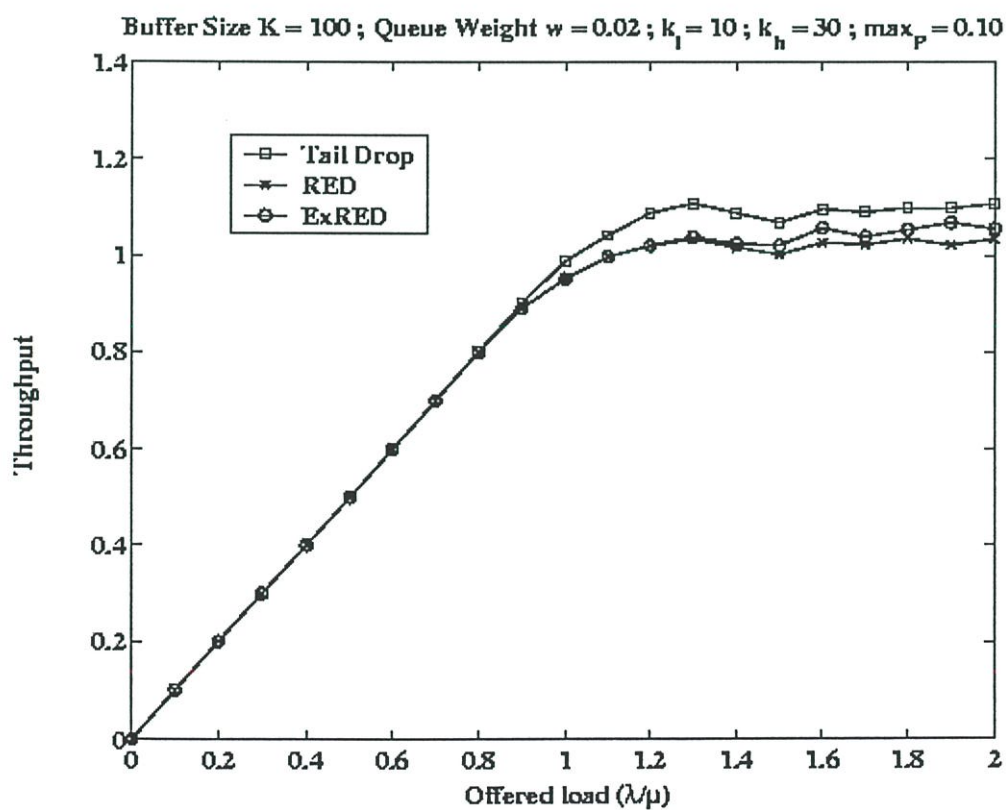
จากตารางที่ 6.1 เปรียบเทียบสมการ (3.3) และ (3.16) โดยแสดงให้เห็นว่าที่สถานะระบบคงตัว (Steady-State) กลไกแบบ RED จะมีการครี้อปแพ็คเกจมากกว่ากลไกแบบ Tail Drop ซึ่งเป็นผลจากการกระจายการครี้อปล่วงหน้าเพื่อป้องกันการเกิดบัฟเฟอร์ล้นและอีกหลายปัญหาที่จะตามมา แม้จะมีผลดีคือช่วยแก้ปัญหาดังกล่าวของกลไกแบบ Tail Drop แต่ก็ทำให้เกิดผลเสียคือ ทรูพุทต่ำ เมื่อเปรียบเทียบสมการที่ (4.12) ซึ่งแสดงการครี้อปของกลไกแบบ ExRED จะเห็นว่าโอกาสที่จะเกิดการครี้อปของกลไกแบบ ExRED จะต่ำกว่า ทำให้ทรูพุทของกลไกแบบ ExRED สูงกว่ากลไกแบบ RED แต่ยังคงกระจายการครี้อปและป้องกันการเกิดบัฟเฟอร์ล้นได้ รูปที่ 6.3 และ 6.4 แสดงทรูพุทเปรียบเทียบกลไกทั้ง 3 แบบโดยการจำลองทางคณิตศาสตร์และการจำลองเหตุการณ์ตามลำดับ

ตารางที่ 6.1 แสดงสมการเปรียบเทียบ โอกาสเกิดการครี้อปของกลไกทั้ง 3 แบบ

กลไกจัดการคิว	ความน่าจะเป็นที่จะเกิดการครี้อป
Tail Drop	$P_{drop} = \pi(K)$
RED	$P_{drop} = \sum_{\hat{q}=k_h}^K \pi(\hat{q}) + \sum_{\hat{q}=k_l}^{k_h-1} (p_d(\hat{q}) \times \pi(\hat{q}))$
ExRED (กลไกที่นำเสนอ)	$P_{drop} = \sum_{\hat{q}=k_h}^K \pi(\hat{q}) \times (a_2 \hat{q}^2 + a_1 \hat{q} + a_0) + \sum_{\hat{q}=k_l}^{k_h-1} p_d(\hat{q}) \times \pi(\hat{q})$ <p>โดยที่</p> $a_2 = \frac{(k_h - k_l) - (K - k_l) \max_p}{(k_h - k_l)(K - k_h)^2}$ $a_1 = \frac{(k_h^2 + K^2 - 2k_h k_l) \max_p - 2k_h^2 + 2k_h k_l}{(k_h - k_l)(K - k_h)^2}$ $a_0 = \frac{k_h^3 - k_h^2 k_l - (k_h^2 + 2k_h k_l + k_l K) K \max_p}{(k_h - k_l)(K - k_h)^2}$



รูปที่ 6.3 แสดงทราฟฟิคเปรียบเทียบกันระหว่างกลไกทั้ง 3 แบบ จากการจำลองทางคณิตศาสตร์



รูปที่ 6.4 แสดงทราฟฟิคเปรียบเทียบกันระหว่างกลไกทั้ง 3 แบบ จากการจำลองเหตุการณ์

จากรูปที่ 6.3 ภาระงานที่ถูกจ่ายให้เครือข่าย (Offered Load) ถูกแสดงแทนด้วย  $\lambda/\mu$  โดยที่  $\lambda$  คือ อัตราเฉลี่ยของการมาของแพ็คเกจ และ  $\mu$  คือ อัตราเฉลี่ยของการให้บริการของเกตเวย์ เมื่อมีภาระงานต่ำทุกแพ็คเกจจะได้รับการบริการ ดังนั้นทรูพุทที่ออกไปจึงมีค่าเท่ากับอัตราการมาถึงของแพ็คเกจทั้งหมด แต่เมื่อภาระงานสูงขึ้น จะเริ่มมีแพ็คเกจ โคนบล็อกและครีอป

ในกรณีของกลไกแบบ Tail Drop แพ็คเกจจะเริ่มถูกครีอปเมื่อบัฟเฟอร์ล้น ซึ่งบัฟเฟอร์ล้นจะเกิดเมื่อมีการมาถึงของแพ็คเกจในอัตราสูงกว่าอัตราการให้บริการของเกตเวย์ในระยะเวลาหนึ่ง เมื่อเกิดบัฟเฟอร์ล้นจะทำให้การบล็อกและมีการครีอปจะเกิดอย่างต่อเนื่อง เกิดผลกระทบในวงกว้างดังที่ได้กล่าวไปแล้วในบทก่อนหน้านี้ ดังนั้นแม้ว่าจะมีทรูพุทโดยเฉลี่ยสูง แต่ก็มีปัญหาที่เกิดขึ้นมาก

ในกรณีของกลไกแบบ RED และ ExRED แพ็คเกจจะเริ่มถูกครีอปแบบสุ่มเมื่อขนาดคิวเฉลี่ยในบัฟเฟอร์  $\hat{Q}$  ถึงค่ากำหนดขั้นต่ำ ดังนั้นการครีอปจะเกิดก่อนบัฟเฟอร์ล้น เป็นการครีอปแบบไม่ต่อเนื่อง ซึ่งไม่ทำให้เกิดผลกระทบที่รุนแรงและเป็นวงกว้างเท่าผลจากบัฟเฟอร์ล้น ในการพิจารณาโอกาสเกิดการบล็อกจากบัฟเฟอร์ล้น จากสมการที่ (2.19) ถ้าโอกาสเกิดการบล็อกจากบัฟเฟอร์ล้นเท่ากับ  $P_{Overflow} = \{\pi(q) | q = K\}$  และจากสมการที่ (2.17) และ (2.18) จะได้ว่า

$$\sum_{q=0}^K \pi(q) = 1 \quad (6.4)$$

สำหรับกลไกแบบ Tail Drop เมื่อ  $Q = \{q | 0 \leq q < K\}$  จะมี  $p_d = 0$  เมื่อ  $Q = \{q | q = K\}$  จะมี  $p_d = 1$  ในขณะที่กลไกแบบ RED และ ExRED เมื่อ  $Q = \{q | \hat{q} < k_l\}$  จะมี  $p_d = 0$  แต่เมื่อ  $Q = \{q | \hat{q} \geq k_l\}$  จะมี  $0 < p_d \leq 1$  จากสมการที่ (6.4) จะแสดงโดยนัยได้ว่า  $P_{Overflow}$  ของกลไกแบบ RED และ ExRED จะต่ำกว่ากลไกแบบ Tail Drop ดังรูปที่ 6.2 ซึ่งสรุปได้ว่าทั้งกลไกแบบ RED และ ExRED กระจายการครีอปและป้องกันการเกิดบัฟเฟอร์ล้นได้ ส่วนการเปรียบเทียบระหว่างกลไกแบบ RED และ ExRED จะแสดงโดยการจำลองเหตุการณ์การเกิดการครีอปต่อเนื่อง

ในกรณีกลไกแบบ RED แม้ว่าเมื่อขนาดคิวเฉลี่ยในบัฟเฟอร์  $\hat{Q}$  สูงถึงค่ากำหนดขั้นสูง ซึ่งแพ็คเกจจะถูกครีอปหมดซึ่งผลที่ปรากฏออกมาจะเหมือนการเกิดบัฟเฟอร์ล้น แต่เมื่อเปรียบเทียบกับกลไก Tail Drop กว่าที่จะถึงจุดนี้ ก็ต้องผ่านหลายขั้นตอนในการพยายามที่จะชะลออัตราการเพิ่มของขนาดคิวมาแล้ว อีกทั้งการเปลี่ยนแปลงของขนาดคิวเฉลี่ยในบัฟเฟอร์  $\hat{Q}$  นั้นมีความไว้น้อยกว่าการเปลี่ยนแปลงของขนาดคิวจริงเพราะ โคนหน่วงด้วยน้ำหนักถ่วง  $w$  จึงสรุปได้โดยนัยว่าการเกิดเหตุการณ์นี้ของกลไกแบบ RED มีโอกาสน้อยกว่าการเกิดบัฟเฟอร์ล้นในกลไกแบบ Tail Drop

แต่อย่างไรก็ตามเมื่อเปรียบเทียบกับกลไกแบบ ExRED จะเห็นว่าทรูพุทของกลไกแบบ ExRED สูงกว่า จากหัวใจสำคัญในการปรับปรุงข้อเสียของกลไกแบบ RED ที่ถูกนำเสนอในรูปแบบของกลไกแบบ ExRED คือการปรับปรุงความยืดหยุ่นในการใช้บัฟเฟอร์ จากสมมติฐานที่ว่ากรณีทรูพุทต่ำของกลไกแบบ RED นั้นอาจมาจากการครอบที่มากเกินไปจนทำให้บัฟเฟอร์เต็ม กลไกแบบ ExRED จึงปรับรูปแบบการครอบโดยให้ความน่าจะเป็นที่แพ็คเกจจะถูกครอบมีลักษณะเพิ่มขึ้นแบบฟังก์ชันโพลิโนเมียลกำลังสอง ทำให้บัฟเฟอร์สามารถถูกใช้ได้มากขึ้นในขณะที่เดียวกันก็ยังสามารถป้องกันการเกิดบัฟเฟอร์เต็มและการเกิดบัฟเฟอร์ล้นได้ ดังนั้นเมื่อเกิดการครอบจากการบล็อกโดยกลไกน้อยลง ก็จะทำให้ทรูพุทสูงขึ้น

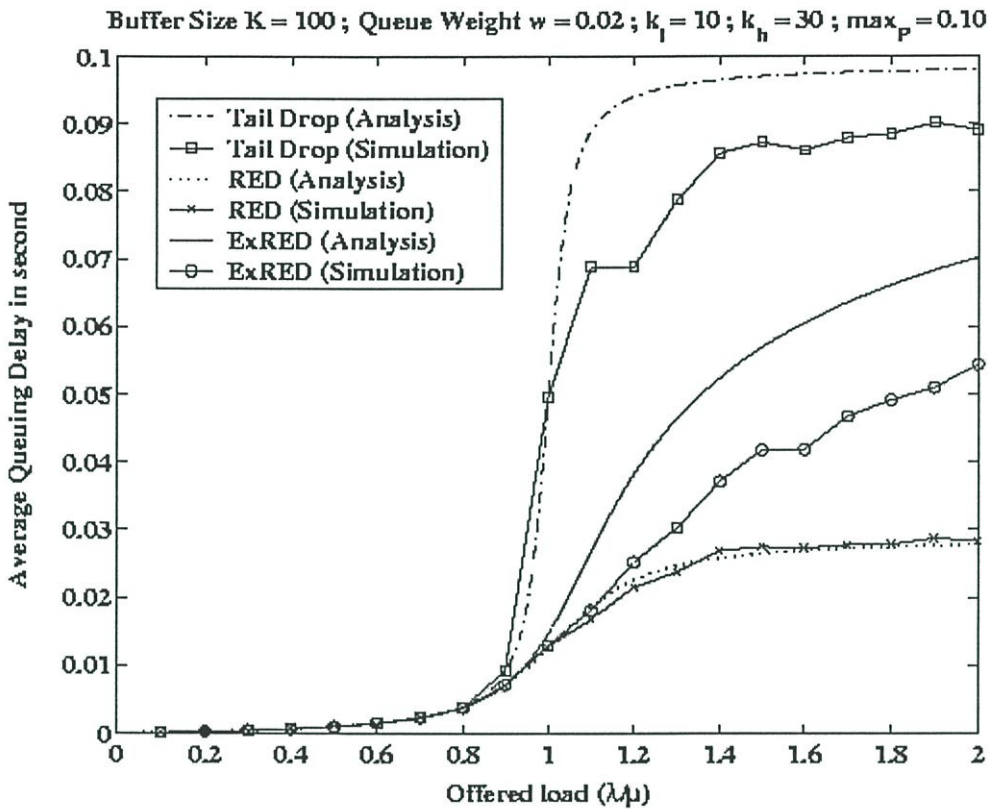
นอกจากนี้ผลการจำลองเหตุการณ์ในรูปที่ 6.4 ก็แสดงให้เห็นว่าทรูพุทของกลไกแบบ ExRED สูงกว่ากลไกแบบ RED ซึ่งมีแนวโน้มในทางเดียวกันกับผลการจำลองทางคณิตศาสตร์ในรูปที่ 6.3 และสนับสนุนการวิเคราะห์ข้างต้น

### 6.3 การวิเคราะห์ค่าประวิงเวลาจากการรอคอยในคิว

ก่อนจะแสดงการวิเคราะห์และผลการครอบที่ติดต่อกันซึ่งแสดงข้อดีอีกส่วนหนึ่งของกลไกแบบ ExRED ในส่วนถัดไป ในส่วนนี้จะแสดงผลและการวิเคราะห์ค่าประวิงเวลาจากการรอคอยในคิวในบัฟเฟอร์ ซึ่งเป็นสิ่งที่ต้องแลกเปลี่ยน (Trade-Off) ที่หลีกเลี่ยงไม่ได้ระหว่างทรูพุทที่ดีขึ้น แต่มีค่าประวิงเวลาที่มากขึ้น

สมการที่ (2.22) แสดงให้เห็นว่าค่าประวิงเวลา  $D$  แปรผันตรงกับค่าเฉลี่ยคิว  $E(Q)$  เมื่อพิจารณารูปที่ 6.5 ซึ่งแสดงค่าประวิงเวลาที่เกิดจากการรอคอยในคิวในบัฟเฟอร์ จะเห็นว่าเมื่อภาระงานยังไม่สูงมากเกิดเวลายังสามารถให้บริการได้ทัน มีการสะสมของแพ็คเกจที่ต้องรอคอยในคิวต่ำ ทำให้มีค่าประวิงเวลาต่ำ แต่เมื่อภาระงานเริ่มสูงขึ้นมีการสะสมมากขึ้นทำให้ขนาดคิวสูงขึ้น และทำให้เกิดการรอคอยยาวนานขึ้น จนกว่าบัฟเฟอร์จะเต็ม หรือเกิดการบล็อกโดยสิ้นเชิง

กลไกแบบ Tail Drop ปล่อยให้มีการใช้บัฟเฟอร์อย่างเต็มที่ จนกว่าจะเต็ม แล้วจึงค่อยบล็อกและครอบแพ็คเกจที่เกินมา ซึ่งในลักษณะนี้จะทำให้เกิดคิวในบัฟเฟอร์สูงจนกระทั่ง  $E(Q) \rightarrow K$  ซึ่งเป็นขนาดบัฟเฟอร์ ดังนั้นค่าประวิงเวลาในการรอคอยต่อคิวในบัฟเฟอร์จะเข้าสู่ค่า  $\frac{K-1}{\mu}$



รูปที่ 6.5 แสดงค่าประวิงเวลาเปรียบเทียบกันระหว่างกลไกทั้ง 3 แบบ

กลไกแบบ RED จะครอบแพคเก็ตเพื่อรักษาระดับคิวในบัฟเฟอร์ จึงทำให้ขนาดคิวถูกควบคุมให้อยู่ในระดับหนึ่ง ซึ่งเป็นผลให้ค่าประวิงเวลาถูกควบคุมให้อยู่ในระดับต่ำไปด้วย เนื่องจากหัวใจสำคัญในการปรับปรุงข้อเสียของกลไกแบบ RED ที่ถูกนำเสนอในรูปแบบของกลไกแบบ ExRED คือ การปรับปรุงทำให้ใช้บัฟเฟอร์ได้มากขึ้น ทำให้มีการครอบแพคเก็ตการบดบังโดยกลไกน้อยลง ซึ่งจะส่งผลให้มีทรูพุทสูงขึ้นดังที่ได้แสดงไปแล้ว และจากเหตุผลที่กล่าวไปในตอนต้น จึงทำให้ค่าประวิงเวลาจากการรอคอยในคิวของกลไกแบบ ExRED สูงกว่ากลไกแบบ RED แม้ว่ากลไกแบบ ExRED จะมีลักษณะยืดหยุ่นกว่ากลไกแบบ RED แต่เป็นกลไกที่มีการใช้บัฟเฟอร์ที่มีการควบคุมจึงทำให้ค่าประวิงเวลาดำกว่าของกลไกแบบ Tail Drop อยู่มาก

แต่ถ้าพิจารณาค่าประวิงเวลาจากปลายข้างหนึ่งถึงอีกข้างหนึ่ง (End-to-End Delay) ในสถานะที่เครือข่ายมีค่าประวิงเวลาในส่วนอื่นอันได้แก่ ค่าประวิงเวลาจากการแพร่กระจาย (Propagation Delay) ค่าประวิงเวลาจากการส่ง (Transmission Delay) ฯลฯ ของช่วงใดช่วงหนึ่งในเส้นทางปลายถึงปลาย (End-to-End Path) มีค่าสูง อาจทำให้ค่าประวิงเวลารวมของกลไกแบบ ExRED มีค่าน้อยกว่ากลไกแบบ RED ได้

## 6.4 การวิเคราะห์การเกิดการครีโปปติดต่อกัน

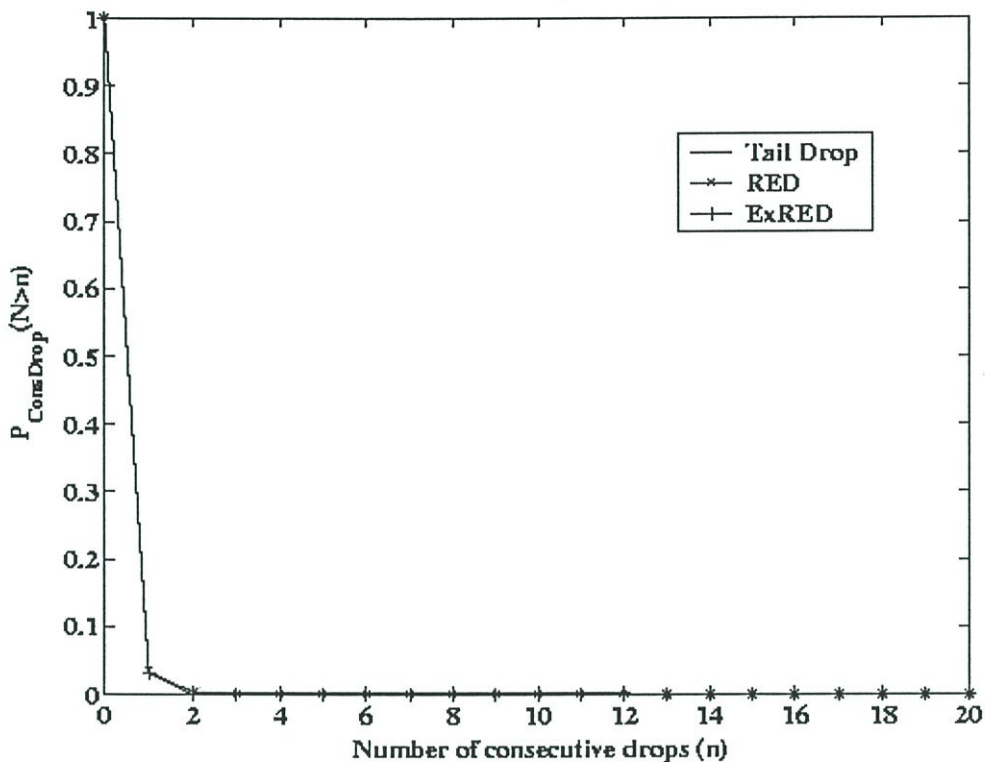
เมื่อพิจารณาความน่าจะเป็นที่จะเกิดการครีโปปติดต่อกันของกลไกแบบ Tail Drop จากสมการที่ (3.7) จะพบว่าถ้าภาระงาน (Offered Load) สูงจะทำให้ความน่าจะเป็นที่แพ็คเกจในลำดับถัดไปที่มาถึงจะถูกครีโปป ( $p$ ) มีค่าเท่ากับหนึ่งดังสมการที่ (6.5)

$$\lim_{\frac{\lambda}{\mu} \rightarrow \infty} p = \lim_{\frac{\lambda}{\mu} \rightarrow \infty} \frac{\frac{\lambda}{\mu}}{1 + \frac{\lambda}{\mu}} = 1 \quad (6.5)$$

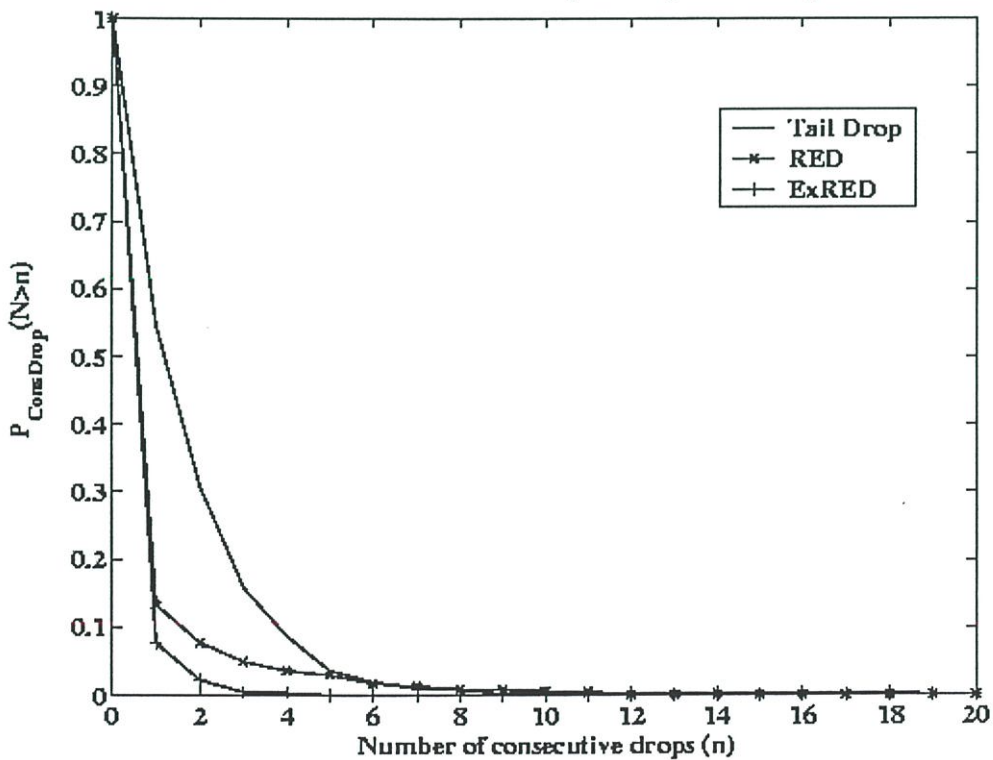
ซึ่งจะทำให้ความน่าจะเป็นที่จะเกิดการครีโปปต่อเนื่องอย่างยาวนาน (ค่า  $n$  มากๆ) จากสมการ (3.8) มีค่ายิ่งเข้าใกล้หนึ่งมากขึ้น ซึ่งแสดงว่ามีโอกาสเกิดมาก นอกจากนี้สมการที่ (3.9) ยังแสดงให้เห็นว่าจำนวนแพ็คเกจเฉลี่ยที่ถูกครีโปปติดต่อกันก็เกิดขึ้นได้อย่างไม่มีขีดจำกัดเช่นกัน ( $\lim_{\frac{\lambda}{\mu} \rightarrow \infty} E(N) = \infty$ )

สำหรับกลไกแบบ RED มีการกระจายการครีโปปและชะลออัตราการมาถึงของแพ็คเกจ ทำให้มีโอกาที่จะเกิดการครีโปปติดต่อกันน้อยลง และลดจำนวนแพ็คเกจที่ถูกครีโปปติดต่อกันเฉลี่ยลงด้วย แต่อย่างไรก็ตามเมื่ออัตราการมาถึงสูงอย่างต่อเนื่องทำให้ขนาดคิวในบัฟเฟอร์ในขณะนั้นสูงอย่างต่อเนื่องด้วย และทำให้ขนาดคิวเฉลี่ย  $\hat{Q}$  มีค่าสูงถึงค่ากำหนดขั้นสูง  $k_h$  และคงระดับไว้หรือสูงขึ้นไปกว่านั้น แพ็คเกจที่มาถึงหลังจากนั้นจะถูกครีโปปทั้งหมดโดยการทำงานของกลไกแบบ RED จนกว่าอัตราการมาถึงจะลดลง จนทำให้ขนาดคิวเฉลี่ย  $\hat{Q}$  ลดลงต่ำกว่าค่ากำหนดขั้นสูง  $k_h$  ในกรณีนี้มีข้อดีคือจะรักษาระดับขนาดคิวไว้ได้ทำให้มีค่าประวิงเวลาที่เกิดจากการรอคอยในคิวในบัฟเฟอร์ที่ค่อนข้างคงที่และอยู่ในระดับต่ำ แต่มีข้อเสียคืออาจเป็นการใช้บัฟเฟอร์ที่น้อยเกินไป และจะทำให้มีทรูพุทในช่วงขณะใดๆต่ำ ดังนั้นการลดการเกิดการครีโปปติดต่อกันก็เป็นอีกวัตถุประสงค์หนึ่งในการนำเสนอกลไกแบบ ExRED ซึ่งพิสูจน์ได้ดังรูปที่ 6.6 ถึง 6.9

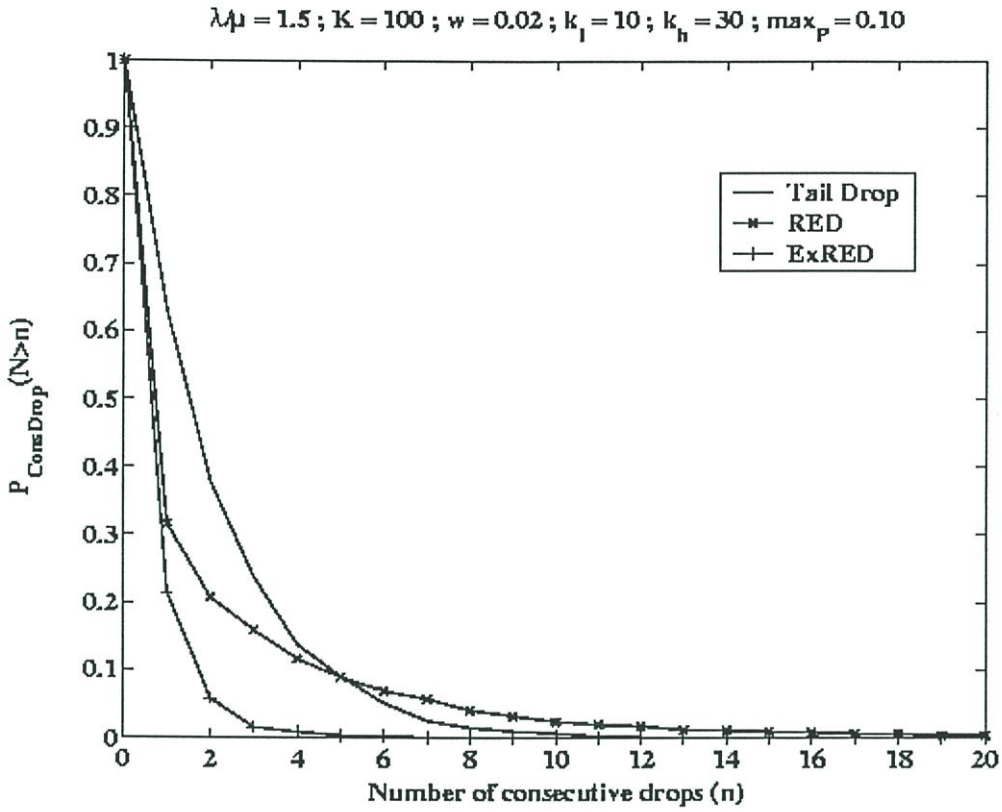
เมื่อภาระงานยังไม่สูงเกินความสามารถในการรองรับของบัฟเฟอร์ ทำให้มีโอกาสน้อยมากที่จะเกิดการครีโปปติดต่อกันยาวต่อเนื่องที่หลายแพ็คเกจดังกราฟในรูปที่ 6.6 ซึ่งได้แสดงให้เห็นว่า  $\lim_{n \rightarrow \infty} P_{drop}^{cons}(N > n) \rightarrow 0$  และแสดงให้เห็นอีกด้วยว่าแต่ละกลไกให้ผลลัพธ์ออกมาไม่แตกต่างกัน

$\lambda\mu = 0.9 ; K = 100 ; w = 0.02 ; k_l = 10 ; k_h = 30 ; \max_p = 0.10$ 


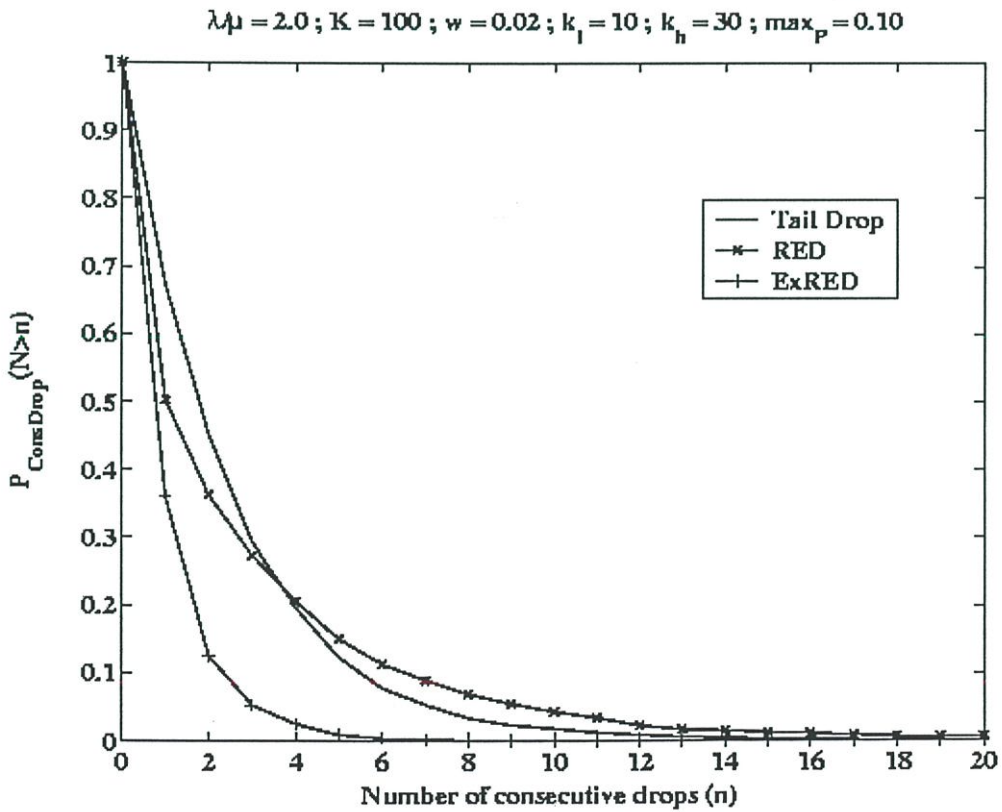
รูปที่ 6.6 แสดงความน่าจะเป็นที่จะเกิดแพกเก็ตต่อเนื่องติดต่อกัน  $n$  แพกเก็ต ( $\frac{\lambda}{\mu} = 0.9$ )

 $\lambda\mu = 1.2 ; K = 100 ; w = 0.02 ; k_l = 10 ; k_h = 30 ; \max_p = 0.10$ 


รูปที่ 6.7 แสดงความน่าจะเป็นที่จะเกิดแพกเก็ตต่อเนื่องติดต่อกัน  $n$  แพกเก็ต ( $\frac{\lambda}{\mu} = 1.2$ )



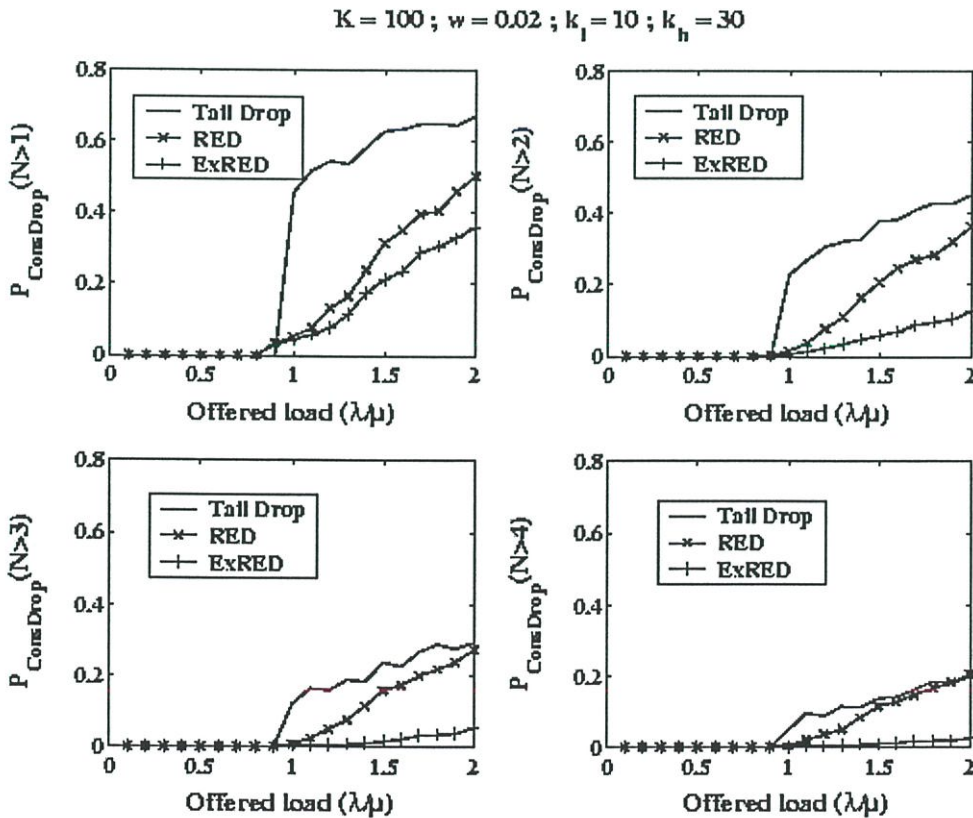
รูปที่ 6.8 แสดงความน่าจะเป็นที่จะเกิดแพคเกจติดต่อกัน  $n$  แพคเกจ ( $\frac{\lambda}{\mu} = 1.5$ )



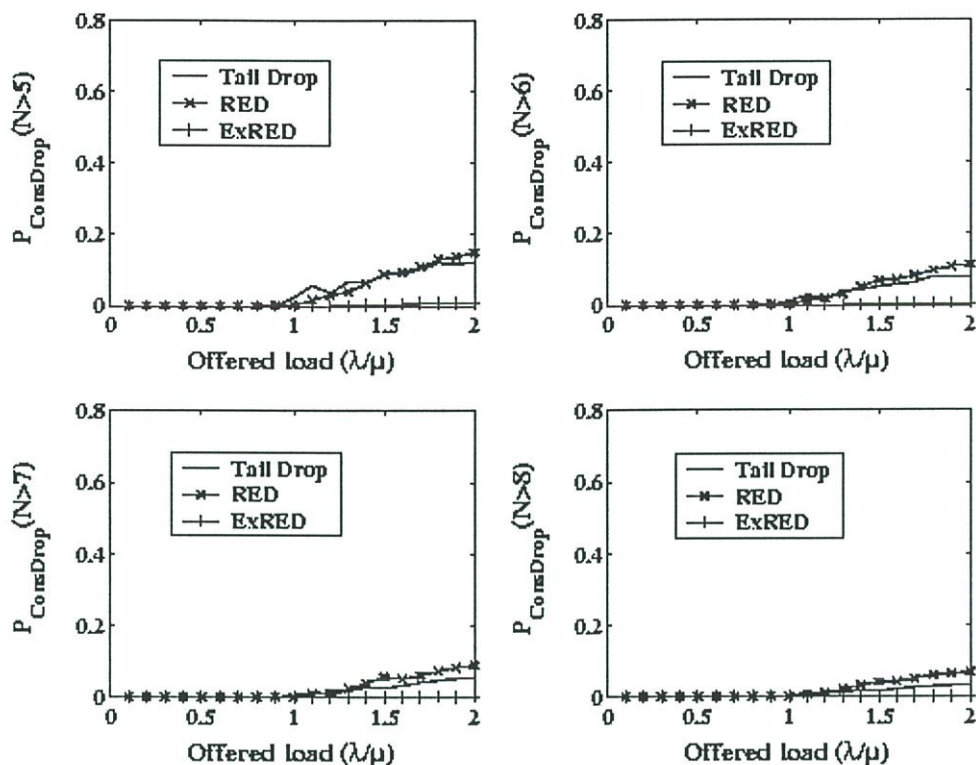
รูปที่ 6.9 แสดงความน่าจะเป็นที่จะเกิดแพคเกจติดต่อกัน  $n$  แพคเกจ ( $\frac{\lambda}{\mu} = 2.0$ )

จากกราฟรูปที่ 6.7 กลไกแบบ Tail Drop ซึ่งปล่อยให้มีการใช้บัฟเฟอร์ได้อย่างเต็มที่ ดังนั้นเมื่อภาระงานสูงจนเกินกว่าที่บัฟเฟอร์จะรองรับได้จะเกิดปัญหาบัฟเฟอร์ล้น และมีการบล็อกทำให้เกิดการครอบตัดต่อกันอย่างต่อเนื่องที่หลายแพคเกจมากขึ้น เมื่อพิจารณาจากรูปที่ 6.7 จะเห็นว่าผลจากการชะลออัตราการมาถึงโดยการครอบตัดแพคเกจล่วงหน้าก่อนที่บัฟเฟอร์จะล้นของกลไกแบบ RED ช่วยลดการเกิดแพคเกจครอบตัดต่อกันที่หลายแพคเกจได้ แต่เมื่อเปรียบเทียบกับกลไกแบบ ExRED ซึ่งมีความยืดหยุ่นกว่าในการบล็อกให้แพคเกจครอบตัด จึงทำให้กลไกแบบ ExRED ให้ผลในการลดการเกิดแพคเกจครอบตัดต่อกันที่หลายแพคเกจที่ดีกว่ากลไกแบบ RED

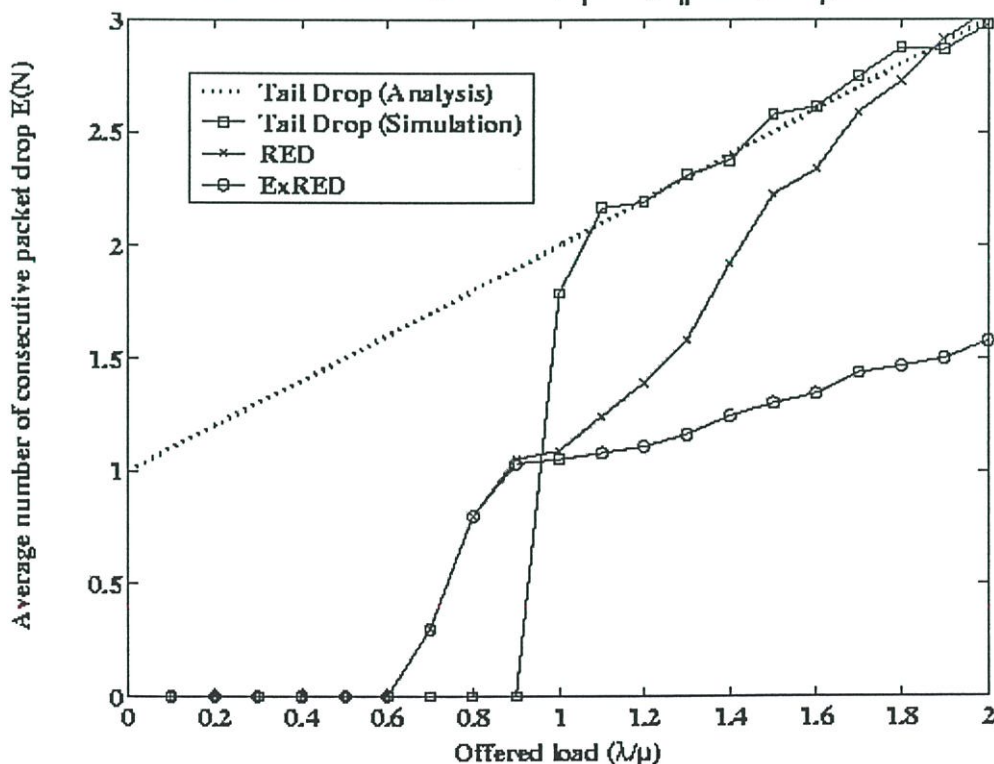
จากกราฟรูปที่ 6.8 ถึง 6.9 ปัญหาการเกิดการครอบตัดต่อกันที่หลายแพคเกจ (เมื่อ  $n$  มากๆ) ของกลไกแบบ RED จะมีความรุนแรงมากขึ้นจนเป็นปัญหาที่ถูกกล่าวถึงโดยงานวิจัยอื่นมาแล้ว เช่น [4] เป็นต้น ทำให้มีโอกาสพบจำนวนแพคเกจที่ถูกครอบตัดต่อกันต่อเนื่องที่หลายแพคเกจมากขึ้น เพื่อให้ชัดเจนยิ่งขึ้น รูปที่ 6.10 ถึง 6.11 แสดงกราฟเปรียบเทียบความแตกต่างของโอกาสเกิดการครอบตัดต่อกันของกลไกแบบ Tail Drop กลไกแบบ RED และกลไกแบบ ExRED ที่ค่า  $n$  ต่างๆเมื่อเปลี่ยนแปลงภาระงานที่สร้างให้กับเครือข่าย ส่วนรูปที่ 6.12 แสดงจำนวนแพคเกจเฉลี่ยที่จะถูกครอบตัดต่อกันอย่างต่อเนื่อง



รูปที่ 6.10 แสดงความน่าจะเป็นที่จะเกิดแพคเกจครอบตัดต่อกัน  $n$  แพคเกจ ( $n = 1$  ถึง 4)

$K = 100 ; w = 0.02 ; k_l = 10 ; k_h = 30$ 


รูปที่ 6.11 แสดงความน่าจะเป็นที่จะเกิดแพคเกจตกรอบติดต่อกัน  $n$  แพคเกจ ( $n = 5$  ถึง  $8$ )

 $\text{Buffer Size } K = 100 ; w = 0.02 ; k_l = 10 ; k_h = 30 ; \max_p = 0.10$ 


รูปที่ 6.12 ค่าเฉลี่ยของจำนวนแพคเกจที่จะถูกตกรอบติดต่อกัน  $E(N)$

จากกราฟรูปที่ 6.12 เมื่อพิจารณากราฟแบบ Tail Drop จะเห็นว่าค่าเฉลี่ยของจำนวนแพคเกจที่ถูกครีโอดักตักกันอย่างต่อเนื่องจะสูงขึ้นมากในช่วงที่ภาระงาน (Offered Load) สูงจนทำให้เกิดการสะสมของแพคเกจในบัฟเฟอร์จนเกินที่จะรองรับได้ ซึ่งเป็นผลโดยนัยจากการเกิดบัฟเฟอร์ล้น แต่กลไกแบบ RED และ ExRED มีการกระจายการครีโอดักโดยการบล็อกให้แพคเกจถูกครีโอดักล่วงหน้าก่อนที่บัฟเฟอร์จะล้น ดังนั้นเมื่อพิจารณากราฟรูปที่ 6.12 จะเห็นว่ามีความถี่ของแพคเกจที่ถูกครีโอดักแบบกระจาย (ที่สามารถระบุได้ว่ากระจายหรือมีการครีโอดักหนึ่งครั้งต่อหลายแพคเกจที่มาถึง เพราะ  $E(N) < 1$ ) ก่อนที่ขนาดคิวจะสูงจนเกินกว่าบัฟเฟอร์จะรองรับได้ ช่วยป้องกันการเกิดบัฟเฟอร์ล้น

แต่เมื่อภาระงานมากขึ้นเรื่อยๆจนขนาดคิวเฉลี่ยสูงถึงค่ากำหนดขั้นสูง  $k_h$  กลไกแบบ RED จะบล็อกทำให้ทุกแพคเกจที่มาถึงหลังจากนั้นถูกครีโอดักอย่างต่อเนื่อง ซึ่งเกิดผลคล้ายกับผลจากการเกิดบัฟเฟอร์ล้น ทำให้ค่าเฉลี่ยจำนวนแพคเกจที่จะถูกครีโอดักตักกันอย่างต่อเนื่องสูงขึ้นต่อเนื่องดังกราฟในรูปที่ 6.12 ในขณะที่กลไกแบบ ExRED ช่วยแก้ปัญหาดังกล่าวด้วยการบล็อกและครีโอดักแพคเกจอย่างยืดหยุ่น ซึ่งระดับความยืดหยุ่นแปรตามขนาดบัฟเฟอร์ที่เหลืออยู่ จึงทำให้ค่าเฉลี่ยจำนวนแพคเกจที่จะถูกครีโอดักตักกันอย่างต่อเนื่องของกลไกแบบ ExRED มีค่าน้อย

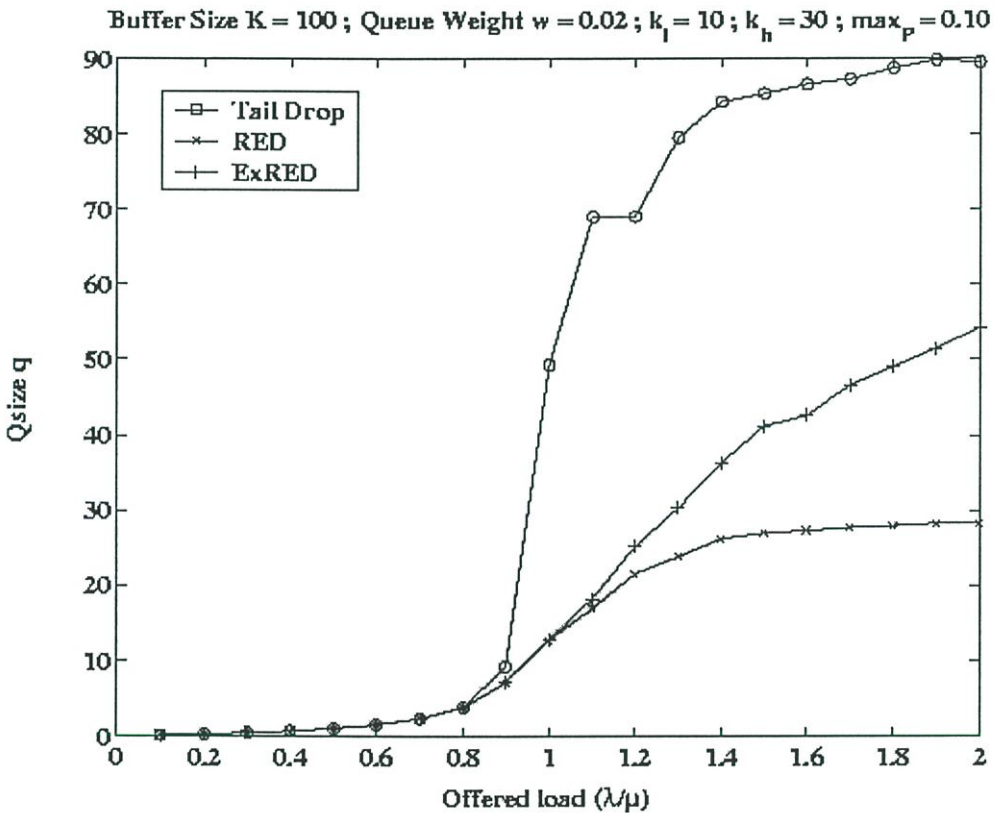
## 6.5 ตัวอย่างผลการจำลองเมื่อมีการปรับค่าพารามิเตอร์ที่ควบคุมการใช้บัฟเฟอร์

จากพารามิเตอร์ที่ใช้ในหัวข้อที่ 6.1 ถึง 6.4 มีการกระจายของคิวเป็นดังรูปที่ 6.2 และขนาดคิวเฉลี่ย  $E(Q)$  จะมีลักษณะเป็นดังรูปที่ 6.13 ซึ่งจะได้ผลดังรูปที่ 6.4 ถึง 6.10 ซึ่งในส่วนต่อไปจะทำการทดสอบการเปลี่ยนแปลงค่าพารามิเตอร์เพื่อพิจารณาว่ากลไกแบบ ExRED จะมีประสิทธิภาพกว่ากลไกแบบ RED หรือไม่ แต่อย่างไรก็ตามหลักการที่ชัดเจนในการกำหนดค่าพารามิเตอร์นั้นยังอยู่ในระหว่างการศึกษาวิจัยของผู้ที่กำลังศึกษาและพัฒนาการทำงานของกลไกแบบ RED ดังนั้นในตัวอย่างการทดสอบต่อไปนี้ จะกำหนดค่าพารามิเตอร์ให้สอดคล้องตามสมการที่ (6.1) ถึง (6.3) โดยเลือกค่าอ้างอิงตามงานวิจัยอื่น เช่น [1], [4], [5], [6] และ [7] เป็นต้น การทดสอบมี 4 แบบ ดังนี้

- แบบที่ 1 ทดลองลดค่ากำหนดขั้นต่ำ ( $k_l$ ) และขั้นสูง ( $k_h$ ) ลดลงเหลือ 5 แพคเกจ และ 20 แพคเกจตามลำดับ ซึ่งหมายความว่ากลไกแบบ RED และ ExRED จะอนุญาตให้ใช้บัฟเฟอร์ได้น้อยลงเมื่อเปรียบเทียบกับพารามิเตอร์ชุดเดิม
- แบบที่ 2 ทดลองเพิ่มค่ากำหนดขั้นต่ำ ( $k_l$ ) และขั้นสูง ( $k_h$ ) ขึ้นเป็น 15 แพคเกจ และ 50 แพคเกจตามลำดับ ซึ่งหมายความว่ากลไกแบบ RED และ ExRED จะอนุญาตให้ใช้บัฟเฟอร์ได้มากขึ้นเมื่อเปรียบเทียบกับพารามิเตอร์ชุดเดิม

- แบบที่ 3 ทดลองเปลี่ยนแปลงอัตราส่วนระหว่างค่ากำหนดชั้นสูงต่อค่ากำหนดชั้นต่ำ โดยให้ค่ากำหนดชั้นต่ำ ( $k_l$ ) และชั้นสูง ( $k_h$ ) เป็น 10 แพคเกจ และ 50 แพคเกจตามลำดับ ซึ่งหมายความว่ากลไกแบบ RED และ ExRED จะอนุญาตให้ใช้บัฟเฟอร์ได้มากขึ้นเมื่อเปรียบเทียบกับพารามิเตอร์ชุดเดิม

- แบบที่ 4 ทดลองเพิ่มความน่าจะเป็นสูงสุดที่แพคเกจจะถูกบดบังและครีอปโดยตัวกลไก ( $\max_p$ ) เป็น 0.15 ซึ่งหมายความว่ากลไกแบบ RED และ ExRED จะยอมให้เกิดโอกาสที่แพคเกจจะถูกบดบังและครีอปมากขึ้นได้ เมื่อเปรียบเทียบกับพารามิเตอร์ชุดเดิม



รูปที่ 6.13 ค่าเฉลี่ยขนาดคิว  $E(Q)$  เมื่อ  $k_l = 10$  แพคเกจ และ  $k_h = 30$  แพคเกจ

จากการทดลองแบบที่ 1 เป็นการทดลองลดค่าพารามิเตอร์เพื่อให้กลไกยอมให้ใช้บัฟเฟอร์ได้น้อยลง โดยมีรูปที่ 6.14 แสดงการกระจายของขนาดคิวหรือจำนวนแพคเกจในบัฟเฟอร์ที่สถานะคงตัว (Stationary Probability Distribution) ส่วนรูปที่ 6.15 แสดงขนาดคิวเฉลี่ย  $E(Q)$  จากการกำหนดพารามิเตอร์ใหม่ ซึ่งทั้งสองรูปแสดงให้เห็นว่าแพคเกจจะสามารถเข้าพักในบัฟเฟอร์ได้น้อยลง

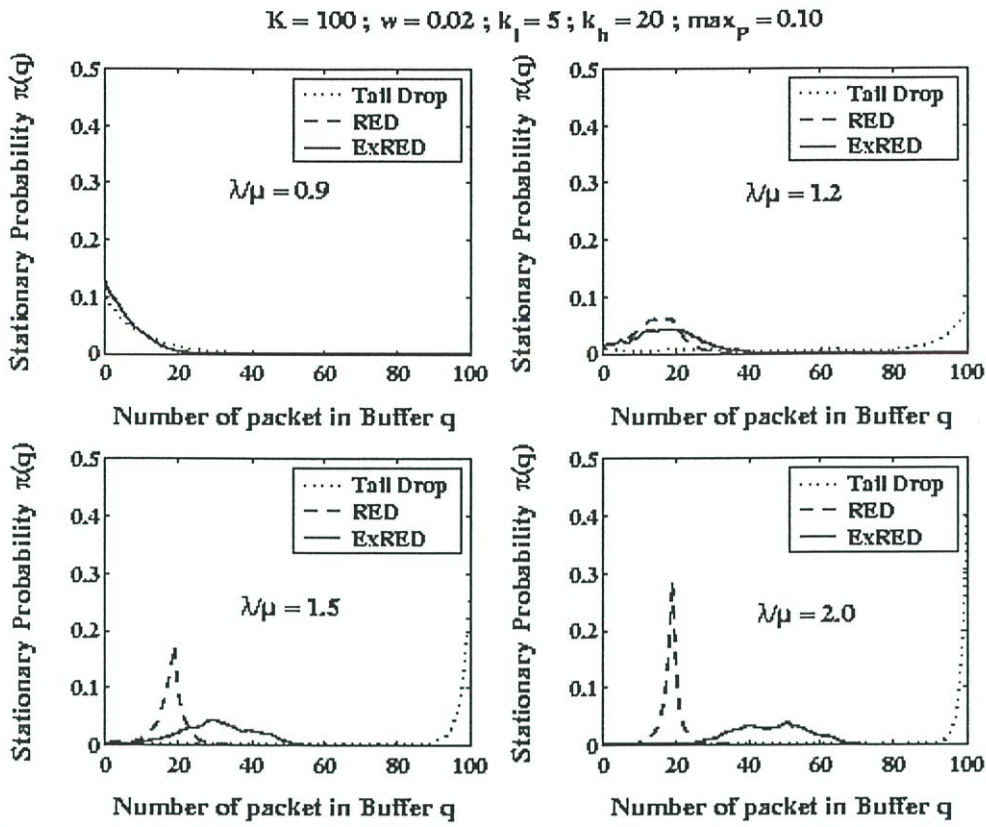
การทดลองแบบที่ 2 เป็นการทดลองเพิ่มค่าพารามิเตอร์เพื่อให้กลไกยอมให้ใช้บัฟเฟอร์ได้มากขึ้น โดยมีรูปที่ 6.16 แสดงการกระจายของขนาดคิวหรือจำนวนแพ็คเกจในบัฟเฟอร์ที่สถานะคงตัว (Stationary Probability Distribution) ส่วนรูปที่ 6.17 แสดงขนาดคิวเฉลี่ย  $E(Q)$  จากการกำหนดพารามิเตอร์ใหม่ ซึ่งทั้งสองรูปแสดงให้เห็นว่าแพ็คเกจจะสามารถเข้าพักในบัฟเฟอร์ได้มากขึ้น

การทดลองแบบที่ 3 เป็นการทดลองเพื่อศึกษาผลเมื่อมีการเปลี่ยนแปลงอัตราส่วนระหว่างค่ากำหนดขั้นสูงต่อค่ากำหนดขั้นต่ำ ( $k_h/k_l$ ) หรือยอมให้ใช้บัฟเฟอร์ได้มากขึ้น รูปที่ 6.18 แสดงการกระจายของขนาดคิวหรือจำนวนแพ็คเกจในบัฟเฟอร์ที่สถานะคงตัว (Stationary Probability Distribution) ส่วนรูปที่ 6.19 แสดงขนาดคิวเฉลี่ย  $E(Q)$  จากการกำหนดพารามิเตอร์ใหม่ ซึ่งทั้งสองรูปแสดงให้เห็นว่าแพ็คเกจจะสามารถเข้าพักในบัฟเฟอร์ได้มากขึ้น

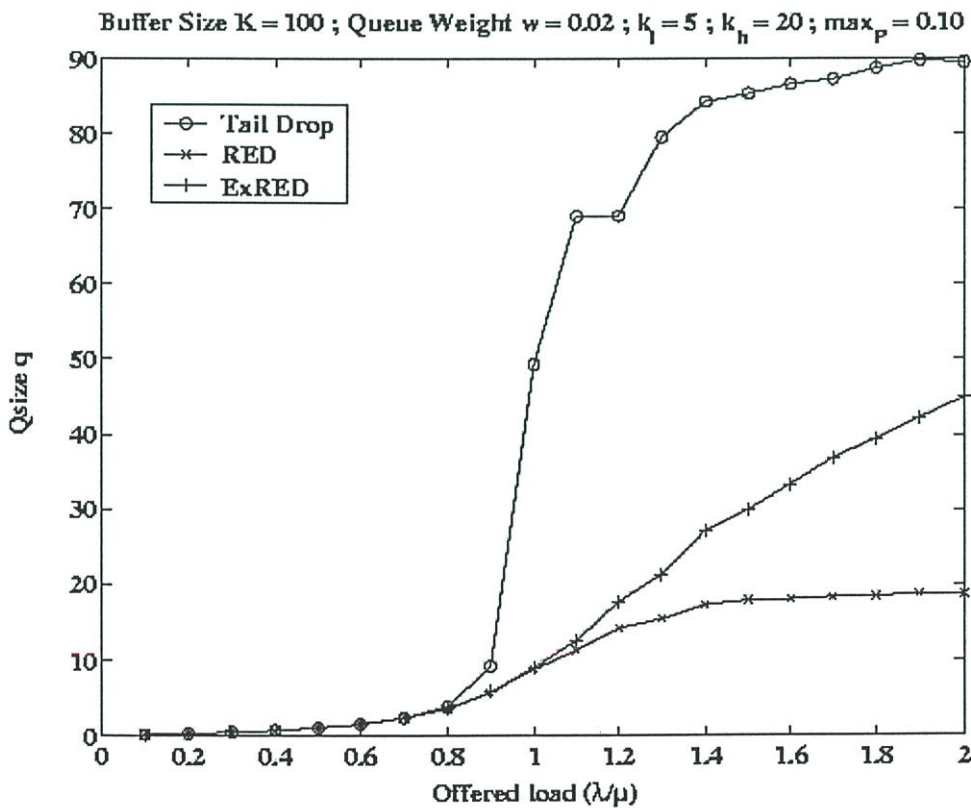
การทดลองแบบที่ 4 เป็นการทดลองเพื่อศึกษาผลเมื่อมีการเปลี่ยนแปลงค่าความน่าจะเป็นสูงสุดที่แพ็คเกจจะถูกครีโปกโดยตัวกลไก ( $\max_p$ ) หรือยอมให้ใช้บัฟเฟอร์ได้น้อยลง รูปที่ 6.20 แสดงการกระจายของขนาดคิวหรือจำนวนแพ็คเกจในบัฟเฟอร์ที่สถานะคงตัว (Stationary Probability Distribution) ส่วนรูปที่ 6.21 แสดงขนาดคิวเฉลี่ย  $E(Q)$  จากการกำหนดพารามิเตอร์ใหม่ ซึ่งทั้งสองรูปแสดงให้เห็นว่าแพ็คเกจจะสามารถเข้าพักในบัฟเฟอร์ได้น้อยลง

การเกิดบัฟเฟอร์ล้นสามารถพิจารณาได้จากรูปที่ 14, 16, 18 และ 20 ซึ่งแสดงการกระจายของจำนวนแพ็คเกจในบัฟเฟอร์ จากที่ได้นิยามไปแล้วว่า  $P_{Overflow} = \{\pi(q) | q = K\}$  แสดงโอกาสจะเกิดบัฟเฟอร์ล้น เมื่อพิจารณาจากรูปดังกล่าว แสดงให้เห็นว่ากลไกแบบ RED และ ExRED ก็ยังสามารถป้องกันการเกิดบัฟเฟอร์ล้นได้เมื่อเปรียบเทียบกับกลไกแบบ Tail Drop และเมื่อเปรียบเทียบกันระหว่างกลไกแบบ ExRED และ ExRED จะเห็นว่ากลไกแบบ ExRED ยอมให้ใช้บัฟเฟอร์ได้มากกว่ากลไกแบบ RED แต่ก็มีโอกาสเกิดบัฟเฟอร์ล้นได้มากกว่า

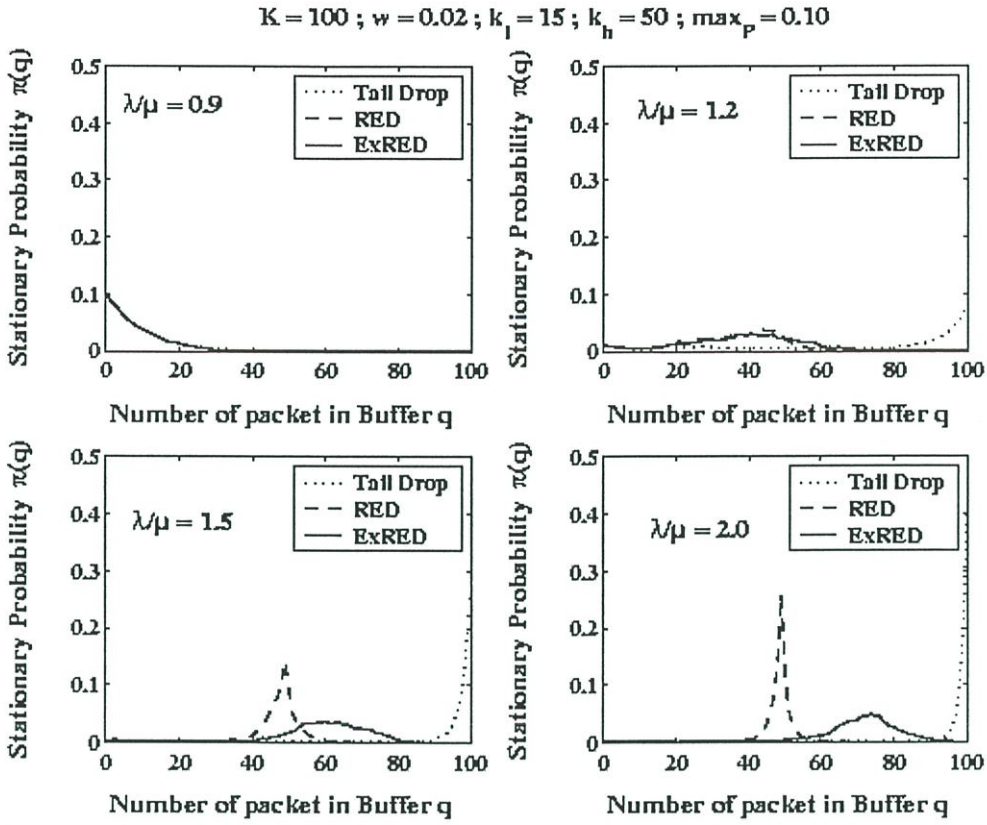
แต่เมื่อพิจารณาแนวโน้มของจำนวนแพ็คเกจในบัฟเฟอร์ของกลไกแบบ ExRED จากรูปที่ 6.16 และ 6.18 ซึ่งกำหนดพารามิเตอร์ค่ากำหนดขั้นสูง  $k_h$  เอาไว้ใกล้ขนาดความจุของบัฟเฟอร์ พบว่าหากกำหนดค่ากำหนดขั้นสูง  $k_h$  เอาไว้สูงกว่านี้ หรือกำหนดค่าความน่าจะเป็นสูงสุดที่แพ็คเกจจะถูกครีโปกโดยกลไก  $\max_p$  ให้ต่ำกว่านี้ ในสถานะที่มีภาระงานสูงมากกลไกแบบ ExRED อาจประสบภาวะบัฟเฟอร์ล้นได้ อย่างไรก็ตามเมื่อเปรียบเทียบกับผลจากกลไกแบบ Tail Drop แล้วก็ยังถือว่ามีโอกาสน้อยกว่ามาก อีกทั้งหากการกำหนดพารามิเตอร์เป็นไปโดยสอดคล้องกับเงื่อนไขตามสมการที่ (6.1) ถึง (6.3) และให้มีค่าห่างจากค่าขอบเขตของเงื่อนไขออกไป ก็จะทำให้โอกาสเกิดบัฟเฟอร์ล้นในกลไกแบบ ExRED ยิ่งน้อยลงไปอีก



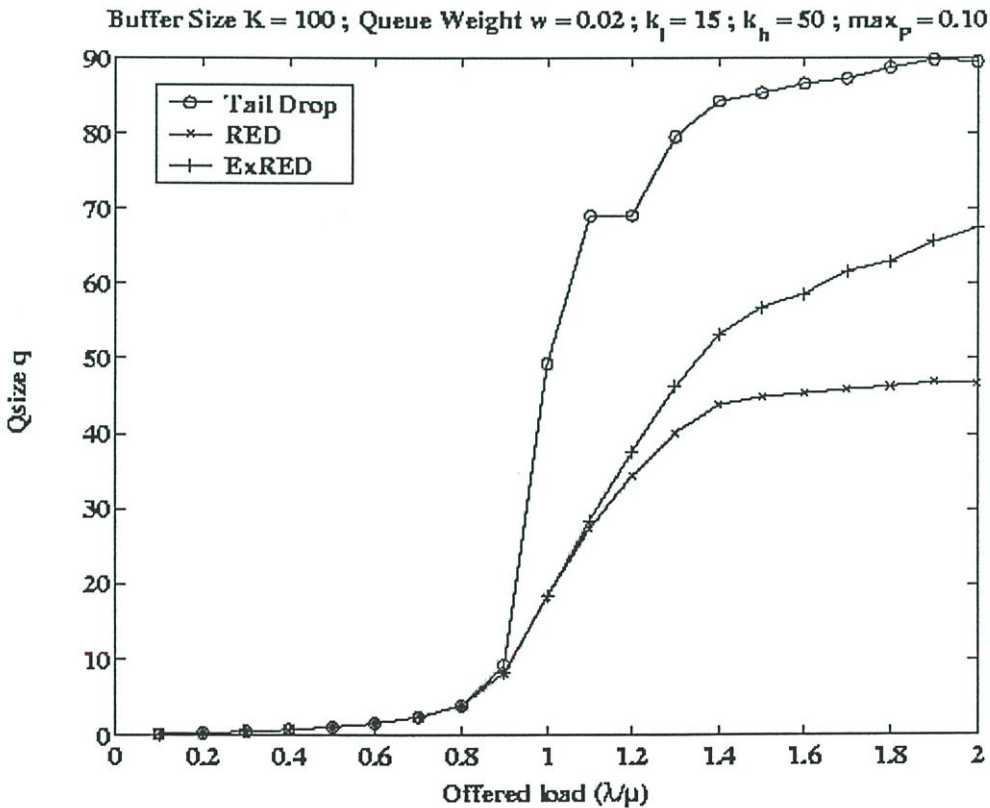
รูปที่ 6.14 การกระจายของขนาดคิวในการทดลองแบบที่ 1 โดย  $k_l = 5$  และ  $k_h = 20$  แพคเกจ



รูปที่ 6.15 ค่าเฉลี่ยขนาดคิว  $E(Q)$  ในการทดลองแบบที่ 1 โดย  $k_l = 5$  และ  $k_h = 20$  แพคเกจ

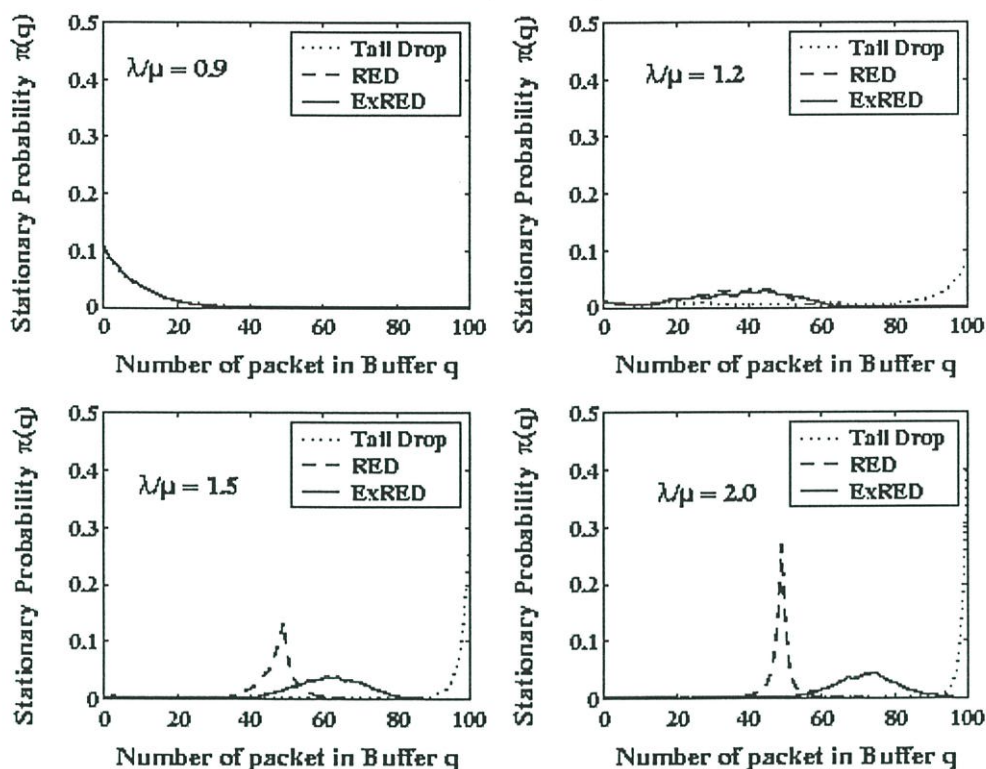


รูปที่ 6.16 การกระจายของขนาดคิวในการทดลองแบบที่ 2 โดย  $k_l = 15$  และ  $k_h = 50$  แพคเกจ

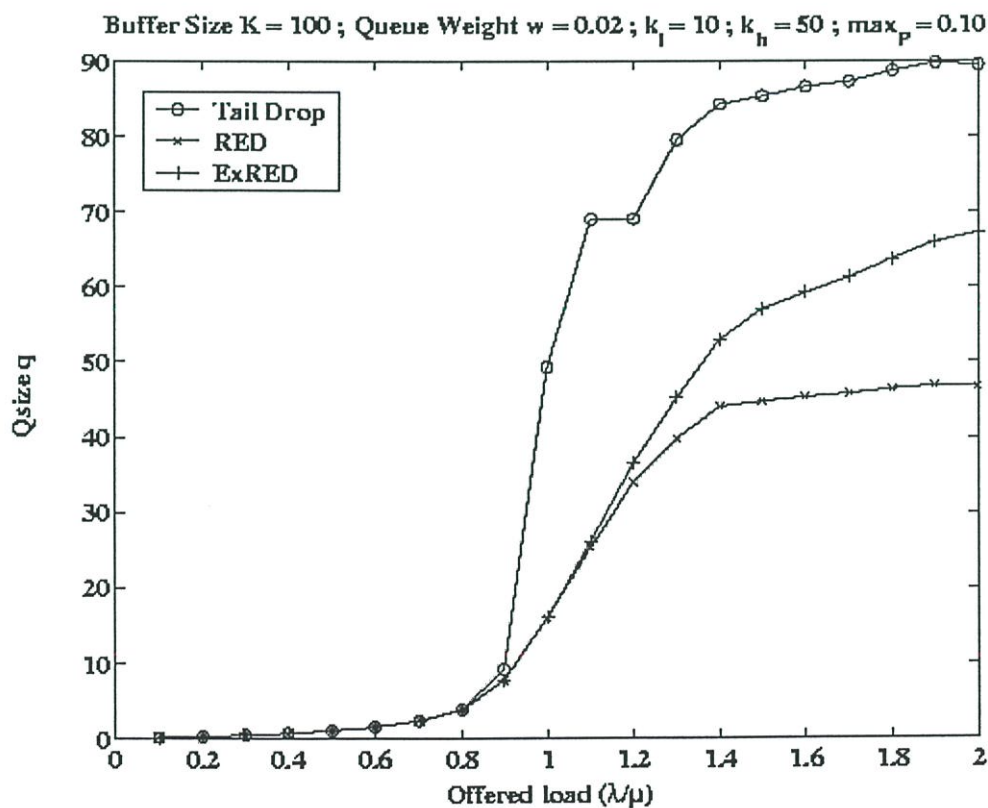


รูปที่ 6.17 ค่าเฉลี่ยขนาดคิว  $E(Q)$  ในการทดลองแบบที่ 2 โดย  $k_l = 15$  และ  $k_h = 50$  แพคเกจ

$K = 100$ ;  $w = 0.02$ ;  $k_l = 10$ ;  $k_h = 50$ ;  $\max_p = 0.10$

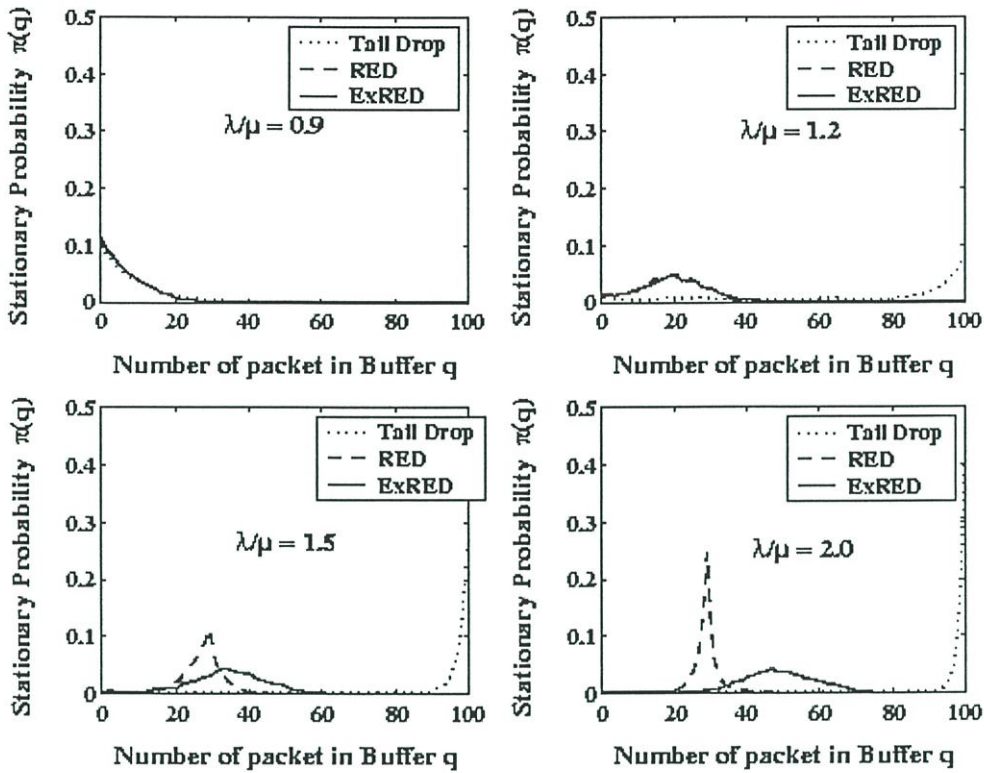


รูปที่ 6.18 การกระจายของขนาดคิวในการทดลองแบบที่ 3 โดย  $k_l = 10$  และ  $k_h = 50$  แพคเกจ

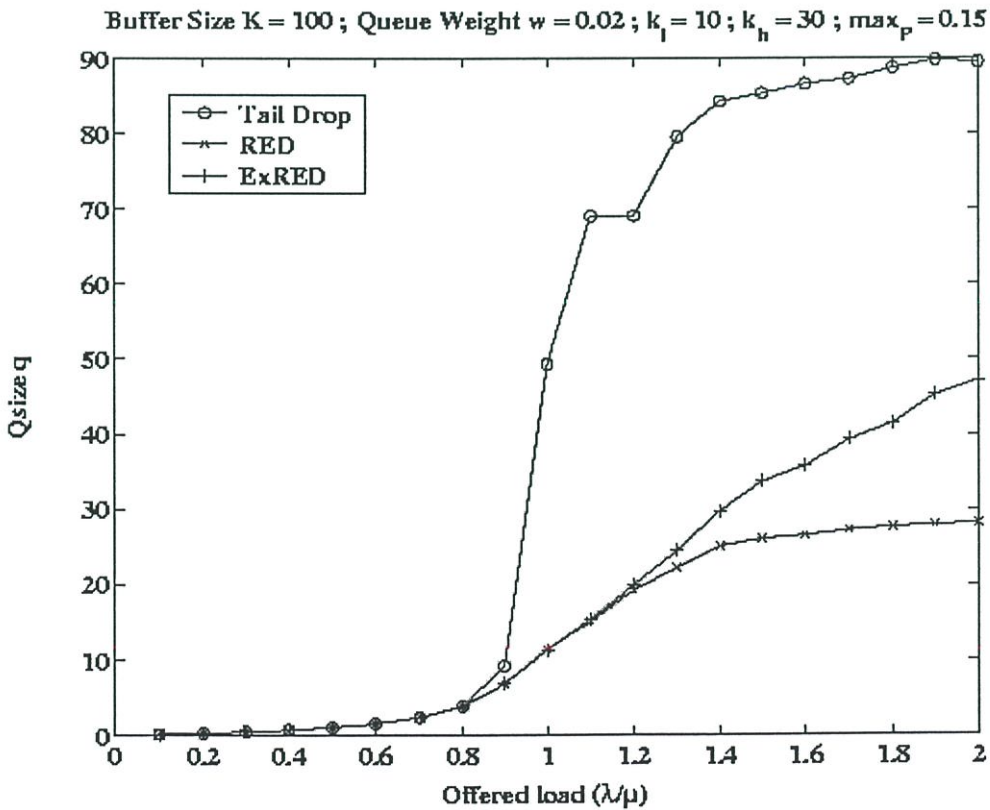


รูปที่ 6.19 ค่าเฉลี่ยขนาดคิว  $E(Q)$  ในการทดลองแบบที่ 3 โดย  $k_l = 10$  และ  $k_h = 50$  แพคเกจ

$K = 100$ ;  $w = 0.02$ ;  $k_l = 10$ ;  $k_h = 30$ ;  $\max_p = 0.15$



รูปที่ 6.20 การกระจายของขนาดคิวในการทดลองแบบที่ 4 โดย  $\max_p = 0.15$

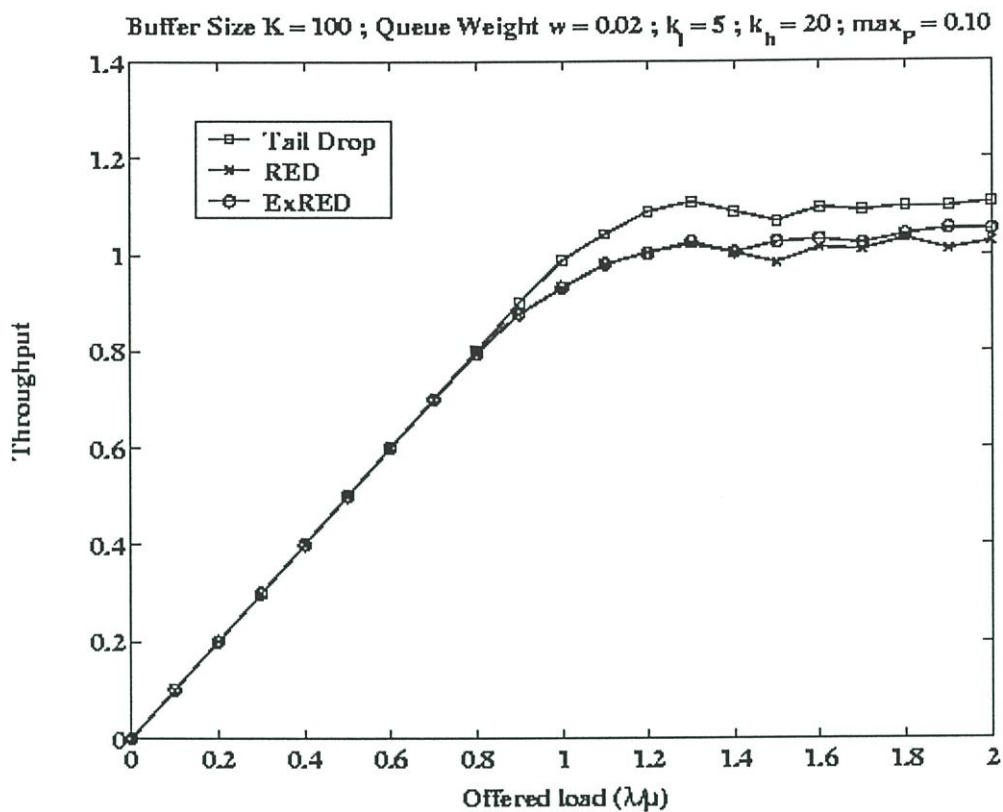


รูปที่ 6.21 ค่าเฉลี่ยขนาดคิว  $E(Q)$  ในการทดลองแบบที่ 4 โดยกำหนด  $\max_p = 0.15$

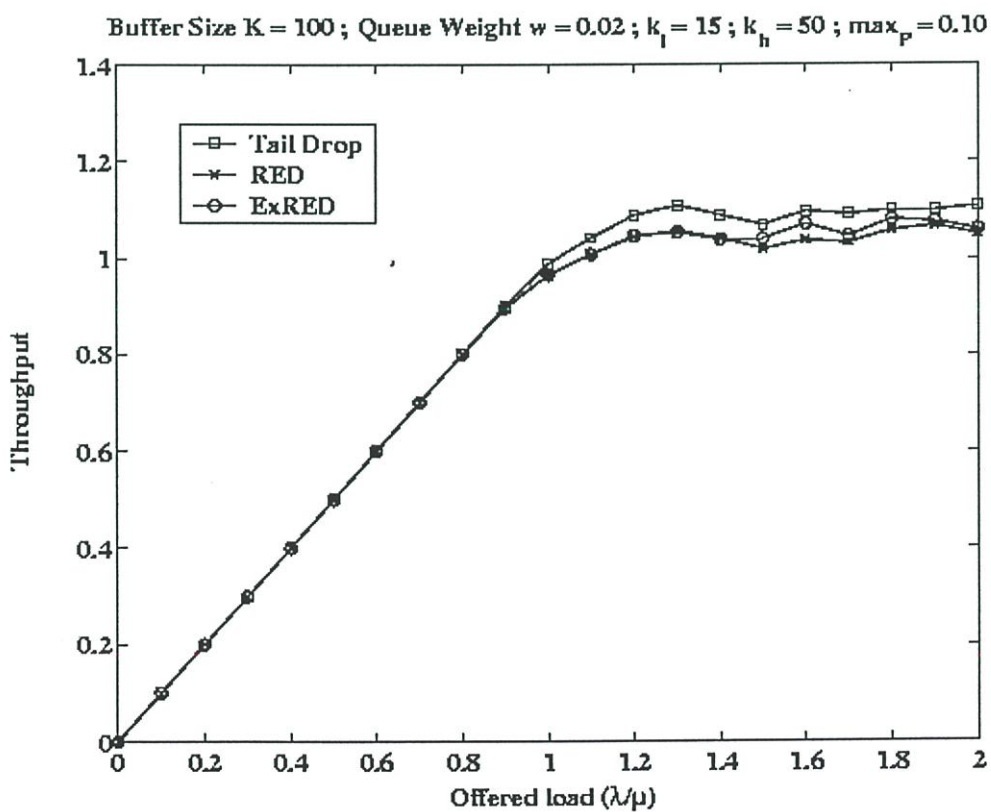
ในประเด็นต่อไปคือการพิจารณาทรูพุท การประวิงเวลาจากการรอคอยในคิวในบัฟเฟอร์ และการเกิดการครีโพลิดต่อกันอย่างต่อเนื่อง ซึ่งผลของทรูพุทสามารถแสดงได้ดังรูปที่ 6.22 ถึง 6.25 ผลของการประวิงเวลาจากการรอคอยในคิวในบัฟเฟอร์สามารถแสดงได้ดังรูปที่ 6.26 ถึง 6.29 ส่วนผลของการเกิดการครีโพลิดต่อกันอย่างต่อเนื่องสามารถแสดงได้ดังรูปที่ 6.30 และรูปที่ 6.33

รูปที่ 6.22 และ 6.25 ซึ่งล้วนเป็นการยอมให้ใช้บัฟเฟอร์ได้น้อยลงเมื่อเปรียบเทียบกับรูปที่ 6.4 และเมื่อเปรียบเทียบกับรูปที่ 6.23 และ 6.24 ซึ่งเป็นการยอมให้ใช้บัฟเฟอร์ได้มากขึ้น จะสังเกตได้ว่าเมื่อกลไกแบบ RED และ ExRED ยอมให้ใช้บัฟเฟอร์ได้มากขึ้น ทรูพุทของทั้งสองกลไกก็จะสูงขึ้น (ดูชำระระดับทรูพุทของกลไกแบบ Tail Drop เมื่อเอาทรูพุทของกลไกแบบ Tail Drop เป็นขอบเขตอ้างอิงด้านบน) และแสดงให้เห็นว่ากลไกแบบ ExRED มีทรูพุทสูงกว่ากลไกแบบ RED

เมื่อเปรียบเทียบกับรูปที่ 6.5 ทั้งรูปที่ 6.26 และ 6.29 แสดงให้เห็นว่าเมื่อกลไกแบบ RED และ ExRED ยอมให้ใช้บัฟเฟอร์น้อยลง ค่าประวิงเวลาจากการรอคอยในคิวก็จะลดลง แต่ถ้ายอมให้มีการใช้บัฟเฟอร์มากขึ้น ค่าประวิงเวลาจากการรอคอยในคิวก็จะสูงขึ้นด้วย ในขณะที่รูป 6.27 และ 6.28 แสดงให้เห็นว่าการที่กลไกทั้งสองยอมให้ใช้บัฟเฟอร์มากขึ้น ก็ทำให้มีค่าประวิงเวลาจากการรอคอยในคิวมากขึ้นด้วย ซึ่งแนวโน้มดังกล่าวสอดคล้องกับการวิเคราะห์ในหัวข้อที่ 6.3 แต่อย่างไรก็ตามค่าประวิงเวลาของกลไกแบบ ExRED ยังถือว่าน้อยเมื่อเปรียบเทียบความต่างระหว่างกลไกแบบ Tail Drop และกลไกแบบ RED

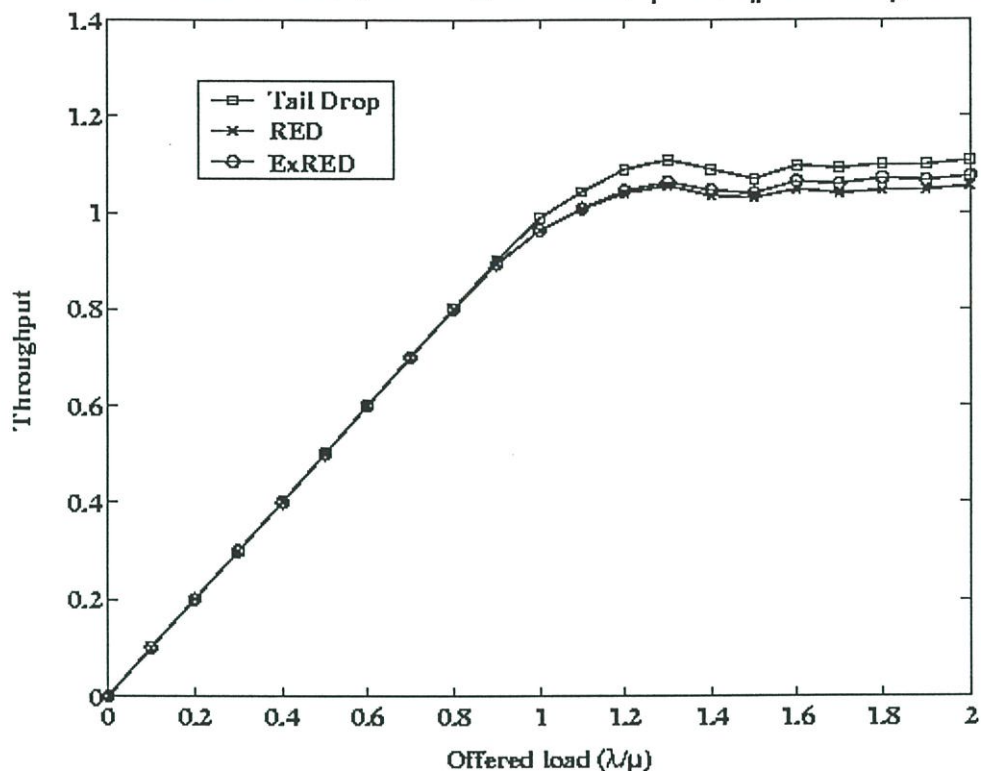


รูปที่ 6.22 ทฤษฎีในการทดลองแบบที่ 1 (ยอมให้ใช้บัฟเฟอร์น้อยลง)



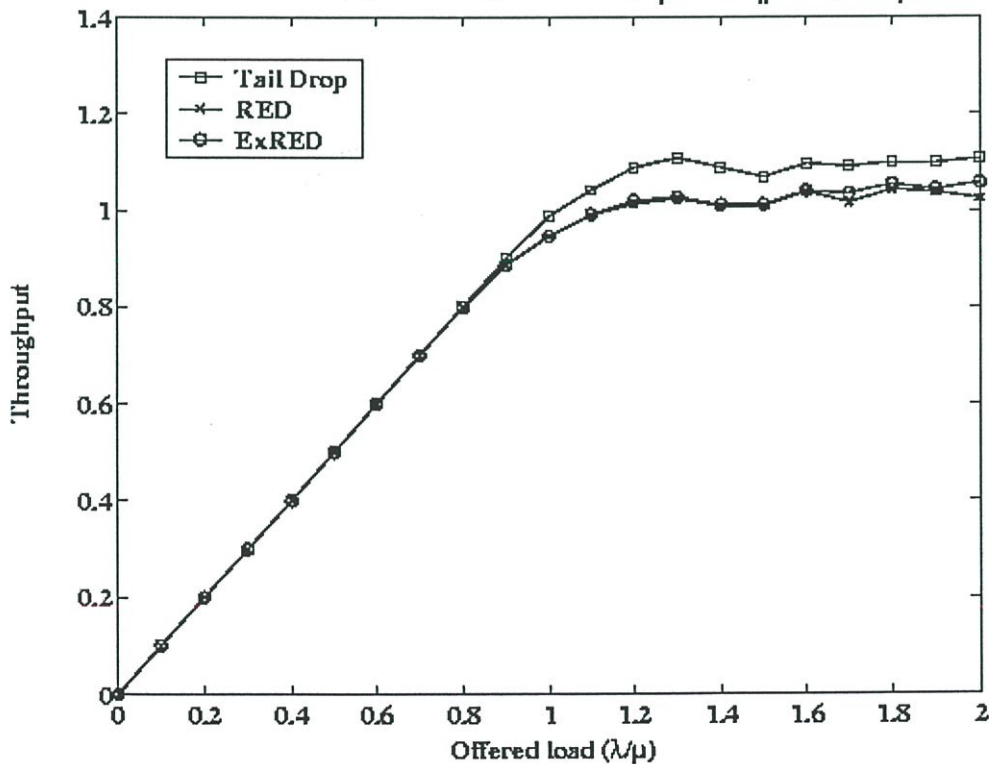
รูปที่ 6.23 ทฤษฎีในการทดลองแบบที่ 2 (ยอมให้ใช้บัฟเฟอร์มากขึ้น)

Buffer Size  $K = 100$ ; Queue Weight  $w = 0.02$ ;  $k_l = 10$ ;  $k_h = 50$ ;  $\max_p = 0.10$

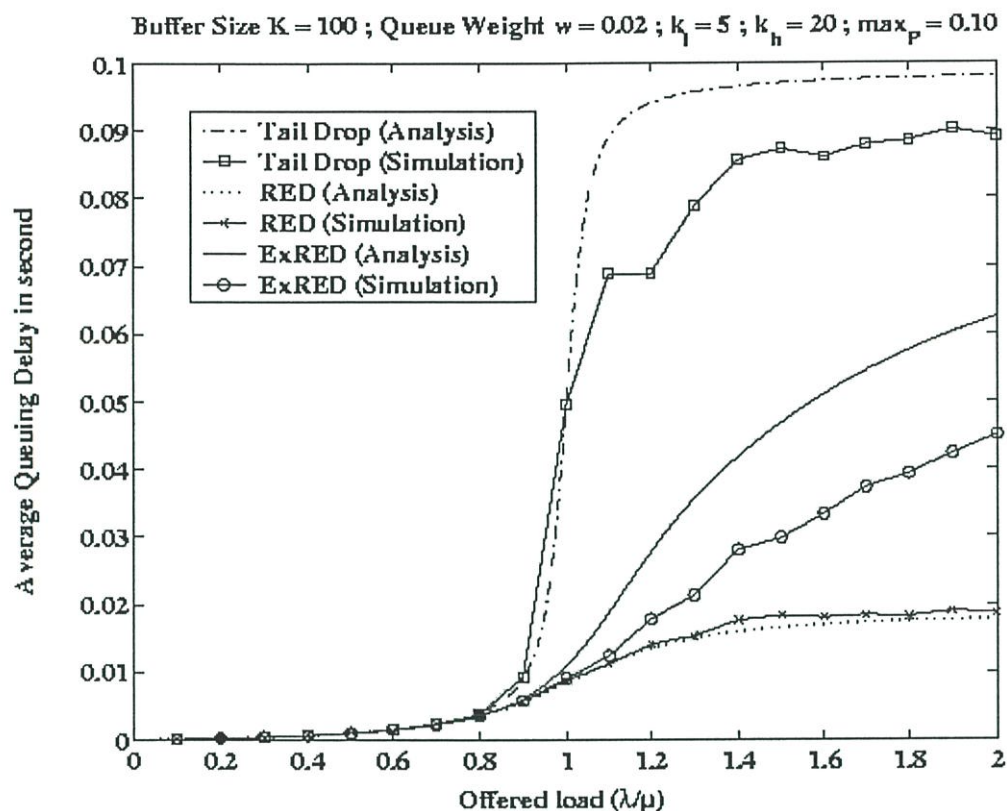


รูปที่ 6.24 ทฤษฎีในการทดลองแบบที่ 3 (เมื่อเปลี่ยนแปลง  $k_h / k_l$ )

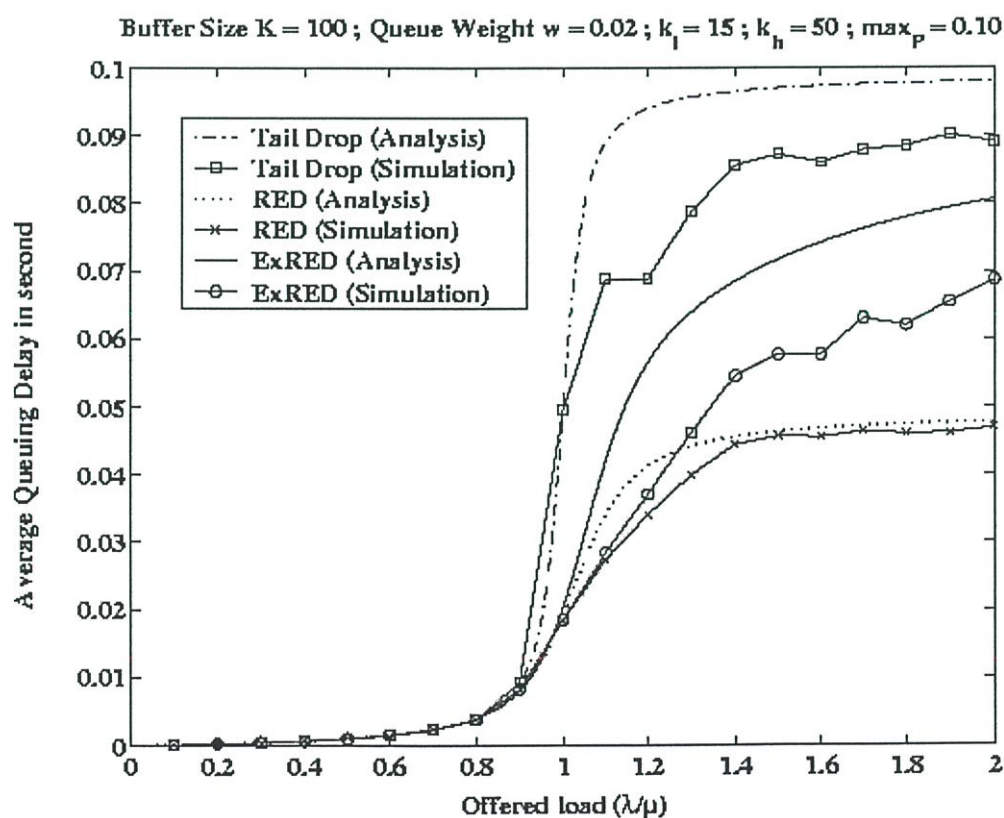
Buffer Size  $K = 100$ ; Queue Weight  $w = 0.02$ ;  $k_l = 10$ ;  $k_h = 30$ ;  $\max_p = 0.15$



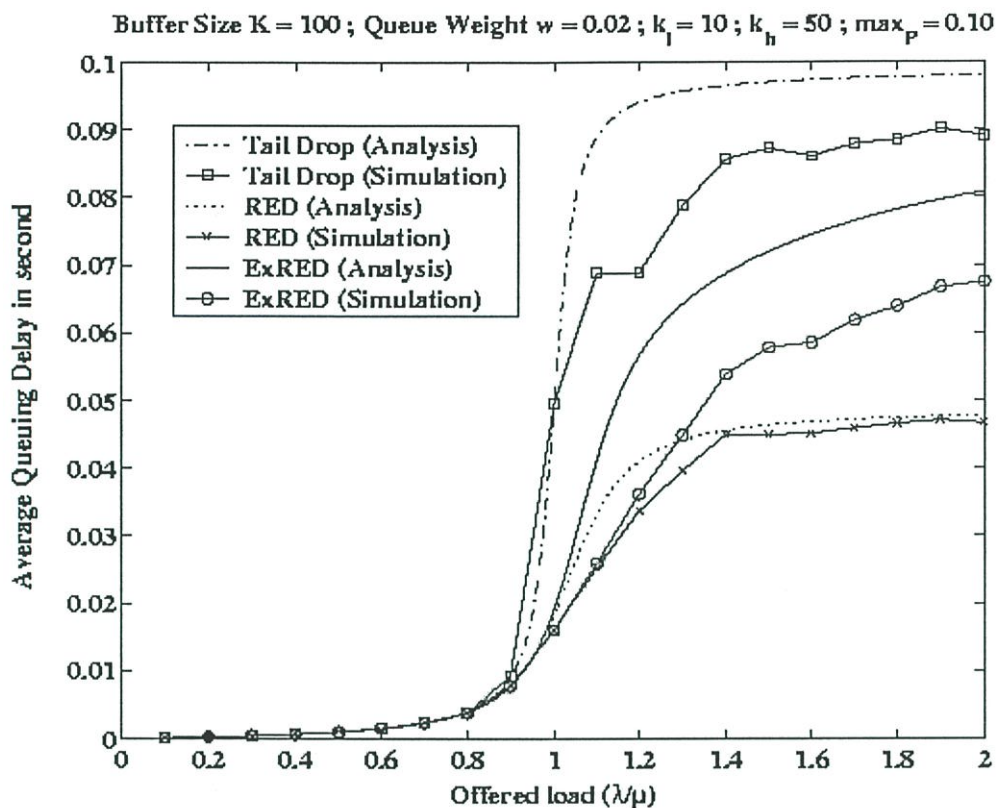
รูปที่ 6.25 ทฤษฎีในการทดลองแบบที่ 4 (เมื่อเปลี่ยนแปลง  $\max_p$ )



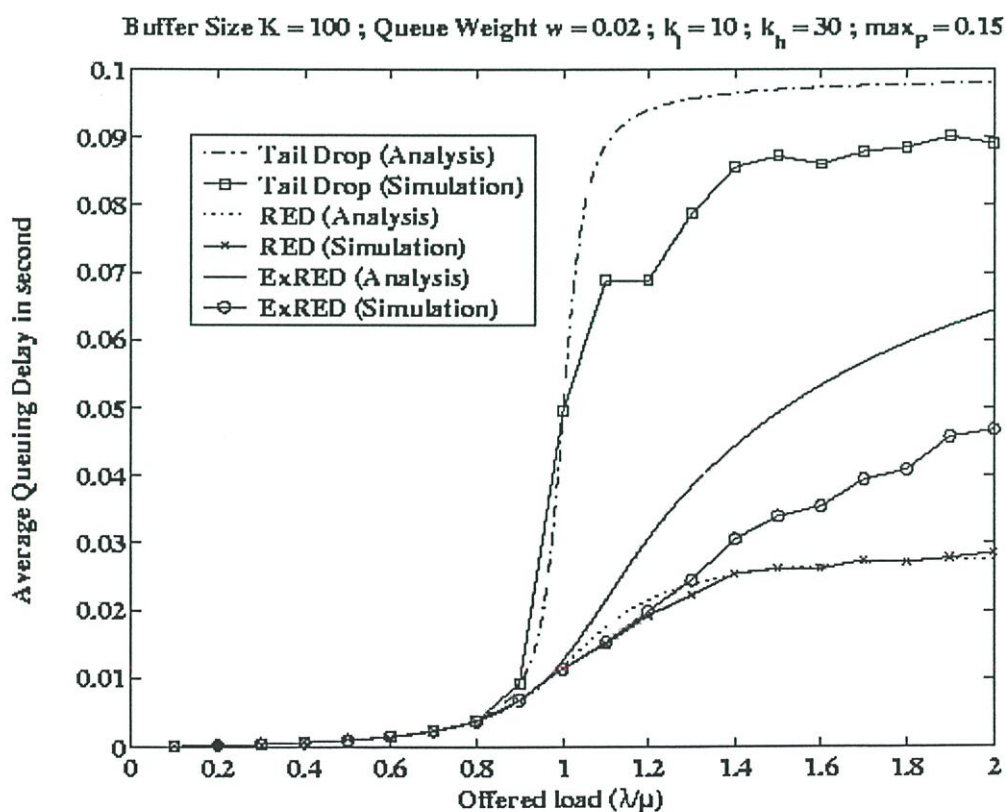
รูปที่ 6.26 ค่าประวิงเวลาจากการรอคอยในคิวในการทดลองแบบที่ 1 (ขอมให้ใช้บัฟเฟอร์น้อยลง)



รูปที่ 6.27 ค่าประวิงเวลาจากการรอคอยในคิวในการทดลองแบบที่ 2 (ขอมให้ใช้บัฟเฟอร์มากขึ้น)



รูปที่ 6.28 ค่าประวิงเวลาจากการรอคอยในคิวในการทดลองแบบที่ 3 (เมื่อเปลี่ยนแปลง  $k_h / k_l$ )



รูปที่ 6.29 ค่าประวิงเวลาจากการรอคอยในคิวในการทดลองแบบที่ 4 (เมื่อเปลี่ยนแปลง  $\max_p$ )

ในการพิจารณาที่สถานะคงตัวเพื่อหาความสัมพันธ์กับภาระงานที่เพิ่มขึ้น เมื่อเปรียบเทียบรูปที่ 6.12 กับรูปที่ 6.30 ถึงรูปที่ 6.33 จะแสดงผลการวิเคราะห์ได้ดังนี้

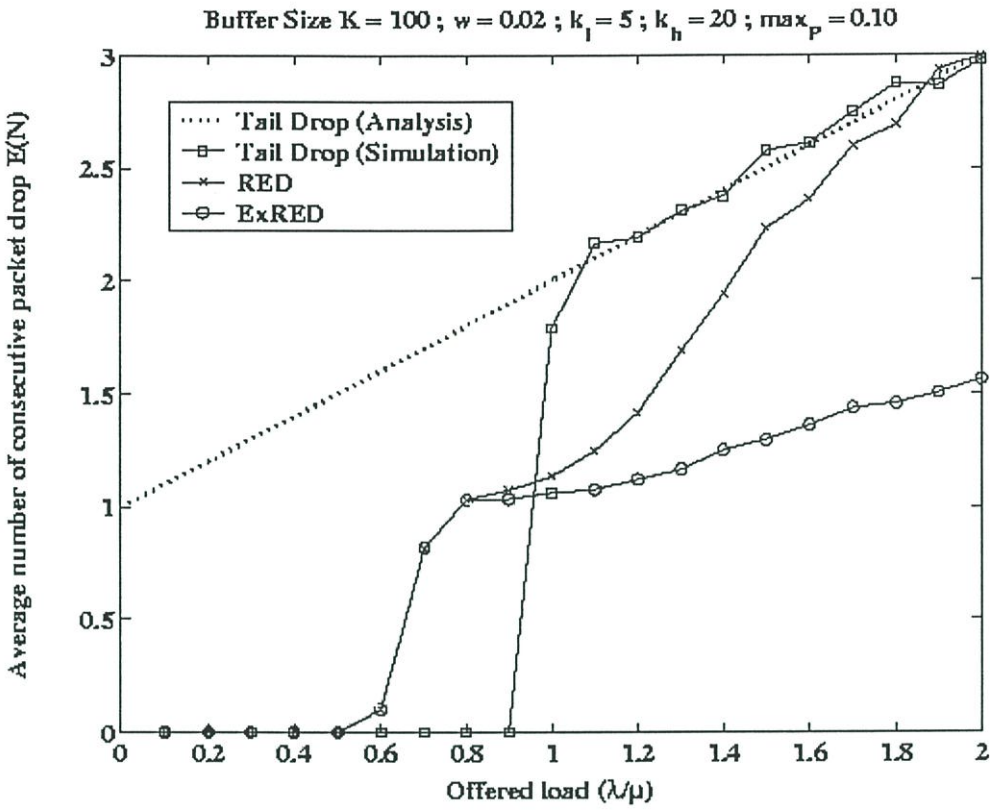
- จากการเปรียบเทียบรูปที่ 6.12 กับรูปที่ 6.30 แสดงให้เห็นว่าเมื่อกลไกทั้งแบบ RED และ ExRED ยอมให้ใช้บัฟเฟอร์น้อยลงดังการทดลองแบบที่ 1 ค่าเฉลี่ยของจำนวนแพ็คเกจที่จะถูกครีโปกติดต่อกันอย่างต่อเนื่องจึงมีมากกว่า

- เมื่อเปรียบเทียบรูปที่ 6.12 กับรูปที่ 6.31 การยอมให้มีการใช้บัฟเฟอร์มากขึ้นดังในการทดลองแบบที่ 2 จะให้ผลตรงกันข้ามกับแบบที่ 1 คือ ค่าเฉลี่ยของจำนวนแพ็คเกจที่ถูกครีโปกติดต่อกันอย่างต่อเนื่องจะมีค่าน้อยกว่า

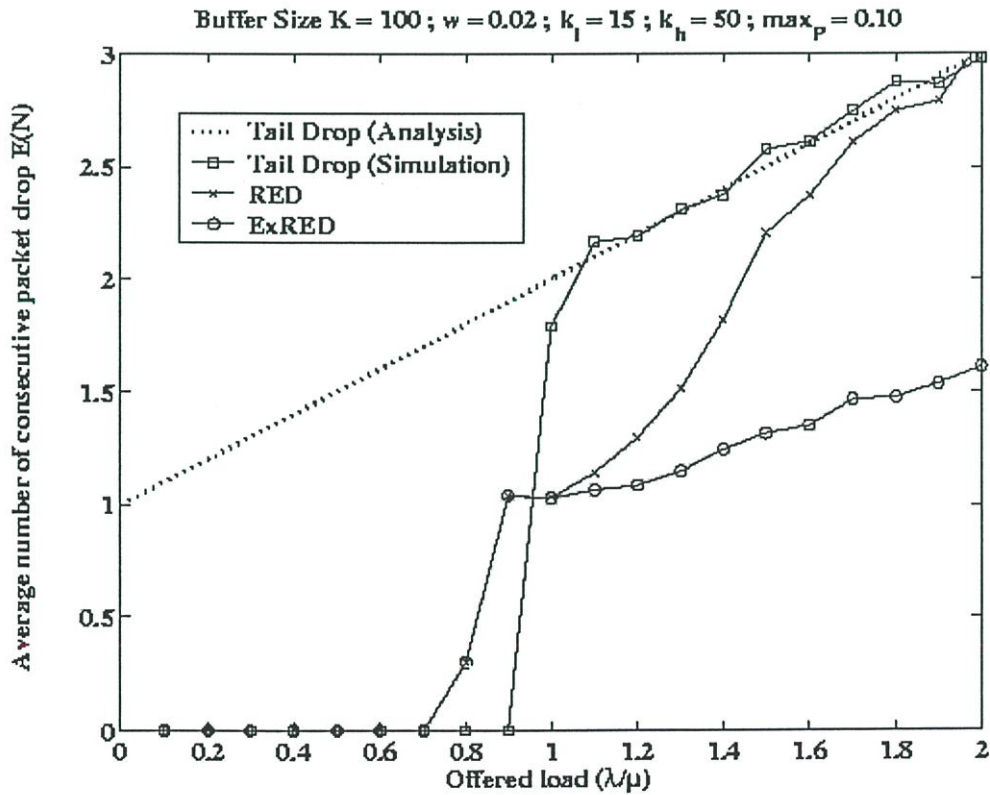
- เมื่อเปรียบเทียบรูปที่ 6.12 กับรูปที่ 6.32 จากในการทดลองแบบที่ 3 ซึ่งเป็นการเพิ่มความยืดหยุ่นในการควบคุมอัตราการเข้าบัฟเฟอร์ของแพ็คเกจ ซึ่งเป็นการลดอัตราการเพิ่มของความน่าจะเป็นที่แพ็คเกจจะถูกบล็อกและครีโปก แต่ความน่าจะเป็นดังกล่าวจะเพิ่มขึ้นด้วยอัตราการเพิ่มที่สูงกว่าเดิมเมื่อที่ว่างในบัฟเฟอร์ใกล้หมดลง ทำให้ค่าเฉลี่ยจำนวนแพ็คเกจที่จะถูกครีโปกติดต่อกันอย่างต่อเนื่องน้อยกว่า เมื่อที่ว่างในบัฟเฟอร์ใกล้หมดลงค่าเฉลี่ยดังกล่าวจะมีค่ามากขึ้นจนเกือบเท่าเดิม

- เมื่อเปรียบเทียบรูปที่ 6.12 กับรูปที่ 6.33 จากการทดลองแบบที่ 4 ซึ่งเป็นการเร่งอัตราการเพิ่มของความน่าจะเป็นที่แพ็คเกจจะถูกบล็อกและครีโปกในช่วงที่อัตราการมาถึงได้อยู่ในระหว่างการควบคุมแล้ว ดังนั้นค่าเฉลี่ยของจำนวนแพ็คเกจที่ถูกครีโปกติดต่อกันอย่างต่อเนื่องจะมากกว่าเดิมในช่วงต้น แต่เมื่อภาระงานเพิ่มขึ้น ค่าเฉลี่ยจำนวนแพ็คเกจที่จะถูกครีโปกติดต่อกันอย่างต่อเนื่องจะน้อยกว่าเดิม ทั้งนี้เป็นผลจากการที่ถูกกลไกควบคุมอัตราการมาถึงของแพ็คเกจในช่วงการสุ่มบล็อกและครีโปกแพ็คเกจไปแล้วอย่างเข้มงวด (เพราะถูกบล็อกและครีโปกไปด้วยความน่าจะเป็นที่สูงกว่า)

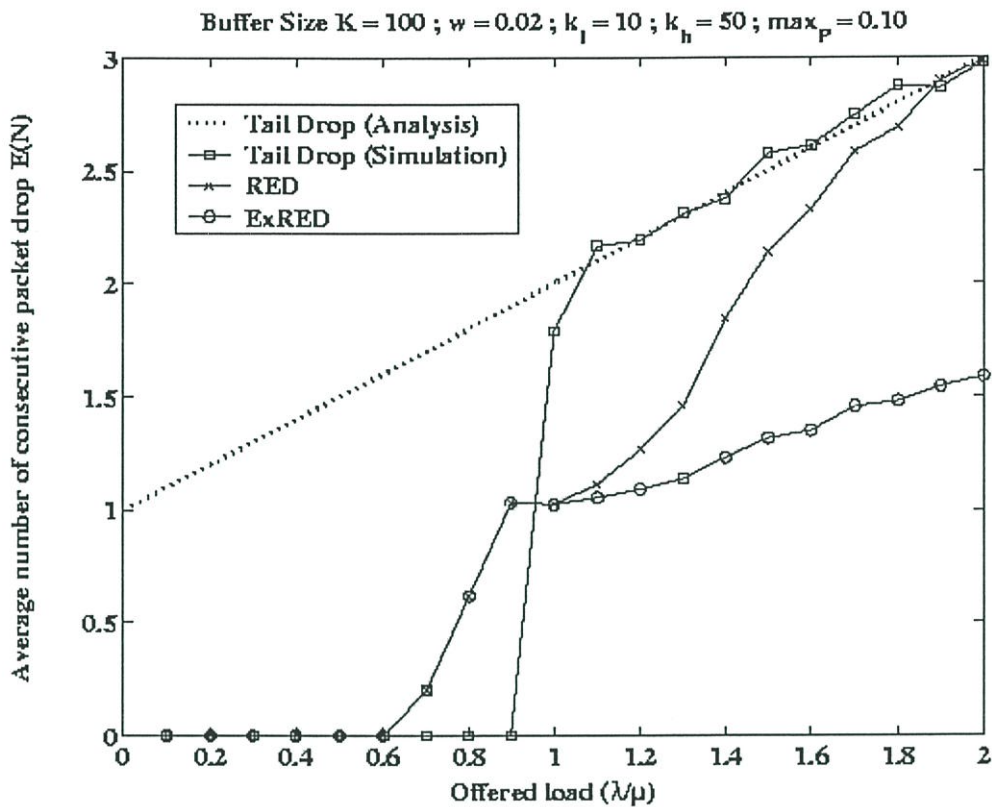
เมื่อเปรียบเทียบระหว่างกลไกแบบ RED และ ExRED แล้วจากกราฟผลการทดลองทั้ง 4 แบบ จะแสดงให้เห็นว่าเส้นกราฟแสดงค่าเฉลี่ยจำนวนแพ็คเกจที่ถูกครีโปกติดต่อกันของกลไกแบบ ExRED ก็ยังมีลักษณะที่ดีกว่าของกลไกแบบ RED เสมอ



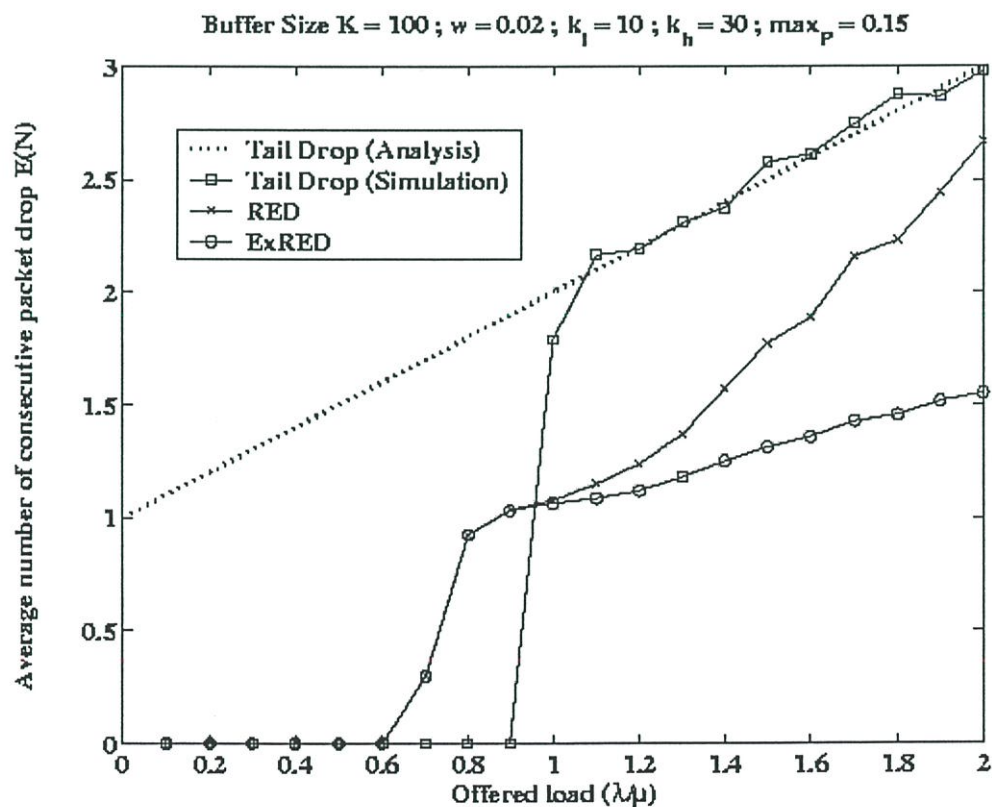
รูปที่ 6.30 การเกิดการรื้อปัดต่อกันในการทดลองแบบที่ 1 (ยอมให้ใช้บัฟเฟอร์น้อยลง)



รูปที่ 6.31 การเกิดการรื้อปัดต่อกันในการทดลองแบบที่ 2 (ยอมให้ใช้บัฟเฟอร์มากขึ้น)



รูปที่ 6.32 การเกิดการคร้อปิดต่อกันในการทดลองแบบที่ 3 (เมื่อเปลี่ยนแปลง  $k_h / k_l$ )



รูปที่ 6.33 การเกิดการคร้อปิดต่อกันในการทดลองแบบที่ 4 (เมื่อเปลี่ยนแปลง  $\max_p$ )

## 6.6 สรุปผล

จากที่ได้วิเคราะห์ในหัวข้อ 6.2 ถึง 6.5 สามารถสรุปได้ว่ากลไกแบบ ExRED นำข้อดีของกลไกแบบ RED คือการติดตามและควบคุมการใช้บัฟเฟอร์ป้องกันการเกิดบัฟเฟอร์ล้นมาใช้ และปรับปรุงข้อเสียคือการครีโปกมากเกินไปทำให้ทรูพุตต่ำออกไปได้ จึงทำให้กลไกแบบ ExRED สามารถป้องกันปัญหาบัฟเฟอร์ล้นที่เป็นสาเหตุของอีกหลายปัญหาที่จะตามมาอีกได้ นอกจากนี้คุณสมบัติในการกระจายการครีโปกที่ยังคงอยู่ในกลไกแบบ ExRED ก็ช่วยแก้ปัญหา Lock-Out [2] ได้อีกด้วย ในขณะที่ปัญหาทรูพุตต่ำของกลไกแบบ RED ได้ถูกปรับปรุงให้ดีขึ้นดังการวิเคราะห์ในหัวข้อ 6.2

ถึงแม้ว่าค่าประวิงเวลาเนื่องจากการรอคอยในคิวของกลไกแบบ ExRED จะมากกว่าเมื่อเปรียบเทียบกับกลไกแบบ RED แต่เมื่อเปรียบเทียบกับกลไกดั้งเดิมคือ Tail Drop ด้วยแล้ว สามารถกล่าวได้ว่าเป็นการแลกเปลี่ยนที่ยอมรับได้ อย่างไรก็ตามขึ้นกับการประยุกต์ใช้ด้วยว่างานที่เครือข่ายนั้นๆ รอรับอยู่ด้วยว่าจำเป็นต้องมีการควบคุมค่าประวิงเวลาแบบกลไก RED หรือไม่ ในขณะที่ยังมีองค์ประกอบอื่นๆ เช่น ค่าประวิงเวลาจากปัจจัยต่างๆบนเส้นทางย่อยๆที่ประกอบกันนับจากต้นทางถึงปลายทางดังที่ได้กล่าวเอาไว้ในการวิเคราะห์ในหัวข้อ 6.3 ซึ่งอาจทำให้ค่าประวิงเวลารวมของกลไกแบบ ExRED

นอกจากทรูพุตเฉลี่ยที่ดีขึ้นที่ได้วิเคราะห์ในหัวข้อ 6.2 แล้ว จากการวิเคราะห์ในหัวข้อ 6.4 ยังแสดงได้ว่าทรูพุตขณะใดขณะหนึ่งของกลไกแบบ ExRED ก็น่าจะดีกว่า เนื่องจากค่าเฉลี่ยของจำนวนแพคเกจที่ถูกครีโปกติดต่อกันอย่างต่อเนื่องของกลไกแบบ ExRED น้อยกว่า

จึงสรุปได้ว่ากลไกแบบ ExRED ทำให้การใช้บัฟเฟอร์มีประสิทธิภาพ เนื่องจากสามารถแก้ไขและป้องกันปัญหาที่เกิดจากกลไกแบบดั้งเดิมคือปัญหาบัฟเฟอร์ล้นและปัญหาจากปรากฏการณ์ Lock-Out ได้จากการอาศัยข้อดีของกลไกแบบ RED ในขณะที่มีการปรับปรุงข้อเสียของกลไกแบบ RED คือการครีโปกมากเกินไปทำให้มีทรูพุตต่ำซึ่งในวิทยานิพนธ์นี้ได้แสดงหลักการในการนำเสนอกลไกดังกล่าว พร้อมทั้งวิเคราะห์ผลจากคิวในบัฟเฟอร์ที่มีต่อประสิทธิภาพของเกตเวย์และเครือข่ายด้วยผลการจำลองทางคณิตศาสตร์โดยใช้ทฤษฎีคิวและผลการจำลองเหตุการณ์เปรียบเทียบกลไกทั้งสาม ซึ่งถือว่าบรรลุวัตถุประสงค์ของงานวิจัยในวิทยานิพนธ์นี้

## 6.7 แนวทางการนำไปใช้และการปรับปรุงจากกลไกเดิม

การทำงานของกลไกแบบแอคทีฟในการติดตามสถานการณ์การทำงานของเครือข่ายล้วนต้องหาข้อมูลสำหรับความเป็นไปที่เกิดขึ้นในขณะใดๆ [13] ซึ่งสิ่งเหล่านี้ต้องแตกต่างกับการทำงานเพิ่มขึ้นของเกตเวย์ และการสร้างภาระงานเพิ่มขึ้นให้กับระบบเครือข่าย ซึ่งกลไกที่ดีควรทำให้การเพิ่มขึ้นเหล่านี้เป็นไปน้อยที่สุด

เมื่อพิจารณาจากอัลกอริทึมการทำงานของกลไก ExRED ในบทที่ 4 แสดงให้เห็นได้ว่าการนำกลไกแบบ ExRED ไปใช้นั้น เมื่อเปรียบเทียบกับกลไกแบบดั้งเดิมคือ Tail Drop จะมีสิ่งเพิ่มขึ้นก็คือ การคำนวณขนาดคิวเฉลี่ยและความน่าจะเป็นในการบล็อก/ครีอปแพคเก็ต กลไกแบบ ExRED จะนับจำนวนแพคเก็ตในบัฟเฟอร์คิว และคำนวณขนาดคิวเฉลี่ยทุกครั้งที่มีแพคเก็ตเข้า การนับจำนวนแพคเก็ตก็จะทำทุกครั้งที่มีแพคเก็ตเข้าและออกจากบัฟเฟอร์ ซึ่งโดยปกติเกตเวย์จะมีการนับค่าเหล่านี้อยู่แล้วเพื่อเก็บเป็นสถิติสำหรับนำไปใช้ในการวิเคราะห์ในการดูแลระบบเครือข่าย อีกทั้งการคำนวณที่เกิดเฉพาะทุกครั้งที่มีแพคเก็ตมาถึง ไม่ต้องคำนวณทุกคาบเวลา สำหรับรายละเอียดของการคำนวณนั้น การคำนวณค่าความน่าจะเป็นในการบล็อกและครีอปแพคเก็ตของกลไกจัดการคิวแบบ ExRED สามารถคำนวณล่วงหน้าได้และเก็บในตารางในหน่วยความจำ ดังได้ถูกนำเสนอใน [1] ซึ่งจำนวนค่าเหล่านี้ขึ้นอยู่กับขนาดของบัฟเฟอร์ และเงื่อนไขการกำหนดขนาดบัฟเฟอร์ดีดข้อจำกัดเรื่องการประวิงเวลา จึงทำให้ขนาดบัฟเฟอร์มีขนาดใหญ่มากไม่ได้ ทำให้ขนาดหน่วยความจำที่จะเก็บค่าเหล่านี้ก็ไม่ต้องใหญ่ด้วยเช่นกัน จากที่แสดงมาข้างต้นแสดงให้เห็นว่ากลไกแบบ ExRED ไม่สร้างภาระงานที่มากเกินไปแก่เกตเวย์และระบบเครือข่าย ซึ่งสามารถนำไปใช้จริงได้

เมื่อเปรียบเทียบกับอัลกอริทึมการทำงานของกลไกแบบ RED และ ExRED ในบทที่ 3 และ 4 ตามลำดับ แสดงให้เห็นได้ว่าการนำกลไกแบบ ExRED ไปใช้นั้น มีความยุ่งยากกว่ากลไกแบบ RED เพียงเล็กน้อยดังที่กล่าวไว้ในบทที่ 4 ดังนั้นในการนำไปใช้จริงการปรับปรุงจากกลไกแบบ RED มาเป็นกลไกแบบ ExRED ก็แทบไม่มีความยุ่งยากเพิ่ม เพราะเป็นเพียงการขยายส่วนการคำนวณความน่าจะเป็นที่จะทำการบล็อกและครีอปแพคเก็ตจากช่วงขอบเขตเดิมออกไป ซึ่งกลไกย่อยในการคำนวณก็มีใช้อยู่แล้ว เพียงแต่เปลี่ยนสมการในการคำนวณใหม่ (หรือเปลี่ยนตารางเก็บค่าใหม่) ในส่วนที่ขยายออกไปเท่านั้น

## เอกสารอ้างอิง

- [1] S. Floyd and V. Jacobson. "Random Early Detection Gateways for Congestion Avoidance." IEEE/ACM Trans. on Networking, vol. 1, no. 4, Aug. 1993. pp. 397-413.
- [2] "Recommendations on Queue Management and Congestion Avoidance in the Internet." **RFC 2309**, April, 1999.
- [3] "Assured Forwarding PHB Group." **RFC 2597**, Jun. 1999.
- [4] T. Bonald, M. May, and J-C. Bolot "Analytic Evaluation of RED Performance." Proc. IEEE-INFOCOM, Tel-Aviv, Israel, March 26-30, 2000.
- [5] R. Laalaoua, T. Czachorski, and T. Atmaca "Markovian Model of RED Mechanism." Proc. IEEE/ACM International Symposium on Cluster Computing and the Grid, 2001. pp. 610-617.
- [6] Prabhavat S., Varakulsiripunth R., Noppanakeepong S. "Throughput Improvement on RED Mechanism." Proc. IEEE-ICCS, Singapore, Nov. 2002. 2P-01-05.
- [7] S. Floyd "RED: Discussions of Setting Parameters." [Online]. Available: <http://www.aciri.org/floyd/REDparameters.txt>. 1997.
- [8] B. R. Haverkort. **Performance of Computer Communication Systems: A Model-Based Approach**. John Wiley & Sons, Inc. 1998.
- [9] V. Jacobson "Congestion Avoidance and Control." Proc. ACM-SIGCOMM, Stanford, CA, Aug. 1988.
- [10] W. Stevens. **TCP/IP illustrated**. Addison Wesley, Reading, MA, 1994.
- [11] M. Christiansen, K. Jeffay, D. Ott, and F. D. Smith "Tuning RED for Web Traffic." IEEE/ACM Trans. on Networking, vol. 9, no. 3, Jun. 2001. pp. 249-264.
- [12] L. Kleinrock. **Queueing Systems VOL. 1: Theory**. John Wiley & Sons, Inc. 1975.
- [13] R. Jain, K. K. Ramakrishnan. "Congestion Avoidance in Computer Networks with a Connectionless Network Layer: Concepts, Goals and Methodology." Proc. IEEE-Computer Networking Symposium, Washington, D.C., April, 1988. pp. 134-143.
- [14] A. M. Law, W. D. Kelton. **Simulation Modeling & Analysis**. 2<sup>nd</sup> Ed. McGraw-Hill. 1991.
- [15] J. R. Norris. **Markov Chains**. Cambridge University Press. 1998.
- [16] S. Ross. **Stochastic Process**. 2<sup>nd</sup> Ed. John Wiley & Sons, Inc. 1996.

ภาคผนวก

## ภาคผนวก ก.

## ตัวอย่างการหาการกระจายของความน่าจะเป็นของขนาดคิว

ในตัวอย่างต่อไปนี้จะเป็นการหาความน่าจะเป็นในการเปลี่ยนสถานะ (Transition Probability) และความน่าจะเป็นของขนาดคิวที่สถานะคงตัว (Stationary Probability) เพื่อพิสูจน์สมการที่ (2.19)

สมมติให้เกตเวย์มีบัฟเฟอร์ขนาดเท่ากับ 5 มีบัฟเฟอร์ว่างเมื่อเวลาเริ่มต้น  $t = 0$  มีลักษณะที่จำลองได้ตามรูปที่ 2.6 และมีการเปลี่ยนสถานะของคิวตามรูปที่ 2.5 ถ้าอัตราการมาถึงเกตเวย์ของแพคเกจมีค่าเฉลี่ยเท่ากับ  $\lambda$  และอัตราการให้บริการของเกตเวย์มีค่าเฉลี่ยเท่ากับ  $\mu$  จะถือได้ว่าอัตราการมาถึงของแพคเกจที่เกตเวย์  $\lambda_i = \lambda$  และอัตราการให้บริการของเกตเวย์  $\mu_{i+1} = \mu$  ที่  $i = 0, 1, 2, \dots, K-1$  ซึ่งจะคำนวณหาความน่าจะเป็นในการเปลี่ยนสถานะของขนาดคิว  $P(t)$  ตามขั้นตอนในหัวข้อ 2.4 ซึ่งจะได้ผลดังนี้

$$\begin{aligned}
 P_{00}(t) = & \frac{\mu^5}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} + \frac{\frac{1}{4} \lambda e^{t(-\lambda - \mu + \sqrt{\lambda \mu})} (\lambda + \mu + \sqrt{\lambda \mu})}{\lambda^2 + \lambda \mu + \mu^2} \\
 & - \frac{1}{4} \frac{\lambda e^{t(-\lambda - \mu - \sqrt{\lambda \mu})} (-\lambda - \mu + \sqrt{\lambda \mu})}{\lambda^2 + \lambda \mu + \mu^2} + \left( \frac{1}{36} \sqrt{3} \mu^4 \lambda + \frac{1}{12} \lambda \sqrt{\lambda \mu} \mu^3 \right. \\
 & + \frac{1}{12} \mu^3 \lambda^2 \sqrt{3} + \frac{1}{6} \mu^2 \lambda^2 \sqrt{\lambda \mu} + \frac{1}{9} \lambda^3 \sqrt{3} \mu^2 + \frac{1}{6} \lambda^3 \mu \sqrt{\lambda \mu} + \frac{1}{12} \lambda^4 \sqrt{3} \mu \\
 & \left. + \frac{1}{36} \lambda^5 \sqrt{3} + \frac{1}{12} \lambda^4 \sqrt{\lambda \mu} \right) \sqrt{3} e^{t(-\lambda - \mu + \sqrt{3} \sqrt{\lambda \mu})} / ( \\
 & \mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5) + \left( \frac{1}{36} \sqrt{3} \mu^4 \lambda - \frac{1}{12} \lambda \sqrt{\lambda \mu} \mu^3 \right. \\
 & + \frac{1}{12} \mu^3 \lambda^2 \sqrt{3} - \frac{1}{6} \mu^2 \lambda^2 \sqrt{\lambda \mu} + \frac{1}{9} \lambda^3 \sqrt{3} \mu^2 - \frac{1}{6} \lambda^3 \mu \sqrt{\lambda \mu} + \frac{1}{12} \lambda^4 \sqrt{3} \mu \\
 & \left. + \frac{1}{36} \lambda^5 \sqrt{3} - \frac{1}{12} \lambda^4 \sqrt{\lambda \mu} \right) \sqrt{3} e^{t(-\lambda - \mu - \sqrt{3} \sqrt{\lambda \mu})} / ( \\
 & \mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5) + \frac{\frac{1}{3} \lambda e^{t(-\mu - \lambda)}}{\mu + \lambda}
 \end{aligned}$$

$$p_{01}(t) =$$

$$\begin{aligned} & \frac{\lambda \mu^4}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} + \frac{\frac{1}{4} \lambda e^{t(-\lambda - \mu + \sqrt{\lambda \mu})} (-\mu^2 + \sqrt{\lambda \mu} \lambda)}{\mu (\lambda^2 + \lambda \mu + \mu^2)} \\ & - \frac{1}{4} \frac{\lambda e^{t(-\lambda - \mu - \sqrt{\lambda \mu})} (\sqrt{\lambda \mu} \lambda + \mu^2)}{\mu (\lambda^2 + \lambda \mu + \mu^2)} - \frac{1}{72} e^{t(-\lambda - \mu + \sqrt{3} \sqrt{\lambda \mu})} (2 \sqrt{3} \mu^5 \lambda \\ & - 6 \mu^3 \lambda^2 \sqrt{\lambda \mu} - 4 \mu^3 \lambda^3 \sqrt{3} - 12 \lambda^3 \sqrt{\lambda \mu} \mu^2 - 6 \lambda^4 \mu^2 \sqrt{3} \\ & - 12 \sqrt{\lambda \mu} \lambda^4 \mu - 4 \lambda^5 \mu \sqrt{3} - 6 \lambda^5 \sqrt{\lambda \mu}) \sqrt{3} / (\mu \\ & (\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5)) - \frac{1}{72} e^{t(-\lambda - \mu - \sqrt{3} \sqrt{\lambda \mu})} ( \\ & 2 \sqrt{3} \mu^5 \lambda + 6 \mu^3 \lambda^2 \sqrt{\lambda \mu} - 4 \mu^3 \lambda^3 \sqrt{3} + 12 \lambda^3 \sqrt{\lambda \mu} \mu^2 - 6 \lambda^4 \mu^2 \sqrt{3} \\ & + 12 \sqrt{\lambda \mu} \lambda^4 \mu - 4 \lambda^5 \mu \sqrt{3} + 6 \lambda^5 \sqrt{\lambda \mu}) \sqrt{3} / (\mu \\ & (\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5)) - \frac{1}{3} \frac{\lambda e^{t(-\mu - \lambda)}}{\mu + \lambda} \end{aligned}$$

$$p_{02}(t) =$$

$$\begin{aligned} & \frac{\lambda^2 \mu^3}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} \\ & - \frac{1}{4} \frac{\lambda e^{t(-\lambda - \mu + \sqrt{\lambda \mu})} (\lambda + \mu + \sqrt{\lambda \mu}) \sqrt{\lambda \mu}}{\mu (\lambda^2 + \lambda \mu + \mu^2)} \\ & - \frac{1}{4} \frac{\lambda e^{t(-\lambda - \mu - \sqrt{\lambda \mu})} (-\lambda - \mu + \sqrt{\lambda \mu}) \sqrt{\lambda \mu}}{\mu (\lambda^2 + \lambda \mu + \mu^2)} - \\ & \frac{1}{36} \frac{\lambda e^{t(-\lambda - \mu + \sqrt{3} \sqrt{\lambda \mu})} \sqrt{3} (3 \mu \sqrt{\lambda \mu} + \lambda \sqrt{3} \mu - 3 \sqrt{\lambda \mu} \lambda - 2 \lambda^2 \sqrt{3})}{\mu (\mu^2 - \lambda \mu + \lambda^2)} \\ & + \\ & \frac{1}{36} \frac{\lambda e^{t(-\lambda - \mu - \sqrt{3} \sqrt{\lambda \mu})} \sqrt{3} (3 \mu \sqrt{\lambda \mu} - \lambda \sqrt{3} \mu - 3 \sqrt{\lambda \mu} \lambda + 2 \lambda^2 \sqrt{3})}{\mu (\mu^2 - \lambda \mu + \lambda^2)} \\ & - \frac{1}{3} \frac{\lambda^2 e^{t(-\mu - \lambda)}}{\mu (\mu + \lambda)} \end{aligned}$$

$$p_{03}(t) =$$

$$\begin{aligned} & \frac{\lambda^3 \mu^2}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} \\ & - \frac{1}{4} \frac{\lambda^2 e^{t(-\lambda - \mu + \sqrt{\lambda \mu})} (\lambda + \mu + \sqrt{\lambda \mu}) \sqrt{\lambda \mu}}{\mu^2 (\lambda^2 + \lambda \mu + \mu^2)} \\ & - \frac{1}{4} \frac{\lambda^2 e^{t(-\lambda - \mu - \sqrt{\lambda \mu})} (-\lambda - \mu + \sqrt{\lambda \mu}) \sqrt{\lambda \mu}}{\mu^2 (\lambda^2 + \lambda \mu + \mu^2)} \\ & + \frac{1}{36} \frac{\lambda^2 e^{t(-\lambda - \mu + \sqrt{3} \sqrt{\lambda \mu})} \sqrt{3} (2 \mu^2 \sqrt{3} + 3 \mu \sqrt{\lambda \mu} - \lambda \sqrt{3} \mu - 3 \sqrt{\lambda \mu} \lambda)}{\mu^2 (\mu^2 - \lambda \mu + \lambda^2)} \\ & + \frac{1}{36} \frac{\lambda^2 e^{t(-\lambda - \mu - \sqrt{3} \sqrt{\lambda \mu})} \sqrt{3} (-2 \mu^2 \sqrt{3} + 3 \mu \sqrt{\lambda \mu} + \lambda \sqrt{3} \mu - 3 \sqrt{\lambda \mu} \lambda)}{\mu^2 (\mu^2 - \lambda \mu + \lambda^2)} \\ & + \frac{\frac{1}{3} \lambda^2 e^{t(-\mu - \lambda)}}{\mu (\mu + \lambda)} \end{aligned}$$

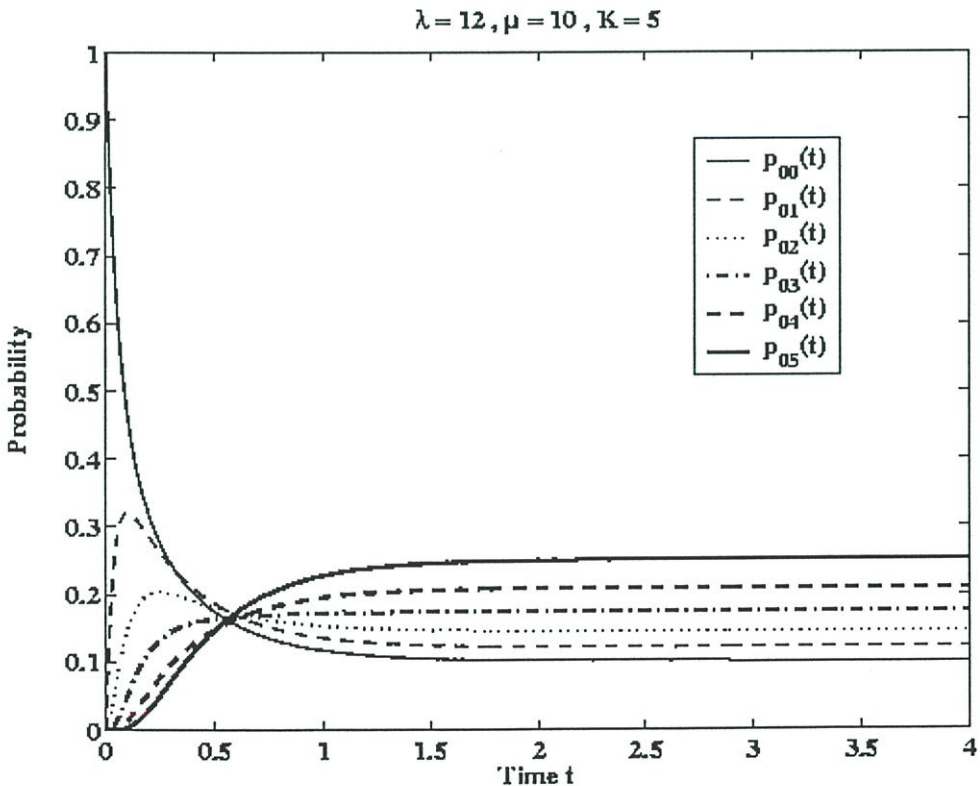
$$p_{04}(t) =$$

$$\begin{aligned} & \frac{\lambda^4 \mu}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} + \frac{\frac{1}{4} \lambda^2 e^{t(-\lambda - \mu + \sqrt{\lambda \mu})} (\mu \sqrt{\lambda \mu} - \lambda^2)}{\mu^2 (\lambda^2 + \lambda \mu + \mu^2)} \\ & - \frac{1}{4} \frac{\lambda^2 e^{t(-\lambda - \mu - \sqrt{\lambda \mu})} (\mu \sqrt{\lambda \mu} + \lambda^2)}{\mu^2 (\lambda^2 + \lambda \mu + \mu^2)} + \frac{1}{72} \lambda^2 e^{t(-\lambda - \mu + \sqrt{3} \sqrt{\lambda \mu})} ( \\ & -6 \sqrt{\lambda \mu} \mu^5 - 4 \sqrt{3} \mu^5 \lambda - 12 \sqrt{\lambda \mu} \mu^4 \lambda - 6 \lambda^2 \mu^4 \sqrt{3} - 12 \mu^3 \lambda^2 \sqrt{\lambda \mu} \\ & - 4 \mu^3 \lambda^3 \sqrt{3} - 6 \lambda^3 \sqrt{\lambda \mu} \mu^2 + 2 \lambda^5 \mu \sqrt{3}) \sqrt{3} / (\mu^3 \\ & (\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5)) + \frac{1}{72} \lambda^2 e^{t(-\lambda - \mu - \sqrt{3} \sqrt{\lambda \mu})} ( \\ & 6 \sqrt{\lambda \mu} \mu^5 - 4 \sqrt{3} \mu^5 \lambda + 12 \sqrt{\lambda \mu} \mu^4 \lambda - 6 \lambda^2 \mu^4 \sqrt{3} + 12 \mu^3 \lambda^2 \sqrt{\lambda \mu} \\ & - 4 \mu^3 \lambda^3 \sqrt{3} + 6 \lambda^3 \sqrt{\lambda \mu} \mu^2 + 2 \lambda^5 \mu \sqrt{3}) \sqrt{3} / (\mu^3 \\ & (\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5)) + \frac{\frac{1}{3} \lambda^3 e^{t(-\mu - \lambda)}}{\mu^2 (\mu + \lambda)} \end{aligned}$$

$$p_{05}(t) =$$

$$\frac{\lambda^5}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} + \frac{\frac{1}{4} \lambda^3 e^{(t(-\lambda - \mu + \sqrt{\lambda \mu}))} (\lambda + \mu + \sqrt{\lambda \mu})}{\mu^2 (\lambda^2 + \lambda \mu + \mu^2)} - \frac{1}{4} \frac{\lambda^3 e^{(t(-\lambda - \mu - \sqrt{\lambda \mu}))} (-\lambda - \mu + \sqrt{\lambda \mu})}{\mu^2 (\lambda^2 + \lambda \mu + \mu^2)} - \frac{1}{72} \lambda^3 e^{(t(-\lambda - \mu + \sqrt{3} \sqrt{\lambda \mu}))} (8 \mu^3 \lambda^2 \sqrt{3} + 6 \sqrt{3} \mu^4 \lambda + 12 \lambda \sqrt{\lambda \mu} \mu^3 + 6 \mu^4 \sqrt{\lambda \mu} + 2 \mu^5 \sqrt{3} + 12 \mu^2 \lambda^2 \sqrt{\lambda \mu} + 6 \lambda^3 \sqrt{3} \mu^2 + 6 \lambda^3 \mu \sqrt{\lambda \mu} + 2 \lambda^4 \sqrt{3} \mu) \sqrt{3} / (\mu^3 (\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5)) - \frac{1}{72} \lambda^3 e^{(t(-\lambda - \mu - \sqrt{3} \sqrt{\lambda \mu}))} (6 \sqrt{3} \mu^4 \lambda + 8 \mu^3 \lambda^2 \sqrt{3} - 12 \lambda \sqrt{\lambda \mu} \mu^3 - 6 \mu^4 \sqrt{\lambda \mu} + 2 \mu^5 \sqrt{3} - 12 \mu^2 \lambda^2 \sqrt{\lambda \mu} + 6 \lambda^3 \sqrt{3} \mu^2 - 6 \lambda^3 \mu \sqrt{\lambda \mu} + 2 \lambda^4 \sqrt{3} \mu) \sqrt{3} / (\mu^3 (\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5)) - \frac{1}{3} \frac{\lambda^3 e^{(t(-\mu - \lambda))}}{\mu^2 (\mu + \lambda)}$$

และแสดงกราฟความน่าจะเป็นของขนาดคิวได้ดังรูปที่ ก.1



รูปที่ ก.1 ความน่าจะเป็นในการเปลี่ยนสถานะของขนาดคิวที่เวลา t ใดๆ

ซึ่งผลจากกราฟรูปที่ ก.1 แสดงให้เห็นว่าที่ระบบเข้าสู่สถานะคงตัว ( $t \rightarrow \infty$ ) ความน่าจะเป็นในการเปลี่ยนสถานะจะคงที่ที่ค่าหนึ่ง ซึ่ง  $\pi(q)$  ที่หาได้จากสมการที่ (2.18) โดยการหาค่าลิมิตของ  $p_{00}(t), p_{01}(t), p_{02}(t), p_{03}(t), p_{04}(t), p_{05}(t)$  เมื่อ  $t \rightarrow \infty$  ดังในสมการ (2.13) จะได้

$$\pi(0) = \lim_{t \rightarrow \infty} p_{00}(t) = \frac{\mu^5}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} = \frac{1}{1 + \frac{\lambda}{\mu} + \frac{\lambda^2}{\mu^2} + \frac{\lambda^3}{\mu^3} + \frac{\lambda^4}{\mu^4} + \frac{\lambda^5}{\mu^5}}$$

$$\pi(1) = \lim_{t \rightarrow \infty} p_{01}(t) = \frac{\lambda \mu^4}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} = \frac{\frac{\lambda}{\mu}}{1 + \frac{\lambda}{\mu} + \frac{\lambda^2}{\mu^2} + \frac{\lambda^3}{\mu^3} + \frac{\lambda^4}{\mu^4} + \frac{\lambda^5}{\mu^5}}$$

$$\pi(2) = \lim_{t \rightarrow \infty} p_{02}(t) = \frac{\lambda^2 \mu^3}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} = \frac{\frac{\lambda^2}{\mu^2}}{1 + \frac{\lambda}{\mu} + \frac{\lambda^2}{\mu^2} + \frac{\lambda^3}{\mu^3} + \frac{\lambda^4}{\mu^4} + \frac{\lambda^5}{\mu^5}}$$

$$\pi(3) = \lim_{t \rightarrow \infty} p_{03}(t) = \frac{\lambda^3 \mu^2}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} = \frac{\frac{\lambda^3}{\mu^3}}{1 + \frac{\lambda}{\mu} + \frac{\lambda^2}{\mu^2} + \frac{\lambda^3}{\mu^3} + \frac{\lambda^4}{\mu^4} + \frac{\lambda^5}{\mu^5}}$$

$$\pi(4) = \lim_{t \rightarrow \infty} p_{04}(t) = \frac{\lambda^4 \mu}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} = \frac{\frac{\lambda^4}{\mu^4}}{1 + \frac{\lambda}{\mu} + \frac{\lambda^2}{\mu^2} + \frac{\lambda^3}{\mu^3} + \frac{\lambda^4}{\mu^4} + \frac{\lambda^5}{\mu^5}}$$

$$\pi(5) = \lim_{t \rightarrow \infty} p_{05}(t) = \frac{\lambda^5}{\mu^2 \lambda^3 + \mu^3 \lambda^2 + \mu \lambda^4 + \mu^4 \lambda + \lambda^5 + \mu^5} = \frac{\frac{\lambda^5}{\mu^5}}{1 + \frac{\lambda}{\mu} + \frac{\lambda^2}{\mu^2} + \frac{\lambda^3}{\mu^3} + \frac{\lambda^4}{\mu^4} + \frac{\lambda^5}{\mu^5}}$$

ซึ่งเป็นไปตามสมการที่ (2.19) จากนั้นใช้สมการที่ (2.20) ถึง (2.23) เพื่อคำนวณหาความน่าจะเป็นที่จะเกิดการครีโอล, ค่าทรูพุท, ค่าเฉลี่ยขนาดคิว และค่าประวิงเวลาที่เกิดจากการรอดอยในคิวได้

ภาคผนวก ข.

ผลงานวิจัยและบทความทางวิชาการที่ได้รับการตีพิมพ์

1. S. Prabhavat, R. Varakulsiripunth, S. Noppanakeepong, "Throughput Improvement on RED Mechanism.", In Proceeding of the 8<sup>th</sup> IEEE International Conference on Communication Systems (ICCS 2002), Singapore, November 25-28, 2002.
2. S. Prabhavat, S. Noppanakeepong, R. Varakulsiripunth, "Effect of The Bottleneck Bandwidth Sharing Fairness Control to Network Round-Trip Delay.", In Proceeding of IEEE Student Conference On Research and Development (SCOReD 2001), Malaysia, February 2001.
3. S. Prabhavat, S. Noppanakeepong, et. al. "Gigabit Satellite: Tele-Training and Tele-Education Application.", In Proceeding of the 2<sup>nd</sup> International Forum on Advanced Satellite Communications in the Asia-Pacific Region, Japan, November 2000.
4. สุเมธ ประภาวัต, สุทธิชัย นพนาถิพงษ์, "การจำลองภาระงานเพื่อทดสอบสถานภาพของเครือข่ายท้องถิ่น.", วิศวกรรมลาดกระบัง , ปีที่ 15 ฉบับที่ 2, ธันวาคม 2541. หน้า 82-87.

THE 8th INTERNATIONAL CONFERENCE ON  
COMMUNICATION SYSTEMS

**ICCS 2002**

25-28 November 2002

Carlton Hotel

Singapore

**FINAL PROGRAMME**

Edited By

**A. Nallanathan**

*Organized by:*

IEEE Singapore Section  
Communications Chapter

*In Cooperation with:*

ECE Department  
National University of Singapore

Institute for  
Communications Research



**IEEE**

*Networking the World™*

IEEE Singapore Section  
Communications Chapter



Institute for  
Communications  
Research

A\*STAR

## THROUGHPUT IMPROVEMENT ON RED MECHANISM

Sumet Prabhavat, Ruttikorn Varakulsiripunth, and Suthichai Noppanakeepong

Faculty of Engineering and Research Center for Communications and Information Technology (ReCCIT)  
King Mongkut's Institute of Technology Ladkrabang, Chalongkrung Road, Ladkrabang, Bangkok 10520, Thailand  
s1061169@kmitl.ac.th, sumetp@usa.net

**Abstract** - Internet Engineering Task Force (IETF) had recommended the active queue management (AQM) scheme to solve the global synchronization problems in TCP/IP based networks for next generation routers. Since then, Random Early Detection (RED) has been proposed as one of the AQM schemes. However, the major problems of RED mechanism are low throughput achievement and high number of consecutive drop. In this paper, we have proposed the Extended Drop Slope Random Early Detection (ExRED) mechanism in order to overcome these problems. The ExRED can reduce both the average number of dropped packets and the consecutive drop probability, hence network throughput is improved while all other advantages of RED are retained.

**Keywords** – Throughput, Drop Probability, Random Early Detection, Active Queue Management

### I. INTRODUCTION

Previously, buffers are a key component of a packet switched network, as they absorb arrivals of packet and hence reduce losses. However, Internet technology is developed and growth very fast. Higher speed and larger size network tend to build up at high load and increase network delays. The traditional technique for managing router queue length is to set a maximum length for each buffer queue, accept packets in the queue until the maximum length is exceeded, then drop subsequent incoming packets until the queue decreases below its maximum value. This buffer management scheme is referred to as Tail Drop. But it possible causes the important problems such as buffer overflow, lock-out phenomenon, and global synchronization problem. The higher speed and larger size network cause performance is getting worse.

To solve the problem and improve performance, active queue management (AQM) is recommended by Internet Engineering Task Force (IETF) for using in the routers of next generation Internet [1][2]. Therefore, Random Early Detection (RED) [3], an active queue management, was proposed. However, RED has several problems such as low throughput and high number of consecutive drop as stated in [4].

The objective of our research is to improve throughput by decreasing a packet drop and consecutive drop probability of RED mechanism. This will in turn improve the QoS of Internet gateways. In this paper, we propose our mechanism

and evaluate performance compare to the existed mechanism by simulation.

The rest of paper is organized as the follows. In Section II, we briefly review RED. In Section III, we propose ExRED model. Then, we compare the ExRED to the traditional and existed mechanism in Section IV. In Section V, we simulate our mechanism and compare to the existed mechanism. Section VI concludes the paper.

### II. RED MECHANISM

RED [3] itself consists of two main parts, i.e., the estimation of an average queue size and the decision of whether or not to drop an incoming packet. An average queue size  $\hat{q}$  is calculated by a current queue size (or instantaneous queue size)  $q$  using an exponentially weighted moving average (EWMA) [3] as shown below.

$$\hat{q} \leftarrow \begin{cases} (1-w)\hat{q} + wq & ; \quad q \neq 0 \\ (1-w)^{\frac{1}{\alpha}} \hat{q} & ; \quad \text{otherwise} \end{cases} \quad (1)$$

where  $w$  is an EWMA parameter which is a small constant value and defined by [3] as queue weight.

In the 2<sup>nd</sup> portion of RED algorithm, RED decides whether or not to drop an incoming packet. It is RED's particular algorithm for dropping that results in performance improvement for responsive flows. There are two thresholds figure prominently in this decision process. Minimum threshold  $k_l$  specifies the average queue size below which no packet will be dropped and maximum threshold  $k_h$  specifies the average queue size above which all packets will be dropped. When the average queue size varies from minimum to maximum threshold level, the packets will be dropped with probability that vary linearly from 0 to  $\max_p$ ,

where  $\max_p$  is the maximum drop probability parameter.

Then, the packet drop probability distribution function is defined as shown below [3]. The suitable value for parameters in this function were discussed in [3] and [7].

$$P_d(\hat{q}) = \begin{cases} 0 & ; \quad \hat{q} < k_l \\ \frac{\hat{q} - k_l}{k_h - k_l} \times \max_p & ; \quad k_l \leq \hat{q} < k_h \\ 1 & ; \quad \hat{q} \geq k_h \end{cases} \quad (2)$$

By using the PASTA property [5] with arrival rate  $\lambda$  and service rate  $\mu$ , RED performance can be presented in average packet drop probability, consecutive drop probability, and queuing delay of finite buffer size  $K$  based on Markovian model [4][6] as the following:

#### A. Drop Probability

Drop probability in a RED router can be approximated by

$$P_{drop} = \sum_{\hat{q}=k_h}^K \pi(\hat{q}) + \sum_{\hat{q}=k_l}^{k_h-1} (p_d(\hat{q}) \times \pi(\hat{q})). \quad (3)$$

Where  $\pi(\hat{q})$  is stationary probability distribution of the average queue size and  $p_d(\hat{q})$  is packet drop probability.

#### B. Consecutive Dropped Packet

$P_{drop}^{cons}(N)$  = Probability that  $N$  packets will be dropped consecutively and

$$\forall n \geq 0, \quad P_{drop}^{cons}(N > n) = \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) (p_d(\hat{q}))^{n+1}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) p_d(\hat{q})}. \quad (4)$$

Equation (4) allows us in particular to evaluate the mean of the number of consecutive dropped packets, as shown below.

$$E(N) = 1 + \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) \frac{(p_d(\hat{q}))^2}{1-p_d(\hat{q})}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) p_d(\hat{q})}. \quad (5)$$

#### C. Average Queuing Delay

By using Little's theorem, average queuing delay is given by

$$D = \frac{E(\hat{Q})}{(1-P_{drop})\lambda} - \frac{1}{\mu}. \quad (6)$$

$E(\hat{Q})$  is mean of average queue size and given by

$$E(\hat{Q}) = \sum_{k=0}^K \frac{k \left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1-p_d(\hat{q}))}{\sum_{k=0}^K \left(\left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1-p_d(\hat{q}))\right)}. \quad (7)$$

### III. EXRED PRINCIPLE

The proposed scheme of our research is called Extended Drop Slope Random Early Detection (ExRED). The model of ExRED is shown in Fig.1,

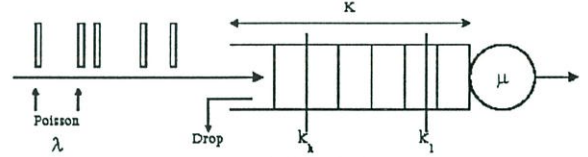


Fig. 1. Analytical model for ExRED

According to (2) to (7), we have observed that  $p_d(\hat{q})$  influence all above performance issues especially drop probability. Packets will be dropped consecutively when  $\hat{q} \geq k_h$  and a packet will be dropped randomly when  $k_l \geq \hat{q} > k_h$ . It is common sense that, under high load, most of packet drops occur when  $\hat{q} \geq k_h$ . In order to reduce a number of packet drop and consecutive drop, we modify drop distribution function and decrease drop probability in case of  $\hat{q} \geq k_h$ . To keep packet drop rate increasing smoothly but continue with a higher rate when queue size is more closed to limit of buffer size,  $k_h \leq \hat{q} \leq K$ , the function of  $p_d$  is modified to be a second order polynomial function of  $\hat{q}$  and a new drop distribution must satisfy three conditions as below.

- $p_d(\hat{q} = k_h) = \max_p$
- $p_d(\hat{q} = K) = 1$
- $\left. \frac{d}{d\hat{q}} p_d(\hat{q}) \right|_{\hat{q}=k_h} = \frac{\max_p}{k_h - k_l}$

Thus, new packet drop distribution can be expressed as:

$$p_d(\hat{q}) = \begin{cases} 0 & ; \hat{q} < k_l \\ \frac{\hat{q} - k_l}{k_h - k_l} \times \max_p & ; k_l \leq \hat{q} < k_h \\ a_2 \hat{q}^2 + a_1 \hat{q} + a_0 & ; k_h \leq \hat{q} \leq K \end{cases} \quad (8)$$

where

$$a_2 = \frac{(k_h - k_l) - (K - k_l) \max_p}{(k_h - k_l)(K - k_h)^2}$$

$$a_1 = \frac{(k_h^2 + K^2 - 2k_h k_l) \max_p - 2k_h^2 + 2k_h k_l}{(k_h - k_l)(K - k_h)^2}$$

$$a_0 = \frac{k_h^3 - k_h^2 k_l - (k_h^2 + 2k_h k_l + k_l K) K \max_p}{(k_h - k_l)(K - k_h)^2}$$

Packet drop probability will increase with higher rate for more seriously lack of available buffer space, the parameter setting must satisfy the following condition:

$$\max_p \leq \frac{k_h - k_l}{K - k_l}$$

Note that the recommended parameter setting in [3] and [7] is also satisfied above condition. When average queue size exceeds the maximum threshold, packet drop probability vs. average queue size by varying the difference of minimum and maximum threshold can be illustrated in Fig. 2.

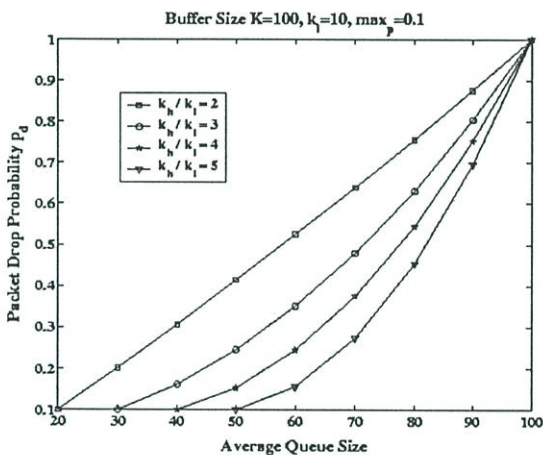


Fig. 2. Drop function of ExRED when average queue size exceeds maximum threshold

#### IV. COMPARISON AND ANALYSIS

We compare the drop function curve of ExRED to RED and illustrate as shown in Fig. 3. Performance will be presented in simulation result next section. However, in this section, we can roughly estimate the result using (3). When  $p_d$  is decreased, it follows then from (3) that the average packet drop is written as (9).

$$P_{drop} = \sum_{\hat{q}=k_h}^K \pi(\hat{q}) \times (a_2 \hat{q}^2 + a_1 \hat{q} + a_0) + \sum_{\hat{q}=k_l}^{k_h-1} p_d(\hat{q}) \times \pi(\hat{q}) \quad (9)$$

If  $P_{drop}^{RED}$  and  $P_{drop}^{ExRED}$  are drop probability of RED obtained by (3) and ExRED obtained by (9), respectively. Fig. 4 illustrates  $P_{drop}^{ExRED} \leq P_{drop}^{RED}$  and therefore ExRED throughput gets better. Similarly, the decreasing of  $p_d(\hat{q})$ ,

the consecutive dropped packet probability  $P_{drop}^{cons}(N)$  from (4) can be expressed by

$$\forall n, 1 < n < K, P_{drop}^{cons}(N_{ExRED} > n) < P_{drop}^{cons}(N_{RED} > n).$$

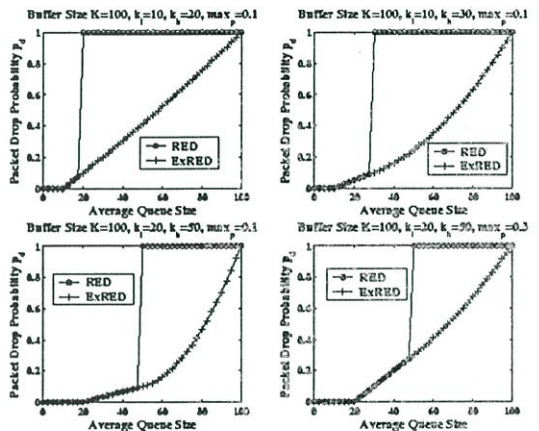


Fig. 3. Comparing of RED and ExRED Drop function

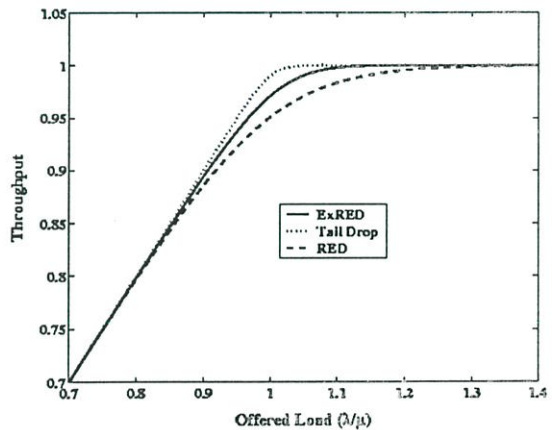


Fig. 4. Estimation of Throughput vs. Offered load

#### V. SIMULATION RESULTS

To compare the simulation result of our proposed mechanism, we model the discrete event simulation by using concept in [8] and [9]. We suppose that the arrivals occur randomly according to a Poisson Process with rate  $\lambda$  and service times of single-server with buffer size  $K$  is exponentially distributed at rate  $\mu$ . We select the value of parameters setting in the simulation as the follows:

- $\mu = 1000$  packets per second,
- $K = 100$  packets,
- $w = 0.02$ ,
- $k_l = 10, k_h = 30$ : The values were chosen such that  $k_l \leq \frac{1}{4}K$  and  $k_h \geq 2k_l$  as suggested in [3].
- $\max_p = 0.1$ : The value was chosen as suggested in [7].

Throughput of Tail Drop, RED, and ExRED with varying offered load can be illustrated in Fig. 5.

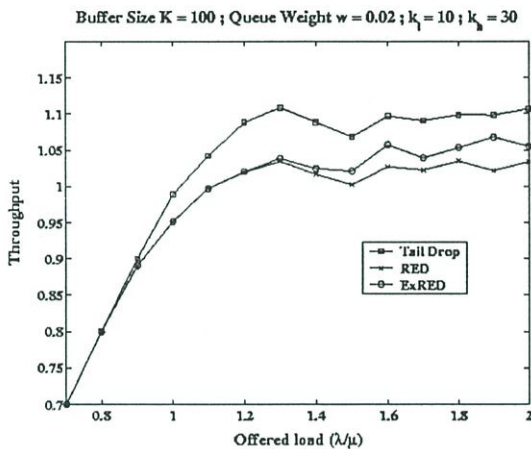


Fig. 5. Throughput of Tail Drop, RED, and ExRED vs. Offered load

Consecutive drop probability and its average of Tail Drop, RED, and ExRED can be illustrated in Fig. 6 and Fig. 7. Average queuing delay of Tail Drop, RED, and ExRED with varying offered load can be illustrated in Fig. 8.

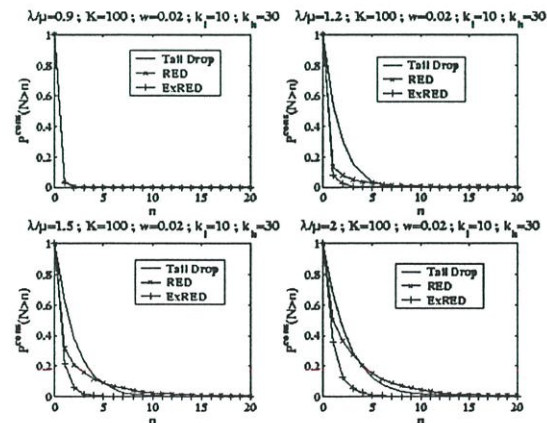


Fig. 6. Consecutive Drop Probability vs. Offered load at 0.9, 1.2, 1.5, and 2.0

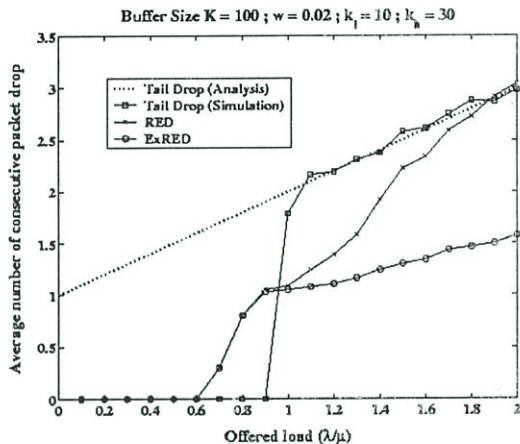


Fig. 7. Average number of Consecutive Drop vs. Offered load

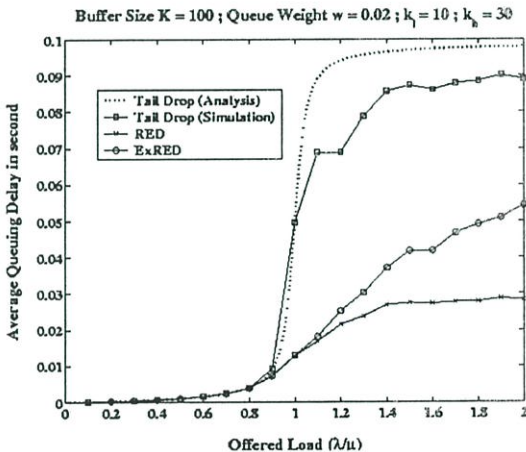


Fig. 8. Average Queuing Delay vs. Offered load

### VI. CONCLUSIONS

In order to relax an aggressive drop in case of the average queue size is over maximum threshold and not exceeds limit of buffer size, ExRED reduces packet drop probability. The 2<sup>nd</sup> order polynomial function shape of packet drop function provides a flexible increasing of drop probability. When average queue size exceeds maximum threshold, drop probability is set to the maximum drop probability parameter. The decreasing of available buffer space, the increasing rate of drop probability is accelerated higher. Not only the average packet drop is reduced, but also the consecutive drop. The simulation results in Fig. 5, Fig. 6, and Fig. 7, of section IV present that ExRED reduces packet drop and consecutive drop probability, hence throughput get better. Fig. 8 illustrates average queuing delay brought about by ExRED. It is a simple tradeoff between delay and throughput. However, we can conclude that ExRED

performs a much lower delay when compare to Tail Drop while it performs a higher throughput when compare to RED. Moreover, end-to-end delay might be better if propagation and transmission delay of a retransmitted dropped packet is a long time.

Next, the complexity of implementation will be briefly discussed on this section. For average queue size exceeds the maximum threshold, packet drop probability will be computed for an incoming packet. This is unavoidably more complicate on ExRED than RED. Indeed, it does not add more difficulty. The algorithm in this part similar to what both RED and ExRED mechanism have done when average queue size is higher than minimum threshold but does not exceed maximum threshold. Packet drop probability could be computed on high performance processor or gotten from the table of previously calculated value stored in memory.

## VII. FUTURE WORKS

In the model with Poisson traffic source, we have shown that throughput and consecutive drop have been improved. We suppose that burst and smooth traffic cause the same drop probability in according to previous researches, i.e.[4] and [6], have stated as the advantages of RED. However, we will study how responsive and non-responsive traffic source effect to ExRED Gateway comparing to Tail Drop and RED by using Markovian model. In addition, the explicit congestion notification (ECN) mechanism will be an extended part that we interest.

## REFERENCES

- [1] "Recommendations on queue management and congestion avoidance in the Internet", *RFC2309*, April, 1999.
- [2] "Assured Forwarding PHB Group", *RFC 2597*, June, 1999.
- [3] S. Floyd and V. Jacobsan, "Random Early Detection Gateways for Congestion Avoidance", In *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397-413, August, 1993.
- [4] T.Bonald, M. May, and J-C.Bolot, "Analytic Evaluation of RED Performance", In *IEEE INFOCOM2000*, Tel-Aviv, Israel, March, 26-30, 2000.
- [5] R. W. Wolf, "Poisson Arrivals see Time Average", *Operations Research*, 20, 223-231, 1982.
- [6] R.Laalaloua, T.Czachorski, and T.Atmaca, "Markovian Model of RED Mechanism", In *Proceeding of First IEEE/ACM International Symposium on Cluster Computing and the Grid 2001*, pp. 610-617, 2001.
- [7] S. Floyd, RED: "Discussions of Setting Parameters", <http://www.aciri.org/floyd/REDparameters.txt>, Nov, 1997
- [8] K.Kant, *Introduction to Computer System Performance Evaluation*, McGraw-Hill 1992.
- [9] Boudewijn R.Haverkort, *Performance of Computer Communication Systems*, Wiley, 1998.

## APPENDIX

In the section IV and V, we assume that arrivals occur randomly according to a Poisson Process with rate  $\lambda$  and service times of single-server with buffer size  $K$  is exponentially distributed at rate  $\mu$ . Thus, performance model of Tail Drop mechanism can be estimated by using M/M/1/K queuing model.

Equation (10) expresses drop probability.

$$P_{drop} = \begin{cases} \frac{\left(1 - \frac{\lambda}{\mu}\right) \left(\frac{\lambda}{\mu}\right)^K}{1 - \left(\frac{\lambda}{\mu}\right)^{K+1}} & ; \lambda \neq \mu \\ \frac{1}{K+1} & ; \lambda = \mu \end{cases} \quad (10)$$

Equation (11) and (12) express consecutive drop probability and average number of consecutive drop, respectively.

$$P_{drop}^{cons}(N > n) = \left( \frac{\frac{\lambda}{\mu}}{1 + \frac{\lambda}{\mu}} \right)^n \quad (11)$$

$$E(N) = 1 + \frac{\lambda}{\mu} \quad (12)$$

Equation (13) expresses queuing delay.

$$D = \frac{E(Q)}{(1 - P_{drop})\lambda} - \frac{1}{\mu}, \quad (13)$$

where  $E(Q)$  is mean of queue size and given by (14).

$$E(Q) = \begin{cases} \frac{\frac{\lambda}{\mu} \left( 1 - (K+1) \left(\frac{\lambda}{\mu}\right)^K + K \left(\frac{\lambda}{\mu}\right)^{K+1} \right)}{\left(1 - \left(\frac{\lambda}{\mu}\right)^{K+1}\right) \left(1 - \frac{\lambda}{\mu}\right)} & ; \lambda \neq \mu \\ \frac{K}{2} & \lambda = \mu \end{cases} \quad (14)$$

## ประวัติผู้เขียน

นาย สุเมธ ประภาวัต เกิดเมื่อวันที่ 19 กันยายน พ.ศ. 2515 ที่จังหวัดกรุงเทพมหานคร สำเร็จ การศึกษาระดับประถมถึงมัธยมศึกษาจากโรงเรียนสาธิตแห่งมหาวิทยาลัยเกษตรศาสตร์ และสำเร็จ การศึกษาระดับปริญญาตรี วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า (ไฟฟ้าสื่อสาร) จาก มหาวิทยาลัยเชียงใหม่ในปีการศึกษา 2537 เข้าทำงานบริษัทฟูจิตส์ซิสเต็มบิสซิเนสประเทศไทย ใน ตำแหน่งวิศวกรเมื่อปีพ.ศ. 2537 และเข้าศึกษาต่อระดับปริญญาโท วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า (โทรคมนาคม) สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เมื่อปีพ.ศ. 2541