



## รายงานการวิจัยฉบับสมบูรณ์

การพัฒนากลไกควบคุมการใช้บัฟเฟอร์ของเกตเวย์

บนระบบเครือข่ายแบบทีซีพีไอพี

Modified RED Mechanism for Buffer Management on  
TCP/IP Network Gateway

ผศ.ดร. สุเมธ ประภาวัต

ได้รับทุนสนับสนุนงานวิจัยจากงบประมาณเงินรายได้ ประจำปีงบประมาณ พ.ศ. 2547

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



# รายงานการวิจัยฉบับสมบูรณ์

การพัฒนากลไกควบคุมการใช้บัฟเฟอร์ของเกตเวย์

บนระบบเครือข่ายแบบทีซีพีไอพี

**Modified RED Mechanism for Buffer Management on  
TCP/IP Network Gateway**

ผศ.ดร. สุเมธ ประภาวัต

RCH

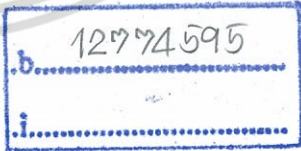
ธ843ก

2547

เลขพญ. ....

เลขทะเบียน 142229

รับเดือนปี 27 โท.ย. 2559



ได้รับทุนสนับสนุนงานวิจัยจากงบประมาณเงินรายได้ ประจำปีงบประมาณ พ.ศ. 2547

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## รายละเอียดโครงการวิจัย

ชื่อโครงการ (ภาษาไทย) การพัฒนากลไกควบคุมการใช้บัฟเฟอร์ของเกตเวย์บนระบบเครือข่าย  
แบบที่ซีพีไอพี

ชื่อโครงการ(ภาษาอังกฤษ) Modified RED Mechanism for Buffer Management on TCP/IP  
Network Gateway

แหล่งเงิน เงินรายได้ คณะเทคโนโลยีสารสนเทศ สจล.

ประจำปีงบประมาณ 2547 จำนวนเงินที่ได้รับการสนับสนุน 57,750 บาท

ระยะเวลาทำการวิจัย 12 เดือน ตั้งแต่ 1 มกราคม 2547 ถึง 31 ธันวาคม 2547

ชื่อ-สกุล หัวหน้าโครงการ และผู้ร่วมโครงการวิจัยพร้อมระบุ หน่วยงานต้นสังกัดและอีเมล

1. ผศ.ดร. สุเมธ ประภาวัต คณะเทคโนโลยีสารสนเทศ สจล.

Email: sumet@it.kmitl.ac.th

คำสำคัญ (Keywords): Active queue management, Buffer queue management, RED, ExRED

## บทคัดย่อ

เกตเวย์เป็นจุดเชื่อมต่อระหว่างเครือข่าย และเป็นทางเข้าออกร่วมกันของผู้ใช้งานเครือข่าย การทำงานของเกตเวย์ในการส่งต่อแพ็คเก็ตข้ามเครือข่ายจึงมีผลต่อคุณภาพการติดต่อสื่อสาร หากไม่มีประสิทธิภาพอาจทำให้เกิดปัญหา เช่น เวลาที่ต้องใช้ในการส่งข้อมูลหรือดีเลย์ยาวนาน แพ็คเก็ตสูญหาย ฯลฯ ซึ่งปัญหาดังกล่าวทำให้กลไกควบคุมความคับคั่งของเครือข่าย TCP/IP ปรับลดอัตราการส่งข้อมูล ทำให้ทรูพุทเครือข่ายลดลง และอาจนำไปสู่ปัญหาอื่นตามที่หน่วยงานที่กำกับดูแลมาตรฐานวิศวกรรมอินเทอร์เน็ต (IETF) เคยศึกษาและสรุปเอาไว้ว่ามีผลเสียต่อประสิทธิภาพของเครือข่าย TCP/IP อย่างมาก ดังนั้นกลไกควบคุมจัดการคิวหรือการใช้บัฟเฟอร์ของเกตเวย์จึงเป็นสิ่งสำคัญและมีการศึกษาวิจัยอย่างต่อเนื่อง ในงานวิจัยนี้จะศึกษาตามแนวทางดังนี้ 1) กลไกควบคุมจัดการคิวด้วยการดรอปแพ็คเก็ต 2) กลไกควบคุมด้วยการเปลี่ยนเส้นทางแพ็คเก็ต และ 3) กลไกควบคุมด้วยการแจ้งเตือนผู้ส่ง ซึ่งงานวิจัยทั้งสามนี้ถูกเสนอเพื่อปรับปรุงการทำงานเครือข่าย TCP/IP

การควบคุมจัดการคิวด้วยการดรอปแพ็คเก็ต จะป้องกันการเกิดบัฟเฟอร์เต็มโดยการดรอปแพ็คเก็ตบางส่วนทิ้งก่อนที่บัฟเฟอร์จะเต็ม เพื่อป้องกันบัฟเฟอร์ล้นและแพ็คเก็ตดรอปต่อเนื่อง ซึ่งเกิดขึ้นในการจัดการคิวแบบพาசிฟที่ใช้กันมา เช่น Tail-Drop ซึ่งต่อมา IETF จึงได้เสนอกฎจัดการคิวแบบแอ็กทีฟอย่าง RED แต่ปัญหาของกลไก RED คือการดรอปแพ็คเก็ตมากเกินไป และมีโอกาสเกิดการดรอปต่อเนื่องแม้ว่าบัฟเฟอร์ยังไม่เต็ม งานวิจัยนี้จึงเสนอกฎ ExRED ที่ยอมให้ใช้บัฟเฟอร์มากกว่าหากเนื้อที่ว่างยังเหลือ แต่ก็มีเพิ่มอัตราการดรอปแพ็คเก็ตที่เร็วพอจะป้องกันการเกิดบัฟเฟอร์ล้น ผลการวิเคราะห์และจำลองการทำงานแสดงให้เห็นว่า ExRED มีทรูพุทดีกว่าและมีการดรอปต่อเนื่องน้อยกว่ามาก เมื่อเทียบกับ RED ในขณะที่มีดีเลย์การรอคิวต่ำเมื่อเทียบกับ Tail-Drop

สำหรับการควบคุมการใช้บัฟเฟอร์ด้วยการเปลี่ยนเส้นทางการส่งแพ็คเก็ต เพื่อกระจายทราฟฟิกบนเครือข่าย งานวิจัยนี้เสนอกฎ LDM/RED และ LDM/ExRED เมื่อเกิดความคับคั่งในเครือข่าย จนกระทั่งขนาดคิวในบัฟเฟอร์เพิ่ม จะทำให้เส้นทางสำหรับการส่งแพ็คเก็ตเปลี่ยน เมื่อเปรียบเทียบกับควบคุมด้วยการกระจายทราฟฟิกแบบอื่นๆ ผลการจำลองเหตุการณ์แสดงให้เห็นการลดลงของขนาดคิวสูงสุด (ซึ่งแสดงนัยถึงการลดลงของดีเลย์การรอคิวในบัฟเฟอร์) และมีโอกาสน้อยที่จะเกิด Packet reordering (ที่เป็นปัญหากระทบประสิทธิภาพเครือข่าย TCP/IP) แต่เมื่อเปรียบเทียบกันเองระหว่าง LDM/RED และ LDM/ExRED จะพบว่า LDM/ExRED มีขนาดคิวสูงสุดต่ำกว่าราวๆ 10%-20%

ส่วนการควบคุมการใช้บัฟเฟอร์โดยการแจ้งเตือนผู้ส่ง งานวิจัยนี้เสนอกฎ Mark-reverse ECN ที่ช่วยให้ผู้ส่งปรับการส่งให้เหมาะสมกับสถานการณ์ได้อย่างรวดเร็ว โดยใช้กลไก RED และกลไก ECN ที่มีการปรับปรุง เป้าหมายของงานวิจัยนี้คือ การปรับปรุงให้เครือข่าย TCP/IP สามารถใช้บนเครือข่ายสื่อสารดาวเทียมที่มีดีเลย์สูง ผลการวิเคราะห์และผลทดสอบโดยการจำลองแสดงให้เห็นว่าอัตราการดรอปน้อยกว่าและทรูพุทสูงกว่า เมื่อเปรียบเทียบกับกลไก TCP ECN (หรือ TCP Mark-tail ECN)

## Abstract

Network gateway is an internetworking system that joins multiple networks together. It connects the networks and forwards packets among them. Queue management plays an important role in congestion control of network. It affects network quality, e.g., packet loss, throughput, and network delay. Its inefficiency can cause problems studied by Internet Engineering Task Force (IETF) that have significant impact on TCP/IP network. We study queue management and control based on the following schemes: 1) Packet drop scheme, 2) Traffic distribution scheme, and 3) Explicit congestion feedback scheme.

Since a traditional passive queue management (i.e. Tail-Drop) can cause buffer overflow and packet loss problems, IETF considers the deployment of active queue management (i.e. RED) to mitigate the problems. Although in theory RED mechanism can prevent buffer overflow and packet loss, in practice it cannot effectively reduce the loss rate because it drops packet unnecessarily while buffer is far from full. In addition, if an average queue size is calculated inaccurately, it can cause consecutive drops even though buffer space is actually available. To overcome these problems, we have proposed a new adaptive queue management mechanism, ExRED. It allows high utilization of buffer space when buffer space is available. But it rapidly increases drop probability when the free buffer space is reduced close to the critical point. The numerical and simulation results show that ExRED mechanism can reduce drop probability and burst drops. ExRED has much lower delay as compared to Tail Drop, while it has higher throughput as compared to RED.

In addition to dropping a packet when buffer space is low (or congestion occurs), there are possible solutions studied in this work: 1) Distributing traffic from a congested path to the other path and 2) Sending feedback (i.e. congestion notification) to TCP sources. LDM/RED and LDM/ExRED are proposed as traffic distribution-based queue management. Traffic is shifted from a path incurring congestion to a low utilized path. They can achieve load balancing and maintain low queue occupancy while not causing packet reordering problem. As compared to LDM/RED, LDM/ExRED can reduce maximum queue size by 10%–20%. Mark-Reverse ECN is proposed as congestion feedback-based queue management for TCP/IP over (long delay) satellite networks. It applies RED and ECN mechanisms to notify TCP senders when congestion occurs. As compared to widely known TCP ECN (i.e. Mark-Tail ECN), our work can reduce packet drop rate and can increase throughput in TCP/IP network.

## กิตติกรรมประกาศ

โครงการวิจัยนี้ได้รับทุนสนับสนุนเงินวิจัยจากเงินรายได้ ประจำปีงบประมาณ 2547 คณะ  
เทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

คณะผู้วิจัย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
รายละเอียดของโครงการวิจัย	i
บทคัดย่อภาษาไทย	ii
บทคัดย่อภาษาอังกฤษ	iii
กิตติกรรมประกาศ	iv
สารบัญ	v
สารบัญภาพ	vii
สารบัญตาราง	viii
<b>บทที่ 1 บทนำ</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตงานวิจัย	2
1.4 ระเบียบวิจัย	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
<b>บทที่ 2 ทฤษฎีที่เกี่ยวข้อง</b>	<b>3</b>
2.1 กลไกควบคุมความคับคั่งทีซีพี (TCP Congestion Control)	3
2.2 การจัดการคิวแบบ Tail-Drop	3
2.3 การจัดการคิวแบบ Random Early Detection (RED)	4
2.4 การกระจายภาระงานเพื่อควบคุมความคับคั่ง	7
2.5 กลไกการแจ้งเตือนผู้ส่งเมื่อเกิดความคับคั่ง	7
<b>บทที่ 3 กลไกควบคุมการใช้บัฟเฟอร์ด้วยการตรวจจับ</b>	<b>9</b>
3.1 คำอธิบายกลไก ExRED	9
3.2 ผลเชิงวิเคราะห์	11
3.3 ผลการจำลองการทำงาน	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำไปใช้

## สารบัญ(ต่อ)

	หน้า
<b>บทที่ 4 กลไกควบคุมการใช้บัฟเฟอร์ด้วยการเปลี่ยนเส้นทางแพ็คเก็ต</b>	<b>15</b>
4.1 คำอธิบายกลไก LDM/ExRED	15
4.2 ผลการจำลองการทำงาน	16
<b>บทที่ 5 กลไกควบคุมการใช้บัฟเฟอร์ด้วยการแจ้งเตือนผู้ส่ง</b>	<b>19</b>
5.1 คำอธิบายกลไก Mark-Reverse ECN	19
5.2 ผลการจำลองการทำงาน	19
<b>บทที่ 6 สรุปผลงานวิจัย</b>	<b>22</b>
6.1 กลไก ExRED	22
6.2 กลไก LDM/RED และ LDM/ExRED	22
6.3 กลไก Mark-Reverse ECN	22
<b>บรรณานุกรม</b>	<b>23</b>
<b>ภาคผนวก</b>	
ภาคผนวก ก ผลงานส่งเผยแพร่ในงานประชุมวิชาการระดับนานาชาติ International Conference on Control, Automation and Systems (ICCAS 2004)	
ภาคผนวก ข ผลงานส่งเผยแพร่ในงานประชุมวิชาการระดับนานาชาติ International Symposium on Technology for Sustainability (ISTS 2011)	
ภาคผนวก ค ผลงานส่งเผยแพร่ในงานประชุมวิชาการระดับนานาชาติ International Conference on Information Technology and Electrical Engineering (ICITEE 2013)	
ภาคผนวก ง ผลงานส่งเผยแพร่ในงานประชุมวิชาการระดับนานาชาติ International Symposium on Technology for Sustainability (ISTS 2012)	
ภาคผนวก จ ผลงานส่งเผยแพร่ในรายงานทางเทคนิค IEICE Technical Report 2012	

### ประวัตินักวิจัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญภาพ

		หน้า
รูปที่ 2.1	กลไก TCP ECN หรือ Mark-Tail ECN	8
รูปที่ 3.1	แบบจำลองคิวของกลไก ExRED	9
รูปที่ 3.2	ความน่าจะเป็นในการดรอปแพ็คเก็ตของกลไก ExRED	10
รูปที่ 3.3	ความน่าจะเป็นในการดรอปแพ็คเก็ตของกลไก ExRED เมื่อพารามิเตอร์แตกต่างกัน	11
รูปที่ 3.4	กราฟเปรียบเทียบทรูพุทจากการวิเคราะห์ระหว่าง Tail-Drop, RED และ ExRED	11
รูปที่ 3.5	กราฟเปรียบเทียบทรูพุทจากการจำลองระหว่าง Tail-Drop, RED และ ExRED	12
รูปที่ 3.6	กราฟเปรียบเทียบโอกาสเกิดการดรอปแพ็คเก็ตแบบต่อเนื่องระหว่าง Tail-Drop, RED และ ExRED	13
รูปที่ 3.7	กราฟเปรียบเทียบค่าเฉลี่ยของจำนวนแพ็คเก็ตที่ถูกดรอปแบบต่อเนื่องระหว่าง Tail-Drop, RED และ ExRED	13
รูปที่ 3.8	กราฟเปรียบเทียบค่าเฉลี่ยของระยะเวลาในการรอคิวระหว่าง Tail-Drop, RED และ ExRED	14
รูปที่ 4.1	ฟังก์ชันการทำงานของกลไก LDM/ExRED	15
รูปที่ 4.2	อัลกอริทึมการคำนวณ โอกาสเปลี่ยนเส้นทาง	16
รูปที่ 4.3	โอกาสในการเกิดปัญหา Packet reordering	18
รูปที่ 5.1	กลไก TCP Mark-Reverse ECN หรือ Mark-Reverse ECN	19
รูปที่ 5.2	กราฟเปรียบเทียบอัตราการทำเครื่องหมายที่แพ็คเก็ตระหว่าง Mark-Tail ECN และ Mark-Reverse ECN	20
รูปที่ 5.3	กราฟเปรียบเทียบอัตราการดรอปแพ็คเก็ตระหว่าง Mark-Tail ECN และ Mark-Reverse ECN	20
รูปที่ 5.4	กราฟเปรียบเทียบทรูพุทระหว่าง Mark-Tail ECN และ Mark-Reverse ECN	21

## สารบัญตาราง

		หน้า
ตารางที่ 3.1	พารามิเตอร์การจำลองเหตุการณ์	12
ตารางที่ 4.1	แสดงโปรไฟล์ของกราฟฟิคจาก Trace ต่างๆ	17
ตารางที่ 4.2	ความน่าจะเป็นที่บัฟเฟอร์ในเส้นทางจะถูกใช้	17
ตารางที่ 4.3	ขนาดคิวสะสมสูงสุดที่เกิดขึ้นในแต่ละเส้นทาง	17



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

เกตเวย์เป็นทางผ่านร่วมของผู้ใช้ทุกคน เพื่อไปสู่เครือข่ายภายนอกหรืออินเทอร์เน็ต ประสิทธิภาพการส่งต่อแพ็คเก็ตของเกตเวย์ จึงกระทบคุณภาพการติดต่อสื่อสารของเครือข่ายอย่างหลีกเลี่ยงไม่ได้ หากเกิดความคับคั่งที่เกตเวย์ อาจทำให้เกิดปัญหาแพ็คเก็ตต้อคิวยาวจนเต็มบัฟเฟอร์ของเกตเวย์ และทำให้เกิดปัญหาตามมาหลายประการนอกเหนือจากการมีดีเลย์ยาวนาน เช่น ปัญหาแพ็คเก็ตสูญหายจากการเกิดการดรอปแพ็คเก็ตนั้นๆ โดยเกตเวย์เพราะเกิดบัฟเฟอร์ล้น และการเกิดแพ็คเก็ตต้อคิวอย่างต่อเนื่องซึ่งมีผลกระทบต่อคุณภาพการทำงานของเครือข่าย การดรอปของแพ็คเก็ตทำให้กลไกควบคุมความคับคั่งของเครือข่าย TCP/IP ชะลอและลดอัตราการส่งแพ็คเก็ต จึงทำให้เครือข่าย TCP/IP มีทรูพุตต่ำ และการดรอปต่อเนื่องทำให้เกิดการลดอัตราการส่งอย่างต่อเนื่องและเกิดกับหลายผู้ใช้ ซึ่งหากการดรอปต่อเนื่องยาวนานมาก ก็อาจทำให้ผู้ใช้ทุกคนมีอัตราการส่งต่ำ ผู้ใช้ทุกคนมีทรูพุตต่ำทั้งเครือข่ายจนกว่าสถานการณ์จะคลี่คลาย หลังจากนั้นผู้ใช้ทุกคนจะแข่งขันกันเพิ่มอัตราการส่ง จนกระทั่งเครือข่ายคับคั่งและเกิดเหตุการณ์ลดอัตราการส่งพร้อมกันอีกครั้งวนลูบไปเรื่อยๆ (TCP global synchronization) [1]

หน่วยงานอย่าง Internet engineering task force (IETF) จึงได้เสนอกลไกควบคุมการใช้บัฟเฟอร์ที่เกตเวย์แบบ Random early detection (RED) [2] เป็น Active queue management technique [1] ที่สามารถป้องกันการเกิดบัฟเฟอร์ล้นและการเกิดแพ็คเก็ตต้อคิวต่อเนื่อง โดยการดรอปแพ็คเก็ตแต่เนิ่นๆเพื่อลดขนาดคิวก่อนจะเต็มบัฟเฟอร์และเกิดบัฟเฟอร์ล้นแล้วลามไปถึงเกิดการดรอปต่อเนื่อง ถึงแม้ว่าการทำแบบนี้จะลดการเกิดปัญหาดังกล่าว แต่กลับทำให้มีโอกาสเกิดการดรอปของแพ็คเก็ตก่อนปัญหาดังกล่าวจะเกิดเสียอีก และหากมีการกำหนดพารามิเตอร์ที่ไม่พอดีกับสภาพเครือข่าย ยิ่งทำให้เกิดการแพ็คเก็ตต้อคิวสูง และนำไปสู่การดรอปแพ็คเก็ตแบบต่อเนื่องได้ [3] ซึ่งการดรอปแพ็คเก็ตแบบต่อเนื่องกระทบการทำงานของกลไกจัดการคับคั่งที่ได้กล่าวมาข้างต้น ส่งผลให้คุณภาพการสื่อสารของเครือข่ายลดลง

งานวิจัยนี้จึงได้ศึกษา และหาแนวทางการปรับปรุงกลไกควบคุมการใช้บัฟเฟอร์ที่เกตเวย์ โดยเริ่มจาก 1) ศึกษาการจัดการคิวในบัฟเฟอร์แบบ Tail-Drop และ RED แล้วหาแนวทางพัฒนาปรับปรุงกลไกจัดการคิวดังกล่าวเพื่อให้การดรอปแพ็คเก็ตเกิดเท่าที่จำเป็น 2) การจัดการคิวโดยการกระจายภาระงานไปเส้นทางอื่นเมื่อเกิดความคับคั่ง และ 3) การจัดการคิวโดยการแจ้งเตือนอุปกรณ์ผู้ใช้ให้ลดอัตราการส่งเมื่อเกิดความคับคั่ง โดยตลอดทั้งงานวิจัยนี้ ความคับคั่งจะวัดจากการที่บัฟเฟอร์ใกล้เต็ม

## 1.2 วัตถุประสงค์

- เพื่อศึกษาการทำงานและประสิทธิภาพของกลไกจัดการคิว
- เพื่อเสนอแนวทางปรับปรุงประสิทธิภาพกลไกจัดการคิว

## 1.3 ขอบเขตงานวิจัย

- งานวิจัยขั้นนี้ศึกษาโดยมีพื้นฐานมาจากเทคนิคการจัดการเกตเวย์แบบ Tail-Drop และ RED
- งานวิจัยนี้เน้นปัญหาที่มีผลกระทบต่อกลไก TCP/IP เช่น การเกิดแพ็คเก็ตดรอป การดรอปต่อเนื่อง และความยาวขนาดคิวซึ่งทำให้เกิดดีเลย์
- ผลสรุปของงานวิจัยจะแสดงผลเชิงวิเคราะห์ และผลจากการจำลองการทำงาน

## 1.4 ระเบียบวิจัย

- 1.ศึกษากลไกการจัดการคิวในบัฟเฟอร์ที่มีการใช้งานและเป็นมาตรฐาน
- 2.พัฒนาเทคนิคใหม่เพื่อปรับปรุงประสิทธิภาพการควบคุมการรับบัฟเฟอร์
- 3.ทดสอบการทำงานโดยการวิเคราะห์และจำลองเหตุการณ์

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

งานวิจัยนี้เสนอผลการศึกษาประสิทธิภาพของกลไกจัดการคิวในบัฟเฟอร์ และนำเสนอกลไกควบคุมการรับบัฟเฟอร์ที่มีการพัฒนาปรับปรุง โดยมีการเปรียบเทียบประสิทธิภาพกับกลไกที่มีอยู่เดิม

## บทที่ 2

### ทฤษฎีที่เกี่ยวข้อง

#### 2.1 กลไกควบคุมความคับคั่งทีซีพี (TCP Congestion Control)

กลไก TCP ในเครือข่าย TCP/IP ถูกพัฒนาในช่วงต้นของยุค 1980 โดยถูกออกแบบมาเพื่อสร้างการส่งข้อมูลที่มีความน่าเชื่อถือ มีการพัฒนา TCP เวอร์ชันต่างๆให้สามารถรับมือจัดการความคับคั่งในเครือข่าย [4]–[7] ในงานวิจัยนี้จะเน้นที่ TCP Reno ซึ่งมีการใช้งานอย่างแพร่หลาย เมื่อเกิดความคับคั่งในเครือข่าย โหนดต่างๆจะชะลอการส่งข้อมูล และหากปัญหายังดำเนินต่อไปจนวิกฤติจะเกิดแพ็คเก็ตตกรูป อัตราการส่งข้อมูลจะลดลงต่ำสุดทันที เมื่อสถานการณ์คลี่คลาย อัตราการส่งข้อมูลจะค่อยๆเพิ่มขึ้น กลไกควบคุมความคับคั่งจะใช้การเกิดแพ็คเก็ตสูญหาย (ซึ่งส่วนหนึ่งเกิดจากการถูกรับโดยเกตเวย์) เป็นสัญญาณที่แสดงถึงการเกิดความคับคั่ง เมื่อโหนดผู้ส่งตรวจพบการสูญหายของแพ็คเก็ตที่ส่งไปจะลดอัตราการส่ง ดังนั้นถ้าการจัดการบัฟเฟอร์ไม่มีประสิทธิภาพ ปล่อยให้มีการสะสมของคิวจนกระทั่งบัฟเฟอร์เต็มและเกิดการตกรูปแพ็คเก็ตอย่างต่อเนื่อง กลไกควบคุมความคับคั่งในเครือข่าย TCP จะปรับลดอัตราการส่งของผู้ใช้ทุกคนอย่างต่อเนื่อง ทำให้ทรูพุตต่ำ มีการใช้แบนด์วิดท์เครือข่ายสูงเกินไปและต่ำเกินไปเป็นระยะๆ [1]

#### 2.2 การจัดการคิวแบบ Tail-Drop

แพ็คเก็ตที่มาถึงบัฟเฟอร์ในขณะที่เกตเวย์ยังไม่พร้อมให้บริการ จะถูกเก็บพักไว้ที่บัฟเฟอร์ จนกว่าเกตเวย์จะอยู่ในสถานการณ์ที่สามารถส่งแพ็คเก็ตนั้นออกไปได้ ซึ่งแพ็คเก็ตที่เข้ามาใหม่ในระหว่างนั้นก็จะถูกเก็บต่อคิวไปเรื่อยๆ จนกระทั่งไม่เหลือพื้นที่บัฟเฟอร์ แพ็คเก็ตที่เข้ามาถัดจากนั้นก็จะถูกตกรูปทิ้ง

เทคนิคการจัดการคิวแบบ Tail-Drop มีรูปแบบการรองรับแพ็คเก็ตบนบัฟเฟอร์ในกรณีที่มีพื้นที่เหลืออยู่ในบัฟเฟอร์เท่านั้น ถ้าไม่เหลือพื้นที่บนบัฟเฟอร์ แพ็คเก็ตที่เข้ามาใหม่จะถูกตกรูปจนกว่าจะมีพื้นที่ว่างบนบัฟเฟอร์ ตามสมการที่ 2.1

$$q \leftarrow \begin{cases} q+1 & ; \quad q < BufferSize \\ q & ; \quad otherwise \end{cases} \quad (2.1)$$

โดยค่า  $q$  คือขนาดของคิวบนบัฟเฟอร์ และ  $BufferSize$  คือพื้นที่ของบัฟเฟอร์ ในการจำลองทดสอบประสิทธิภาพ ได้ทำการกำหนดให้มีแพ็คเก็ตที่เข้ามาในระบบแบบสุ่มด้วยรูปแบบการกระจายแบบ Poisson ด้วยอัตรา  $\lambda$  แทนค่าระยะเวลาในการเซอร์วิสของเซิร์ฟเวอร์ด้วยบัฟเฟอร์ที่มีขนาดเท่ากับ  $K$  และมีรูปแบบการกระจายแบบเอ็กโปเนนเชียลด้วยอัตรา  $\mu$  จากนั้นจะสามารถหาประสิทธิภาพของการทำงานของเทคนิคแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tail-Drop ได้ด้วย M/M/1/K queuing model ในส่วนสมการที่ 2.2 จะแสดงถึงความน่าจะเป็นในการรื้อปแพ็คเก็ตเกิดของเทคนิคแบบ Tail-Drop

$$P_{drop} = \begin{cases} \frac{\left(1 - \frac{\lambda}{\mu}\right) \left(\frac{\lambda}{\mu}\right)^K}{1 - \left(\frac{\lambda}{\mu}\right)^{K+1}} & ; \lambda \neq \mu \\ \frac{1}{K+1} & ; \lambda = \mu \end{cases} \quad (2.2)$$

ในส่วนของสมการที่ 2.3 และ 2.4 แสดงถึงโอกาสในการรื้อปของแพ็คเก็ตแบบต่อเนื่อง และค่าเฉลี่ยของการรื้อปแพ็คเก็ตแบบต่อเนื่อง

$$P_{drop}^{cons}(N > n) = \left(\frac{\lambda}{\mu}\right)^n \left(1 + \frac{\lambda}{\mu}\right) \quad (2.3)$$

$$E(N) = 1 + \frac{\lambda}{\mu} \quad (2.4)$$

ส่วนสมการที่ 2.5 จะแสดงถึงความล่าช้าที่เกิดขึ้นจากการรอคิว (queuing delay) โดยสามารถหาค่า  $E(Q)$  ได้จากสมการที่ 2.6

$$D = \frac{E(Q)}{(1 - P_{drop})\lambda} - \frac{1}{\mu} \quad (2.5)$$

$$E(Q) = \begin{cases} \frac{\frac{\lambda}{\mu} \left(1 - (K+1) \left(\frac{\lambda}{\mu}\right)^K + K \left(\frac{\lambda}{\mu}\right)^{K+1}\right)}{\left(1 - \left(\frac{\lambda}{\mu}\right)^{K+1}\right) \left(1 - \frac{\lambda}{\mu}\right)} & ; \lambda \neq \mu \\ \frac{K}{2} & \lambda = \mu \end{cases} \quad (2.6)$$

### 2.3 การจัดการคิวแบบ Random Early Detection (RED)

วิธีการจัดการเกตเวย์แบบอาร์อีดี (Random Early Detection : RED) [2] มีหลักการการทำงานโดยเกตเวย์จะตรวจสอบความหนาแน่นของแพ็คเก็ตในเครือข่าย จากการคำนวณขนาดของคิวโดยเฉลี่ยในขณะนั้น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากเครือข่ายมีการจราจรที่แออัด RED จะจัดการกับปัญหาดังกล่าวโดยการดรอปแพ็คเก็ตทิ้งเพื่อลดแพ็คเก็ตในระบบ (หรือทำเครื่องหมายที่แพ็คเก็ตนั้นๆ เพื่อให้หนอตทางผ่านต่างๆรู้ว่า แพ็คเก็ตใดถูกตัดสินให้ถูกรดรอป หากต้องมีการรดรอปจริงขึ้นมา) เมื่อขนาดของคิวโดยเฉลี่ยเกินค่าที่กำหนดจะมีโอกาสที่เกตเวย์จะทำการรดรอปหรือทำเครื่องหมายลงไปบนแพ็คเก็ตนั้นๆ ซึ่งการสุ่มการจัดการทั้งสองรูปแบบนั้นเป็นกลไกหนึ่งที่ได้จากการคำนวณขนาดของคิวแบบเฉลี่ย

RED จะพยายามรักษาขนาดของคิวให้มีขนาดเล็ก ยอมให้มีการส่งข้อมูลจำนวนมากในระยะเวลาสั้นๆ ในระบบ แต่เมื่อระบบมีความหนาแน่นสูงขึ้น จะมีโอกาสที่เกตเวย์ส่งสัญญาณระหว่างแพ็คเก็ตที่มีการเชื่อมต่อในขณะนั้นให้ลดขนาดของหน้าต่างลงเพื่อลดการใช้งานแบนด์วิธในระบบ RED ถูกออกแบบมาให้ใช้งานบนโปรโตคอลในชั้นทรานส์พอร์ต เช่น TCP เป็นต้น โดยการจัดการเกตเวย์แบบ RED ถูกเสนอเพื่อช่วยลดการเกิดการลดย่างต่อเนื่องของอัตราการส่งแพ็คเก็ตของแต่ละโหนด และป้องกันไม่ให้เกิดการล่มถล่มพร้อมๆกันในทุกโหนด

ในกระบวนการทำงานของ RED [2] นั้นจะแบ่งการทำงานออกเป็นสองส่วนคือ 1) การตัดสินใจว่าจะรดรอปแพ็คเก็ตที่เข้ามาในบัฟเฟอร์หรือไม่ และ 2) การหาขนาดของคิวเฉลี่ย (Average queue size) โดยคำนวณจากขนาดของคิวในขณะนั้นๆโดยใช้ Exponentially weighted moving average (EWMA) [2] ดังสมการที่ 2.7

$$\hat{q} \leftarrow \begin{cases} (1-w)\hat{q} + wq & ; q \neq 0 \\ (1-w)^x \hat{q} & ; otherwise \end{cases} \quad (2.7)$$

โดยที่  $w$  คือ ตัวแปรจากการถ่วงน้ำหนักแบบ EWMA ซึ่งเป็นค่าคงที่ซึ่งใช้ในการกำหนดน้ำหนักของคิว (Queue weight)

ในส่วนการตัดสินใจที่จะรดรอปแพ็คเก็ตที่กำลังเข้ามาในบัฟเฟอร์หรือไม่ กระบวนการทำงานของ RED ที่ใช้ในส่วนนี้แสดงให้เห็นถึงประสิทธิภาพในการพัฒนากระบวนการไหลของข้อมูล ซึ่งในกระบวนการทำงานของ RED มีค่าที่ถูกกำหนดเอาไว้สองค่าซึ่งมีส่วนสำคัญต่อการทำงานดังนี้

ค่าต่ำสุด  $k_l$  คือ ขนาดของคิวโดยเฉลี่ยซึ่งกำหนดไว้ว่าระบบจะไม่ทำการรดรอปแพ็คเก็ตเมื่อบัฟเฟอร์มีขนาดน้อยกว่า หรือเท่ากับ  $k_l$

ค่าสูงสุด  $k_h$  คือขนาดของคิวโดยเฉลี่ยเมื่อบัฟเฟอร์มีค่ามากกว่า  $k_h$  แพ็คเก็ตที่เข้ามาในบัฟเฟอร์ทั้งหมดจะถูกดรอป

เมื่อค่าคิวโดยเฉลี่ยมีค่าแตกต่างกันไปตามตัวแปรสองตัวข้างต้น แพ็คเก็ตจึงมีโอกาสถูกรดรอปตั้งแต่ 0 จนถึง  $max_p$  โดยที่  $max_p$  คือโอกาสที่จะเกิดการรดรอปมากที่สุด ดังนั้นโอกาสในการรดรอปของแพ็คเก็ตจึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถแสดงได้ด้วยสมการที่ 2.8 โดยที่ค่าพารามิเตอร์ที่เหมาะสมสำหรับสมการดังกล่าวได้ถูกกลั่นกรองมาจากงานวิจัย [2], [3]

$$p_d(\hat{q}) = \begin{cases} 0 & ; \hat{q} < k_l \\ \frac{\hat{q} - k_l}{k_h - k_l} \times \max_p & ; k_l \leq \hat{q} < k_h \\ 1 & ; \hat{q} \geq k_h \end{cases} \quad (2.8)$$

โดยการใช้คุณสมบัติ PASTA [8] ที่มีอัตราไหลเข้าของแพ็คเก็ต  $\lambda$  และอัตราการส่งแพ็คเก็ตออก (Service rate)  $\mu$  ประสิทธิภาพการทำงานของ RED จะสามารถแสดงให้เห็นได้จากโอกาสการรีอปแพ็คเก็ตโดยเฉลี่ยที่เกิดขึ้นบนบัฟเฟอร์ โอกาสที่มีการรีอปแพ็คเก็ตอย่างต่อเนื่อง และ ความล่าช้าจากการรอคิวที่ขนาดของบัฟเฟอร์เท่ากับ  $K$  ซึ่งอ้างอิงจาก Markovian model [9], [10] ดังสมการต่อไปนี้

$$P_{drop} = \sum_{\hat{q}=k_h}^K \pi(\hat{q}) + \sum_{\hat{q}=k_l}^{k_h-1} (p_d(\hat{q}) \times \pi(\hat{q})) \quad (2.9)$$

โอกาสการรีอปแพ็คเก็ตโดยเฉลี่ยสามารถประมาณค่าได้จากสมการที่ 2.9 โดยค่า  $\pi(\hat{q})$  คือโอกาสการกระจายแบบ Stationary ของขนาดคิวโดยเฉลี่ย และ  $P_d(\hat{q})$  คืออัตราการรีอปแพ็คเก็ต

$$\forall n \geq 0, \quad P_{drop}^{cons}(N > n) = \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) (p_d(\hat{q}))^{n+1}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) p_d(\hat{q})} \quad (2.10)$$

สมการที่ 2.10 แสดงถึงการรีอปแพ็คเก็ตแบบต่อเนื่องซึ่งแทนค่าด้วย  $P_{drop}^{cons}(N)$  โดย  $N$  เท่ากับจำนวนแพ็คเก็ตที่ถูกรีอป จากสมการที่ 2.10 จะสามารถประเมินค่าเฉลี่ยของจำนวนแพ็คเก็ตที่ถูกรีอปอย่างต่อเนื่องได้ด้วยสมการที่ 2.11

$$E(N) = 1 + \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) \frac{(p_d(\hat{q}))^2}{1 - p_d(\hat{q})}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) p_d(\hat{q})} \quad (2.11)$$

$$D = \frac{E(\hat{Q})}{(1 - P_{drop})\lambda} - \frac{1}{\mu} \quad (2.12)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$E(\hat{Q}) = \frac{\sum_{k=0}^K k \left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1-p_d(\hat{q}))}{\sum_{k=0}^K \left(\left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1-p_d(\hat{q}))\right)} \quad (2.13)$$

สมการที่ 2.12 แสดงถึงความล่าช้าจากการรอคิวบนบัฟเฟอร์โดยใช้ Little's Theorem มาคำนวณ โดย  $E(\hat{Q})$  เป็นค่าเฉลี่ยของขนาดคิว ซึ่งสามารถหาได้จากสมการที่ 2.13

## 2.4 การกระจายภาระงานเพื่อควบคุมความคับคั่ง

ภาระงานของเครือข่าย คือ การนำส่งแพ็คเก็ตจากต้นทางไปถึงปลายทาง ปริมาณภาระงานวัดได้จาก ปริมาณทราฟฟิกที่วิ่งผ่านเครือข่าย การมีภาระงานมากทำให้เกิดความคับคั่งและเกิดดีเลย์สูงในการส่งแพ็คเก็ต ถึงปลายทาง ซึ่งส่งผลกระทบต่อโลก TCP Congestion Control ถ้ามีเครือข่ายมีดีเลย์สูง จะทำให้อัตราการส่ง ข้อมูลต่ำ และทำให้ทรูพุตต่ำ ถ้าเส้นทางไปยังปลายทางนั้นมีมากกว่าหนึ่งเส้นทาง และมีการกระจายภาระงาน ส่งไปเส้นทางอื่น จะช่วยลดความคับคั่งของเส้นทางเดิมที่ใช้งาน ซึ่งช่วยลดดีเลย์ ทำให้โลก TCP Congestion Control ยอมให้แต่ละโหนดเพิ่มอัตราการส่ง ทำให้ดีเลย์และทรูพุตดีขึ้น การกระจายภาระงานที่ดีจะช่วยให้ เกิดการใช้บัฟเฟอร์แบบสมดุล ขนาดคิวในบัฟเฟอร์ของแต่ละเส้นทางจะมีขนาดน้อยพอกัน (ไม่มีเส้นทาง ใดที่มีขนาดคิวยาวมาก ในขณะที่อีกเส้นทางมีขนาดคิวสั้นมาก) แต่การกระจายภาระงานก็อาจนำมาซึ่งปัญหา Packet reordering ซึ่งทำให้ลำดับของแพ็คเก็ตที่ไปถึงปลายทางอาจสลับกันได้ ซึ่งกระทบต่อประสิทธิภาพ ของเครือข่าย TCP/IP ตามที่ได้มีการอภิปรายใน [11]–[13]

การเลือกเส้นทางใหม่เพื่อการกระจายภาระงานจากเส้นทางเดิม (Path selection หรือบางที่ใช้คำว่า Path scheduling) มีหลักการเบื้องต้นในการเลือกดังนี้ 1) เลือกเส้นทางแบบ Round robin 2) เลือกเส้นทาง ที่มีภาระงานต่ำที่สุด และ 3) เลือกเส้นทางตามเงื่อนไขพิเศษอื่นๆ เช่น เลือกตามลำดับ Priority เลือกเส้นทาง ตามนโยบาย (Policy-based routing) หรือเลือกเส้นทางที่มีคุณสมบัติสอดคล้องเงื่อนไขพิเศษบางประการ ฯลฯ ซึ่งความซับซ้อนของอัลกอริทึมในการเลือกก็จะเรียงน้อยไปหามากตามลำดับ รายละเอียดสามารถอ่านได้ จาก [11]

## 2.5 กลไกการแจ้งเตือนผู้ส่งเมื่อเกิดความคับคั่ง

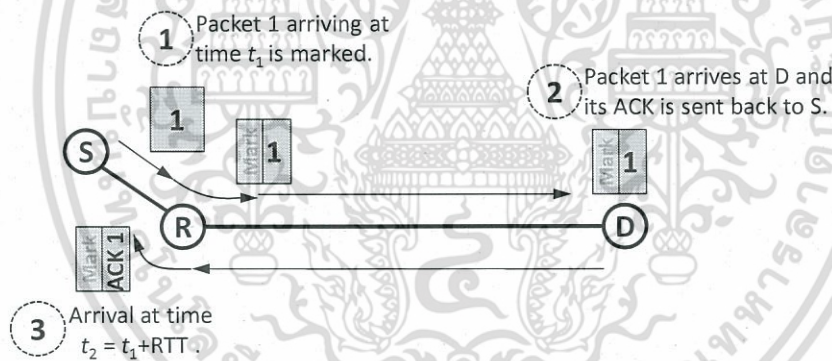
การจัดการคิวโดยการแจ้งเตือนผู้ส่งให้รับทราบสถานะความคับคั่งของเครือข่าย มี 2 แนวทาง คือ แนวทางการแจ้งเตือนโดยอ้อม (Implicit feedback) และการแจ้งเตือนโดยตรง (Explicit feedback) เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5.1 การแจ้งเตือนโดยนัยหรือโดยอ้อม (Implicit Feedback)

เมื่อโหนดผู้ส่งมีการส่งข้อมูลและผู้รับได้รับข้อมูลสำเร็จ ผู้รับจะส่งกลับ ACK มาให้ผู้ส่ง การแจ้งเตือนโดยอ้อมจะอาศัยลักษณะการกลับมาของแพ็คเก็ต ACK หากเครือข่ายไม่คับคั่ง แพ็คเก็ต ACK จะกลับมาได้เร็ว โหนดผู้ส่งก็จะสามารถส่งข้อมูลลำดับต่อไปได้เร็ว แต่หากเครือข่ายคับคั่ง แพ็คเก็ต ACK กลับมาช้า โหนดผู้ส่งก็ต้องชะลอการส่งให้ช้าลง และถ้าเครือข่ายคับคั่งมากจนกระทั่งแพ็คเก็ต ACK กลับมาช้ากว่าเวลา Timeout โหนดผู้ส่งก็จะลดอัตราการส่งลงสู่ระดับต่ำสุด

### 2.5.2 การแจ้งเตือนโดยตรง (Explicit Feedback)

เมื่อเกิดความคับคั่งในเครือข่าย เกตเวย์หรืออุปกรณ์เราเตอร์จะแจ้งสถานการณ์ดังกล่าวมาให้ผู้ส่ง ซึ่งมีได้หลายวิธี เช่น 1) การสร้างแพ็คเก็ตสำหรับแจ้งเตือนเช่น ICMP source quench [14] หรือ 2) การทำเครื่องหมายลงในแพ็คเก็ตข้อมูลเพื่อให้ผู้รับทำเครื่องหมายลงในแพ็คเก็ต ACK เมื่อได้รับแพ็คเก็ตข้อมูลและพบเครื่องหมายดังกล่าว ซึ่งแบบแรกจะทำให้เสียแบนด์วิดท์ในเครือข่ายไปกับการส่งแพ็คเก็ตแจ้งเตือน ในขณะที่แบบที่สองไม่มีการสร้างแพ็คเก็ตเพิ่ม มีเพียงการทำเครื่องหมายในส่วนหัวของ IP Packet เท่านั้น [15], [16] กลไกในแบบที่ 2 คือกลไก TCP Explicit Congestion Notification (TCP ECN) หรือ Mark-Tail ECN



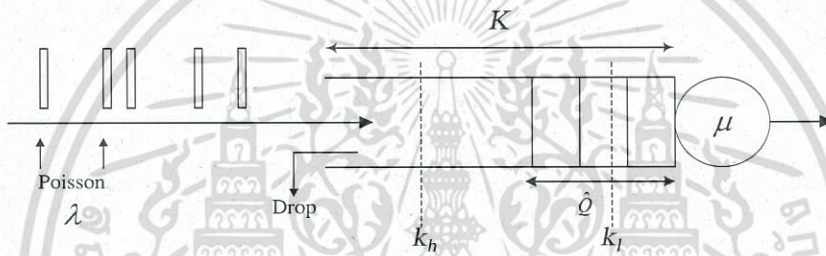
รูปที่ 2.1 กลไก TCP ECN หรือ Mark-Tail ECN

## บทที่ 3

### กลไกควบคุมการใช้บัฟเฟอร์ด้วยการดรอปแพ็คเก็ต

#### 3.1 คำอธิบายกลไก ExRED

จากการศึกษาเทคนิคต่างๆจากบทที่ 2 ผู้วิจัยสามารถสรุปได้ว่ากลไก Tail-Drop นั้นขาดประสิทธิภาพเนื่องจากส่งผลให้เกิดการดรอปแพ็คเก็ตอย่างต่อเนื่องทุกครั้งที่บัฟเฟอร์เต็ม ในส่วน RED นั้น ถึงแม้จะลดอัตราการเกิดการดรอปแพ็คเก็ตแบบต่อเนื่องได้ แต่ก็ยังไม่สามารถป้องกันกรณีที่ความหนาแน่นของเน็ตเวิร์คเพิ่มขึ้นอย่างรวดเร็ว และใช้งานบัฟเฟอร์ได้ไม่เต็มประสิทธิภาพ ดังนั้นผู้วิจัยจึงทำการคิดค้นเทคนิค ExRED [17] ซึ่งมีพื้นฐานมาจาก RED รายละเอียดตามเอกสารแนบในภาคผนวก ก

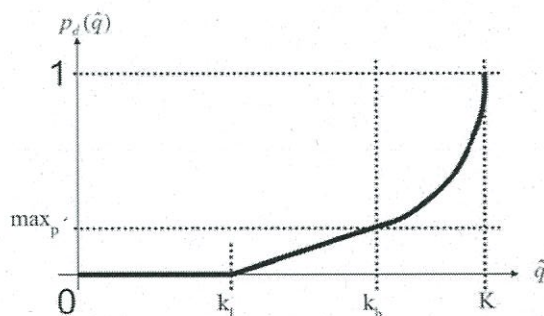


รูปที่ 3.1 แบบจำลองคิวของกลไก ExRED

รูปที่ 3.1 แสดงแบบจำลองคิวในบัฟเฟอร์ของเกตเวย์ที่มีขนาดบัฟเฟอร์  $K$  ส่วน  $\hat{Q}$  เป็นตัวแปรสุ่มขนาดคิวซึ่งมีค่าสุ่มแทนด้วย  $\hat{q}$  มีแบนด์วิธลิงก์ขาออกเท่ากับ  $\mu$  และมีอัตราการไหลเข้าของทราฟฟิกเท่ากับ  $\lambda$  สำหรับพารามิเตอร์ของกลไก ExRED คือ  $k_l$  และ  $k_h$  เป็น Threshold ของขนาดคิวในบัฟเฟอร์ ถ้าขนาดคิวต่ำกว่า  $k_l$  แพ็คเก็ตที่เข้ามาในขณะนั้นจะผ่านเข้ามาได้โดยไม่ถูกดรอป แต่ถ้าขนาดคิวสูงกว่า  $k_h$  แพ็คเก็ตที่เข้ามานั้นจะถูกดรอปด้วยอัตราการดรอปที่เพิ่มขึ้นอย่างรวดเร็ว

จากสมการ 2.8–2.13 จะสังเกตได้ว่า  $P_d(\hat{q})$  นั้นส่งผลต่อประสิทธิภาพของการทำงานของ RED เมื่อ  $\hat{q} \geq k_h$  แพ็คเก็ตทั้งหมดจะถูกดรอปอย่างต่อเนื่อง ผู้วิจัยจึงปรับฟังก์ชันการดรอปดังกล่าวเพื่อลดความน่าจะเป็นที่จะเกิดการดรอปแพ็คเก็ตในกรณีที่  $\hat{q} \geq k_h$  ขะลอกโดยให้อัตราการดรอปนั้นเพิ่มขึ้นอย่างช้าๆในตอนต้นแต่จะเพิ่มขึ้นในอัตราที่สูงขึ้นเมื่อขนาดของคิวมีขนาดใกล้เคียงกับขนาดของบัฟเฟอร์ดังรูปที่ 3.2

ฟังก์ชัน  $p_d$  ถูกแก้ไขให้อยู่ในรูปแบบ Polynomial function ของ  $\hat{q}$  และการดรอปแพ็คเก็ตจะเกิดขึ้นได้ต่อเมื่อตรงตามข้อกำหนดดังสมการที่ 3.1 และรูปแบบในการดรอปแพ็คเก็ตจะเปลี่ยนรูปแบบตามชุดสมการที่ 3.2



รูปที่ 3.2 ความน่าจะเป็นในการดริอปแพ็คเก็ตเกิดของกลไก ExRED

$$\left. \begin{aligned} p_d(\hat{q} = k_h) &= \max_p \\ p_d(\hat{q} = K) &= 1 \end{aligned} \right\} (3.1)$$

$$\left. \begin{aligned} \frac{d}{d\hat{q}} p_d(\hat{q}) \Big|_{\hat{q}=k_h} &= \frac{\max_p}{k_h - k_l} \\ p_d(\hat{q}) &= \begin{cases} 0 & ; \hat{q} < k_l \\ \frac{\hat{q} - k_l}{k_h - k_l} \times \max_p & ; k_l \leq \hat{q} < k_h \\ a_2 \hat{q}^2 + a_1 \hat{q} + a_0 & ; k_h \leq \hat{q} \leq K \end{cases} \end{aligned} \right\} (3.2)$$

where

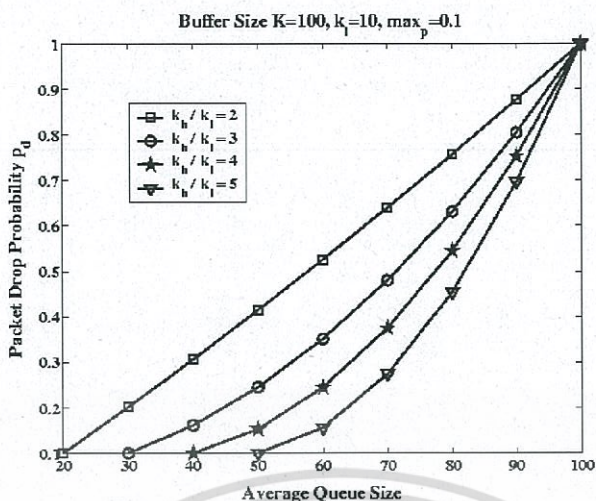
$$\left. \begin{aligned} a_2 &= \frac{(k_h - k_l) - (K - k_l) \max_p}{(k_h - k_l)(K - k_h)^2} \\ a_1 &= \frac{(k_h^2 + K^2 - 2k_h k_l) \max_p - 2k_h^2 + 2k_h k_l}{(k_h - k_l)(K - k_h)^2} \\ a_0 &= \frac{k_h^3 - k_h^2 k_l - (k_h^2 + 2k_h k_l + k_l K) K \max_p}{(k_h - k_l)(K - k_h)^2} \end{aligned} \right\} (3.2)$$

จากสมการที่ 3.2 โอกาสในการดริอปแพ็คเก็ตจะเพิ่มขึ้นด้วยอัตราที่สูงขึ้นเมื่อพื้นที่ในบัฟเฟอร์มีขนาดเล็กมากๆ โดยการตั้งค่าในบัฟเฟอร์จะต้องมีข้อกำหนดดังนี้

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

โดยในงานวิจัย [2], [3] นั้น ค่าพารามิเตอร์ต่างๆ มีการกำหนดให้ตรงตามเงื่อนไขดังกล่าวอยู่แล้ว โดยเมื่อนำค่าขนาดของคิวโดยเฉลี่ย และ โอกาสในการดริอปของแพ็คเก็ต มาพล็อตกราฟโดยมีเงื่อนไขว่า ขนาดของคิวโดยเฉลี่ยมีขนาดเพิ่มขึ้นเรื่อยๆ จะได้ผลลัพธ์ดังรูป 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



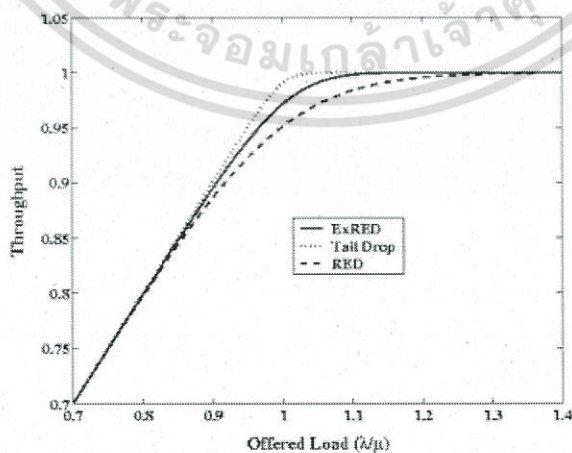
รูปที่ 3.3 ความน่าจะเป็นในการดรอปแพ็คเก็ตของกลไก ExRED เมื่อพารามิเตอร์แตกต่างกัน

จากรูปสามารถสังเกตได้ว่า เมื่อขนาดของคิวโดยเฉลี่ยเพิ่มขึ้นเมื่อเทียบกับขนาดของคิวน้อยที่สุดที่ไม่ก่อให้เกิดการดรอปแพ็คเก็ต ( $k_l$ ) จะส่งผลให้อัตราการดรอปของแพ็คเก็ตลดลง

### 3.2 ผลเชิงวิเคราะห์

เมื่อเปรียบเทียบฟังก์ชันในการดรอปแพ็คเก็ตของ RED และ ExRED โดยฟังก์ชันในการดรอปของ RED จะใช้สมการที่ 2.9 และ ExRED ใช้สมการที่ 3.3 จะสามารถพล็อตกราฟทรูพุท (throughput) เปรียบเทียบของทั้งสองกลไกได้ดังรูปที่ 3.4

$$P_{drop} = \sum_{\hat{q}=k_h}^K \pi(\hat{q}) \times (a_2 \hat{q}^2 + a_1 \hat{q} + a_0) + \sum_{\hat{q}=k_l}^{k_h-1} p_d(\hat{q}) \times \pi(\hat{q}) \tag{3.3}$$



รูปที่ 3.4 กราฟเปรียบเทียบทรูพุทจากการวิเคราะห์ระหว่าง Tail-Drop, RED และ ExRED

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.4 แสดงให้เห็นว่าการดริอปแพ็คเก็ตของกลไก ExRED น้อยกว่า RED ส่วนการดริอปแบบต่อเนื่อง สามารถอธิบายได้จากสมการที่ 3.4

$$\forall n, 1 < n < K, P_{drop}^{cons}(N_{ExRED} > n) < P_{drop}^{cons}(N_{RED} > n) \quad (3.4)$$

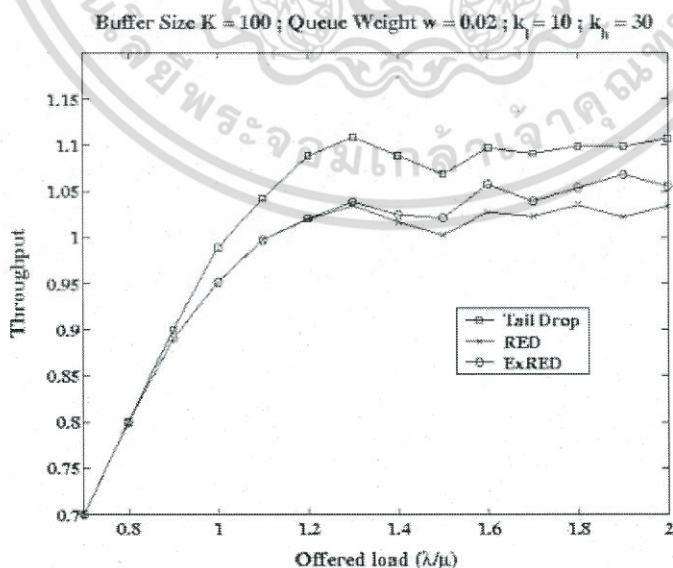
### 3.3 ผลการจำลองการทำงาน

การทดสอบการทำงานของกลไกโดยเปรียบเทียบทั้งกลไก Tail-Drop, RED และ ExRED โดยให้การไหลของแพ็คเก็ตใช้การกระจายแบบ Poisson ด้วยอัตรา  $\lambda$  ให้บัฟเฟอร์มีขนาดเท่ากับ  $K$  ที่มีการกระจายแบบ Exponential ด้วยอัตรา  $\mu$  โดยใช้พารามิเตอร์ต่างๆดังต่อไปนี้

ตารางที่ 3.1 พารามิเตอร์การจำลองเหตุการณ์

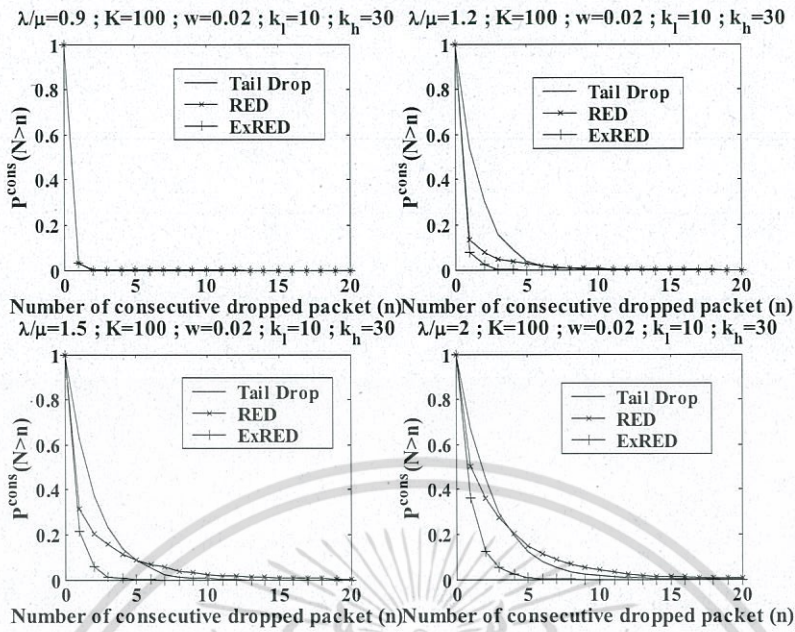
$\mu$	100 packets/second
$K$	100 packets
$w$	0.02
$k_l$	10
$k_h$	30
$max_p$	0.1

การกำหนดค่า  $k_l$  ต้องกำหนดให้น้อยกว่า 4 เท่าของ  $K$  และกำหนดให้  $k_h$  มากกว่าหรือเท่ากับสองเท่าของ  $k_l$  ตามงานวิจัย [2] ในส่วนของ  $max_p$  ทำการกำหนดค่าตามคำแนะนำในงานวิจัย [3] โดยผลลัพธ์การทดลองจะแสดงตามรูปภาพที่ 3.5–3.8

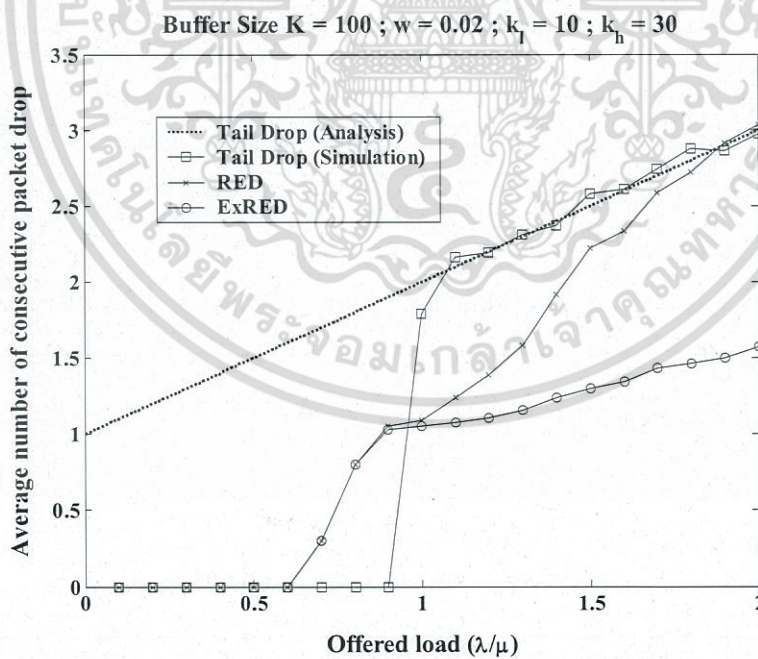


รูปที่ 3.5 กราฟเปรียบเทียบทราฟฟิคจากการจำลองระหว่าง Tail-Drop, RED และ ExRED

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

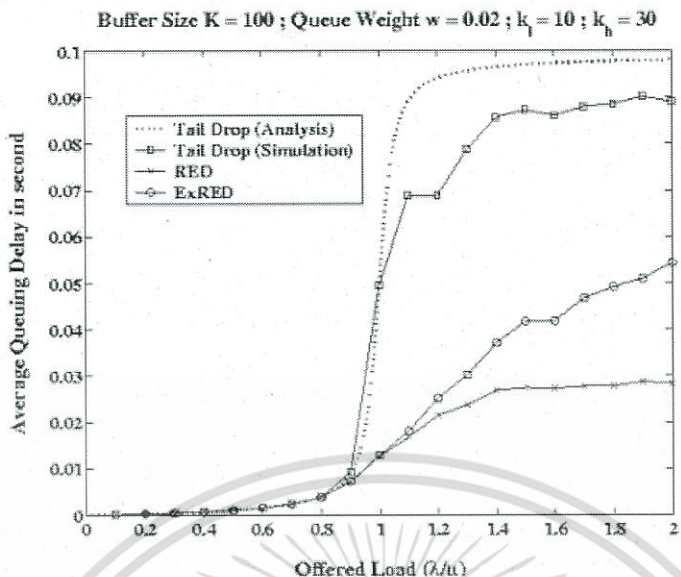


รูปที่ 3.6 กราฟเปรียบเทียบโอกาสเกิดการรื้อแพ็คเกิดแบบต่อเนื่องระหว่าง Tail-Drop, RED และ ExRED



รูปที่ 3.7 กราฟเปรียบเทียบค่าเฉลี่ยของจำนวนแพ็คเกิดที่ถูกรื้อแบบต่อเนื่องระหว่าง Tail-Drop, RED และ ExRED

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 กราฟเปรียบเทียบค่าเฉลี่ยของระยะเวลาในการรอคิว ระหว่าง Tail-Drop, RED และ ExRED

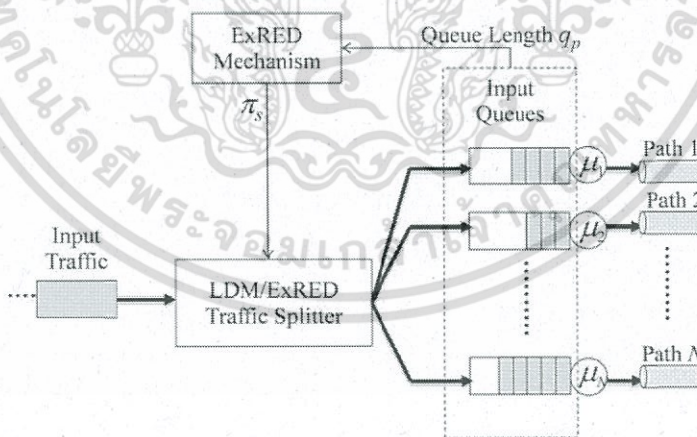
จากรูปที่ 3.5 แสดงให้เห็นว่ากลไกแบบ ExRED มีทรูพุที่สูงกว่า RED และเมื่ออัตราส่วนระหว่าง  $\lambda/\mu$  มีค่ามากขึ้นจะแสดงให้เห็นได้ชัดว่ากลไก ExRED มีโอกาสที่จะดริอปแพ็คเก็ตแบบต่อเนื่องลดลงมากกว่ากลไกอื่นๆ จากรูปที่ 3.6-3.7 แสดงแพ็คเก็ตที่ถูกดริอปอย่างต่อเนื่องโดยเฉลี่ย กลไก ExRED มีจำนวนน้อยที่สุดเมื่อเทียบกับกลไกอื่นๆทั้งหมด ในขณะที่ค่าเฉลี่ยในการรอคิวของ ExRED มีค่าสูงขึ้นเพียงเล็กน้อยเมื่อเทียบกับกลไก RED ดังแสดงในรูปที่ 3.8

## บทที่ 4

### กลไกควบคุมการใช้บัฟเฟอร์ด้วยการเปลี่ยนเส้นทางแพ็คเก็ต

บทนี้จะนำเสนอกลไกการควบคุมการใช้บัฟเฟอร์ โดยการกระจายภาระงานบนเครือข่ายที่มีเส้นทางไปปลายทางมากกว่าหนึ่งเส้นทาง เมื่อเส้นทางที่ใช้อยู่เกิดความคับคั่ง ภาระงานจะถูกย้ายไปอีกเส้นทางหนึ่งที่มีภาระงานน้อยกว่า ทำให้บัฟเฟอร์แต่ละเส้นทางถูกใช้อย่างสมดุลเพื่อลดขนาดคิวของแพ็คเก็ตที่อยู่ในบัฟเฟอร์ของแต่ละเส้นทาง ในงานวิจัยนี้จะเลือกการกระจายที่เลือกเส้นทางจากการพิจารณาภาระงาน ซึ่งมี 2 รูปแบบ Least-loaded first (LFF) [18]–[20] และ Load distribution over multipath (LDM) [21] ทั้งสองเป็นรูปแบบการกระจายแบบ Adaptive load distribution ซึ่งเป็นที่รู้จักอย่างกว้างขวาง การเลือกเส้นทางจะพิจารณาจากความคับคั่งที่สะท้อนจากขนาดคิวในบัฟเฟอร์ของลิงก์เส้นทาง เส้นทางใดมีขนาดคิวยาว หมายถึงเส้นทางนั้นมีความคับคั่งมาก แต่เนื่องจาก LFF มีข้อเสียเรื่องการทำให้เกิดปัญหา Packet reordering ตามที่ศึกษาใน [11] งานวิจัยนี้จึงเลือก LDM เป็นรูปแบบการกระจายภาระงานที่ใช้ RED และ ExRED มาควบคุมการตัดสินใจการเปลี่ยนเส้นทาง เรียกว่ากลไก LDM/RED[22] และ LDM/ExRED [23] ซึ่งในบทนี้จะอธิบายกลไก LDM/ExRED ซึ่งเป็นผลงานที่วิจัยและพัฒนาขึ้นมาล่าสุด (สามารถอ่านรายละเอียด LDM/RED จากภาคผนวก ข และ LDM/ExRED จากภาคผนวก ค)

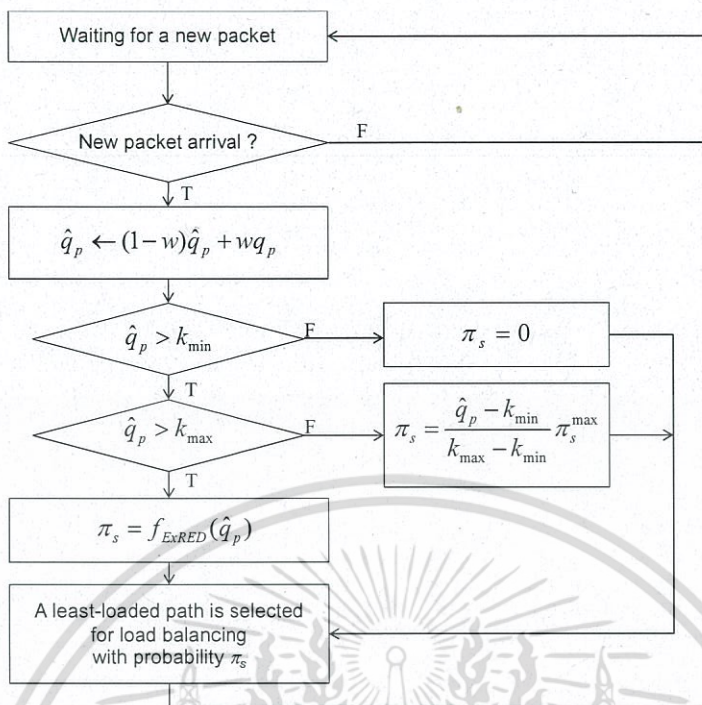
#### 4.1 คำอธิบายกลไก LDM/ExRED



รูปที่ 4.1 ฟังก์ชันการทำงานของกลไก LDM/ExRED

จากรูปที่ 4.1 กลไก ExRED จะตรวจสอบสถานะความคับคั่งของเครือข่าย และเมื่อพบว่าความคับคั่งมากเกินไปในระดับที่กำหนด จะสั่งให้กลไก Traffic splitter เปลี่ยนเส้นทางสำหรับการส่งแพ็คเก็ต (ที่มาถึงเกตเวย์ในขณะนั้น) อัลกอริทึมการทำงานแสดงดังรูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 อัลกอริทึมการคำนวณโอกาสเปลี่ยนเส้นทาง

เมื่อมีแพ็คเก็ตมาถึงเกตเวย์ ความน่าจะเป็นที่เส้นทางจะถูกเปลี่ยน  $\pi_s$  ที่คำนวณจากสมการที่ 3.2 จะถูกนำมาใช้ในการตัดสินใจว่าแพ็คเก็ตดังกล่าวควรถูกส่งออกไปเส้นทางที่ใช้อยู่เดิม หรือควรเปลี่ยนไปเส้นทางที่มีภาระงานต่ำสุดในขณะนั้น ค่าความน่าจะเป็นดังกล่าวเป็นไปตามรูปที่ 3.2 นั่นคือ ถ้าเครือข่ายยังคับคั่งไม่มาก เส้นทางเดิมจะยังถูกใช้ต่อไปเพื่อหลีกเลี่ยงการเกิดปัญหา Packet reordering แต่ถ้าความคับคั่งเพิ่มขึ้นโอกาสที่เส้นทางจะถูกเปลี่ยนจะสูงขึ้นอย่างรวดเร็วเพื่อป้องกันการเกิดปัญหาบัฟเฟอร์ล้น

จากการที่กลไก RED หรือ ExRED มีความสามารถในการตรวจสอบสถานะความคับคั่งในเครือข่ายได้ และทำการดรอปแพ็คเก็ตเพื่อบรรเทาความคับคั่ง ซึ่งผลการวิเคราะห์ในบทที่ 3 ได้แสดงให้เห็นประสิทธิภาพดังกล่าว ดังนั้นการใช้กลไกเหล่านี้มาตรวจจับความคับคั่งและตัดสินใจว่าควรเปลี่ยนเส้นทางหรือไม่ จึงมีความเป็นไปได้ว่าการตัดสินใจดังกล่าวจะเหมาะสมกับสถานการณ์ความคับคั่ง เพื่อให้การเปลี่ยนเส้นทางควรเกิดขึ้นเฉพาะในสภาวะการณ์ที่จำเป็นเท่านั้น

## 4.2 ผลการจำลองการทำงาน

การประเมินประสิทธิภาพจะใช้การจำลองจากเหตุการณ์ที่เครือข่ายมี 3 เส้นทางไปปลายทางเดียวกัน โดยมีแบนด์วิธ 1, 4, และ 7 Mbps. ตามลำดับ ซึ่งมีผลรวมแบนด์วิธเท่ากับ 12 Mbps. และมีการจำลอง 3 เหตุการณ์ โดยสมมติว่าแต่ละเหตุการณ์ทดสอบมีทราฟฟิกที่ถูกสร้างจาก Data set ที่แตกต่างกัน คือ DSet1, DSet2 และ DSet3 ซึ่งมีลักษณะทราฟฟิกที่แตกต่างกันดังตารางที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 แสดงโปรไฟล์ของกราฟฟิกจาก Trace ต่างๆ

Trace ID	# Packets $\times 10^6$	Traffic Rate (Mbps.)			# Different Flows	Flow Size (Packets)		Flow Rate (Flows/Second)		
		Mean	Min.	Max.		Mean	CV	Mean	Min.	Max.
DSet 1	2.66	5.91	2.07	13.65	5865	453.87	7.52	137.89	77	204
DSet 2	2.87	6.38	0.46	12.24	12903	222.71	5.98	175.32	44	247
DSet 3	3.86	8.58	1.86	15.45	12710	303.88	7.11	184.50	90	269

การทดสอบโดยกราฟฟิก DSet1, DSet2, และ DSet3 คือ การทดสอบการทำงานในสภาวะที่เครือข่ายมีความคับคั่งน้อย, ปานกลาง (แต่มี Peak load บ้างเป็นประปราย แต่ยังไม่เกินผลรวมแบนด์วิธของทั้งสามเส้นทาง), และสูง (Peak load สูงเกินผลรวมแบนด์วิธของทั้งสามเส้นทาง) ตามลำดับ [11] ซึ่งผลการทดลองเป็นดังนี้

ตารางที่ 4.2 ความน่าจะเป็นที่บัฟเฟอร์ในเส้นทางจะถูกใช้

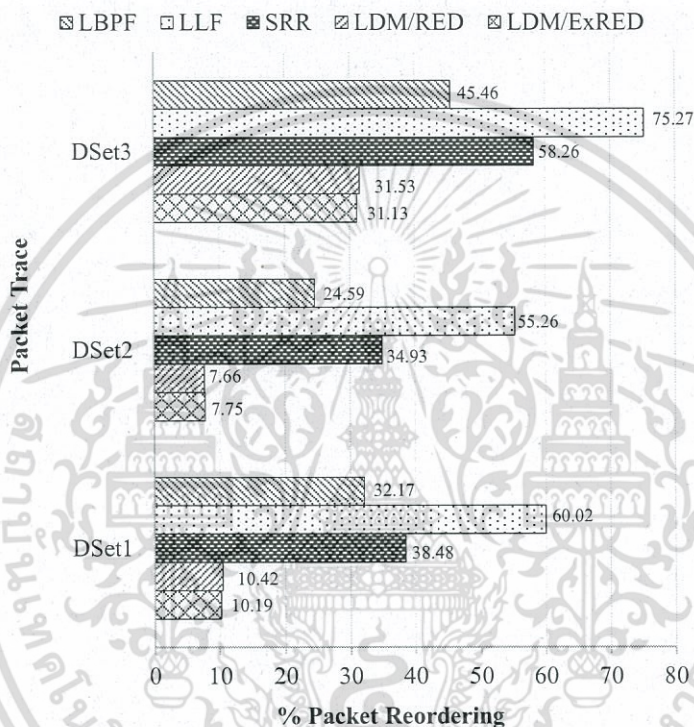
Load Distribution Model	Dset 1			Dset 2			Dset 3		
	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3
LDM/ExRED	0.71	0.66	0.65	0.73	0.68	0.65	0.86	0.82	0.80
LDM/RED	0.70	0.66	0.65	0.73	0.67	0.65	0.87	0.82	0.80
LDM	0.86	0.71	0.59	0.89	0.73	0.58	0.95	0.85	0.77
LLF	0.77	0.53	0.47	0.80	0.55	0.48	0.91	0.74	0.70
SRR	0.35	0.66	0.66	0.37	0.68	0.68	0.61	0.81	0.82

ตารางที่ 4.3 ขนาดคิวสะสมสูงสุดที่เกิดขึ้นในแต่ละเส้นทาง

Load Distribution Model	Dset 1			Dset 2			Dset 3		
	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3
LDM/ExRED	83	81	82	85	83	82	319	318	317
LDM/RED	100	98	100	104	97	102	376	375	375
LDM	1685	422	180	3845	378	76	5976	2780	522
LLF	92	92	93	83	83	83	293	293	293
SRR	43	86	185	33	88	165	110	350	557

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 และ 4.3 แสดงผลเปรียบเทียบการใช้บัฟเฟอร์ของแต่ละเส้นทางเมื่อใช้กลไกการจัดการคิวในบัฟเฟอร์ที่มีการกระจายภาระงานในแบบต่างๆ จะเห็นได้ว่ากลไกจัดการคิวที่มีการประยุกต์ RED และ ExRED สามารถบรรลุผลสำเร็จในการสร้างความสมดุลของการใช้บัฟเฟอร์แต่ละเส้นทาง ทำให้ไม่เกิดทราฟฟิกค้ำคั่งบนเส้นทางใดเส้นทางหนึ่ง แต่เมื่อเปรียบเทียบกันแล้ว กลไก LDM/ExRED มีขนาดคิวสูงสุดที่น้อยที่สุดซึ่งหมายความว่าดีเลย์ก็จะน้อยกว่าด้วย



รูปที่ 4.3 โอกาสในการเกิดปัญหา Packet reordering

รูปที่ 4.3 แสดงให้เห็นว่าการควบคุมการใช้บัฟเฟอร์โดยการกระจายภาระงานด้วยกลไก LDM/ExRED และ LDM/RED สามารถป้องกันการเกิดปัญหา Packet reordering ได้ ซึ่งปัญหาดังกล่าวทำให้เกิดผลกระทบต่อการทำงานของเครือข่าย TCP/IP [11], [12], [13]

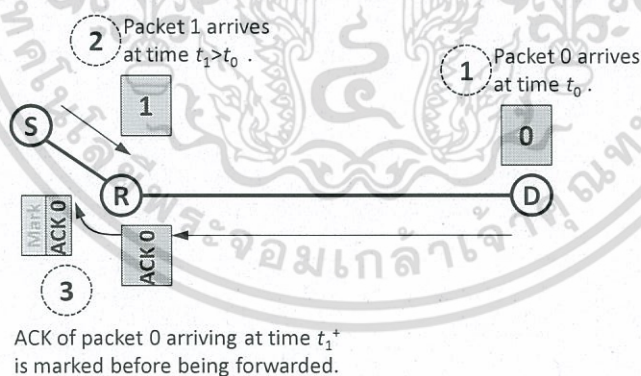
## บทที่ 5

### กลไกควบคุมการใช้บัฟเฟอร์ด้วยการแจ้งเตือนผู้ส่ง

เมื่อเครือข่ายคับคั่งทำให้บัฟเฟอร์ของเกตเวย์มีขนาดคิวเพิ่มขึ้น การควบคุมที่ทำได้อย่างหนึ่งก็คือการแจ้งเตือนผู้ส่งให้ปรับลดการส่ง ความท้าทายอยู่ที่ความรวดเร็วในการได้รับการแจ้งเตือน เนื่องจากกลไก TCP ECN หรือ TCP Mark-Tail ECN ต้องรอนาน้อย 1 Round trip time (RTT) โหนดผู้ส่งจึงจะรับการแจ้งเตือน ทำให้มีการปรับตัวตอบสนองได้ช้า ทั้งการปรับลด (เมื่อคับคั่ง) และปรับเพิ่ม (เมื่อคลี่คลายจากความคับคั่ง) กลไก Mark-Reverse ECN [24], [25] จึงถูกนำเสนอเพื่อทำให้โหนดผู้ส่งรับรู้สถานการณ์เครือข่ายได้เร็ว ทำให้สามารถปรับตัวได้สอดคล้องกับสถานการณ์อย่างรวดเร็ว

#### 5.1 คำอธิบายกลไก Mark-Reverse ECN

กลไก Mark-Reverse ECN พัฒนามาจาก TCP Mark-Tail ECN (รูป 2.1) โดยใช้กลไก RED ตรวจสอบความคับคั่งซึ่งสะท้อนด้วยขนาดคิวในบัฟเฟอร์ที่เกตเวย์ และแจ้งเตือนโหนด (ที่ส่งแพ็คเก็ตออกมา) ด้วยความน่าจะเป็นที่สอดคล้องกับขนาดคิว โดยการทำเครื่องหมายที่แพ็คเก็ต ACK ที่ถูกส่งกลับจากปลายทางและมาถึงเกตเวย์ในขณะนั้น (แทนที่จะทำเครื่องหมายที่แพ็คเก็ตข้อมูลเหมือน Mark-Tail ECN) ก่อนที่เกตเวย์จะส่งต่อแพ็คเก็ต ACK ดังกล่าวไปให้โหนดต้นทาง (ที่เป็นปลายทางของแพ็คเก็ต ACK) ดังรูปที่ 5.2 เพื่อให้โหนดผู้ส่งรับทราบสถานการณ์ความคับคั่งได้เร็ว (สามารถอ่านรายละเอียดจากภาคผนวก ง และ ภาคผนวก จ)

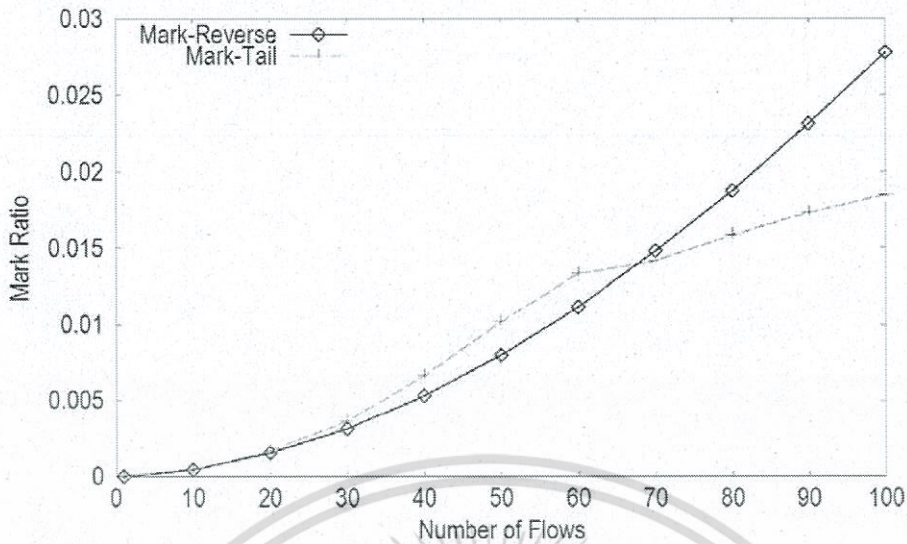


รูปที่ 5.1 กลไก TCP Mark-Reverse ECN หรือ Mark-Reverse ECN

#### 5.2 ผลการจำลองการทำงาน

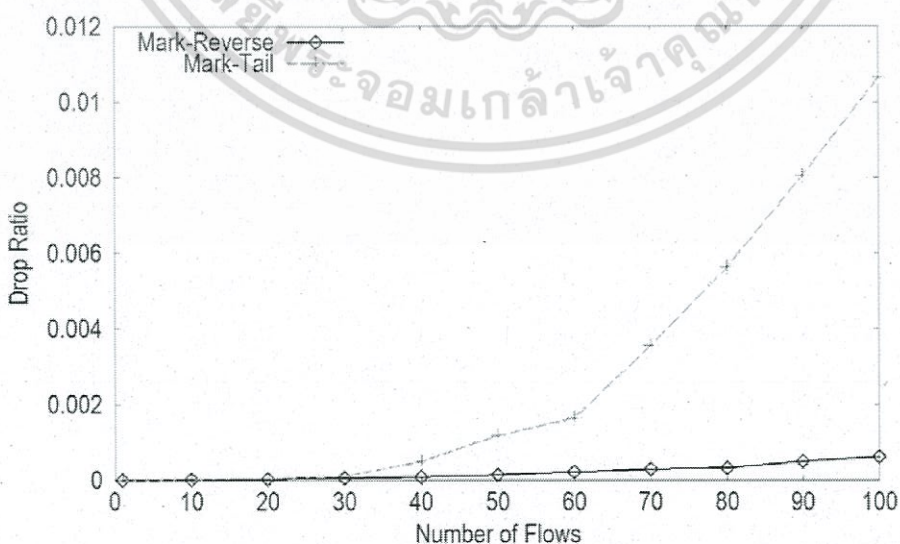
ในการสร้างแบบจำลองเพื่อประเมินประสิทธิภาพ พารามิเตอร์ต่างๆอธิบายใน [24], [25] การทดสอบนี้ควบคุมความคับคั่งด้วยจำนวน Flow ของแพ็คเก็ต แล้วสังเกตอัตราการเกิดทำเครื่องหมายที่แพ็คเก็ต อัตราการเกิดแพ็คเก็ตต้อป และทรูพุท เมื่อจำนวน Flow เพิ่มขึ้นจนทำให้เครือข่ายคับคั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 ผลการเปรียบเทียบอัตราการทำเครื่องหมายที่แพ็คเก็ต ระหว่าง Mark-Tail ECN และ Mark-Reverse ECN

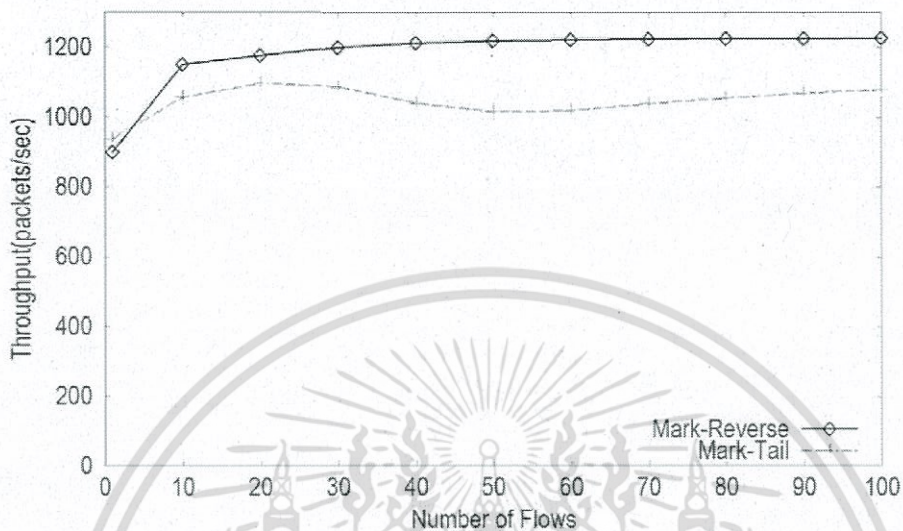
เนื่องจาก Mark-Reverse ECN ใช้แพ็คเก็ต ACK ในการฝากเครื่องหมายแสดงการเกิดความคับคั่ง จึงไม่ต้องรอรเวลาอย่างน้อย 1 RTT ที่ผู้ส่งจะได้รับการแจ้งดังกล่าว เมื่อเครือข่ายคับคั่งมากขึ้นจะทำให้ RTT เพิ่มขึ้น รูปที่ 5.3 แสดงให้เห็นว่าการเพิ่มขึ้นของ RTT ไม่มีผลทำให้เกิดความล่าช้าในการรับแจ้งดังกล่าว ดังนั้นกราฟ Mark-Reverse ECN จึงมีอัตราการทำเครื่องหมายที่แพ็คเก็ตสูงสุดต่อความคับคั่งที่เกิดขึ้น ซึ่งแตกต่างจาก Mark-Tail ECN ที่การเพิ่มขึ้นของ RTT ทำให้การเพิ่มของอัตราแจ้งเตือนลดลง (ทั้งที่ควรเพิ่มขึ้น) การมีอัตราแจ้งเตือนที่ไม่สอดคล้องกับสถานการณ์ ทำให้ผู้ส่งปรับตัวไม่ทันและทำให้เกิดเวลาดับคั่งและเกิดบัฟเฟอร์ล้น กลไก Mark-Reverse ECN จึงมีอัตราการตรวจจับแพ็คเก็ตต่ำกว่า Mark-Tail ECN ดังรูปที่ 5.4



รูปที่ 5.3 ผลการเปรียบเทียบอัตราการตรวจจับแพ็คเก็ตระหว่าง Mark-Tail ECN และ Mark-Reverse ECN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากโหนดผู้ส่งสามารถปรับตัวได้สอดคล้องกับสถานการณ์ความคับคั่ง เมื่อเครือข่ายว่างจะทำให้มีอัตราการส่งสูง และเมื่อเครือข่ายคับคั่งจะทำการปรับลดอัตราการส่งอย่างสอดคล้องเหมาะสม ทำให้เกิดอัตราการดรอปต่ำ จึงทำให้เครือข่าย TCP/IP ที่ใช้ Mark-Reverse ECN มีทรูพุตสูงดังรูปที่ 5.5



รูปที่ 5.4 ผลการเปรียบเทียบทรูพุตระหว่าง Mark-Tail ECN และ Mark-Reverse ECN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 6

### สรุปผลงานวิจัย

#### 6.1 กลไก ExRED

กลไก ExRED [17] ทำให้การดรอปแพ็คเก็ตเกิดความยืดหยุ่นมากขึ้น ถ้าพื้นที่ในบัฟเฟอร์ยังเหลือ จะปล่อยผ่านแพ็คเก็ตเหมือนกลไก Tail-Drop แต่เมื่อขนาดคิวเพิ่มขึ้น อัตราการดรอปก็จะเพิ่มช้าๆในช่วงที่พื้นที่บัฟเฟอร์ยังเหลือมาก หากขนาดคิวเพิ่มขึ้นต่อเนื่องจนกระทั่งพื้นที่ในบัฟเฟอร์เหลือน้อย กลไก ExRED จะเพิ่มอัตราการดรอปที่สูงกว่ากลไก RED เพื่อตอบสนองการเพิ่มขึ้นของโอกาสเกิดบัฟเฟอร์ล้น ทำให้ลดโอกาสเกิดและจำนวนการดรอปได้ดีกว่ากลไก RED ในช่วงที่บัฟเฟอร์ยังเหลือพื้นที่มาก และลดการดรอปจากบัฟเฟอร์ล้นได้ดีกว่ากลไก Tail-Drop ในช่วงที่เครือข่ายคับคั่ง จึงทำให้ทรูพุทสูง นอกจากนี้ยังลดการดรอปต่อเนื่อง ถึงแม้จะต้องแลกกับการเกิดดีเลย์ที่เพิ่มขึ้น แต่เมื่อเปรียบเทียบกับดีเลย์ของ Tail-Drop และ RED จะเห็นได้ว่าดีเลย์ของ ExRED น้อยกว่าของกลไก Tail-Drop อยู่มาก และใกล้เคียงกับของกลไก RED ที่เป็นเช่นนี้ก็เนื่องจากการลดการดรอปช่วยทางอ้อมในการลดเวลาที่ใช้ในการส่งข้อมูลสำเร็จ เพราะไม่ต้องส่งแพ็คเก็ตซ้ำแทนแพ็คเก็ตที่ถูกดรอป ในอีกด้านหนึ่งกลไก ExRED อาจจะซับซ้อนมากกว่ากลไก RED ในการคำนวณความน่าจะเป็นในการตัดสินใจปล่อยผ่านหรือดรอปแพ็คเก็ต แต่ความซับซ้อนก็ไม่ได้เพิ่มมากขึ้นจนเกินไป เนื่องจากส่วนอื่นๆของกลไก ExRED และ RED ทำงานเหมือนกัน อีกทั้งเทคโนโลยีฮาร์ดแวร์ปัจจุบันก้าวหน้าไปไกลสวนทางกับราคาที่ลดลงเรื่อยๆ จึงมีความเป็นไปได้ที่จะหาอุปกรณ์มารับการทำงานได้

#### 6.2 กลไก LDM/RED และ LDM/ExRED

กลไก LDM/RED [22] และ LDM/ExRED [23] ควบคุมการใช้บัฟเฟอร์โดยการวัดระดับความคับคั่งในเครือข่ายจากขนาดคิว ไปควบคุมการเปลี่ยนเส้นทางส่งต่อแพ็คเก็ตเพื่อกระจายภาระงานบนเครือข่าย ซึ่งจะมีการเปลี่ยนเส้นทางเมื่อขนาดคิวสะสมเพิ่มขึ้น ทั้งสองกลไกสามารถควบคุมการใช้บัฟเฟอร์ได้อย่างสมดุลทำให้มีขนาดคิวต่ำ โดยไม่ทำให้เกิดปัญหา Packet reordering จึงช่วยปรับปรุงประสิทธิภาพการทำงานเครือข่าย TCP/IP ในด้านโอกาสเกิดแพ็คเก็ตดรอปและดีเลย์ แต่เมื่อเปรียบเทียบกับ กลไก ExRED มีประสิทธิภาพการกระจายภาระงานที่ดีกว่าทำให้เกิดขนาดคิวในบัฟเฟอร์ต่ำกว่า

#### 6.3 กลไก Mark-Reverse ECN

กลไก Mark-Reverse ECN [24], [25] ช่วยให้การควบคุมการใช้บัฟเฟอร์มีประสิทธิภาพ โดยใช้กลไก RED วัดขนาดคิว และกลไก ECN แจ้งเตือนผู้ส่งโดยการทำเครื่องหมายที่แพ็คเก็ต ACK ทำให้ผู้ส่งสามารถปรับอัตราการส่งได้สอดคล้องกับสถานะความคับคั่งในเครือข่ายอย่างเหมาะสม งานวิจัยนี้ถูกพัฒนาเพื่อให้เครือข่าย TCP/IP ทำงานได้ดีบนแม่ต้องผ่านลิงก์ที่มีดีเลย์สูงอย่างดาวเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- [1] “Recommendations on queue management and congestion avoidance in the Internet,” RFC2309, April, 1999.
- [2] S. Floyd and V. Jacobson, “Random Early Detection Gateways for Congestion Avoidance,” IEEE/ACM Trans. Networking, vol. 1, no. 4, pp. 397–413, August, 1993.
- [3] S. Floyd, RED: “Discussions of Setting Parameters,”  
<http://www.aciri.org/floyd/REDparameters.txt>, Nov, 1997
- [4] V. Jacobson, “Congestion avoidance and control,” In Proc. SIGCOMM '88, (ACM), pp. 314–329, 1988.
- [5] J. Postel, “Transmission control protocol,” RFC793, 1981.
- [6] W. R. Steven, “TCP/IP-illustrated,” vol. 1, Addison Wesley, New York, 1994.
- [7] J. C. Hoe, “Improving the start-up behaviour of a congestion control scheme for TCP,” In Proc. SIGCOMM '96, (ACM), pp. 314–329, 1996.
- [8] R. W. Wolf, “Poisson Arrivals see Time Average,” Operations Research, 20, 223–231, 1982.
- [9] T. Bonald, M. May, and J-C. Bolot, “Analytic Evaluation of RED Performance,” In Proc. IEEE INFOCOM2000, Tel-Aviv, Israel, March, 26–30, 2000.
- [10] R. Laalaoua, T. Czachorski, and T. Atmaca, “Markovian Model of RED Mechanism,” In Proc. First IEEE/ACM International Symposium on Cluster Computing and the Grid 2001, pp. 610–617, 2001.
- [11] S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, “On Load Distribution over Multipath Networks,” IEEE Communications Surveys and Tutorials, vol. 14, no. 3, pp. 662–680, Jul. 2012.
- [12] S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, “On the Performance Analysis of Traffic Splitting on Load Imbalancing and Packet Reordering of Bursty Traffic,” in Proc. IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC 2009), Beijing, China, Nov. 2009.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [13] S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, "Effective Delay-Controlled Load Distribution over Multipath Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 22, no. 10, pp. 1730–1741, Oct. 2011.
- [14] J. Postel, "Internet control Message protocol," RFC792, 1981.
- [15] J. Postel, "A Proposal to add Explicit Congestion Notification (ECN) to IP," RFC2481, 1999.
- [16] K. Ramakrishnan, S. Floyd, and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP," RFC3168, 2001.
- [17] S. Prabhavat, R. Varakulsiripunth, "Performance Improvement on RED Based Gateway in TCP Communication Network," In *Proc. International Conference on Control, Automation and Systems (ICCAS 2004)*, Aug. 25–27, 2004, Bangkok, Thailand, pp.782–787.
- [18] K. G. Shin and C. J. Hou, "Design and Evaluation of Effective Load Sharing in Distributed Real-Time Systems," *IEEE Trans. Parallel and Distributed Systems*, vol. 5, no. 7, pp. 704–719, Jul. 1994.
- [19] O. Kremien and J. Kramer, "Methodical analysis of adaptive load sharing algorithms," *IEEE Trans. Parallel Distribution Systems*, vol. 3, no. 6, pp. 747–760, Nov. 1992.
- [20] C. C. Hui and S. T. Chanson, "Hydrodynamic load balancing," *IEEE Trans. Parallel and Distributed Systems*, vol. 10, no. 11, pp. 1118–1137, Nov. 1999.
- [21] J. Song, S. Kim, M. Lee, H. Lee, and T. Suda, "Adaptive load distribution over multipath in MPLS networks," in *Proc. IEEE ICC*, Anchorage, Alaska, May 2003, pp. 233–237.
- [22] S. Prabhavat, N. Kato, and R. Varakulsiripunth, "Load Distribution Mechanism for Multipath TCP Communication," in *Proc. International Symposium on Technology for Sustainability (ISTS 2011)*, Bangkok, Thailand, Jan. 2012.
- [23] S. Prabhavat, R. Varakulsiripunth, and S. Utsumi, "Load Distribution using Modified RED for Multipath TCP Communication," in *Proc. IEEE International Conference on*

Information Technology and Electrical Engineering (ICITEE 2013), Yogyakarta, Indonesia, Oct. 2013.

- [24] M. Inoue, P. Tangdumrongrat, S. Prabhavat, S. Utsumi, S. Zabir, and N. Shiratori, "Mark-Reverse Explicit Congestion Notification for Satellite IP Networks," in *Proc. International Symposium on Technology for Sustainability (ISTS 2012)*, Bangkok, Thailand, Nov. 2012.
- [25] M. Inoue, P. Tangdumrongrat, S. Prabhavat, S. Utsumi, S. M. S. Zabir, N. Shiratori, "Mark-Reverse Explicit Congestion Notification: Concept, Analysis and Evaluation," *IEICE Technical Report*, vol. 112, no. 208, NS2012-79, pp. 153-158, Sep. 2012.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก

ผลงานส่งเผยแพร่ในงานประชุมวิชาการระดับนานาชาติ  
International Conference on Control, Automation and Systems (ICCAS 2004)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Performance Improvement on RED Based Gateway in TCP Communication Network

Sumet Prabhavat\* and Ruttikorn Varakulsiripunth \*\*

Faculty of Information Technology\* and Faculty of Engineering\*\*  
and Research Center for Communications and Information Technology (ReCCIT)\*,\*\*  
King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand  
sumet@it.kmitl.ac.th\*, kvruttik@kmitl.ac.th\*\*

**Abstract:** Internet Engineering Task Force (IETF) has been considering the deployment of the Random Early Detection (RED) in order to avoid the increasing of packet loss rates which caused by an exponential increase in network traffic and buffer overflow. Although RED mechanism can prevent buffer overflow and hence reduce an average values of packet loss rates, but this technique is ineffective in preventing the consecutive drop in the high traffic condition. Moreover, it increases a probability and average number of consecutive dropped packet in the low traffic condition (named as "uncritical condition"). RED mechanism effects to TCP congestion control that build up the consecutive of the unnecessary transmission rate reducing; lead to low utilization on the link and consequently degrade the network performance. To overcome these problems, we have proposed a new mechanism, named as Extended Drop slope RED (ExRED) mechanism, by modifying the traditional RED. The numerical and simulation results show that our proposed mechanism reduces a drop probability in the uncritical condition.

**Keywords:** Active queue management, Buffer queue management, RED, ExRED

### 1. INTRODUCTION

Recently, Internet Engineering Task Force (IETF) has been considering the deployment of the Random Early Detection (RED) [5] in order to avoid the increasing of packet loss rates caused by an exponential increment of network traffic and buffer overflow. RED is an active queue management technique [6] that can prevent buffer overflow and reduce an average of packet loss rates. But this technique is not only ineffective in preventing high rate of consecutive drop but also increases a probability and mean number of consecutive dropped packets [11]. The consecutive drop will cause consecutive loss and low utilization of a link and it consequently degrades link performance due to TCP congestion control.

Most commonly used TCP congestion control, i.e. TCP Reno / New Reno, assume that every packet loss is an indication of network congestion. This interaction between the error recovery and the congestion control procedures results in a low utilization of the link. In this paper, we have proposed new approach for modifying RED, in order to overcome the above-mentioned problems.

In section 2, we will describe a TCP congestion control to describe how the consecutive drops cause the low utilization on a link. A tradition queue management, i.e. Tail Drop and RED mechanism, will be described in the section 3. And we have shown the analytical and simulation results to compare a probability and a mean number of consecutive dropped packets between Tail Drop and RED. Our simulation results illustrates that RED reduces a packet loss rates by increasing a packet drop probability and a mean number of consecutive dropped packets to prevent buffer overflow. However, an increasing of average queue size results a higher increasing rate of mean number of consecutive dropped packets. It is an

inefficiency of RED in preventing the consecutive drops. With the consecutive drops, TCP congestion control mechanism will assume that network congestion occurs and consequently TCP link utilization is reduced. It could be more critical and leads to the TCP global synchronization problem [11]. In section 4, we propose our algorithm to overcome the problems. Our key of the solution is the modification of drop probability characteristic line. The 2nd order polynomial function is applied to the drop probability characteristic line. We assume that the problem occurs because of inefficiently and unsuitable queue management. If we have a more applicable queue management, buffer will be more efficiently utilized and the consecutive drops will be decreased. We call our proposed algorithm as the Extended drop slope RED (ExRED) mechanism. Finally, we have compared ExRED to the prior mechanism to show that it can improve the performance. The result of our research will be shown in the last section of the paper include the conclusion. We also present that our proposed mechanism improve throughput while keeps an advantage of RED mechanism in the issue of queuing delay.

### 2. TCP CONGESTION CONTROL

TCP was first introduced in early 1980s to provide reliable operation over a variety of transmission media. The efficiency of the TCP has been improved through a series of TCP reference implementations (e.g. Tahoe, Reno), which have refined TCP behavior. The congestion control (slow start [2]) and congestion avoidance (multiplicative decrease [2]) techniques introduced from Tahoe TCP use the principle of self-clocking.

On detection of congestion, slow start procedure implemented in Tahoe TCP 'drains the pipe' before transmission of more data [2]. This achieves network stability but is also unduly conservative. The fast recovery algorithm [3, 4] has therefore been introduced by the Reno TCP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

implementation to drain only a half of the pipe and then recommence transmission, assuming the reception of each duplicate ACK is an indication of a packet leaving the network. Although most of the operating systems released in early 1990s have implemented Tahoe, the current implementations of TCP are based on the Reno reference implementation or have the same functions of Reno.

TCP congestion control mechanism assumes that every packet loss is an indication of network congestion and takes measures to avoid further congestion in the network by reducing the transmission rate. These also illustrate that a single loss of packet effect to transmission rate. Moreover, the consecutive packet losses result a very low utilization of the TCP link due to buffer overflow or buffer control mechanism action.

### 3. BUFFER QUEUE MANAGEMENT

#### 3.1 Tail-Drop

An arrival packet will be allowed to accommodate the buffer queue only if a space of buffer is available. We can illustrate as the following:

$$q \leftarrow \begin{cases} q+1 & ; q < BufferSize \\ q & ; otherwise \end{cases} \quad (1)$$

Where  $q$  is buffer queue size and the BufferSize is a space in buffer. In our simulation, 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 (2) expresses drop probability.

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

Equation (3) and (4) express consecutive drop probability and average number of consecutive drop, respectively.

$$P_{drop}^{cons} (N > n) = \left( \frac{\frac{\lambda}{\mu}}{1 + \frac{\lambda}{\mu}} \right)^n \quad (3)$$

$$E(N) = 1 + \frac{\lambda}{\mu} \quad (4)$$

Equation (5) expresses queuing delay.

$$D = \frac{E(Q)}{(1 - P_{drop})\lambda} - \frac{1}{\mu} \quad (5)$$

where  $E(Q)$  is mean of queue size and given by (6).

$$E(Q) = \begin{cases} \frac{\frac{\lambda}{\mu} \left(1 - (K+1) \left(\frac{\lambda}{\mu}\right)^K + K \left(\frac{\lambda}{\mu}\right)^{K+1}\right)}{\left(1 - \left(\frac{\lambda}{\mu}\right)^{K+1}\right) \left(1 - \frac{\lambda}{\mu}\right)} & ; \lambda \neq \mu \\ \frac{K}{2} & \lambda = \mu \end{cases} \quad (6)$$

#### 3.2 RED Mechanism

RED [5] 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 is calculated by a current queue size (or instantaneous queue size) using an exponentially weighted moving average (EWMA) [5] as shown below.

$$\hat{q} \leftarrow \begin{cases} (1-w)\hat{q} + wq & ; q \neq 0 \\ (1-w)^{\frac{w}{\lambda}} \hat{q} & ; otherwise \end{cases} \quad (7)$$

where  $w$  is an EWMA parameter which is a small constant value and defined by [5] 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 [5]. The suitable value for parameters in this function were discussed in [5] and [11].

$$p_d(\hat{q}) = \begin{cases} 0 & ; \hat{q} < k_l \\ \frac{\hat{q} - k_l}{k_h - k_l} \times \max_p & ; k_l \leq \hat{q} < k_h \\ 1 & ; \hat{q} \geq k_h \end{cases} \quad (8)$$

By using the PASTA property [9] 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 [7][10] as the following:

#### Drop Probability

Drop probability in a RED router can be approximated by

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

**Consecutive Dropped Packet**

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

$$\forall n \geq 0, P_{drop}^{cons}(N > n) = \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q})(p_d(\hat{q}))^{n+1}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q})p_d(\hat{q})} \quad (10)$$

Equation (10) allows us in particular to evaluate the mean of the number of consecutive dropped packets, as shown below.

$$E(N) = 1 + \frac{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q}) \frac{(p_d(\hat{q}))^2}{1-p_d(\hat{q})}}{\sum_{\hat{q}=0}^{K-1} \pi(\hat{q})p_d(\hat{q})} \quad (11)$$

**Average Queuing Delay**

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

$$D = \frac{E(\hat{Q})}{(1-P_{drop})\lambda} - \frac{1}{\mu} \quad (12)$$

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

$$E(\hat{Q}) = \frac{\sum_{k=0}^K k \left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1-p_d(\hat{q}))}{\sum_{k=0}^K \left(\left(\frac{\lambda}{\mu}\right)^k \prod_{\hat{q}=0}^{k-1} (1-p_d(\hat{q}))\right)} \quad (13)$$

**4. ExRED MECHANISM**

**4.1 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,

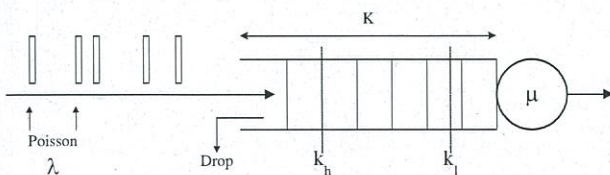


Fig. 1. Analytical model for ExRED

According to (8) to (13), 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:

$$p_d(\hat{q}) = \begin{cases} 0 & ; \hat{q} < k_l \\ \frac{\hat{q} - k_l}{k_h - k_l} \times \max_p & ; k_l \leq \hat{q} < k_h \\ a_2 \hat{q}^2 + a_1 \hat{q} + a_0 & ; k_h \leq \hat{q} \leq K \end{cases} \quad (14)$$

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 [5] and [11] 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.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญูาตเ็นหาไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

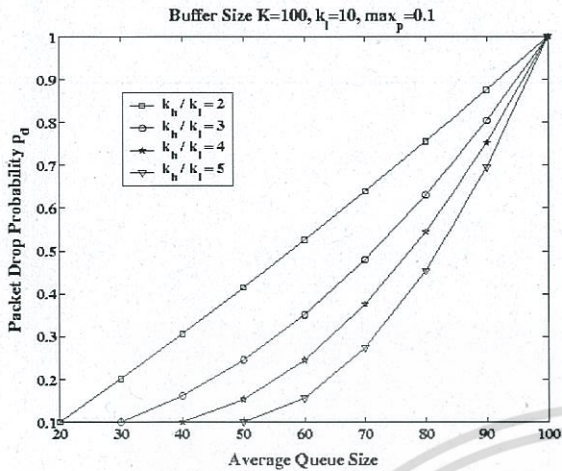


Fig. 2. Drop function of ExRED when average queue size exceeds maximum threshold

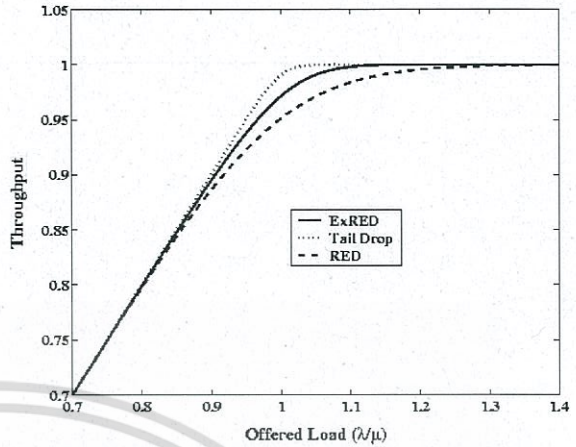


Fig. 4. Estimation of Throughput vs. Offered load

### 4.3 Performance 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 (9). When  $P_d$  is decreased, it follows then from (9) that the average packet drop is written as (15).

$$P_{drop} = \sum_{\hat{q}=k_l}^K \pi(\hat{q}) \times (a_2 \hat{q}^2 + a_1 \hat{q} + a_0) + \sum_{\hat{q}=k_l}^{k_h-1} P_d(\hat{q}) \times \pi(\hat{q}) \quad (15)$$

If  $P_{drop}^{RED}$  and  $P_{drop}^{ExRED}$  are drop probability of RED obtained by (9) and ExRED obtained by (15), 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 (10) can be expressed by

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

### 4.3 Simulation results

To compare the simulation result of our proposed mechanism, we model the discrete event simulation. 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 [5].
- $\max_p = 0.1$ : The value was chosen as suggested in [11].

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

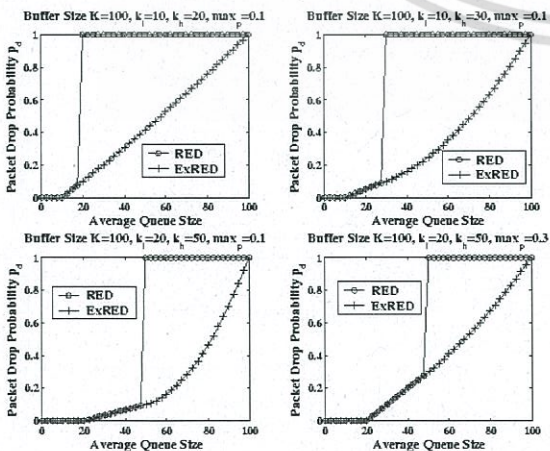


Fig. 3. Comparing of RED and ExRED Drop function

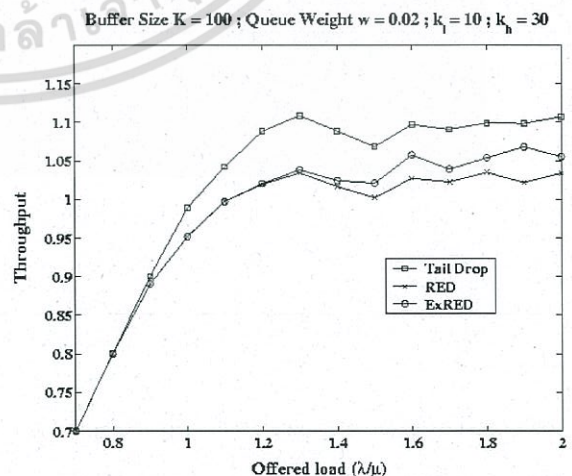


Fig. 5. Throughput of Tail Drop, RED, and ExRED vs. Offered load

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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.

## 5. 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.

## REFERENCES

- [1] J. Postel, "Transmission control protocol," Information Sciences Institute, *RFC793*, 1981.
- [2] V. Jacobson, "Congestion avoidance and control," *Proceeding of SIGCOMM '88*, (ACM), pp. 314-329, 1988.
- [3] W. R. Steven, "TCP/IP illustrated", vol. 1, Addison Wesley, New York, 1994.
- [4] J. C. Hoe, "Improving the start-up behaviour of a congestion control scheme for TCP", *Proceeding of SIGCOMM '96*, (ACM), pp. 314-329, 1996.
- [5] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance", In *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397-413, August, 1993.
- [6] "Recommendations on queue management and congestion avoidance in the Internet", *RFC2309*, April, 1999.
- [7] T. Bonald, M. May, and J-C. Bolot, "Analytic Evaluation of RED Performance", In *IEEE INFOCOM2000*, Tel-Aviv, Israel, March, 26-30, 2000.
- [8] "Assured Forwarding PHB Group", *RFC 2597*, June, 1999.

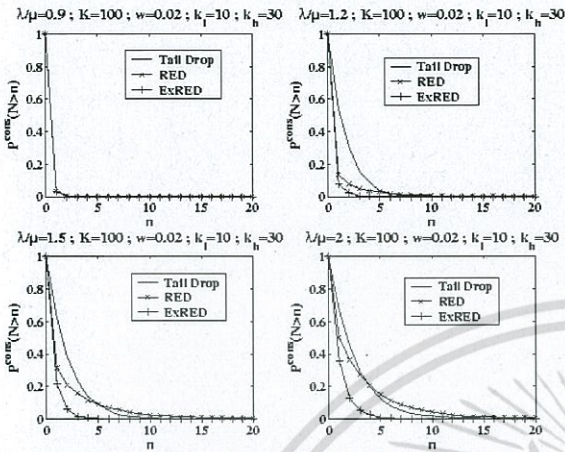


Fig. 6. Consecutive Drop Probability vs. Offered load at 0.9, 1.2, 1.5, and 2.0

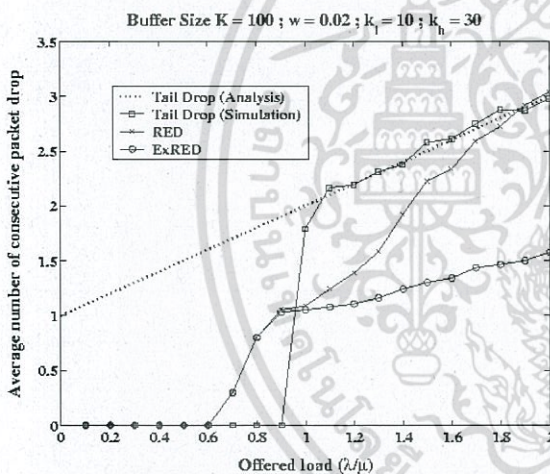


Fig. 7. Average number of Consecutive Drop vs. Offered load

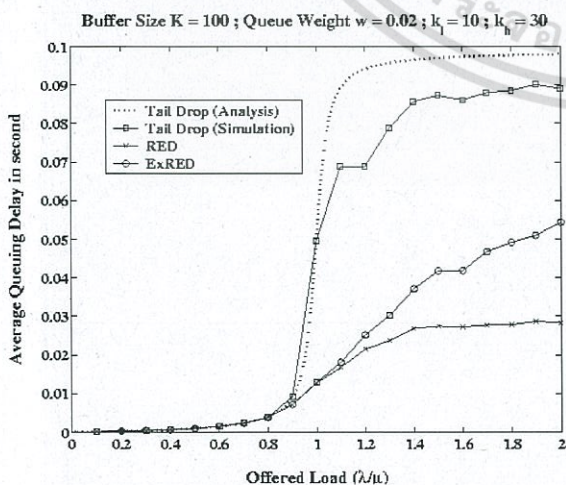


Fig. 8. Average Queuing Delay vs. Offered load

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [9] R. W. Wolf, "Poisson Arrivals see Time Average", *Operations Research*, 20, 223-231, 1982.
- [10] R.Laalaoua, 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.
- [11] S. Floyd, RED: "Discussions of Setting Parameters", <http://www.aciri.org/floyd/REDparameters.txt>, Nov, 1997



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ข

ผลงานส่งเผยแพร่ในงานประชุมวิชาการระดับนานาชาติ  
International Symposium on Technology for Sustainability (ISTS 2011)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Load Distribution Mechanism for Multipath TCP Communication

Sumet Prabhavat<sup>1</sup>, Nei Kato<sup>2</sup>, Ruttikorn Varakulsiripunth<sup>3</sup>

<sup>1</sup>Faculty of Information Technology, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand

<sup>2</sup>Graduate School of Information Sciences, Tohoku University, Sendai, Japan

<sup>3</sup>Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand

Email: <sup>1</sup>sumet@it.kmitl.ac.th, <sup>2</sup>kato@it.ecei.tohoku.ac.jp, <sup>3</sup>kvrttik@kmitl.ac.th

**Abstract**— In today's communication networks, demand for a large bandwidth network connection is continuously growing while network connections via multiple paths are likely to exist because of high degree of connectivity. Utilization of the multiple paths is an economical solution for provisioning network capacity to meet the users' demand but it brings about major issues such as load imbalance (causing a large queue and buffer overflow at a gateway) and packet reordering problems that have significant impacts on TCP connections. In this paper, we propose a load distribution model for TCP transmission, called Load Distribution over Multipath with Random Early Detection (LDM/RED). LDM/RED takes advantages of RED mechanism to maintain a small queue, avoid buffer overflow at a gateway, and mitigate packet reordering problem. Our proposed model is evaluated and compared to the other existing models by simulations under realistic traffic conditions.

**Index Terms**—Load Distribution, Packet Reordering, Multipath TCP.

## I. INTRODUCTION

The demand for a wide variety of network services has been the major driving force for development of networking technologies [1]. Multipath configurations can be established in several different ways [2]. A primary objective of multiple paths is to improve network reliability by increasing network availability (i.e., reducing network downtime); a main path was used for data transmission while the other ones were backups which would be activated when the main path became unavailable. Currently, the exploitation of multiple paths no longer aims only at circumventing single point of failure scenarios but also focuses on facilitating network provision [1]–[4] where its effectiveness is essential to maximize high quality network services and guarantee Quality of Service (QoS) at high data rates. Using multiple paths as a single path with aggregate bandwidth is a practical solution which is preferable rather than provisioning a large-bandwidth path because it offers a possibility to establish a very large-bandwidth connection. This improves both scalability to support the future growth in bandwidth demand and affordability for network users. It also provides flexibility in bandwidth management within the communication protocol over the multipath network. Bandwidth aggregation and network-load balancing are major issues that have attracted a large amount of research, and a number of load distribution approaches have been proposed. The rest of this paper is organized as follows. Section II briefly describes existing load distribution models, some studies on

packet reordering problems, and an active queue management for TCP networks. Section III presents a new load distribution model with active queue management. Section IV discusses performance evaluation under real traffic conditions. Concluding remarks are then given in Section V.

## II. RELATED WORKS

### A. Load Distribution Models

Load distribution can be classified into packet-based model and flow-based models. Packet-based models can be further classified into non-adaptive and adaptive models. Surplus Round Robin (SRR) [5] is an enhanced version of round robin family model which is widely used schemes in the Internet because it can achieve starvation-free (i.e., no non-work-conserving idle time) and competent load balancing efficiency. Least-loaded first (LLF) [6]–[8] is one of the most well known load-sharing approaches introduced to handle task loads with heavy-tailed distribution, where a task is assigned to the least-loaded server. In load distribution over multiple paths, with this scheme, a path having the smallest load or the shortest queue will be selected for an arrived packet. Their major drawback is that they do not consider the order of tasks (i.e., do not keep packet ordering) which can result in the packet reordering problem [9].

On the other hand, flow-based models such as Direct Hashing (DH), Table-based Hashing (TH) [10], [11], and Fast Switching (FS) [12] can completely prevent packet reordering but they can cause load imbalance due to its inability to deal with variation of the flow size distribution. Adaptive load distribution models, e.g., Load Distribution over Multipath (LDM) and Load Balancing for Parallel Forwarding (LBPF) [13]–[15] were introduced to mitigate the load imbalance problem. However, there is trade-off between improving load balancing and maintaining a low risk of packet reordering depends on the respective algorithms as well as their set parameters.

### B. Packet Reordering Problem

Several works have addressed the packet reordering problem [16]–[22]. Packet reordering may occur more frequently with higher probabilities of splitting a flow and switching a path [18], [19]. On the other hand, the possibility of packet reordering will decrease if the interarrival time of two successive packets belonging to the same flow is greater than the maximum time required to deliver a packet via the parallel paths [23]. The packet interarrival time must be

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

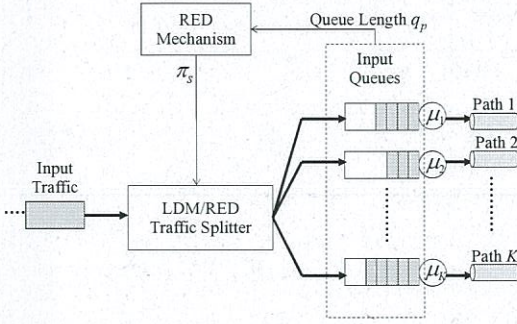


Figure 1. Description of the proposed model, LDM/RED

greater than the difference between the maximum and the minimum delays among parallel paths to ensure preservation of packet order. If the load is not properly distributed among the parallel paths, some paths may experience congestion (leading to large queues) while the others are idle; consequently, the delay difference can be very large.

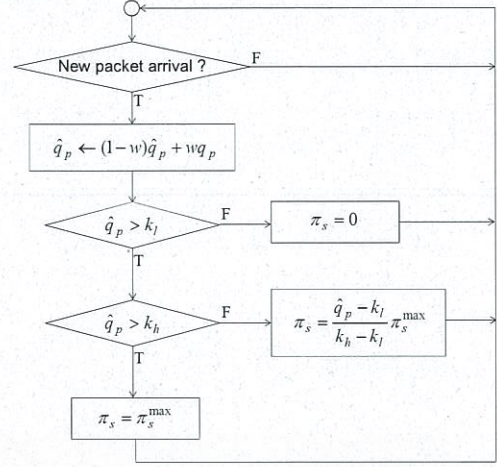
### C. Active Queue Management for TCP Gateway

Random Early Detection (RED) [24] is one of the most well-known active queue management which is introduced to improve performance of TCP communication. RED 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 is calculated by a current queue size using an exponentially weighted moving average (EWMA). Based on the average queue size, RED decides whether or not to drop an incoming packet. Packet arrival rate of a flow plays an important role in determining packet drop probability. The higher the packet arrival rate, the higher the packet drop rate. This feature of RED is to prevent two major problems that have a significant impact on TCP connections: buffer overflow and synchronization of TCP flows [25] due to burst traffic.

## III. LOAD DISTRIBUTION (OVER MULTIPATH NETWORKS) WITH RANDOM EARLY DETECTION

Fig. 1 describes our approach called Load Distribution over Multipath with Random Early Detection (LDM/RED). The input traffic is split into flows, each of which independently takes a path determined by an adaptive path selection component. LDM/RED is a variant version of FS [12] and LDM [13], a table-based model which selects paths according to information in the flow-path mapping table. A packet belonging to an existing flow is sent via the same path as its preceding one. When a new flow emerges, a packet belonging to the new flow will be sent via a new path having the shortest queue.

In LDM/RED, a mapping table can be modified and the path for an existing flow will be changed. The existing flow is switched (to a new path having the shortest queue) with probability  $\pi_s$  which is determined by RED mechanism, as illustrated in Fig. 2. Based on an average queue length  $\hat{q}_p$ , LDM/RED makes a decision of whether or not to split the flow. There are two queue thresholds: minimum and max-


 Figure 2. Calculation of  $\pi_s$  in traffic splitting algorithm

imum thresholds, i.e.,  $k_l$  and  $k_h$ . When  $\hat{q}_p > k_l$ , the flow is split with probability  $\pi_s > 0$ . The larger the average queue size  $\hat{q}_p$ , the higher the probability  $\pi_s$ . When  $\hat{q}_p > k_h$ ,  $\pi_s$  increases to a pre-defined parameter of maximum probability  $\pi_s^{\max}$ . If  $\pi_s^{\max} = 1$ , all flows are always moved to a new path.

## IV. PERFORMANCE EVALUATION

### A. Simulation model and parameters

In this section, comparative performance under different conditions of real traffics is demonstrated. Three simulation scenarios are conducted to show the performance of each load distribution model, by using 1-hour long real traffic traces [26], i.e., DSet1, DSet2, and DSet3, which contain wide-area traffics at primary Internet access point between Digital Equipment Corporation and the rest of the world, where characteristics of the traces are listed in Table I. Bandwidth capacities (or mean service rates) of path 1, path 2, and path 3 are 1, 4, and 7 Mbps, respectively; the total bandwidth capacity of the multiple paths is 12 Mbps. As compared to the bandwidth capacities, traffics generated from trace DSet1 and DSet2 incur heavy load and some load-spikes. Moreover, we use trace DSet3 to generate extremely heavy traffic, having maximum offered load much higher than the total bandwidth capacity, thus incurring overload on the multiple paths.

With the set-up simulation environment, LDM/RED, LDM, LLF, and SRR are evaluated. In LDM/RED, the parameters for RED are chosen: EWMA parameter  $w=0.02$ ,  $k_l=10$ ,  $k_h=30$ , and  $\pi_s^{\max}=0.1$ . These parameters setting are recommended in [24], [27], and [28]. In SRR, the numbers of credits assigned for path 1, path 2, and path 3 are 1, 4, and 7, respectively. In LBPf, the size of the table is 20,  $W=1000$ , and  $P=20$ .

### B. Load Balancing Efficiency

Tables II shows that LDM/RED can achieve load balancing, as compared to the other models since probabilities of buffer occupancy of all paths are almost equal. This can be implied that all paths are fairly utilized. Moreover, Table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TABLE I. PROFILE OF TRAFFIC TRACES

Trace ID	# Packets $\times 10^6$	Traffic Rate (Mbps.)			# Different Flows	Flow Size (Packets)		Flow Rate (Flows/Second)		
		Mean	Min.	Max.		Mean	CV	Mean	Min.	Max.
DSet 1	2.66	5.91	2.07	13.65	5865	453.87	7.52	137.89	77	204
DSet 2	2.87	6.38	0.46	12.24	12903	222.71	5.98	175.32	44	247
DSet 3	3.86	8.58	1.86	15.45	12710	303.88	7.11	184.50	90	269

TABLE II. PROBABILITY OF BUFFER OCCUPANCY

Load Distribution Model	Dset 1			Dset 2			Dset 3		
	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3
LDM/RED	0.70	0.66	0.65	0.73	0.67	0.65	0.87	0.82	0.80
LDM	0.86	0.71	0.59	0.89	0.73	0.58	0.95	0.85	0.77
LLF	0.77	0.53	0.47	0.80	0.55	0.48	0.91	0.74	0.70
SRR	0.35	0.66	0.66	0.37	0.68	0.68	0.61	0.81	0.82

TABLE III. MAXIMUM QUEUE SIZE (PACKETS)

Load Distribution Model	Dset 1			Dset 2			Dset 3		
	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3
LDM/RED	100	98	100	104	97	102	376	375	375
LDM	1685	422	180	3845	378	76	5976	2780	522
LLF	92	92	93	83	83	83	293	293	293
SRR	43	86	185	33	88	165	110	350	557

III shows that LDM/RED can prevent buffer overflow; maximum queue size incurred by LDM/RED is significantly smaller than that incurred by LDM and SRR. This is because burst traffic (which can cause a large queue) is detected by RED and distributed over multiple paths.

C. Packet Order Preservation

Fig. 3 illustrates that LDM/RED can efficiently alleviate packet reordering which inherently exists in packet-based models such as SRR and LLF. SRR, which sends packets in a round robin manner, causes a high risk of packet reordering. LLF, which chooses only the path with the shortest queue size, also causes a very high risk of packet reordering. LBPf splits a group of largest flows, thus causing the risk of packet reordering. The discussion on this issue can be found in [23] and [29].

V. CONCLUDING REMARKS

This paper presents a load distribution model for TCP transmission. LDM/RED takes advantages from RED mechanism in traffic splitting and distribute load subject to necessary condition to prevent packet reordering. Therefore, LDM/RED can prevent buffer overflow (due to burstiness of TCP flows) at a gateway and maintain a low risk of packet reordering. The superior performances of LDM/RED are justified by comparative performance among LDM/RED and the current existing models by simulations under various traffic conditions.

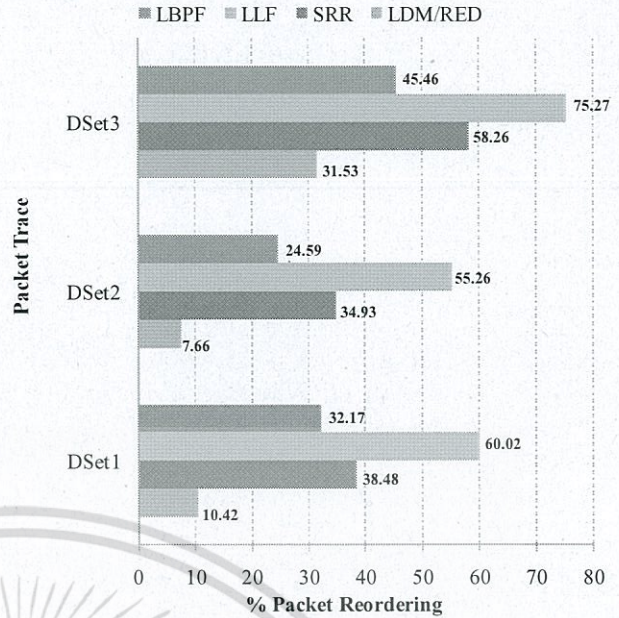


Figure 3. Risk of Packet Reordering

VI. REFERENCES

- [1] L. Golubchik, J. Lui, T. Tung, A. Chow, W. Lee, G. Franceschinis, and C. Anglano, "Multi-path continuous media streaming: What are the benefits?," *Performance Evaluation*, vol. 49, pp. 429-449, Sep. 2002.
- [2] S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, "On Load Distribution over Multipath Networks," *IEEE Communications Surveys and Tutorials*. (to be published)
- [3] M. Menth, R. Martin, A. Koster, and S. Orlowski, "Overview of resilience mechanisms based on multipath structures," in *Proc. the 6th International Workshop on Design and Reliable Communication Networks (DRCN)*, La Rochelle, France, Oct. 2007.
- [4] N. Wang, K. Ho, G. Pavlou, and M. Howarth, "An overview of routing optimization for Internet traffic engineering," *IEEE Communications Surveys and Tutorials*, vol. 10, no. 1, pp. 36-56, 2008.
- [5] H. Adishesu, G. Parulkar, and G. Varghese, "A reliable and scalable striping protocol," *ACM SIGCOMM Computer Communication Review*, vol. 26, no. 4, pp. 131-141, Oct. 1996.
- [6] K. G. Shin and C. J. Hou, "Design and Evaluation of Effective Load Sharing in Distributed Real-Time Systems," *IEEE Trans. Parallel and Distributed Systems*, vol. 5, no. 7, pp. 704-719, Jul. 1994.
- [7] O. Kremien and J. Kramer, "Methodical analysis of adaptive load sharing algorithms," *IEEE Trans. Parallel Distribution Systems*, vol. 3, no. 6, pp. 747-760, Nov. 1992.
- [8] C. C. Hui and S. T. Chanson, "Hydrodynamic load balancing," *IEEE Trans. Parallel and Distributed Systems*, vol. 10, no. 11, pp. 1118-1137, Nov. 1999.
- [9] Z. Tari, J. Broberg, A. Zomaya, and R. Baldoni, "A least flow-time first load sharing approach for distributed server farm," *Journal of Parallel and Distributed Computing*, vol. 65, no. 7, pp. 832-842, Jul. 2005.
- [10] C. Villamizar, "OSPF optimized multipath (OSPF-OMP)," Internet draft draft-ietf-ospf-omp-02.txt, Feb. 1999.
- [11] D. Thaler and C. Hopps, "Multipath issues in unicast and multicast next-hop selection," RFC 2991, Nov. 2000.
- [12] A. Zinin, *Cisco IP Routing, packet forwarding and intra-domain routing protocols*. Reading, MA: Addison Wesley, 2002.
- [13] J. Song, S. Kim, M. Lee, H. Lee, and T. Suda, "Adaptive load distribution over multipath in MPLS networks," in *Proc. IEEE ICC*, Anchorage, Alaska, May 2003, pp. 233-237.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [14] W. Shi, M. H. MacGregor, and P. Gburzynski, "Load balancing for parallel forwarding," *IEEE/ACM Trans. Networking*, vol. 13, no. 4, pp. 790–801, Aug. 2005.
- [15] S. Kandula, D. Katabi, S. Sinha, and A. Berger, "Dynamic load balancing without packet reordering," *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 2, pp. 53–62, Apr. 2007.
- [16] V. Paxson, G. Almes, J. Mahdavi, and M. Mathis, "Framework for IP performance metrics," RFC 2330, May. 1998.
- [17] C. Demichelis and P. Chimento, "IP packet delay variation metric for IP performance metrics (IPPM)," RFC 3393, Nov. 2002.
- [18] N. M. Piratla, A. P. Jayasumana, A. A. Bare, and T. Banka, "Reorder buffer-occupancy density and its application for measurement and evaluation of packet reordering," *Computer Communications*, vol. 30, no. 9, pp. 1980–1993, Jun. 2007.
- [19] N. M. Piratla and A. P. Jayasumana, "Reordering of packets due to multipath forwarding – An analysis," in *Proc. IEEE ICC*, Istanbul, Turkey, Jun. 2006, pp. 829–834.
- [20] A. Morton, L. Ciavattone, G. Ramachandran, S. Shalunov, and J. Perser, "Packet reordering metrics," RFC 4737, Nov. 2006.
- [21] A. Jayasumana, N. Piratla, T. Banka, and R. Whitner, "Improved packet reordering metrics," RFC 5236, Jun. 2008.
- [22] S. Jaiswal, G. Iannaccone, C. Diot, J. Kurose, and D. Towsley, "Measurement and classification of out-of-sequence packets in a tier-1 IP backbone," *IEEE/ACM Trans. Networking*, vol. 15, no. 1, pp. 54–66, Feb. 2007.
- [23] S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, "On the Performance Analysis of Traffic Splitting on Load Imbalancing and Packet Reordering of Bursty Traffic," in *Proc. IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC 2009)*, Beijing, China, Nov. 2009.
- [24] S. Floyd and V. Jacobsan, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.
- [25] S. Floyd and V. Jacobson, "On traffic phase effects in packet-switched gateways," *Internetworking: Research & Experience*, vol. 3, no. 3, Sep. 1992, pp. 115–56, UK.
- [26] (Online Sources) P. Danzig, J. Mogul, V. Paxson, and M. Schwartz. (1995, Mar.). *The Internet Traffic Archive*. Available: <http://ita.ee.lbl.gov/index.html>.
- [27] (Online Sources) S. Floyd (1967, Nov.). "RED: Discussions of Setting Parameters," Available: <http://www.icir.org/floyd/REDparameter.s.txt>.
- [28] S. Prabhavat, R. Varakulsiripunth, and S. Noppanakeepong, "Throughput improvement on RED mechanism," in *Proc. IEEE Communication Systems (ICCS)*, Singapore, Nov. 2002, pp. 599–603.
- [29] S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, "Effective Delay-Controlled Load Distribution over Multipath Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 22, no. 10, pp. 1730–1741, Oct. 2011.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ค

ผลงานส่งเผยแพร่ในงานประชุมวิชาการระดับนานาชาติ  
International Conference on Information Technology and Electrical Engineering  
(ICITEE 2013)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Load Distribution using Modified RED for Multipath TCP Communication

Sumet Prabhavat<sup>1</sup>Ruttikorn Varakulsiripunth<sup>2</sup>Satoshi Utsumi<sup>3</sup>Yasushi Kato<sup>4</sup><sup>1</sup>Faculty of Information Technology, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand<sup>2</sup>Faculty of Information Technology, Thai-Nichi Institute of Technology, Bangkok, Thailand<sup>3,4</sup>Tsuruoka National College of Technology, Yamagata, Japan<sup>1</sup>sumet@it.kmitl.ac.th, <sup>2</sup>kvrttik@gmail.com, <sup>3</sup>u.satoshi0215@gmail.com, <sup>4</sup>president@tsuruoka-nct.ac.jp

**Abstract**— Demand for large bandwidth network connections is growing continuously while network connections via multiple paths can exist because of high degree of connectivity. Utilization of the multiple paths is an economical solution for provisioning large network capacity to meet the users' requirements but it brings about key issues such as load imbalance (causing a large queue and buffer overflow at a gateway) and packet reordering problems that have significant impacts on TCP connections. In this paper, we propose a load distribution model for TCP transmission, called Load Distribution over Multipath with Extended Drop Slope Random Early Detection (LDM/ExRED). LDM/ExRED takes advantages of RED mechanism to maintain a small queue, avoid buffer overflow at a gateway, and mitigate packet reordering problem. In addition, it reduces peak buffer occupancy by using our modified RED (i.e., ExRED). Our work will be evaluated and compared to the other existing models by simulations under realistic traffic conditions.

**Keywords**—Load Distribution; Load Balancing; Multipath TCP

## I. INTRODUCTION

A wide variety of network service requirements demanded by users has been an important driving force for development of networking technologies [1]. By using today's technologies, multipath configurations can be established in various ways [2]. Their primary objectives are to improve network reliability by increasing network availability and reducing network downtime. A main path was used for data transmission while the others were backups which would be activated when the main path became unavailable. Currently, applications of multiple paths aims not only at avoiding a single point of failure but also at facilitating network provision [1–4] where its effectiveness is essential to quality of network services and Quality of Service (QoS) guarantee. Using multiple paths as a single path with aggregate bandwidth is a practical solution which is preferable rather than provisioning a large-bandwidth path because it offers a possibility to establish a very large-bandwidth connection. This improves scalability to support the future growth in bandwidth demand and affordability for network users. It also provides flexibility in bandwidth management within the communication protocol over the multipath network. Bandwidth aggregation and network-load balancing are important issues having been attractive research

topics, and a large amount of load distribution approaches have been proposed. The rest of this paper is organized as follows. Section II briefly describes existing load distribution models, some studies on packet reordering problems, and an active queue management for TCP networks. Section III presents a new load distribution model with our modified active queue management mechanism. Section IV discusses performance evaluation under real traffic conditions. Conclusions are then given in Section V.

## II. RELATED WORKS

### A. Load Distribution Models

Load distribution can be classified into packet-based model and flow-based models. Packet-based models can be further classified into non-adaptive and adaptive models. Surplus Round Robin (SRR) [5] is a simple non-adaptive load distribution model. SRR is an enhanced version of round robin family model which is widely used schemes in the Internet because it can achieve starvation-free and acceptable load balancing efficiency. Least-loaded first (LLF) [6–8] is an adaptive load distribution model. LLF is a well-known load-sharing approach introduced to manage task loads with heavy-tailed distribution, where a task is assigned to the least-loaded server. In multipath load balancing with this scheme, a path having the smallest load or the shortest queue will be selected for an arrived packet. The most important drawback is that they do not consider task sequence, which can result in the packet reordering problem [9].

On the other hand, Direct Hashing (DH) [10], Table-based Hashing (TH) [11], and Fast Switching (FS) [12] are flow-based models which can prevent packet reordering but they can cause load imbalance problem due to inability to cope with variance of the flow size distribution. This problem can cause a large buffer occupancy leading to a high risk of buffer overflow [2]. Adaptive load distribution models, e.g., Load Distribution over Multipath (LDM) and Load Balancing for Parallel Forwarding (LBPF) [13–15] were introduced to mitigate the load imbalance problem. However, there is trade-off between balancing load and maintaining a low risk of packet reordering. This depends on the respective algorithms as well as their set parameters.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

B. Packet Reordering Problem

Several works have studied on packet reordering problem [16–22]. Packet reordering can occur with high probabilities of splitting a flow and switching a path [18], [19]. On the other hand, the possibility of packet reordering will decrease if the interarrival time of two consecutive packets belonging to the same flow is longer than the longest time required to deliver a packet via the parallel paths [23]. The packet interarrival time must be greater than the difference between the longest and the shortest delays among parallel paths to ensure packet order preservation. If the load is not properly distributed among the parallel paths, some paths may experience congestion (leading to large queues) while the others are idle; consequently, the delay difference can be very large.

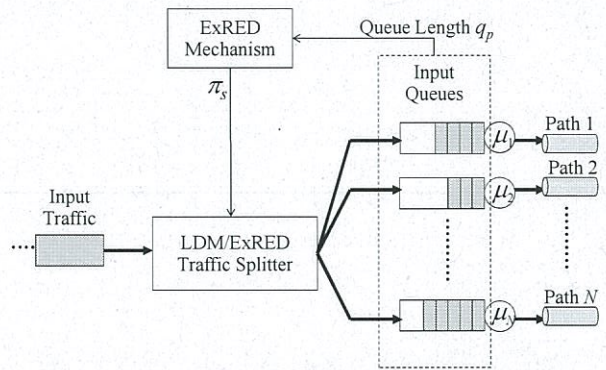


Figure 1. Description of the proposed model, LDM/ExRED

C. Active Queue Management for TCP Gateway

Random Early Detection (RED) [24] is a well-known active queue management mechanism introduced to improve performance of TCP communication. RED 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 is calculated by a current queue size using an exponentially weighted moving average (EWMA). Based on the average queue size, RED decides whether or not to drop an incoming packet. Packet arrival rate of a flow plays an important role in determining packet drop probability. The higher the packet arrival rate, the higher the packet drop rate. This feature of RED is to prevent two major problems that have a significant impact on TCP connections: buffer overflow and synchronization of TCP flows [25] due to burst traffic. In LDM/RED [26], we uses RED for sensing congestion and marking a packet arriving at the time congestion occurs. The marked packet will be switched to a less congested path.

III. LOAD DISTRIBUTION (OVER MULTIPATH NETWORKS) WITH EXTENDED DROP SLOPE RANDOM EARLY DETECTION

Figure 1 describes our approach called Load Distribution over Multipath with Extended Drop Slope Random Early Detection (LDM/ExRED), which is developed from LDM/RED [26] and our modified RED algorithm (i.e., ExRED) [27]. Assume that multiple and parallel connections are established between a source and a destination before data transmission is started. For each path  $p$ , bandwidth is  $\mu_p$ . The other parameters, e.g., hop counts and path delay, are assumed reflected by a measured queue length  $q_p$ . The input traffic is split into flows, each of which independently takes a path determined by a path selection component. LDM/ExRED is a variant version of FS [12] and LDM [13], a table-based model which selects paths based on the flow-path mapping table. A packet belonging to an existing flow is sent via the same path as its previous ones. When a new flow appears, a packet belonging to the new flow will be sent via a new path having the shortest queue.

In LDM/ExRED, a mapping table is modified and the path for an existing flow will be changed. The existing flow is switched (to a new path having the shortest queue) with probability  $\pi_s$  which is determined by ExRED mechanism, as illustrated in Fig. 2. Based on an average queue length  $\hat{q}_p$ ,

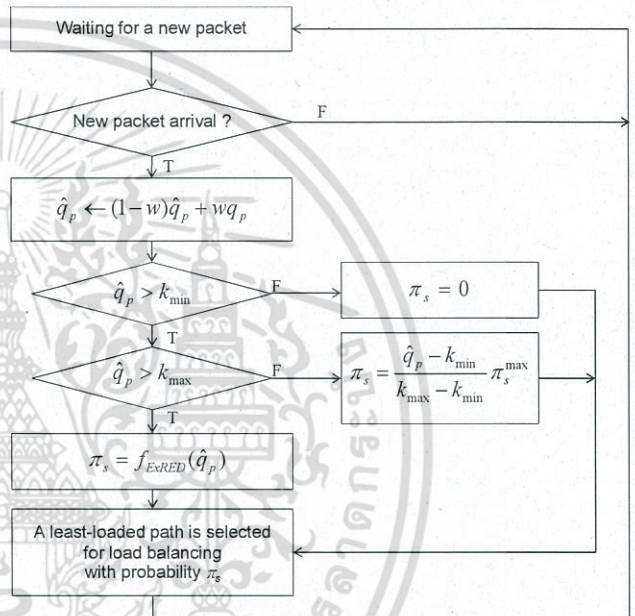


Figure 2. Flowchart to calculate  $\pi_s$  in traffic splitting algorithm

LDM/ExRED makes a decision of whether or not to split the flow. There are two queue thresholds: minimum and maximum thresholds, i.e.,  $k_{min}$  and  $k_{max}$ . When  $\hat{q}_p > k_{min}$ , the flow is split with probability  $\pi_s > 0$ . The larger the average queue size  $\hat{q}_p$ , the higher the probability  $\pi_s$ . When  $\hat{q}_p > k_{max}$ ,  $\pi_s$  increases according to polynomial function in (1) until  $\pi_s = \pi_s^{max}$  (which is a desired maximum value of splitting probability). When  $\pi_s^{max} = 1$  and  $\pi_s = \pi_s^{max}$ , all flows are always moved to a new path.

$$f_{ExRED}(\hat{q}) = a_2\hat{q}^2 + a_1\hat{q} + a_0 \tag{1}$$

where coefficients of the positive increasing function are

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$a_2 = \frac{(k_{max} - k_{min}) - (K - k_{min})\pi_s^{max}}{(k_{max} - k_{min})(K - k_{max})^2}$$

$$a_1 = \frac{(k_{max}^2 + K^2 - 2k_{max}k_{min})\pi_s^{max} - 2k_{max}^2 + 2k_{max}k_{min}}{(k_{max} - k_{min})(K - k_{max})^2}$$

$$a_0 = \frac{k_{max}^3 - k_{max}^2k_{min} - (k_{max}^2 + 2k_{max}k_{min} + k_{min}K)K\pi_s^{max}}{(k_{max} - k_{min})(K - k_{max})^2}$$

TABLE I. PROFILE OF TRAFFIC TRACES

Trace ID	# Packets x10 <sup>6</sup>	Traffic Rate (Mbps.)			# Different Flows	Flow Size (Packets)		Flow Rate (Flows/Second)		
		Mean	Min.	Max.		Mean	CV	Mean	Min.	Max.
DSet1	2.66	5.91	2.07	13.65	5865	453.87	7.52	137.89	77	204
DSet2	2.87	6.38	0.46	12.24	12903	222.71	5.98	175.32	44	247
DSet3	3.86	8.58	1.86	15.45	12710	303.88	7.11	184.50	90	269

TABLE II. PROBABILITY OF BUFFER OCCUPANCY

Load Distribution Model	Dset 1			Dset 2			Dset 3		
	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3
LDM/ExRED	0.71	0.66	0.65	0.73	0.68	0.65	0.86	0.82	0.80
LDM/RED	0.70	0.66	0.65	0.73	0.67	0.65	0.87	0.82	0.80
LDM	0.86	0.71	0.59	0.89	0.73	0.58	0.95	0.85	0.77
LLF	0.77	0.53	0.47	0.80	0.55	0.48	0.91	0.74	0.70
SRR	0.35	0.66	0.66	0.37	0.68	0.68	0.61	0.81	0.82

IV. PERFORMANCE EVALUATION

In this section, comparative performance under different conditions of real traffics is demonstrated.

A. Simulation model and parameters

Three simulation scenarios are conducted to compare the performance of load distribution models, by using 1-hour long real traffic traces [28], i.e., DSet1, DSet2, and DSet3, which are captured from primary Internet access point between Digital Equipment Corporation and the Internet. Characteristics of the traces are listed in Table I. Three paths exist between a pair of source and destination. Bandwidth capacities (or mean service rates) of path 1, path 2, and path 3 are 1, 4, and 7 Mbps, respectively; the total bandwidth capacity is 12 Mbps. As compared to the total bandwidth capacity, traffics generated from trace DSet1 and DSet2 incur heavy load and some load-spikes. Moreover, we use trace DSet3 to generate extremely heavy traffic which peak load is much higher than the total bandwidth capacity, thus incurring network overload.

With the set-up simulation environment, LDM/ExRED, LDM, LLF, and SRR are evaluated. In LDM/RED and LDM/ExRED, the parameters are chosen: EWMA parameter  $w=0.02$ ,  $k_{min}=10$ ,  $k_{max}=30$ , and  $\pi_s^{max}=0.1$ . These parameters setting are recommended in [24], [27], and [29]. In SRR, weights assigned for path 1, path 2, and path 3 are 1, 4, and 7, respectively, which are used for specifying desired forwarding rates on each path [5]. In LBPF, the size of the table is 20,  $W=1000$ , and  $P=20$ , where  $W$  is window size for detecting aggressive flows and  $P$  is period for splitting the aggressive flows. LBPF's parameters can be further read at [14].

B. Load Balancing Efficiency

Tables II shows that LDM/ExRED can achieve load balancing, as compared to the others since probabilities of buffer occupancy of all paths are almost equal. This can be implied that all paths are fairly utilized. Moreover, Table III shows that LDM/RED and LDM/ExRED can prevent buffer overflow; maximum queue sizes incurred by both of them are significantly smaller than those incurred by SRR and LDM. This is because burst traffic (which can cause a large queue) is

TABLE III. MAXIMUM QUEUE SIZE (PACKETS)

Load Distribution Model	Dset 1			Dset 2			Dset 3		
	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3	Path 1	Path 2	Path 3
LDM/ExRED	83	81	82	85	83	82	319	318	317
LDM/RED	100	98	100	104	97	102	376	375	375
LDM	1685	422	180	3845	378	76	5976	2780	522
LLF	92	92	93	83	83	83	293	293	293
SRR	43	86	185	33	88	165	110	350	557

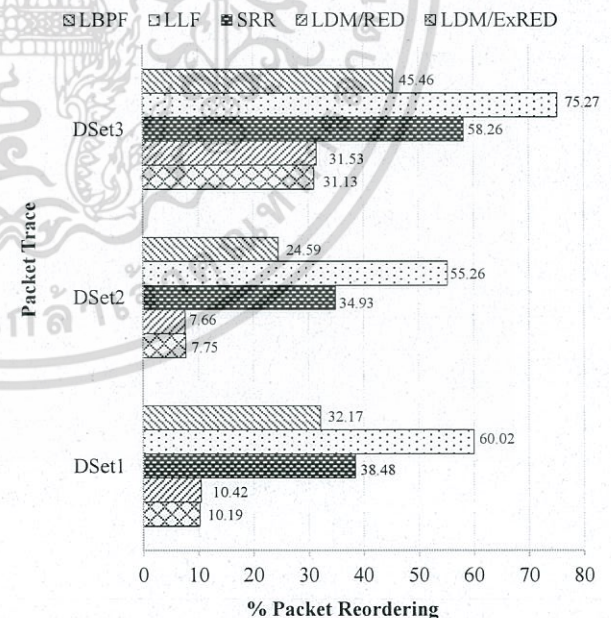


Figure 3. Risk of Packet Reordering

detected by RED and then distributed over multiple paths.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

However, when buffer is highly utilized, RED causes consecutive marked packets while ExRED can prevent such problem. Therefore, LDM/ExRED can reduce peak buffer occupancy. A gateway device requires only a small buffer.

### C. Packet Order Preservation

Fig. 3 illustrates that LDM/ExRED and LDM/RED can efficiently alleviate packet reordering inherently existing in packet-based models such as SRR and LLF. LDM is a flow-based model which can perfectly prevent packet reordering. SRR sends packets in a round robin manner. It can cause a risk of packet reordering. LLF choosing only the path with the shortest queue size can cause a high risk of packet reordering. LBPF splits a group of largest flows, thus leading to the risk of packet reordering. The discussion on this issue can be found in [2], [23], and [30]. As compared to LDM/RED, LDM/ExRED has a potential to maintain low risk of packet reordering because of a smaller path-switching probability. However, difference between delay on high congested path and that on a less congested path is also a factor of the risk of packet reordering [2], [23]. This is an open issue for further studies.

## V. CONCLUDING REMARKS

This paper proposes a load distribution model for TCP transmission. LDM/ExRED takes advantages from RED mechanism in traffic splitting and distribute load subject to necessary condition to prevent packet reordering. With a modified RED, i.e., ExRED, it can prevent buffer overflow (due to burstiness of TCP flows) at a gateway and maintain low risk of packet reordering. The superiority of LDM/ExRED is presented by comparative performance among LDM/ExRED and the current existing load distribution models by simulations under variety of real traffic conditions.

## REFERENCES

- [1] L. Golubchik, J. Lui, T. Tung, A. Chow, W. Lee, G. Franceschinis, and C. Anglano, "Multi-path continuous media streaming: What are the benefits?," *Performance Evaluation*, vol. 49, pp. 429–449, Sep. 2002.
- [2] S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, "On Load Distribution over Multipath Networks," *IEEE Communications Surveys and Tutorials*, vol. 14, no. 3, pp. 662–680, 2012.
- [3] M. Menth, R. Martin, A. Koster, and S. Orlowski, "Overview of resilience mechanisms based on multipath structures," in *Proc. the 6<sup>th</sup> International Workshop on Design and Reliable Communication Networks (DRCN)*, La Rochelle, France, Oct. 2007.
- [4] N. Wang, K. Ho, G. Pavlou, and M. Howarth, "An overview of routing optimization for Internet traffic engineering," *IEEE Communications Surveys and Tutorials*, vol. 10, no. 1, pp. 36–56, 2008.
- [5] H. Adishesu, G. Parulkar, and G. Varghese, "A reliable and scalable striping protocol," *ACM SIGCOMM Computer Communication Review*, vol. 26, no. 4, pp. 131–141, Oct. 1996.
- [6] K. G. Shin and C. J. Hou, "Design and Evaluation of Effective Load Sharing in Distributed Real-Time Systems," *IEEE Trans. Parallel and Distributed Systems*, vol. 5, no. 7, pp. 704–719, Jul. 1994.
- [7] O. Kremien and J. Kramer, "Methodical analysis of adaptive load sharing algorithms," *IEEE Trans. Parallel Distributed Systems*, vol. 3, no. 6, pp. 747–760, Nov. 1992.
- [8] C. C. Hui and S. T. Chanson, "Hydrodynamic load balancing," *IEEE Trans. Parallel and Distributed Systems*, vol. 10, no. 11, pp. 1118–1137, Nov. 1999.
- [9] Z. Tari, J. Broberg, A. Zomaya, and R. Baldoni, "A least flow-time first load sharing approach for distributed server farm," *Journal of Parallel and Distributed Computing*, vol. 65, no. 7, pp. 832–842, Jul. 2005.
- [10] C. Villamizar, "OSPF optimized multipath (OSPF-OMP)," Internet draft draft-ietf-ospf-omp-02.txt, Feb. 1999.
- [11] D. Thaler and C. Hopps, "Multipath issues in unicast and multicast next-hop selection," RFC 2991, Nov. 2000.
- [12] A. Zinin, Cisco IP Routing, packet forwarding and intra-domain routing protocols. Reading, MA: Addison Wesley, 2002.
- [13] J. Song, S. Kim, M. Lee, H. Lee, and T. Suda, "Adaptive load distribution over multipath in MPLS networks," in *Proc. IEEE ICC*, Anchorage, Alaska, May 2003, pp. 233–237.
- [14] W. Shi, M. H. MacGregor, and P. Gburzynski, "Load balancing for parallel forwarding," *IEEE/ACM Trans. Networking*, vol. 13, no. 4, pp. 790–801, Aug. 2005.
- [15] S. Kandula, D. Katabi, S. Sinha, and A. Berger, "Dynamic load balancing without packet reordering," *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 2, pp. 53–62, Apr. 2007.
- [16] V. Paxson, G. Almes, J. Mahdavi, and M. Mathis, "Framework for IP performance metrics," RFC 2330, May. 1998.
- [17] C. Demichelis and P. Chimento, "IP packet delay variation metric for IP performance metrics (IPPM)," RFC 3393, Nov. 2002.
- [18] N. M. Piratla, A. P. Jayasumana, A. A. Bare, and T. Banka, "Reorder buffer-occupancy density and its application for measurement and evaluation of packet reordering," *Computer Communications*, vol. 30, no. 9, pp. 1980–1993, Jun. 2007.
- [19] N. M. Piratla and A. P. Jayasumana, "Reordering of packets due to multipath forwarding – An analysis," in *Proc. IEEE ICC*, Istanbul, Turkey, Jun. 2006, pp. 829–834.
- [20] A. Morton, L. Ciavattone, G. Ramachandran, S. Shalunov, and J. Perser, "Packet reordering metrics," RFC 4737, Nov. 2006.
- [21] A. Jayasumana, N. Piratla, T. Banka, and R. Whitner, "Improved packet reordering metrics," RFC 5236, Jun. 2008.
- [22] S. Jaiswal, G. Iannaccone, C. Diot, J. Kurose, and D. Towsley, "Measurement and classification of out-of-sequence packets in a tier-1 IP backbone," *IEEE/ACM Trans. Networking*, vol. 15, no. 1, pp. 54–66, Feb. 2007.
- [23] S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, "On the Performance Analysis of Traffic Splitting on Load Imbalancing and Packet Reordering of Bursty Traffic," in *Proc. IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC 2009)*, Beijing, China, Nov. 2009.
- [24] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.
- [25] S. Floyd and V. Jacobson, "On traffic phase effects in packet-switched gateways," *Internetworking: Research & Experience*, vol. 3, no. 3, Sep. 1992, pp. 115–56, UK.
- [26] S. Prabhavat, N. Kato, and R. Varakulsiripunth, "Load Distribution Mechanism for Multipath TCP Communication," in *Proc. International Symposium on Technology for Sustainability (ISTS 2011)*, Bangkok, Thailand, Jan. 2012, pp. 321–324.
- [27] S. Prabhavat and R. Varakulsiripunth, "Performance Improvement on RED Based Gateway in TCP Communication Network," in *Proc. International Conference on Control, Automation and Systems (ICCAS 2004)*, Bangkok, Thailand, Aug. 2004, pp. 782–787.
- [28] (Online Sources) P. Danzig, J. Mogul, V. Paxson, and M. Schwartz. (1995, Mar.). *The Internet Traffic Archive*. Available: <http://ita.ee.lbl.gov/index.html>.
- [29] (Online Sources) S. Floyd (1967, Nov.), "RED: Discussions of Setting Parameters," Available: <http://www.icir.org/floyd/REDparameters.txt>.
- [30] S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, "Effective Delay-Controlled Load Distribution over Multipath Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 22, no. 10, pp. 1730–1741, Oct. 2011.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ง

ผลงานส่งเผยแพร่ในงานประชุมวิชาการระดับนานาชาติ  
International Symposium on Technology for Sustainability (ISTS 2012)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Mark-Reverse Explicit Congestion Notification for Satellite IP Networks: Simulation and Analysis Results

Masato Inoue\*, Pakapol Tangdumrongrat\*\*, Sumet Prabhavat\*\*, Satoshi Utsumi\*, Salahuddin Muhammad Salim Zabir\*\*\* and Norio Shiratori\*\*\*\*

\*Tsuruoka National College of Technology, Yamagata, Japan

\*\*King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand

\*\*\*France Telecom Japan, Tokyo, Japan

\*\*\*\*Tohoku University/Waseda University, Sendai/Tokyo, Japan

e-mail: u-satoshi@tsuruoka-nct.ac.jp

**Abstract**—In this paper, we propose a new congestion notification scheme, named as Mark-Reverse ECN (Explicit Congestion Notification) for satellite IP networks. Simulation and analysis are conducted for performance evaluation. This analytical model takes the packet drop ratio and the packet mark ratio into account. We show that the results from the analytical model and the simulation match quite closely.

*Primary Categories* — Informatics.

*Secondary Categories* — Computer System/Network.

*Keywords* — ECN (Explicit Congestion Notification), Congestion Control, Satellite IP Network, Analytical Model.

## I. INTRODUCTION

Explicit Congestion Notification (ECN) has been proposed for the Internet as a means for detecting an incipient congestion before it actually occurs and notifying that explicitly to the corresponding end hosts. This has the potential to improve the network performance under congestion scenarios than conventional packet dropping mechanisms [1]. Random Early Detection (RED) with ECN gives definite improvement in time delays for interactive traffic. In [1], they examined use of a Backward Explicit Congestion Notification (BECN) mechanism to inform the sender of the congestion situation in the network. Because this mechanism uses ICMP Source Quenches (ISQs) for backward notification, the TCP senders would be attacked by the blind throughput-reduction attack [2].

In this paper, we propose a new congestion notification scheme, we name as Mark-Reverse ECN for satellite IP networks. Contrary to the conventional ECN mechanisms, rather than marking the packets in the forward direction, our proposed scheme marks the packets containing the ACKs in the reverse direction bound for the source nodes. The ACKs eventually notify congestion to the source nodes. As such, this mechanism leads to a faster notification of an incipient congestion.

The rest of this paper is organized as follows. In section II, we propose our novel marking scheme, Mark-Reverse ECN following a brief outline of existing schemes. In section III,

we evaluate our scheme by comparing with the conventional scheme through simulations. Section IV describes a new analytical model for ECN. Finally, we conclude this paper in section V.

## II. OUR PROPOSAL: MARK-REVERSE ECN

Before presenting our novel mechanism, we first provide a brief outline of the conventional marking schemes.

### A. Conventional ECN Schemes

Fig. 1 demonstrates Mark-Tail ECN [3]. At time  $t_1$ , packet 1 arrives at router R and suddenly causes the router to experience congestion. It will be marked before forwarded to its destination D. Source S will be notified about the congestion after this packet arrives at destination D and its ACK packet returns (at time  $t_2$ ). Notification delay is almost RTT (Round Trip Time).

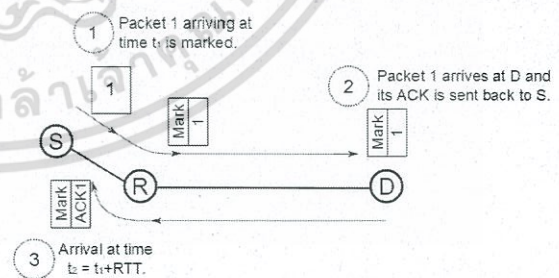


Fig. 1. Mark-Tail ECN Scheme.

### B. Mark-Reverse ECN

Fig. 2 shows how our proposal Mark-Reverse ECN works. At time  $t_1$ , packet 1 arrives at router R and suddenly causes the router to experience congestion. When an ACK packet of any previous packet arrives at the router, e.g., at time  $t_1^+$ ,

it will be marked before being forwarded to its destination (i.e., Source S). Assume that the ACK packet acknowledges the successful delivery of packet 0 which is previously sent at time  $t_0$ . Notification delay  $\Delta t = t_1^+ - t_1$  is very small.

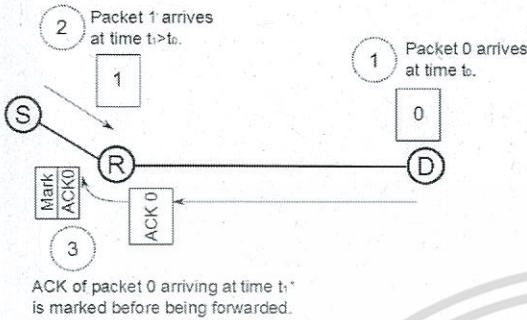


Fig. 2. Mark-Reverse ECN Scheme.

### III. SIMULATION

We use ns-2 [4] for our simulations with respect to the simulation scenario as shown in Fig. 3. We set the packet size 1,000bytes, the maximum window size 1000 segments, the capacity of the satellite link between senders and receivers 10Mbps (about 1,300 packets/sec), round trip time of the satellite link 550ms, and the link error ratio 0%. We changed the number of TCP flows. We set RED for ECN on the forwarding node of the satellite link. We observed the throughput, the packet drop ratio and the packet marking ratio. The simulation time is 500s. Simulation results are in Fig. 4, Fig. 5, and Fig. 6.

### IV. ANALYTICAL MODELS FOR ECN

Our proposed model is composed of four parts according to a typical short-lived flow evolution: the start of the connection (three-way-handshake), the initial slow-start phase, the first loss (or mark) part, and the subsequent losses or marks.

#### A. The Connection Start-Up Phase

Every TCP connection starts with the three-way-handshake process. Assuming that no ACK packets can get lost, this process can be well modeled as follows [5]:

$$E[T_{twhs}] = RTT + T_s \left( \frac{1-p}{1-2p} - 1 \right), \quad (1)$$

where  $T_s$  is the duration of SYN time-out and  $p$  is the packet loss ratio.

#### B. The Initial Slow-Start Phase

In order to derive the latency for this phase,  $E[Y^{SS}]$ , the expected number of packets sent until a loss or a mark occurs is given by following enhanced equation (based on the one given in [5]):

$$E[Y^{SS}] = \frac{(1 - (1 - (p + q))^d)(1 - (p + q))}{p + q}, \quad (2)$$

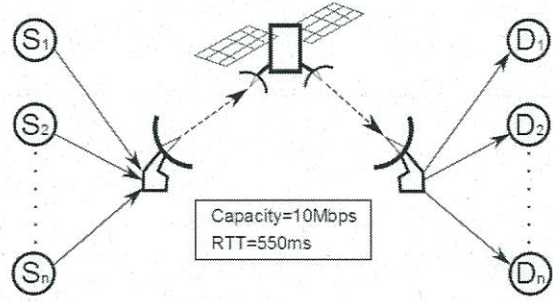


Fig. 3. Simulation scenario.

where  $d$  is the total file size measured in packets that must be transmitted and  $q$  is the packet marking ratio.

Substituting the value of  $E[Y^{SS}]$  in the Equation (2), we get the expected size of congestion window at the end of the low-start phase due to the packet losses or the packet marks:

$$E[W^{SS}] = \frac{(1 - (1 - (p + q))^d)(1 - (p + q)) + 2(p + q)}{(p + q)g^2}, \quad (3)$$

where  $g = 1.61804$  [5].

The expected slow-start latency is calculated as follows [5]:

$$E[n] = \begin{cases} \lceil \log_g \left( \frac{W_m}{C_1} \right) \rceil \\ + \frac{1}{W_m} (E[Y^{SS}] - g^2 W_m - 2), \text{ when } E[W^{SS}] > W_m \\ \lceil \log_g \left( \frac{E[Y^{SS}] + 2}{C_1} \right) \rceil - 2, \text{ when } E[W^{SS}] \leq W_m \end{cases} \quad (4)$$

#### C. The First Loss

Assume that the first indication to detect congestion is a packet loss. In the round where a TD (triple-duplicate) occurs, let  $W^{SS}$  be the current size of  $cwnd$ , which has a value  $w$ . In this round,  $w$  packets were sent. Among them,  $k$  packets are assumed to be ACKed. Since the connection is still in the slow-start phase,  $cwnd$  increases to  $w + k$  and another  $2k$  packets are sent in the next round. If more than three packets from these  $2k$  packets get ACKed, then a TD would occur; otherwise, a TO would take place. Let

$$A(w, k) = \frac{(1-p)^k p}{1 - (1-p)^w}, \quad (5)$$

be the probability that the first  $k$  packets have been successfully transmitted and ACKed in a round of  $w$  packets, provided that there might be one or more packets got lost. Now, let

$$h(m) = \sum_{i=0}^2 (1-p)^i p, \text{ for } m \geq 3, \quad (6)$$

be the probability that no more than 2 packets have been transmitted successfully in a round of  $m$  packets. We the

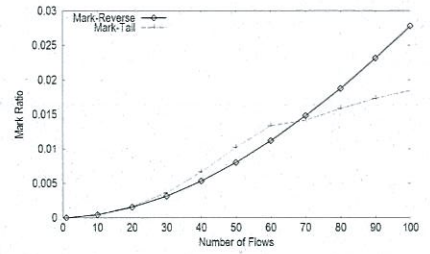
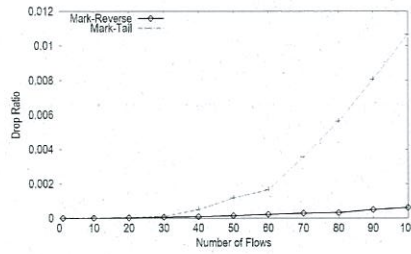
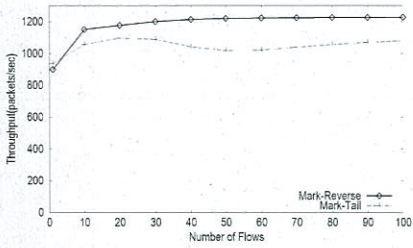


Fig. 4. Throughput comparisons between Mark-Reverse ECN and Mark-Tail ECN.(simulation results)

Fig. 5. Drop ratio comparisons between Mark-Reverse ECN and Mark-Tail ECN.(simulation results)

Fig. 6. Mark ratio comparisons between Mark-Reverse ECN and Mark-Tail ECN.(simulation results)

obtain  $Q(w)$ , the probability that a loss in a window size  $w$  is a TO (Timeout), is given by

$$Q(w) = \begin{cases} 1 & , \text{when } w \leq 3 \\ \sum_{k=0}^2 A(w, k) + \sum_{k=3}^{w-1} A(w, k)h(k) & , \text{otherwise} \end{cases} \quad (7)$$

The expected time that TCP spends in the RTOs is given in [7] as:

$$E[Z^{TO}] = T_0 \frac{f(p)}{1-p} \quad (8)$$

where  $T_0$  is the average duration of the first time-out, and

$$f(p) = 1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + 32p^6 \quad (9)$$

Finally, the expected latency that this loss would incur is

$$T_{loss} = \frac{p}{p+q} \times (1 - (1 - (p+q))^d) \times (Q(W^{SS})E[Z^{TO}] + (1 - Q(W^{SS}))RTT) \quad (10)$$

#### D. Sending the Rest of the Packets

After the first packet loss or the first packet mark, the transmission latency of the rest ( $d - E[Y_{init}]$ ) packets is obtained as follows:

$$T_{rest} = \frac{d - E[Y^{SS}]}{H} \quad (11)$$

where  $H$  is the throughput of a steady-state flow.

The expected value of the number of packets sent in a round when a packet loss occurs,  $E[Y]$ , is

$$E[Y] = E[\alpha] + E[W^{TDM}] - 1, \quad (12)$$

where  $E[\alpha]$  is the expected number of packets sent in a TDMP (triple-duplicate or marking period) up to and including the first packet that is lost or marked. Given our loss and mark model, the probability that  $\alpha_i = k$  is equal to the probability that exactly  $k - 1$  packets are successfully acknowledged before a loss or a mark occurs is

$$P[\alpha = k] = (1 - (p+q))^{k-1} (p+q), k = 1, 2, \dots \quad (13)$$

The mean of  $\alpha$  is thus,

$$E[\alpha] = (1 - (p+q))^{k-1} (p+q)k = \frac{1}{p+q} \quad (14)$$

From (12) and (14) it follows that

$$E[Y] = \frac{1 - (p+q)}{p+q} + E[W^{TDM}] \quad (15)$$

From [8] the expected round where the first loss or the first mark occurs,  $E[X]$ , is

$$E[X] = b \left( \frac{E[W^{TDM}]}{2} + 1 \right) \quad (16)$$

From [8] the expected value of the number of packets sent in a round when a packet loss occurs,  $E[Y]$ , is

$$E[Y] = \frac{E[X]}{2} \left( \frac{E[W^{TDM}]}{2} + E[W^{TDM}] - 1 \right) + E[\beta] \quad (17)$$

Then, from Equations (15) and (17), we get the following expression which involves  $E[W^{TDM}]$ :

$$\begin{aligned} & \frac{1 - (p+q)}{p+q} + E[W^{TDM}] \\ &= \frac{E[X]}{2} \left( \frac{E[W^{TDM}]}{2} + E[W^{TDM}] - 1 \right) + E[\beta]. \end{aligned} \quad (18)$$

For simplicity, we assume  $\beta_i$  to be uniformly distributed between 1 and  $W_i - 1$ . Thus,  $E[\beta] = E[W^{TDM}]/2$ .

Combining (16) and (18), we get:

$$\begin{aligned} E[W^{TDM}] &= \\ &= \frac{2(b-1)}{3b} + \sqrt{\frac{4(b(p+q)+2(1-(p+q)))}{3b(p+q)} + \left(\frac{2(b-1)}{3b}\right)^2}. \end{aligned} \quad (19)$$

From [8], the expected value of duration of a round,  $E[A]$ , is

$$E[A] = RTT \left( \frac{p}{p+q} (E[X] + 1) + \frac{q}{p+q} E[X] \right) \quad (20)$$

### E. Impact of Window Limitation

At the beginning of TCP flow establishment, the receiver advertises a maximum buffer size which determines a maximum congestion window size,  $W_m$ . As a consequence, during a period without loss indications, the window size can grow up to  $W_m$ , but will not grow further beyond this value. During the first TDMP, the window grows linearly up to  $W_m$  for  $U$  rounds, then remains constant for  $V$  rounds, and then a triple-duplicate or marking indication occurs. The expected  $U$  rounds,  $E[U]$ , is as follows [7]:

$$E[U] = \frac{b}{2} W_m. \quad (21)$$

Since  $Y_i$ , the number of packets in the  $i$ th TDMP, does not depend on window limitation,  $E[Y]$  is given by (12):

$$E[Y] = E[\alpha] + W_m - 1. \quad (22)$$

From [7],

$$E[Y] = \frac{3}{4} W_m E[U] + W_m E[V]. \quad (23)$$

Then,

$$\frac{1 - (p + q)}{p + q} + W_m = \frac{3b}{8} W_m^2 + W_m E[V]. \quad (24)$$

Thus,

$$E[V] = \frac{1 - (p + q)}{(p + q)W_m} + 1 - \frac{3b}{8} W_m. \quad (25)$$

Finally, since  $X_i = U_i + V_i$  in  $i$ th TDMP, we have

$$E[X] = E[U] + E[V] = \frac{b}{8} W_m + \frac{1 - (p + q)}{(p + q)W_m} + 1. \quad (26)$$

The expected number of packets that have been received in a TDMP,  $E[Y']$ , is

$$\begin{aligned} E[Y'] &= \frac{p}{p+q} (E[\alpha] + E[\beta] - 1.0) \\ &\quad + \frac{q}{p+q} (E[\alpha] + E[W^{TDM}] - E[\beta] - 1.0) \\ &= E[\alpha] + \frac{E[W^{TDM}]}{2} - 1.0. \end{aligned} \quad (27)$$

The expected number of packets received in the time-out phase,

$$E[R] = 1. \quad (28)$$

Thus, the throughput can be formulated as:

$$H = \frac{E[Y'] + \frac{p}{p+q} Q(W^{TDM}) * E[R]}{E[A] + \frac{p}{p+q} Q(W^{TDM}) * E[Z^{TO}]}. \quad (29)$$

From [5], the total expected latency is

$$T_{latency} = E[T_{twhs}] + E[n]RTT + T_{loss} + T_{rest} - \frac{RTT}{2}. \quad (30)$$

Here, the delay caused by the delayed acknowledgment for the first packet is 0ms. That is, the option of delayed acknowledgment is off.

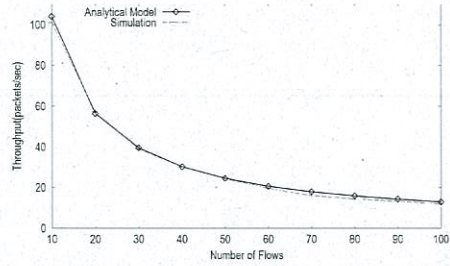


Fig. 7. Throughput comparisons between analytical model and simulation for Mark-Reverse ECN.

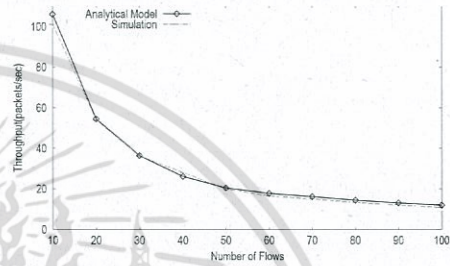


Fig. 8. Throughput comparisons between analytical model and simulation for Mark-Tail ECN.

### F. Verification

Fig. 7 and Fig. 8 show the results of this analytical model comparing with simulation results.

## V. CONCLUSION

In this paper, we propose a novel congestion notification scheme, that we name as Mark-Reverse ECN for satellite IP networks. Simulation results show that Mark-Reverse can ensure higher throughput and lower packet drop ratio. In addition, we propose an analytical model for ECN. Analytical results also verify that Mark-Reverse ECN yields better performance than conventional or Mark-Tail mechanisms.

## REFERENCES

- [1] F. Akujobi, I. Lambadaris, R. Makkar, N. Seddigh, and B. Nandy, "BECN for Congestion Control in TCP/IP Networks: Study and Comparative Evaluation," *IEEE Globecom 2002*, Nov. 2002.
- [2] RFC 5927, "ICMP Attacks against TCP," July 2010.
- [3] S. Floyd, "TCP and Explicit Congestion Notification," *Computer Communications Review*, vol. 24, no. 5, Oct. 1994.
- [4] UCB/LBNL/VINT Network Simulator (NS).
- [5] D. Zheng, G. Y. Lazarou, and R. Hu, "A Stochastic Model for Short-lived TCP Flows," *IEEE International Conference on Communications 2003 (ICC 2003)*, May 2003, pp.11-15.
- [6] N. Cardwell, S. Savage, and T. Anderson, "Modeling TCP Latency," in *Proc. IEEE INFOCOM 2000*, vol. 3, Mar. 2000, pp.1742-1751.
- [7] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, "Modeling TCP Reno Performance: A Simple Model and Its Empirical Validation," *IEEE/ACM Trans. Networking*, vol. 8, no. 2, pp.133-145, Apr. 2000.
- [8] D. Zheng, "On the modeling of TCP latency and throughput," Master's thesis Mississippi State University, 2002.

## ภาคผนวก จ

ผลงานส่งเผยแพร่ในรายงานทางเทคนิค  
IEICE Technical Report 2012



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Mark-Reverse Explicit Congestion Notification: Concept, Analysis and Evaluation

Masato INOUE<sup>†</sup>, Pakapol TANGDUMRONGRAT<sup>††</sup>, Sumet PRABHAVAT<sup>††</sup>, Satoshi UTSUMI<sup>†</sup>,  
Salahuddin MUHAMMAD SALIM ZABIR<sup>†††</sup>, and Norio SHIRATORI<sup>††††</sup>

<sup>†</sup> Tsuruoka National College of Technology 104 Sawada, Inooka, Tsuruoka, Yamagata, 997-8511 Japan

<sup>††</sup> King Mongkut's Institute of Technology Ladkrabang Chalongkrung Rd., Ladkrabang, Bangkok 10520,  
Thailand

<sup>†††</sup> France Telecom Japan Co., Ltd Keio Shinjuku Oiwake Bldg. 9F, 3-1-13 Shinjuku, Shinkjuku-ku,  
Tokyo, 160-0022 Japan

<sup>††††</sup> Tohoku University 2-1-1 Katahira, Aoba-ku, Sendai, 980-8577 Japan

**Abstract** In this paper, we propose a new congestion notification scheme, named as Mark-Reverse ECN (Explicit Congestion Notification) for satellite IP networks. When congestion is beginning to occur on a link, our scheme notifies a source node more quickly than a conventional scheme, i.e., Mark-Tail ECN. Instead of a packet on forwarding path, we marks an acknowledge (ACK) packet on backwarding path which is heading for the source node. We expect that latency time for congestion notification will be reduced and congestion control will thus work more accurately and effectively. Simulation and analysis are conducted for performance evaluation. Through simulation, we compare and evaluate performance of Mark-Reverse ECN and Mark-Tail ECN. Mark-Reverse ECN shows high throughput and low packet drop ratio in satellite IP networks. In analysis, we propose a new analytical model of TCP NewReno supporting ECN. This analytical model takes the packet drop ratio and the packet mark ratio into account for computing delay time and throughput. Analytical results also show superiority of our scheme, correspondingly. A considerably small discrepancy between the results verify our simulation and analysis.

**Key words** ECN (Explicit Congestion Notification), Congestion Control, Satellite IP Network, Analytical Model.

## 1. Introduction

Explicit Congestion Notification (ECN) was proposed for the Internet as a means of explicitly notifying end-hosts of network congestion by marking, instead of dropping packets [1]. Random Early Detection (RED) [2] with ECN gives definite improvement in time delays for interactive traffic. In [1], they examined use of a Backward Explicit Congestion Notification (BECN) mechanism to inform the sender of the congestion situation in the network. Because this mechanism uses ICMP Source Quenches (ISQs) for backward notification, the TCP senders would be attacked by the blind throughput-reduction attack [3].

In this paper, we propose a new congestoin notification scheme, named as Mark-Reverse ECN for satellite IP networks. Our proposal, i.e., Mark-Reverse ECN is a scheme

when congestion is beginning to occur on the link. The ACKs notifies congestion to the distination node.

The rest of this paper is organized as follows: In section 2., we describe a conventional ECN scheme, i.e., Mark-Tail ECN. In section 3., we propose a new ECN scheme, i.e., Mark-Reverse ECN. In section 4., we evaluate our scheme comparing with the conventional scheme through simulations. Section 5. describes a new analytical model for ECN. Finally, section 6. concludes ths paper.

## 2. Conventional Scheme: Mark-Tail ECN [4]

Fig. 1 demonstrates Mark-Tail ECN [4]. At time  $t_1$ , packet 1 arrives at router R and suddenly causes the router to experience congestion. It will be marked before forwarded to its destination. Source S will be notified about the congestion after this packet arrives at destination D and its ACK packet returns (at time  $t_2$ ). Notification delay is almost RTT.

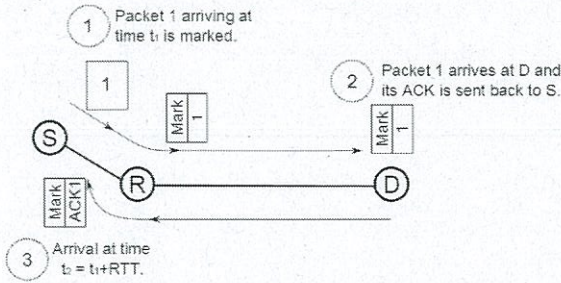


Fig. 1 Mark-Tail ECN Scheme.

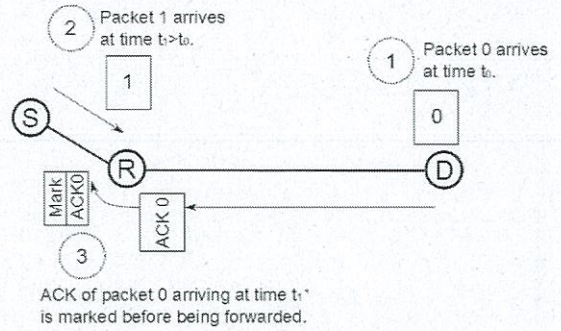


Fig. 2 Mark-Reverse ECN Scheme.

### 3. Our Proposal: Mark-Reverse ECN

Our proposal Mark-Reverse ECN scheme needs Mark-Reverse ECN-Capable Transport (MR-ECT) bit on IP packets. (MR-ECT bit should not be on at once with ECN-Capable Transport (ECT) bit.) Also, Mark-Reverse ECN scheme needs Congestion Experienced (CE) bit on IP packets. According to Random Early Detection (RED) mechanisms on the queue of the positive direction, the CE bit on the packet whose MR-ECT bit is on in the reverse direction of the same link supporting Mark-Reverse ECN becomes on when the packet is passing in the reverse direction. Mark-Tail ECN scheme and Mark-Front ECN scheme [5] need about RTT to notify congestion to the source node. Mark-Reverse ECN scheme needs about 0 msec to notify congestion to the source node, if the congestion link is near the source node. Because of the fast notification, the number of packet losses due to congestion can be minimized. Especially, it is very useful when the bottleneck link is satellite one. Because Mark-Reverse ECN scheme uses the packets of the same connection for congestion notification differing with Backward Explicit Congestion Notification (BECN), Mark-Reverse ECN scheme can avoid the blind throughput-reduction attack. That is, ICMP Source Quench (ISQ) message of BECN can be transmitted from an arbitrary node to the source node, the Mark-Reverse ECN is notified only from the nodes passed through by the flow using the connection. Moreover, there is no overhead by notifying messages as ISQ messages.

Fig. 2 shows how our proposal Mark-Reverse ECN works. At time  $t_1$ , packet 1 arrives at router R and suddenly causes the router to experience congestion. When an ACK packet of any previous packet arrives at the router, e.g., at time  $t_1^+$ , it will be marked before being forwarded to its destination (i.e., Source S). Assume that the ACK packet acknowledges the successful delivery of packet 0 which is previously sent at time  $t_0$ . Notification delay  $\Delta t = t_1^+ - t_1$  is very small.

### 4. Simulation

#### 4.1 Evaluation of Throughput, Drop ratio and Mark ratio

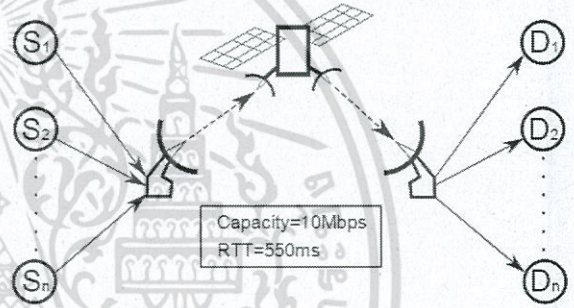


Fig. 3 Simulation scenario.

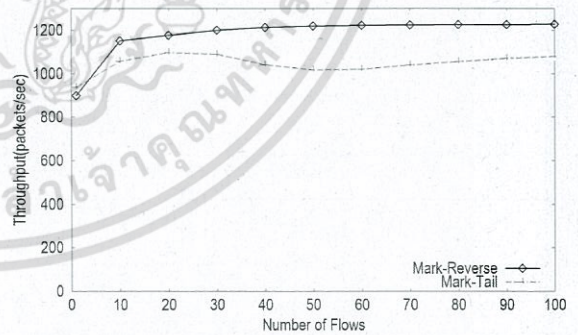


Fig. 4 Throughput comparisons between Mark-Reverse ECN and Mark-Tail ECN. (simulation results)

We use ns-2 [6] for network simulations. We evaluate throughput, drop ratio and mark ratio in the simulation scenario as shown in Fig. 3. We set the packet size 1,000bytes, the maximum window size 1,000 segments, the capacity of the satellite link between senders and receivers 10Mbps (about 1,300 packets/sec), round trip time of the satellite link 550ms, and the link error ratio 0%. We changed the number of TCP flows. We set RED for ECN on the forwarding node of the satellite link. We observed the throughput,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น มิใช่เพื่อเผยแพร่ไปยังบุคคลอื่นโดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

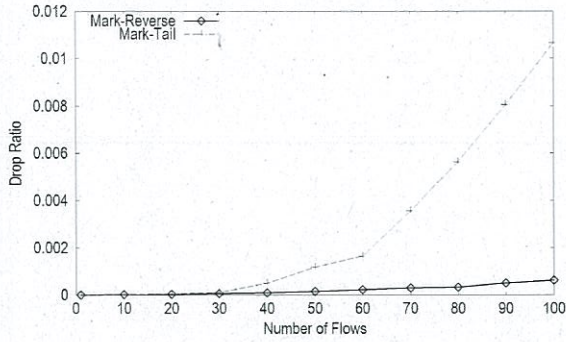


Fig. 5 Drop ratio comparisons between Mark-Reverse ECN and Mark-Tail ECN. (simulation results)

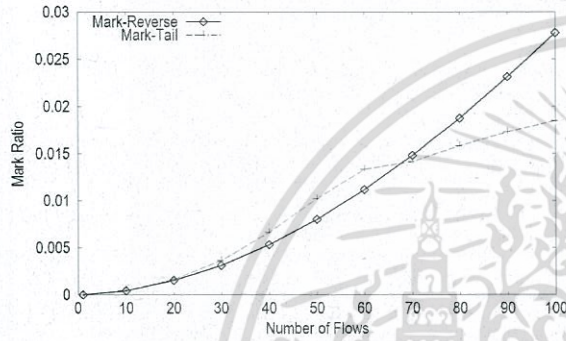


Fig. 6 Mark ratio comparisons between Mark-Reverse ECN and Mark-Tail ECN. (simulation results)

the packet drop ratio and the packet mark ratio. The simulation time is 500s. Simulation results are in Fig. 4, Fig. 5, and Fig. 6.

## 5. Analytical Models for ECN

Our proposed model is composed of four parts according to a typical short-lived flow evolution: the start of the connection (three-way-handshake), the initial slow-start phase, the first loss (or mark) part, and the subsequent losses or marks.

### 5.1 The Connection Start-Up Phase

Every TCP connection starts with the three-way-handshake process. Assuming that no ACK packets can get lost, this process can be well modeled as follows [7]:

$$E[T_{twhs}] = RTT + T_s \left( \frac{1-p}{1-2p} - 1 \right), \quad (1)$$

where  $T_s$  is the duration of SYN time-out and  $p$  is the packet loss ratio.

### 5.2 The Initial Slow-Start Phase

In order to derive the latency for this phase,  $E[Y^{SS}]$ , the expected number of packets sent until a loss or a mark occurs is given by following enhanced equation (based on the one given in [7]):

$$E[Y^{SS}] = \frac{(1 - (1 - (p + q))^d)(1 - (p + q))}{p + q}, \quad (2)$$

where  $d$  is the total file size measured in packets that must be transmitted and  $q$  is the packet mark ratio.

Substituting the value of  $E[Y^{SS}]$  in the Equation (2), we get the expected size of congestion window at the end of the low-start phase due to the packet losses or the packet marks:

$$E[W^{SS}] = \frac{(1 - (1 - (p + q))^d)(1 - (p + q)) + 2(p + q)}{(p + q)g^2}, \quad (3)$$

where  $g = 1.61804$  [7].

The expected slow-start latency is calculated as follows [7]:

$$E[n] = \begin{cases} \left[ \log_g \left( \frac{W_m}{C_1} \right) \right] + \frac{1}{W_m} (E[Y^{SS}] - g^2 W_m - 2) \\ \quad , \text{ when } E[W^{SS}] > W_m \\ \left[ \log_g \left( \frac{E[Y^{SS}] + 2}{C_1} \right) \right] - 2 \\ \quad , \text{ when } E[W^{SS}] \leq W_m \end{cases} \quad (4)$$

### 5.3 The First Loss

Assume that the first indication to detect congestion is a packet loss. In the round where a TD (triple-duplicate) occurs, let  $W^{SS}$  be the current size of *cwnd*, which has a value  $w$ . In this round,  $w$  packets were sent. Among them,  $k$  packets are assumed to be ACKed. Since the connection is still in the slow-start phase, *cwnd* increases to  $w + k$  and another  $2k$  packets are sent in the next round. If more than three packets from these  $2k$  packets get ACKed, then a TD would occur; otherwise, a TO would take place. Let

$$A(w, k) = \frac{(1-p)^k p}{1 - (1-p)^w}, \quad (5)$$

be the probability that the first  $k$  packets have been successfully transmitted and ACKed in a round of  $w$  packets, provided that there might be one or more packets got lost. Now, let

$$h(m) = \sum_{i=0}^2 (1-p)^i p, \quad \text{for } m \geq 3, \quad (6)$$

be the probability that no more than 2 packets have been transmitted successfully in a round of  $m$  packets. As [7], we obtain  $Q^{SS}(W^{SS})$ , the probability that a loss in a window size  $W^{SS}$  of slow-start phase is a TO (Timeout), is given by

$$Q^{SS}(W^{SS}) = \begin{cases} 1 & , \text{ when } w \leq 2 \\ \sum_{k=0}^1 A(W^{SS}, k) \\ + \sum_{k=2}^{w-1} A(W^{SS}, k) h(2k) & , \text{ otherwise} \end{cases} \quad (7)$$

The expected time that TCP spends in the RTOs is given in [9] as:

$$E[Z^{TO}] = T_0 \frac{f(p)}{1-p}, \quad (8)$$

where  $T_0$  is the average duration of the first time-out, and

$$f(p) = 1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + 32p^6, \quad (9)$$

Finally, the expected latency that this loss would incur is

$$T_{loss} = \frac{p}{p+q} (1 - (1 - (p+q))^d) \times (Q(W^{SS})E[Z^{TO}] + (1 - Q(W^{SS}))RTT). \quad (10)$$

#### 5.4 Sending the Rest of the Packets

After the first packet loss or the first packet mark, the transmission latency of the rest  $(d - E[Y_{init}])$  packets is obtained as follows:

$$T_{rest} = \frac{d - E[Y^{SS}]}{H}, \quad (11)$$

where  $H$  is the throughput of a steady-state flow.

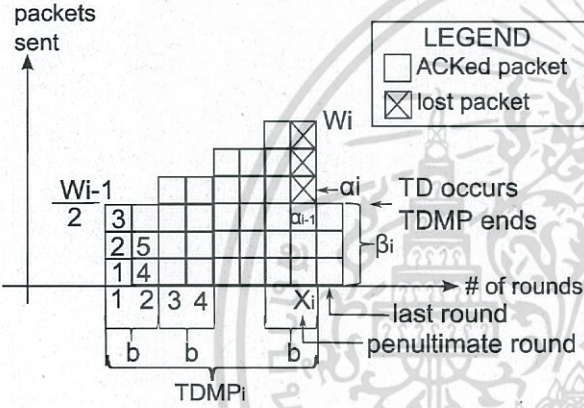


Fig. 7 Packets sent during a TDMP when a packet loss occurs.

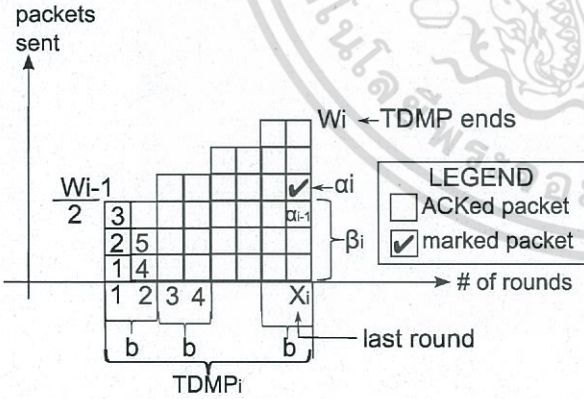


Fig. 8 Packets sent during a TDMP when a marking occurs.

We define TDMP (triple-duplicate or marked period) to be the period between tow triple-duplicate (TD) losses or receipts of marked ACKs. With reference to Fig. 7 when a packet loss occur, and with reference to Fig. 8 when receiving a marked ack, the following enhanced equations are given baesd on [8].

$$Y_i = \begin{cases} \alpha_i + W_i^{TDM} - 1, & \text{when a packet loss} \\ \alpha_i = (\beta_i + 1) + W_i^{TDM}, & \text{when a marked ack} \end{cases} \quad (12)$$

$$A_i = \begin{cases} \sum_{j=1}^{X_i+1} r_{ij}, & \text{when a packet loss} \\ \sum_{j=1}^{X_i} r_{ij}, & \text{when a marked ack} \end{cases} \quad (13)$$

$$W_i^{TDM} = \frac{W_{i-1}^{TDM}}{2} + \frac{X_i}{b} - 1, \quad (14)$$

$$Y_i = \begin{cases} \frac{X_i}{2} \left( \frac{W_{i-1}^{TDM}}{2} + W_i^{TDM} - 1 \right) + \beta_i, & \text{when a packet loss} \\ \frac{X_i}{2} \left( \frac{W_{i-1}^{TDM}}{2} + W_i^{TDM} - 1 \right), & \text{when a marked ack} \end{cases} \quad (15)$$

where  $X_i$  is the penultimate round in the TDMP which experiences packet losses,  $r_{ij}$  is the round trip tim,  $W_i^{TDM}$  is the window size at the end of a TDMP,  $\alpha_i$  is the number of packets sent in a TDMP until the first loss or the first receipt of a marked ACK happens, and  $\beta_i$  is the number of packets sent before the loss or the receipt of a marked ack in the round.

The expected value of the number of packets sent in a round when a packet loss occurs,  $E[Y]$ , is

$$E[Y] = \begin{cases} E[\alpha] + E[W^{TDM}] - 1, & \text{when a packet loss} \\ E[\alpha] - E[\beta] - 1 + E[W^{TDM}], & \text{when a marked ack} \end{cases} \quad (16)$$

where  $E[\alpha]$  is the expected number of packets sent in a TDMP up to and including the first packet that is lost or marked. Given our loss and mark model, the probability that  $\alpha_i = k$  is equal to the probability that exactly  $k - 1$  packets are successfully acknowledged before a loss or a mark occurs is

$$P[\alpha = k] = (1 - (p+q))^{k-1} (p+q), \quad k = 1, 2, \dots \quad (17)$$

The mean of  $\alpha$  is thus,

$$E[\alpha] = (1 - (p+q))^{k-1} (p+q)k = \frac{1}{p+q}. \quad (18)$$

From (16) and (18) it follows that

$$E[Y] = \begin{cases} \frac{1-(p+q)}{p+q} + E[W^{TDM}], & \text{when a packet loss} \\ \frac{1-(p+q)}{p+q} - E[\beta] + E[W^{TDM}], & \text{when a marked ack} \end{cases} \quad (19)$$

From [10] the expected round where the first loss or the first mark occurs,  $E[X]$ , is

$$E[X] = b \left( \frac{E[W^{TDM}]}{2} + 1 \right). \quad (20)$$

From [10] the expected value of the number of packets sent in a round when a packet loss occurs,  $E[Y]$ , is

$$E[Y] = \begin{cases} \frac{E[X]}{2} \left( \frac{E[W^{TDM}]}{2} + E[W^{TDM}] - 1 \right) + E[\beta], & \text{when a packet loss} \\ \frac{E[X]}{2} \left( \frac{E[W^{TDM}]}{2} + E[W^{TDM}] - 1 \right), & \text{when a marked ack} \end{cases} \quad (21)$$

Then, from Equations (20), (19) and (21), we get the following expression which involves  $E[W^{TDM}]$ :

$$\frac{1 - (p + q)}{p + q} + E[W^{TDM}] = \frac{E[X]}{2} \left( \frac{E[W^{TDM}]}{2} + E[W^{TDM}] - 1 \right) + E[\beta]. \quad (22)$$

For simplicity, we assume  $\beta_i$  to be uniformly distributed between 1 and  $W_i - 1$ . Thus,  $E[\beta] = E[W^{TDM}]/2$ .

Combining (20) and (22), we get:

$$E[W^{TDM}] = -\frac{2(b-1)}{3b} + \sqrt{\frac{4(b(p+q) + 2(1-(p+q)))}{3b(p+q)} + \left(\frac{2(b-1)}{3b}\right)^2}. \quad (23)$$

From [10], the expected value of duration of a round,  $E[A]$ , is

$$E[A] = \begin{cases} RTT \cdot (E[X] + 1) & , \text{when a packet loss} \\ RTT \cdot E[X] & , \text{when a marked ack} \end{cases} \quad (24)$$

That is,

$$E[A] = \frac{p}{p+q} RTT \cdot (E[X] + 1) + \frac{q}{p+q} RTT \cdot E[X] \quad (25)$$

### 5.5 Impact of Window Limitation

At the beginning of TCP flow establishment, the receiver advertises a maximum buffer size which determines a maximum congestion window size,  $W_m$ . As a consequence, during a period without loss indications, the window size can grow up to  $W_m$ , but will not grow further beyond this value. During the first TDMP, the window grows linearly up to  $W_m$  for  $U$  rounds, then remains constant for  $V$  rounds, and then a triple-duplicate or marking indication occurs. The expected  $U$  rounds,  $E[U]$ , is as follows [9]:

$$E[U] = \frac{b}{2} W_m. \quad (26)$$

Since  $Y_i$ , the number of packets in the  $i$ th TDMP, does not depend on window limitation,  $E[Y]$  is given by (16):

$$E[Y] = E[\alpha] + W_m - 1. \quad (27)$$

From [9],

$$E[Y] = \frac{3}{4} W_m E[U] + W_m E[V]. \quad (28)$$

Then,

$$\frac{1 - (p + q)}{p + q} + W_m = \frac{3b}{8} W_m^2 + W_m E[V]. \quad (29)$$

Thus,

$$E[V] = \frac{1 - (p + q)}{(p + q)W_m} + 1 - \frac{3b}{8} W_m. \quad (30)$$

Finally, since  $X_i = U_i + V_i$  in  $i$ th TDMP, we have

$$E[X] = E[U] + E[V] = \frac{b}{8} W_m + \frac{1 - (p + q)}{(p + q)W_m} + 1. \quad (31)$$

The expected number of packets that have been received in a TDMP,  $E[Y']$ , is

$$E[Y'] = \begin{cases} E[\alpha] + E[\beta] - 1.0 & , \text{when a packet loss} \\ E[\alpha] + E[W^{TDM}] - E[\beta] - 1.0 & , \text{when a marked ack} \end{cases} \quad (32)$$

That is,

$$E[Y'] = \frac{p}{p+q} (E[\alpha] + E[\beta] - 1.0) + \frac{q}{p+q} (E[\alpha] + E[W^{TDM}] - E[\beta] - 1.0) = E[\alpha] + \frac{E[W^{TDM}]}{2} - 1.0. \quad (33)$$

is

As [9], we obtain  $Q(w)$ , the probability that a loss in a window size  $w$  of congestion-avoidance phase is a TO, is given by

$$Q(w) = \begin{cases} 1 & , \text{when } w \leq 3 \\ \sum_{k=0}^2 A(w, k) + \sum_{k=3}^w A(w, k) h(k) & \\ , \text{otherwise} & \end{cases} \quad (34)$$

The expected number of packets received in the time-out phase,

$$E[R] = 1. \quad (35)$$

Thus, the throughput can be formulated as:

$$H = \frac{E[Y'] + \frac{p}{p+q} Q(W^{TDM}) * E[R]}{E[A] + \frac{p}{p+q} Q(W^{TDM}) * E[Z^{TO}]}. \quad (36)$$

From [7], the total expected latency is

$$T_{latency} = E[T_{whs}] + E[n]RTT + T_{loss} + T_{rest} - \frac{RTT}{2}. \quad (37)$$

Here, the delay caused by the delayed acknowledgment for the first packet is 0ms. That is, the option of delayed acknowledgment is off.

### 5.6 Verification

We evaluate our proposal Mark-Reverse ECN scheme by the analytical model of Section 5. and by the network simulator ns-2 using the parameters used and given in the Section 4.1. Fig. 11 and Fig. 12 show that the results of this analytical model are very close to simulation results.

## 6. Conclusion

In this paper, we proposed a new congestion notification scheme, named as Mark-Reverse ECN for satellite IP networks. Simulation results show that our scheme achieves

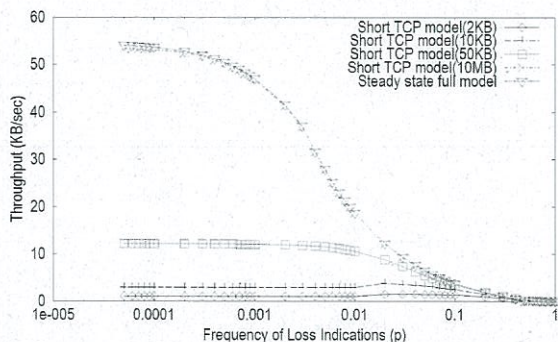


Fig. 9 Throughput predictions given by the short TCP model and the steady state model when drop ratio  $p$  is changing and mark ratio is constant  $q = 0$ . The conditions are:  $RTT = 550ms$ ,  $MSS = 1,000bytes$ ,  $T_S = T_0 = 1sec$ ,  $W_m = 30segments$ ,  $b = 1$ .

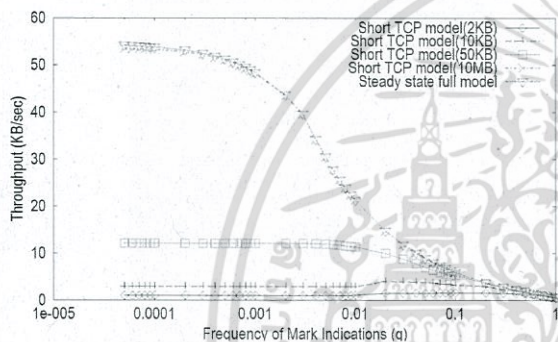


Fig. 10 Throughput predictions given by the short TCP model and the steady state model when drop ratio is constant  $p = 0$  and mark ratio is changing. The conditions are:  $RTT = 550ms$ ,  $MSS = 1,000bytes$ ,  $T_S = T_0 = 1sec$ ,  $W_m = 30segments$ ,  $b = 1$ .

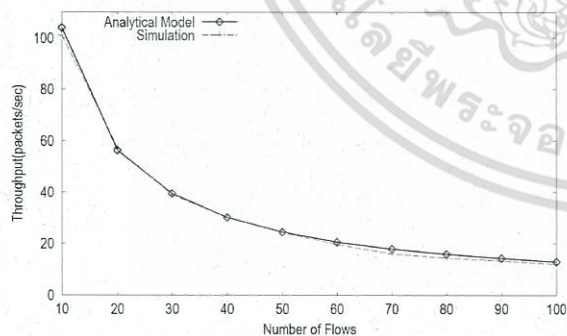


Fig. 11 Throughput comparisons between analytical model and simulation for Mark-Reverse ECN.

higher throughput than the conventional scheme. Analytical results also shows our superiority, correspondingly. A good match between analysis and simulation curves can be observed and validates our results.

#### REFERENCES

[1] F. Akujobi, I. Lambadaris, R. Makkar, N. Seddigh, and B. Nandy, "BECN for Congestion Control in TCP/IP Networks: Study and Comparative Evaluation", *IEEE*

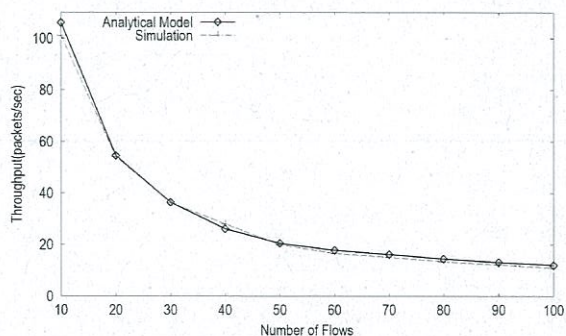


Fig. 12 Throughput comparisons between analytical model and simulation for Mark-Tail ECN.

*Globecom 2002*, Nov. 2002.

[2] S. Floyd, V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance", *IEEE/ACM Trans. Networking*, vol.1, no.4, August 1993.  
 [3] RFC 5927, "ICMP Attacks against TCP", July 2010.  
 [4] S. Floyd, "TCP and Explicit Congestion Notification", *Computer Communications Review*, vol. 24, no. 5, Oct. 1994.  
 [5] C. Liu, R. Jain, "Improving Explicit Congestion Notification with the Mark-Front Strategy", *Computer Networks* (elsevier), vol.35, issue 2-3, Feb. 2001.  
 [6] UCB/LBNL/VINT Network Simulator (NS).  
 [7] D. Zheng, G. Y. Lazarou, and R. Hu, "A Stochastic Model for Short-lived TCP Flows", *IEEE International Conference on Communications 2003 (ICC 2003)*, May 2003, pp.11-15.  
 [8] N. Cardwell, S. Savage, and T. Anderson, "Modeling TCP Latency", in *Proc. IEEE INFOCOM 2000*, vol. 3, Mar. 2000, pp.1742-1751.  
 [9] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, "Modeling TCP Reno Performance: A Simple model and its Empirical Validation", *IEEE/ACM Trans. Networking*, vol. 8, no. 2, pp.133-145, Apr. 2000.  
 [10] D. Zheng, "On the modeling of TCP latency and throughput", Master's thesis Mississippi State University, 2002.

## ข้อมูลประวัติคณะผู้วิจัย

ประวัติส่วนตัว

ชื่อ-สกุล.....สุเมธ ประภาวัต.....  
 ตำแหน่งปัจจุบัน.....ผู้ช่วยศาสตราจารย์.....

ประวัติการศึกษา

ชื่อย่อปริญญา	สาขา	สถาบันที่จบ	ปีที่จบ
Ph.D.	Information Sciences	Tohoku University, Japan	2554
วศ.ม.	วิศวกรรมไฟฟ้า	สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง, กรุงเทพฯ	2546
วศ.บ.	วิศวกรรมไฟฟ้า	มหาวิทยาลัยเชียงใหม่, จ.เชียงใหม่	2537

สาขาวิจัยที่มีความชำนาญพิเศษ (แตกต่างจากวุฒิการศึกษา)

- 1) ..Mobile Computing and Sensor Networks.....
- 2) ..Mobile Ad-hoc Network.....
- 3) ..Internet Traffic Engineering and Congestion Control.....
- 4) ..Internet and Network Appliances.....

รางวัลด้านวิชาการ/ด้านวิจัย/งานสร้างสรรค์ (ด้านศิลปะ หรืออื่นๆ) ที่ได้รับ

ปี พ.ศ.	ชื่อรางวัล	สถาบันที่ให้
2552	บทความดีเด่น (Best Paper Award) สำหรับงานวิจัย “On the Performance Analysis of Traffic Splitting on Load Imbalancing and Packet Reordering of Bursty Traffic”	IEEE IC-NIDC Conference

ทุนการศึกษาและทุนวิจัยที่เคยได้รับ

ปี พ.ศ.	ทุนการศึกษาและทุนวิจัย	สถาบันที่ให้
2550	ทุนการศึกษาระดับปริญญาเอกมหาวิทยาลัยโทโฮกุ ประเทศญี่ปุ่น (Monbukagakusho: MEXT)	รัฐบาลญี่ปุ่น
2550	ทุนวิจัย GCOE Program สำหรับโครงการวิจัย “Traffic Engineering for Next Generation Network”	มหาวิทยาลัยโทโฮกุ ประเทศญี่ปุ่น
2551	ทุนวิจัย GCOE Program สำหรับโครงการวิจัย “Delay-Minimized Load Distribution for Multi-path Networks”	มหาวิทยาลัยโทโฮกุ ประเทศญี่ปุ่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2552	ทุนวิจัย GCOE Program สำหรับโครงการวิจัย “Towards Secure Quality of Services”	มหาวิทยาลัยโทโฮคุ ประเทศญี่ปุ่น
2553	ทุนผู้ช่วยสอนและผู้ช่วยนักวิจัย	มหาวิทยาลัยโทโฮคุ ประเทศญี่ปุ่น
2555	ทุน Visiting Researcher ที่ Tsuruoka NCT	Tsuruoka NCT, ประเทศญี่ปุ่น
2556	ทุนวิจัย “โครงการจัดตั้งห้องปฏิบัติการวิจัยและพัฒนา Mobile Computing and Sensor Networks”	คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
2556	ทุนวิจัย “โครงการวิจัยด้านเทคโนโลยีสารสนเทศ ร่วมกับหน่วยงานนอกคณะ”	คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
2556	ทุน Visiting Researcher ที่ Tsuruoka NCT	National Institute of Information and Communications Technology (NICT), ประเทศญี่ปุ่น
2558	ทุนวิจัย “เครือข่ายเฉพาะกิจแบบดีทีเอ็นบนสมาร์ตโฟนสำหรับการสื่อสารในสถานการณ์ภัยพิบัติ”	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง โดยผ่านการพิจารณาจากสำนักงานคณะกรรมการวิจัยแห่งชาติ (วช.)

#### ผลงานวิจัย/งานสร้างสรรค์

#### ผลงานวิจัย/งานสร้างสรรค์ที่ตีพิมพ์เผยแพร่ (ระดับชาติและนานาชาติ)

#### ผลงานเผยแพร่ในวารสารวิชาการระดับนานาชาติ

- 1) S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, “On Load Distribution over Multipath Networks,” *IEEE Communications Surveys and Tutorials*, vol. 14, no. 3, pp. 662–680, Jul. 2012.
- 2) S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, “Effective Delay-Controlled Load Distribution over Multipath Networks,” *IEEE Trans. Parallel and Distributed Systems*, vol. 22, no. 10, pp. 1730–1741, Oct. 2011.

#### การเสนอผลงานวิชาการ

#### ผลงานเผยแพร่ในรายงานการประชุมระดับนานาชาติ

- 1) T. Choksatid and S. Prabhavat, “An Epidemic Routing with Low Message Exchange Overhead for Delay Tolerant Networks,” in *Progress in Systems Engineering, Advances in Intelligent Systems and Computing*, vol. 1089, H. Selvaraj, Ed. Springer International Publishing, 2015, pp. 429–436.
- 2) T. Thongthavorn, S. Prabhavat, and W. Narongkhachavana, “A Study on Overhead Reduction for GPS-Assisted Mobile Ad-Hoc Networks,” in *Proc. IEEE Region 10 Annual International Conference (TENCON 2014)*, Bangkok, Thailand, Oct. 2014. (Accepted)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3) S. Prabhavat, R. Varakulsiripunth, and S. Utsumi, "Load Distribution using Modified RED for Multipath TCP Communication," in *Proc. IEEE International Conference on Information Technology and Electrical Engineering (ICITEE 2013)*, Yogyakarta, Indonesia, Oct. 2013.
- 4) M. Inoue, P. Tangdumrongrat, S. Prabhavat, S. Utsumi, S. Zabir, and N. Shiratori, "Mark-Reverse Explicit Congestion Notification for Satellite IP Networks," in *Proc. International Symposium on Technology for Sustainability (ISTS 2012)*, Bangkok, Thailand, Nov. 2012.
- 5) S. Prabhavat, N. Kato, and R. Varakulsiripunth, "Load Distribution Mechanism for Multipath TCP Communication," in *Proc. International Symposium on Technology for Sustainability (ISTS 2011)*, Bangkok, Thailand, Jan. 2012.
- 6) S. Prabhavat, H. Nishiyama, N. Ansari, and N. Kato, "On the Performance Analysis of Traffic Splitting on Load Imbalancing and Packet Reordering of Bursty Traffic," in *Proc. IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC 2009)*, Beijing, China, Nov. 2009. (The Best Paper Award)
- 7) S. Prabhavat, H. Nishiyama, Y. Nemoto, N. Ansari, and N. Kato, "Load Distribution with Queuing Delay Bound over Multipath Networks: Rate control using Stochastic Delay Prediction," in *Proc. the 26<sup>th</sup> International Communications Satellite Systems Conference (ICSSC 2008)*, San Diego, CA, Jun. 2008.
- 8) S. Prabhavat, R. Varakulsiripunth, "Performance Improvement on RED Based Gateway in TCP Communication Network", In *Proc. International Conference on Control, Automation and Systems (ICCAS 2004)*, Thailand, Aug. 25-27, 2004.

ผลงานเผยแพร่ในรายงานการประชุมระดับชาติ

- 1) ศุภโชค พงษ์ขวัญ, วรวัชร ณรงค์ชวณะ, และ สุมธ ประภาวัต, "การปรับปรุงการแบ่งโหลดเพื่อกระจายข้อมูลหลายทางบนเครือข่ายเฉพาะกิจเคลื่อนที่," งานประชุมวิชาการระดับประเทศด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศ (National Conference on Computing and Information Technology: NCCIT) ครั้งที่ 10, ภูเก็ต, ประเทศไทย, พ.ค. 2557.
- 2) สุมธ ประภาวัต, สารัช ศรีสวัสดิ์พงษ์, ทะนง ชูณิษานนท์, และ วรวัชร ณรงค์ชวณะ, "กลไกค้นหาเกตเวย์ที่มีโอเวอร์เฮดจากการแลกเปลี่ยนข้อความสำหรับเครือข่ายเคลื่อนที่เฉพาะกิจ," งานประชุมวิชาการระดับประเทศด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศ (National Conference on Computing and Information Technology: NCCIT) ครั้งที่ 10, ภูเก็ต, ประเทศไทย, พ.ค. 2557.
- 3) ธนานพ ทองถาวร, วรวัชร ณรงค์ชวณะ, และ สุมธ ประภาวัต, "การศึกษาความเป็นไปได้ในการลดโอเวอร์เฮดบนระบบเครือข่ายไร้สายเฉพาะกิจ โดยการสร้างเงื่อนไขในการจำกัดการกระจายข้อมูลในขั้นตอนการค้นหาเส้นทาง," รายงานการประชุมวิชาการระดับประเทศด้านเทคโนโลยีสารสนเทศ (National Conference on Information Technology: NCIT) ครั้งที่ 6, นครราชสีมา, ประเทศไทย, ก.พ. 2557.
- 4) อธิพงษ์ โชคสถิตย์, เบญจพร ปาวะริย์, วรวัชร ณรงค์ชวณะ, และ สุมธ ประภาวัต, "การปรับปรุงกลไกการแพร่กระจายข้อความเพื่อลดการแลกเปลี่ยนข้อมูลบนเครือข่ายดีทีเอ็น," รายงานการประชุมวิชาการระดับประเทศด้านเทคโนโลยีสารสนเทศ (National Conference on Information Technology: NCIT) ครั้งที่ 6, นครราชสีมา, ประเทศไทย, ก.พ. 2557.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 5) สุเมธ ประภาวัต, ทะนง ชูวณิชชานนท์, สาริซ ศรีสวัสดิ์พงษ์ และ วรวัชร ฌรงคะชวณะ, “การปรับปรุงกลไกค้นหาเกตเวย์เพื่อลดโอเวอร์เฮดจากการแลกเปลี่ยนข้อมูลในเครือข่ายเฉพาะกิจ,” รายงานการประชุมวิชาการระดับประเทศด้านเทคโนโลยีสารสนเทศ (National Conference on Information Technology: NCIT) ครั้งที่ 6, นครราชสีมา, ประเทศไทย, ก.พ. 2557.
- 6) สุเมธ ประภาวัต, ธนานพ ทองถาวร, และ ณัชชา วัฒนอรุณกิจ, “การประยุกต์ใช้สมาร์ทโฟนในการแจ้งเหตุฉุกเฉิน และตำแหน่งที่อยู่ เพื่อขอรับบริการรถนาส่ง และบริการทางการแพทย์,” รายงานการประชุมวิชาการระดับประเทศด้านเทคโนโลยีสารสนเทศ (National Conference on Information Technology: NCIT) ครั้งที่ 5, กุมภาพันธ์ 2556.
- 7) สุเมธ ประภาวัต, นิติ ชัยกะเสวี, และ ภรภัทร นนทะบรรหาย, “การประยุกต์ใช้สมาร์ทโฟนเพื่อการพิสูจน์ตัวตนด้วยข้อมูลพิกัดจีพีเอส”, รายงานการประชุมวิชาการระดับประเทศด้านเทคโนโลยีสารสนเทศ (National Conference on Information Technology: NCIT) ครั้งที่ 5, กุมภาพันธ์ 2556.

#### ผลงานเผยแพร่อื่นๆ

- 1) M. Inoue, P. Tangdumrongrat, S. Prabhavat, S. Utsumi, S. M. S. Zahir, N. Shiratori, “Mark-Reverse Explicit Congestion Notification: Concept, Analysis and Evaluation,” IEICE Technical Report, vol. 112, no. 208, NS2012-79, pp. 153-158, Sep. 2012.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้