รายการอ้างอิง

1. Crowcroft, J., and Oechslin, O. Differentiated end-to-end Internet service using a weighted proportional fair sharing TCP. Computer Communications Review Vol. 28, Issue 3 (July 1998): 53-67.
2. Stevens, W. TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms. RFC2001 (January 1997).
3. Nichols, K., Jacobson, V., and Zhang, L. A two-bit differentiated service architecture for the internet. Internet draft-nichols-diff-svc-arch-00.txt (November 1997).
4. Wang, Z. User-share differentiation (USD) scalable bandwidth allocation for differentiated services – A case for proportional sharing IEEE/IFIPIWQos (1998).
5. Gerla, M., Weng, W., and Cigno, R. L. Bandwidth feedback control of TCP and real time source in the internet. IEEE GLOBECOM Vol. 1 (2000): 560-565.
6. Gerla, M., Weng, W., Cigno, R. L., and Mascolo., S. Generalized window advertising for TCP congestion control. CSD-TR 990012 (February 1999).
7. Mascolo, S. Smith's principle for congestion control in high speed data networks. IEEE Transaction on Automatic Control Vol. 45 (February 2000): 358-364.
8. Gevros, P., Crowcroft, J., Kristein, P., and Bhatti, S. Congestion control mechanisms and the best effort service model. IEEE Network Vol. 15 (2001): 16-25.
9. Gevros,P., and Crowcroft, J. Experiment result on weight proportional TCP throughput Differentiation. Fourth International Workshop on High Performance Protocol Architectures HIPPARCH-98 (June1998).
10. Allman, M., Paxson, V., and Stevens, W. TCP Congestion control. RFC 2581 (April 1999).
11. Clark, D., and Wroclawski, J. An Approach to Service Allocation in the Internet. draft-clark-diff-svc-alloc-00.txt (August 1997).
12. Floyd, S., and Fall, K. Promoting the use of End-to-End congestion control in the internet. IEEE/ACM Transactions on Networking Vol. 7 (August 1999): 458-472.
13. Vangala, S., and Labrador, M. A. The TCP SACK-Aware-Snoop Protocol for TCP over wireless Network. Proc. IEEE VTC 2003 (October 2003): 2624-2628.

14. Bakre, A., and Badrinath, B. R. I-TCP: Indirect TCP for mobile hosts. iProc. IEEE International Conference on Distributed Computer System ICDC'95 (June 1995): 136-143.

15. Balakrishnan, H., Seshan, S., Amir, E., and .Katz, R. H. Improving TCP/IP Performance over Wireless Networks. Proc. of International Conference on Mobile Computing and Networking MOBICOM (November 1995): 2-11.

16. Balakrishnan, H., Padmanabhan, N., Seshan, S., and Katz, R. H. A Comparison of mechanism for improving TCP performance over wireless links. IEEE/ACM Transactions on Networking Vol. 5 (December 1997): 756-769.

17. Vangala, S., and Labrador, M. Performance of TCP over wireless networks with the Snoop protocol. Proc. IEEE LCN (November 2002): 600-601.

18. ns-2. http://www.isi.edu/nsnam/ns/

19. Zorzi, M., Rao, R. R., and Milstein, L. B. On the accuracy of a first-order Markov model for data transmission on fading channels. Proc. IEEE ICUPC 95 (1995): 211-215.

ภาคผนวก

บทความทางวิชาการที่ได้รับเผยแพร่แล้ว

บทความทางวิชาการจากการประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 28 (28th Electrical Engineering Conference : EECON28) ซึ่งจัดขึ้นในระหว่างวันที่ 21-22 ตุลาคม พ.ศ. 2548 ณ โรงแรมเพิร์ลวิลเลจ หาดในยาง อำเภอ ถลาง จังหวัด ภูเก็ต

การกู้คืนแพ็กเกตบนช่องสัญญาณไร้สายบน TCP/IP ด้วยการส่งข้อมูลของ SACK TCP ทำงานร่วมกับ New Snoop

SACK TCP and New Snoop Protocol Fast Recovery Loss for TCP/IP on Wireless Channel

วัฒนา เกยูรรัศมี, ศักดิ์ เสกขุนทด และลัญฉกร วุฒิสิทธิกุลกิจ
ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
254 ถนนพญาไท เขตปทุมวัน กรุงเทพมหานคร 10330
โทร. 02 218 6908 E-mail: wattana.ke@student.chula.ac.th, sak@nectec.or.th and wlunchak@chula.ac.th

บทคัดย่อ

บทความฉบับนี้เสนอกรรมวิธีการส่งซ้ำในระดับลิงค์เลเยอร์ (Link layer retransmission) รูปแบบใหม่ที่เรียกว่า New Snoop ที่ทำงานร่วมกับวิธีการส่งข้อมูลของ SACK TCP เพื่อแก้ไขข้อจำกัดการกู้คืนแพ็กเกตบนช่องสัญญาณไร้สายแบบดั้งเดิมโดยอาศัยชุดข้อมูลที่อยู่ในบล็อกของ SACK ให้เกิดประโยชน์อย่างเต็มที่นอกเหนือจากการใช้ระบุชุดข้อมูลที่สูญหายส่งไปยังผู้ใช้ต้นทางเพียงอย่างเดียว และวิธีนี้ยังก่อให้เกิดความต่อเนื่องในการส่งออกของข้อมูลเพื่อให้ใช้แบนด์วิดท์อย่างเต็มที่โดยไม่มีการลดขนาดหน้าต่างของความแออัด (congestion window) ลงเนื่องจากการสูญหายของแพ็กเกต และท้ายที่สุดได้แสดงให้เห็นถึงประสิทธิภาพในการส่งข้อมูลของวิธีที่ได้เสนอกับอัตราการสูญหายของช่องสัญญาณไร้สายที่ค่าต่าง ๆ เปรียบเทียบกับการส่งข้อมูลของ TCP รูปแบบอื่น ๆ

คำสำคัญ: การส่งข้อมูลด้วย TCP, การส่งซ้ำของลิงค์เลเยอร์

Abstract

This paper proposes a new link layer retransmission protocol called New Snoop, to work with SACK TCP mechanism, for providing multiple loss recovery on wireless channel. The proposed scheme uses SACK option in the most effective manner not only to inform the lost packets but uses for recovery lost packet faster than the tradition TCP. Packet transmission is allowed to continue without reducing congestion window sizes so as to maximize the bandwidth utilization. Finally the transmission performance evaluations of TCP over wireless network at various error rates are carried out to compare the proposed scheme with other TCP mechanism with and without Snoop protocol.

Keywords: TCP, Link layer retransmission

1. บทนำ

การส่งข้อมูลด้วย TCP (Transmission control Protocol) เป็นโพรโทคอลที่ถูกพัฒนามาตั้งแต่ยุคแรก ๆ ของการสื่อสารบนเครือข่ายอินเทอร์เน็ตและได้ถูกนำมาใช้ในการส่งข้อมูลส่วนใหญ่บนเครือข่ายในปัจจุบัน เนื่องจากมีความน่าเชื่อถือในการส่งข้อมูลและมีความสามารถปรับตัวให้เข้ากับสภาพเครือข่าย โดยในระยะเวลาที่ผ่านมา TCP ได้ถูกพัฒนาและปรับปรุงหลายต่อหลายครั้งจนกระทั่งมีรูปแบบดังที่ใช้อยู่ในปัจจุบันซึ่งได้แก่ Tahoe TCP, Reno TCP และ SACK TCP เป็นต้น การควบคุมความแออัดในเครือข่ายเป็นเรื่องที่มีความสำคัญมากซึ่งมีหลักการทำงานดังนี้ เมื่อทางผู้ใช้ต้นทางทราบถึงแพ็กเกตที่สูญหายจะทำการลดอัตราการส่งลงเพื่อควบคุมจำนวนแพ็กเกตที่วิ่งอยู่ในเครือข่ายให้อยู่ในระดับที่ยังสามารถรองรับได้โดยไม่ก่อให้เกิดการสูญหาย เมื่อนำการส่งข้อมูลด้วย TCP มาใช้ในช่องสัญญาณไร้สายที่มีลักษณะการสูญหายเป็นแบบสุ่ม การกู้คืนแพ็กเกตด้วยการควบคุมความแออัดไม่สามารถทำได้อย่างมีประสิทธิภาพเนื่องจากอัตราการสูญหายแพ็กเกตของช่องสัญญาณไร้สายที่สูงไปกระตุ้นการทำงานของวิธีการควบคุมความแออัด (congestion control mechanism) บ่อยครั้งอย่างไม่จำเป็น เป็นเหตุให้ประสิทธิภาพของปริมาณงาน (throughput) ถูกลดทอนลงทั้งนี้เพราะว่า TCP ที่ใช้อยู่ในปัจจุบัน ยังไม่มีวิธีที่สามารถแยกแยะการสูญหายที่เกิดขึ้นในเครือข่ายเนื่องจากความแออัดและการสูญหายที่เกิดเพราะสัญญาณรบกวนในช่องสัญญาณไร้สายออกจากกันและตอบสนองต่อการสูญหายทั้งสองประเภทอย่างเหมาะสม ซึ่งมีหลายวิธีที่ถูกคิดค้นขึ้นมาเพื่อบรรเทาผลที่เกิดจากการสูญหายแบบสุ่มที่เกิดในโครงข่ายไร้สาย สามารถแบ่งวิธีเหล่านี้ตามลักษณะการส่งข้อมูลออกเป็น

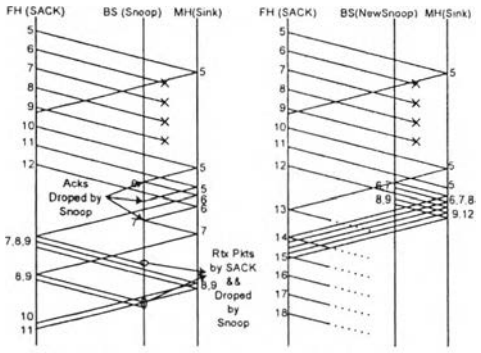
1. การส่งข้อมูลแบบระหว่างปลายทาง (End to end) เช่น WTCP, Explicit Loss Notification, Explicit Bad state Notification และ Explicit Congestion Notification [1]

2. การส่งข้อมูลแบบแบ่งแยกการเชื่อมต่อ (Split connection) เช่น M-TCP และ I-TCP [2]

3. การส่งข้อมูลแบบการส่งซ้ำของลิงค์เลเยอร์ เช่น Snoop [3]

ตามลักษณะของวิธีการส่งข้อมูลแบบระหว่างปลายทางที่ได้กล่าวไว้ข้างต้นคงต้องพูดว่าเคยสามารถปรับปรุงการส่งข้อมูลได้บนช่องสัญญาณไร้สายได้เพราะสามารถแยกแยะประเภทของการสูญหายว่าเกิดจากความแออัดหรือเกิดจากลักษณะของช่องสัญญาณไร้สาย แต่ก็ไม่สามารถทำได้ดีไปกว่าการส่งข้อมูลแบบการส่งซ้ำของลิงค์เลเยอร์ เพราะว่าเวลาประวิงการแพร่กระจาย (propagation delay) จาก BS ไปยัง MH มีค่าน้อยกว่าเมื่อเทียบกับเวลาประวิงการแพร่กระจายระหว่าง FH และ MH ส่วนการส่งข้อมูลแบบแบ่งแยกการเชื่อมต่อนั้นไม่เป็นที่นิยมใช้

กันเพราะว่าไปขัดแย้งกับพฤติกรรมการส่งข้อมูลระหว่างปลายทางของ TCP และจากการเปรียบเทียบประสิทธิภาพของการส่งข้อมูลด้วยวิธีข้างต้นใน [4] ซึ่งแสดงให้เห็นว่าการแก้ไขปัญหาการสูญหายจากช่องสัญญาณไร้สายด้วยวิธีการส่งข้อมูลแบบการส่งซ้ำของลิงค์เลเยอร์หรือ Snoop [3] ให้ประสิทธิภาพทางด้านปริมาณงานมี utilization ที่สูงและมีการตอบสนองต่อการสูญหายแบบสุ่มของช่องสัญญาณไร้สายได้รวดเร็วกว่าวิธีอื่น ๆ เมื่ออัตราการสูญหายของแพ็กเกตในช่องสัญญาณไร้สายเพิ่มมากขึ้น



(a) TCP SACK with Snoop (b) TCP SACK with NewSnoop

รูปที่ 1 (a) การทำงานของ SACK TCP เมื่อทำงานร่วมกับ Snoop
(b) การทำงานของ SACK TCP เมื่อทำงานร่วมกับ New Snoop

ทั้งนี้เพราะว่าการทำงานของ Snoop นั้นจะทำการจัดเก็บแพ็กเกตที่ผ่านเข้ามา BS ไว้ในบัฟเฟอร์ เมื่อแพ็กเกตเกิดการสูญหายในช่องสัญญาณไร้สายหลังจากทางด้าน MH ส่งแพ็กเกตตอบรับ (acknowledgement packet) ไปถึงยัง BS ทางด้าน Snoop จะทำการกู้แพ็กเกตที่สูญหายตามหมายเลขที่ระบุไว้แทนที่ FH พร้อมทั้งทำการลบแพ็กเกตตอบรับที่ผ่าน BS เข้ามาทั้งหมดจนกว่าจะได้แพ็กเกตตอบรับที่มีลำดับหมายเลขค่าใหม่จึงจะส่งแพ็กเกตตอบรับนั้นไปยัง FH แต่อย่างไรก็ตาม Snoop นี้สามารถที่จะกู้คืนแพ็กเกตที่สูญหายในช่องสัญญาณไร้สายใน 1 รอบของ local RTT (round-trip time) ได้เพียง 1 แพ็กเกตเท่านั้น (เพราะข้อจำกัดของการใช้แพ็กเกตตอบรับที่ใช้ลำดับหมายเลขที่ซ้ำในการกู้คืนข้อมูล) ดังนั้นภายใต้อัตราการสูญหายของช่องสัญญาณไร้สายที่เพิ่มมากขึ้น การกู้คืนแพ็กเกตของ Snoop ทำได้ไม่ดีเพราะว่าพฤติกรรมที่ Snoop ลบแพ็กเกตตอบรับที่มีลำดับหมายเลขซ้ำจนกว่าจะมีแพ็กเกตตอบรับที่มีลำดับหมายเลขค่าใหม่จึงจะส่งผ่านไปยัง FH นั้น ไปลดอัตราการส่งข้อมูลของผู้ใช้ต้นทาง

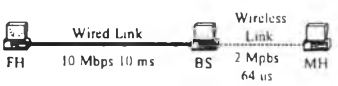
เพื่อแก้ไขข้อจำกัดในการกู้คืนข้อมูลที่สูญหายมากกว่า 1 แพ็กเกตนี้ วิธี Selective Acknowledgement (SACK) ได้ถูกคิดค้นเพื่อนำมาใช้แก้ไขปัญหาเนื่องจากความแออัดตาม RFC 2018 [5] ใน SACK TCP ประกอบไปด้วยบล็อกของ SACK ที่บรรจุข้อมูลของลำดับหมายเลขแพ็กเกตที่ทางด้านผู้รับปลายทางได้รับ ซึ่งทำให้ทางด้านผู้ส่งต้นทางทำการส่งเฉพาะข้อมูลที่สูญหายได้มากกว่า 1 แพ็กเกตในคราวเดียว ซึ่งมีบทความได้ศึกษาและทดลองเกี่ยวกับการทำงานของ SACK TCP เมื่อนำการใช้ร่วมกับ Snoop [1] แสดงให้เห็นว่ามีการส่งแพ็กเกตซ้ำอย่างไม่จำเป็น ทั้ง ๆ ที่ผู้รับปลายทางได้รับแพ็กเกตจากการกู้คืนของ Snoop แล้ว

ในบทความฉบับนี้ได้เสนอวิธีการทำงานของ Snoop รูปแบบใหม่เพื่อทำงานร่วมกับ SACK TCP ที่เหมาะสมกับเครือข่ายไร้สายที่มีอัตราการสูญหายของแพ็กเกตสูงและปรับปรุงประสิทธิภาพโดยรวมของการส่งข้อมูลบนช่องสัญญาณไร้สายให้ดีขึ้น ดังนั้นในหัวข้อต่อไปจะกล่าวถึงแนวคิดและเหตุผลรวมทั้งรายละเอียดของงานที่เกี่ยวข้อง

2. แนวคิดและเหตุผล

บทความ [6] ได้เปรียบเทียบการทำงานของวิธีการส่งข้อมูลด้วย TCP รูปแบบต่าง ๆ ทำงานร่วมกับ Snoop และการส่งข้อมูลของ TCP รูปแบบต่าง ๆ ที่ไม่มี Snoop ได้แสดงให้เห็น SACK TCP เมื่อทำงานร่วมกับ Snoop ให้ผลที่ดีกว่าเมื่อเปรียบเทียบการการส่งข้อมูลร่วมกับ Snoop ของ TCP รูปแบบอื่น ๆ เพราะความสามารถในการกู้คืนแพ็กเกตที่สูญหายมากกว่า 1 แพ็กเกตใน 1 ช่วง RTT ถึงแม้ว่าจะมีผลจากการส่งซ้ำแพ็กเกตโดยไม่จำเป็นขึ้นมาจากแพ็กเกตตอบรับที่ Snoop ส่งผ่านไปยัง FH หลังจากที่กู้คืนแพ็กเกตที่สูญหายตัวแรกได้และทางด้านผู้ใช้ต้นทางทำการส่งแพ็กเกตที่สูญหายตามบล็อกของ SACK ระบุมาในแพ็กเกตตอบรับอันล่าสุด ทั้ง ๆ ที่แพ็กเกตเหล่านั้นได้รับการจัดเก็บไว้ที่บัฟเฟอร์ของ Snoop และกำลังกู้คืนไปยัง MH ดังนั้นเมื่อแพ็กเกตที่ทำการกู้คืนโดย FH ไปถึง Snoop แพ็กเกตที่เพิ่งส่งมาจะถูกลบทิ้งเพราะแพ็กเกตเหล่านั้นยังคงอยู่ในบัฟเฟอร์ รูปที่ 1(a) แสดงให้เห็นถึงการกู้คืนข้อมูลซ้ำโดยไม่จำเป็นของ SACK TCP เมื่อทำงานร่วมกับ Snoop ดังนั้นใน [1] ได้เสนอวิธีที่เรียกว่า TCP SACK-Aware Snoop ขึ้นมาเพื่อแก้ไขปัญหานี้ ซึ่งวิธีนี้อาศัยคุณสมบัติที่สามารถกู้คืนหลายแพ็กเกตของ SACK มาใช้เฉพาะในส่วนของช่องสัญญาณไร้สายเท่านั้นโดยทุกแพ็กเกตตอบรับที่ผ่าน Snoop เพื่อที่จะส่งผ่านไปยังผู้ใช้ต้นทางจะทำการกำหนดค่าของ Permitted bits ให้เป็น 0 ซึ่งเป็นการปิดการทำงานบล็อกของ SACK ทำให้ทางด้านผู้ใช้ต้นทางรับทราบเพียงลำดับหมายเลขของแพ็กเกตตอบรับเท่านั้นเหมือนกับการตอบรับแพ็กเกตของ TCP รูปแบบอื่น ๆ การที่จะทำเช่นนี้ให้มีประสิทธิภาพในเกณฑ์ที่ดีต้องอยู่บนสมมุติฐานที่ว่าไม่มีการสูญหายของแพ็กเกตเนื่องจากความแออัดมากกว่า 1 แพ็กเกตต่อ 1 รอบ RTT ดังนั้นการปิดการทำงานบล็อก SACK สามารถเพิ่มประสิทธิภาพการกู้คืนแพ็กเกตในช่องสัญญาณไร้สายได้ดี แต่สำหรับเครือข่ายที่เกิดความแออัดในเครือข่ายได้ง่าย การปิดการทำงานในส่วนนี้อาจจะลดประสิทธิภาพในการทำงานลง และพฤติกรรมของ Snoop ที่ไม่ส่งผ่านแพ็กเกตตอบรับไปยัง FH จนกว่าที่จะมีการตอบรับของแพ็กเกตที่เพิ่งได้กู้คืนจาก MH ทำให้แพ็กเกตตอบรับไปถึงมือผู้ใช้ต้นทางนั้นขาดช่วงส่งผลกระทบต่อสมรรถนะของเครือข่ายได้ เพราะโดยพื้นฐานของการทำงานของ TCP นั้นเป็นการส่งข้อมูลแบบใช้แพ็กเกตตอบรับในการขับดันให้ผู้ใช้ต้นทางส่งแพ็กเกตออกไปยังผู้ใช้ปลายทาง ถ้าแพ็กเกตตอบรับ

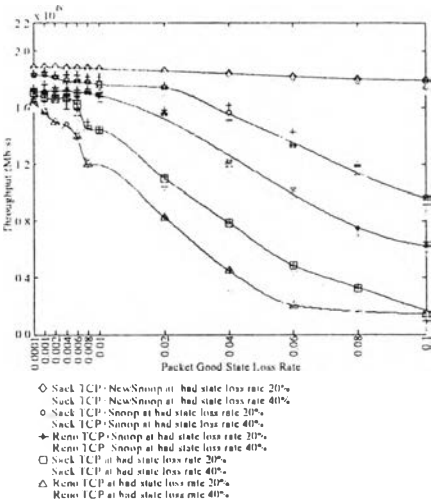
ที่ควรจะได้รับขาดหายไปช่วงระยะเวลาหนึ่งย่อมส่งผลให้สมรรถนะของเครือข่ายลดลงแน่นอน ดังนั้นบทความนี้เสนอวิธีที่กู้คืนแพ็กเกตที่สูญหายเนื่องจากลักษณะของช่องสัญญาณไร้สายเป็นหน้าที่ของ Snoop และปิดบังการสูญหายที่เกิดขึ้นในส่วนนี้ไม่ให้ทางผู้ใช้ต้นทางรู้และทำการกู้คืนแพ็กเกตที่เกิดขึ้น แต่ในขณะเดียวกันต้องไม่ปิดการทำงานบล็อกของ SACK ซึ่งมีความจำเป็นต้องใช้เมื่อมีการสูญเสียของแพ็กเกตมากกว่า 1 แพ็กเกตที่ไม่ใช่มาจากลักษณะของช่องสัญญาณไร้สาย โดยไม่ส่งผลกระทบต่อการส่งออกของแพ็กเกตต้องขาดความต่อเนื่องเพราะการลบแพ็กเกตตอบรับของ Snoop



รูปที่ 2 Topology ที่ใช้จำลองในระบบ

3. หลักการทำงานของวิธีที่ได้เสนอ

จากรายละเอียดที่ได้กล่าวมาข้างต้นในหัวข้อนี้จะพูดถึงหลักการทำงานของวิธีที่ได้นำเสนอ ซึ่งได้นำข้อได้เปรียบของ TCP SACK-Aware Snoop ที่สามารถกู้คืนแพ็กเกตที่สูญหายได้มากกว่า 1 แพ็กเกตใน 1 รอบของ RTT โดยหลักการของวิธีนี้กำหนดให้ Snoop สามารถอ่านบล็อกของ SACK ที่ผ่านเข้ามาเพื่อหาหมายเลขของแพ็กเกตที่สูญหายตามที่ระบุมาในบล็อกของ SACK และทำการเปรียบเทียบลำดับหมายเลขของแพ็กเกตที่สูญหายกับลำดับหมายเลขแพ็กเกตที่อยู่ในบัฟเฟอร์ ถ้ามีลำดับหมายเลขที่สูญหายนั้นอยู่ในบัฟเฟอร์ของ Snoop ซึ่งหมายความว่าเป็นแพ็กเกตที่สูญหายในช่องสัญญาณไร้สายจะทำการกู้คืนทันทีพร้อมทั้งทำการลบบล็อกของ SACK และแทนที่ลำดับหมายเลขค่าเดิมด้วยลำดับหมายเลขค่าที่แท้จริงของแพ็กเกตตอบรับ ซึ่งมีค่าเดียวกับหมายเลขแพ็กเกตในกรณีที่ไม่มีการสูญหาย (ในกรณีที่มีการสูญหายส่วนนี้มีค่าเท่ากับกับค่าของบล็อกทางขวาลำดับที่ 0 ของ SACK ลบด้วย 1) [5] ดังในรูปที่ 1(b) และทำการส่งแพ็กเกตตอบรับตัวนี้ผ่านไปยังผู้ใช้ต้นทาง แต่สำหรับในกรณีที่ลำดับหมายเลขแพ็กเกตที่สูญหายตามที่บล็อกของ SACK ระบุมาตัวใดตัวหนึ่งไม่ได้ถูกจัดเก็บไว้ในบัฟเฟอร์เพราะว่าแพ็กเกตสูญหายระหว่างทางก่อนที่จะถูกส่งมาถึง BS แทนที่จะรอจนกระทั่งการกู้คืนแพ็กเกตที่สูญหายเนื่องจากช่องสัญญาณไร้สายเป็นที่สำเร็จก่อนถึงจะทำการส่งแพ็กเกตตอบรับไปยังผู้ใช้ต้นทางกับบล็อกของ SACK แต่ในที่นี้วิธีที่ได้เสนอทันทีที่แพ็กเกตตอบรับเข้ามาจะถูกตรวจสอบหมายเลขในบล็อกของ SACK และนำมาเปรียบเทียบกับหมายเลขแพ็กเกตที่เก็บไว้ในบัฟเฟอร์ของ Snoop เพื่อนำหมายเลขที่ไม่ได้ถูกจัดเก็บในบัฟเฟอร์ระบุลงในบล็อกของ SACK พร้อมทั้งกำหนดลำดับหมายเลขแพ็กเกตให้สอดคล้องการสูญหายที่เกิดขึ้นระบุไปในบล็อกของ SACK ใหม่ทั้งหมดและไม่ขัดแย้งกับวิธีการส่งข้อมูลของ TCP ในทันทีที่แพ็กเกตตอบรับตัวนี้ไปถึงผู้ใช้ต้นทางและได้ส่งแพ็กเกตที่ไม่ได้จัดเก็บไว้ในบัฟเฟอร์ตามที่ได้กล่าวไว้ข้างต้นไปยัง Snoop ตรงส่วนนี้เมื่อมีแพ็กเกตตอบรับที่จะต้องถูกส่งไปยังผู้ใช้ต้นทาง แพ็กเกตดังกล่าวจะถูกแก้ไขบล็อกของ SACK ใหม่ โดยค่าที่ระบุลงในบล็อกของ SACK ครั้งนี้มีลักษณะเดียวกันกับในกรณีที่ได้รับแพ็กเกต ตอบรับที่ส่งมาจากผู้ใช้ปลายทางเมื่อได้รับแพ็กเกตหมายเลขนั้น พร้อมทั้งกำหนดหมายเลขแพ็กเกตใหม่ให้สอดคล้องกับหมายเลขแพ็กเกตตัวอื่น ๆ ที่ไม่ได้เก็บไว้ในบัฟเฟอร์ ถึงแม้ว่าแพ็กเกตส่วนนี้จะสูญหายอีกครั้งหน้าที่ในการกู้คืนแพ็กเกตในส่วนนี้จะเป็นของ Snoop โดยไม่ยุ่งหรือร้องขอให้ส่งแพ็กเกตจากผู้ใช้ต้นทาง ซึ่งวิธีนี้ก่อให้เกิดความต่อเนื่องในการส่งแพ็กเกตของผู้ใช้ต้นทางเพราะว่าจะทำส่งแพ็กเกตตอบรับที่ผ่านเข้ามาทุกแพ็กเกตโดยแก้ไขบล็อกของ SACK และกำหนดหมายเลขใหม่โดยไม่มีการลบแพ็กเกตตอบรับทิ้งทั้งหมดในระหว่างการกู้คืนแพ็กเกตของ Snoop



รูปที่ 3 ปริมาณงานช่องสัญญาณไร้สายทั้งหมดในช่วงเวลาการทดลองของวิธีการส่งข้อมูลด้วย TCP รูปแบบต่าง ๆ เปรียบเทียบกับวิธีที่ได้เสนอ

4. แบบจำลองของระบบ

ผลการทดลองในบทความฉบับนี้เป็นการทดลองด้วยการจำลองการทำงาน เพื่อเปรียบเทียบประสิทธิภาพต่างๆของวิธีที่ได้นำเสนอกับ TCP รูปแบบต่าง ๆ ที่ทำงานร่วมกับ Snoop โดยใช้ ns-2 network simulator และมี Topology สำหรับทดสอบการกู้คืนแพ็กเกตดังรูปที่ 2 ซึ่งมีผู้ใช้ต้นทางหรือ FH ต่ออยู่กับ BS ผ่านช่องสัญญาณแบบสาย (wired line) ที่มีแบนด์วิดท์ 10 Mbps และ เวลาประวิงการแพร่กระจายมีค่าเท่ากับ 20 ms เชื่อมต่อกับช่องสัญญาณไร้สายที่ต่อผู้ใช้ปลายทางหรือ MH มีแบนวิดท์ขนาด 2 Mbps ซึ่งกำหนดให้ค่าเวลาประวิงการแพร่กระจายมีน้อยมากเมื่อเทียบกับค่าประวิงเวลาโดยรวมจนสามารถละเลยค่าเวลาประวิงการแพร่ในส่วนนี้ได้ และแพ็กเกตมีขนาดคงที่ 1000 ไบต์ ส่วนโมเดลการสูญหายในช่องสัญญาณไร้สายนั้นกำหนดให้เป็นแบบ Continuous time two-state Markov model ซึ่งเป็นโมเดลที่ได้รับการยอมรับในหลายๆบทความสำหรับจำลองการสูญหายของช่องสัญญาณไร้สายบนช่องสัญญาณแบบสายของการจำลองการทำงาน

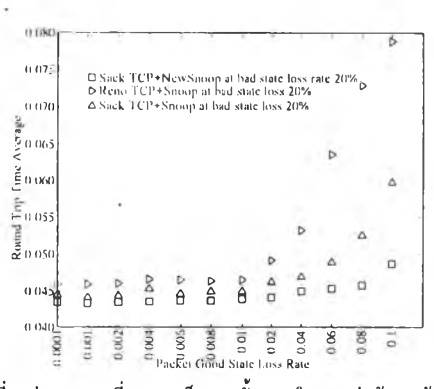
ใน ns-2 ที่ประกอบไปด้วย 2 สองสถานะคือ สถานะที่ช่องสัญญาณมีสภาพดี (good state) และสถานะที่ช่องสัญญาณมีสภาพเลว (bad state) ตามที่การวิเคราะห์ใน [7] เพื่อจำลองถึงสภาพของช่องสัญญาณไร้สาย กำหนดให้คาบของช่วงเวลาที่ช่องสัญญาณมีสภาพเลวและสภาพดีมีค่าเท่ากับ 6 วินาที และ 0.2 วินาที ตามลำดับ ระยะเวลาในการจำลองการทำงานมีค่า 100 วินาที และอัตราการสูญเสียแพ็กเกตในช่องสัญญาณที่มีสภาพดีกำหนดให้มีการสูญหายของแพ็กเกตเปลี่ยนแปลงตั้งแต่ 0.01% ถึง 10% ส่วนอัตราการสูญเสียแพ็กเกตในช่องสัญญาณที่มีสภาพเลวมีค่า 20% และ 40% ตามการทดลองแต่ละครั้ง โดยเปรียบเทียบปริมาณงาน และ ค่า RTT เฉลี่ยของแพ็กเกตทั้งหมดในแต่ละรูปแบบของ TCP รูปแบบต่าง ๆ กันเทียบกับวิธีที่ได้นำเสนอ

5. ผลการทดลอง

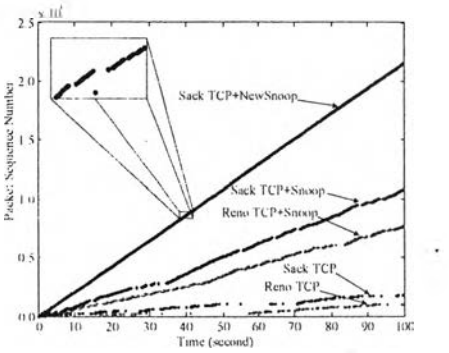
ผลของการส่งข้อมูลในช่วงเวลา 100 วินาทีและมีแบบจำลองของระบบตามรายละเอียดที่ได้กล่าวในหัวข้อที่ 3 ซึ่งทำการเปรียบเทียบกับวิธีที่ได้นำเสนอกับการส่งข้อมูลของ TCP 2 รูปแบบด้วยกัน ได้แก่ Reno TCP และ SACK TCP โดยแบ่งเป็น 2 กรณี คือกรณีที่ทำงานร่วมกับ Snoop และไม่ได้ทำงานร่วมกับ Snoop มีค่าดังรูปที่ 3 และรูปที่ 5 เห็นได้ว่าการทำงานของ Snoop สามารถแก้ไขปัญหาการลดทอนประสิทธิภาพเนื่องจากการสูญหายของช่องสัญญาณได้ดี แต่ยังคงมีผลจากการกู้คืนข้อมูลซ้ำโดยไม่จำเป็นอยู่ เมื่อเทียบกับวิธีที่ได้เสนอจะเห็นได้ว่าการสูญหายในช่องสัญญาณไร้สายไม่มีผลกระทบต่อการส่งข้อมูล เพราะว่าวิธีที่ได้เสนอนั้นสามารถปิดบังการสูญหายที่เกิดในช่องสัญญาณไร้สายได้อย่างดี แต่การสูญหายที่เกิดเนื่องจากความแออัดทำส่งคงส่งผลให้อัตราการส่งลดลง และจากรูปที่ 4 เป็นค่า RTT เฉลี่ยของแพ็กเกตทั้งหมดแสดงให้เห็นว่าวิธีที่ได้เสนอนั้นสามารถลดค่า RTT โดยรวมได้เป็นอย่างดีซึ่งลดการเกิดการส่งซ้ำเนื่องจากเกินช่วงเวลารอคอยได้ดีกว่าวิธีอื่น ๆ ที่นำมาเปรียบเทียบ

6. สรุป

วิธีที่เสนอในบทความนี้เป็นการกู้คืนการสูญหายที่เกิดขึ้นในช่องสัญญาณไร้สายโดยใช้บล็อกของ SACK มาใช้เพื่อเพิ่มความสามารถในการกู้คืนแพ็กเกตและทำแก้ไขข้อมูลในบล็อกของ SACK ให้สอดคล้องกับการสูญหายโดยไม่ขัดกับการทำงานในส่วนอื่นๆ ในการจำลองแบบและวิเคราะห์ถึงหลักการทำงานแสดงให้เห็นว่าวิธีที่ได้นำเสนอที่เรียกว่า New Snoop นั้น เพิ่มสมรรถนะการส่งข้อมูลภายในช่องสัญญาณไร้สายที่มีอัตราการสูญหายที่สูงแต่ต้องกำหนดให้ขนาดบัฟเฟอร์ของ New Snoop มีขนาดใหญ่เพียงพอที่จัดเก็บแพ็กเกตส่วนที่ไม่ได้ทำการกู้คืนได้หมด ดังนั้นในงานที่จะทำครั้งต่อไปคือปรับปรุงการทำงานในด้านการจัดเก็บข้อมูลในบัฟเฟอร์ของ New Snoop ให้มีประสิทธิภาพในการใช้งานได้อย่างเต็มที่และใช้ขนาดบัฟเฟอร์เล็กลง



รูปที่ 4 ค่า RTT เฉลี่ยของแพ็กเกตทั้งหมดในการส่งข้อมูลด้วย TCP รูปแบบต่าง ๆ เปรียบเทียบกับวิธีที่ได้นำเสนอ



รูปที่ 5 อัตราการเพิ่มของลำดับหมายเลขแพ็กเกตเมื่อเทียบกับเวลาของ TCP รูปแบบต่าง ๆ เปรียบเทียบกับวิธีที่ได้นำเสนอ

7. กิตติกรรมประกาศ

ขอขอบคุณทุนสถาบันบัณฑิตวิทยาศาสตร์และเทคโนโลยีไทย (TGIST) ที่สนับสนุนเงินทุนในการวิจัยครั้งนี้

เอกสารอ้างอิง

[1] S. Vangala and M. Labrador, "The TCP SACK-Aware-Snoop Protocol for TCP over wireless Network", in Proc. IEEE VTC 2003, pp. 2624-2628, October 2003.

[2] A. Bakre and B.R. Badrinath., "I-TCP: Indirect TCP for Mobile Hosts", in Proc. IEEE ICDC'95, pp. 136-143, 1995.

[3] H. Balakrishnan, S. Seshan, E. Amir, and R.H. Katz, "Improving TCP/IP Performance over Wireless Networks." in proceedings of ACM Mobicom, Nov 1995.

[4] H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. H. Kaz, "A Comparison of Mechanism for Improving TCP Performance over Wireless Links", IEEE/ACM Transactions on Networking, December 1997.

[5] M. Mathis, J. Mahdavi, S. Floyd and A. Romanow, "TCP Selective Acknowledgement Options", RFC 2018, October 1996.

[6] S. Vangala and M. Labrador, "Performance of TCP over Wireless Networks with the Snoop Protocol", in Proc. IEEE LCN, pp. 600-601, November 2002.

[7] M. Zorzi, R. Rao, L. Milstein, "On the Accuracy of a First-order Markov Model for Data Transmission on Fading Channels", in Proc. IEEE ICUPC 95, pp. 211-215, 1995.

ประวัติผู้เขียนวิทยานิพนธ์

นายวัฒนา เกยูรรัศมี เกิดเมื่อวันที่ 2 มกราคม พ.ศ. 2523 ที่จังหวัดปัตตานี ได้รับการศึกษาในหลักสูตรวิศวกรรมศาสตรบัณฑิต ณ มหาวิทยาลัยสงขลานครินทร์ ในปีการศึกษา 2541 และสำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า เกียรตินิยมอันดับสอง ในปีการศึกษา 2544 จากนั้นจึงเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า ณ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2545