การออกแบบและพัฒนากลไกปรับปรุงเส้นทางสำหรับเครือข่ายเนมดาต้า

A Design and Development of Path-convergence Mechanisms for Named Data Networking

ธงชัย เจือจันทร์[1] , สมนึก พ่วงพรพิทักษ์[2], นพรัตน์ โพธิ์สิงห์[1], สุวัฒน์ กล้วยทอง[3], อภิชาต สมรัตน์[1]

Thongchai Chuachan[1], Somnuk Puangpronpitag[2], Nopparat Posing[1], Suwat Gluaythong[3], Apichart Somrat[1]

Received: 19 May 2017; Accepted: 16 October 2017

บทคัดย่อ

เครือข่ายเนมดาต้าเป็นสถาปัตยกรรมเครือข่ายที่มีแนวโน้มจะถูกนำไปใช้ กับการเชื่อมต่อสื่อสารที่มีความซับซ้อนในปัจจุบัน ด้วยการใช้แพ็กเก็ตประเภทอินเทอร์เรสในการค้นหาข้อมูลจากการใช้ชื่อของข้อมูล และตัวข้อมูลสามารถถูกสำรองไว้บนอุปกรณ์เครือข่ายได้ จึงทำให้การเร้าแพ็กเก็ตประเภทอินเทอร์เรสมีความสำคัญสูงต่อการได้รับข้อมูลในเครือข่ายเนมดาต้า โดยในปัจจุบันอัลกอริธึมเบสรูท ถูกนำมาใช้อย่างแพร่หลาย ทั้งบนระบบเทสเบดของกลุ่มวิจัยเนมดาต้า อย่างไรก็ตามอัลกอริธึมเบสรูทยังมีปัญหาหลายประการ เช่น การเลือกเส้นทางจากค่าที่ต่ำที่สุด ทำให้เปลี่ยนเส้นทางได้ไม่ทันเหตุการณ์ เป็นต้น งานวิจัยนี้จึงพัฒนาการเลือกเส้นทางโดยออกแบบอัลกอริธึม Longest Prefix Reduction (LPR) ที่ช่วยให้การส่งต่อแพ็กเก็ตอินเทอร์เรสมีประสิทธิภาพสูงขึ้น แล้วพัฒนาซอฟต์แวร์ต้นแบบต่อยอดจาก Named Forwarding Daemon ผลการทดสอบพบว่า LPR ช่วยให้เครือข่ายเนมดาต้าใช้เวลาปรับปรุงเส้นทางและมีสัดส่วนการสูญหายของแพ็กเก็ตลดลงได้มากกว่าร้อยละ 90

คำสำคัญ: เครือข่ายเนมดาต้า การทดลองบนเครือข่ายจริง การปรับปรุงเส้นทาง

Abstract

Named Data Networking (NDN) is a promising candidate for future network architecture, and is well suited to today's complex networks. A named prefix of the NDN Interest-packet has been used for searching a desire data. Furthermore, such NDN data can be temporarily stored in intermediate devices without concerning for locations. So, the Interest-packet plays an important role for NDN network communication. Currently, A Best-Route algorithm has been widely used in the major NDN's network Testbed. However, the Best-Route always selects a lowest routing cost, which is unreliable in disruptive networks. In this paper, we have developed a Longest Prefix Reduction (LPR) algorithm to handle NDN path-convergence. LPR actively updates routing paths to the data, and improves several aspects to multi-path connection. We have implemented our LPR in Named Forwarding Daemon (NFD). LPR experimental results show that a network loss has been reduced approximately 90%.

Keywords: Named Data Networking, Test-bed, Path convergence

---

[1] *อาจารย์, สาขาวิทยาการคอมพิวเตอร์, ภาควิชาวิทยาศาสตร์พื้นฐาน, คณะวิทยาศาสตร์และเทคโนโลยี, มหาวิทยาลัยราชภัฏสุรินทร์. E-mail: thongchai.c@srru.ac.th, n_posing@hotmail.com, apichart@comsci.srru.ac.th*

[2] *อาจารย์, สาขาวิทยาการคอมพิวเตอร์, คณะวิทยาการสารสนเทศ, มหาวิทยาลัยมหาสารคาม. E-mail: {somnuk.p@msu.ac.th}*

[3] *อาจารย์, สาขาเทคโนโลยีสารสนเทศ, ภาควิชาวิทยาศาสตร์พื้นฐาน, คณะวิทยาศาสตร์และเทคโนโลยี, มหาวิทยาลัยราชภัฏสุรินทร์. E-mail: suwat_gl@srru.ac.th*

[1] *Lecturer, Department of Computer Science, Faculty of Science and Technology, Surindra Rajabhat University. E-mail: thongchai.c@srru.ac.th, n_posing@hotmail.com, apichart@comsci.srru.ac.th*

[2] *Lecturer, Department of Computer Science, Faculty of Informatics, Mahasarakham University. E-mail: somnuk.p@msu.ac.th*

[3] *Lecturer, Department of Information Technology, Faculty of Science and Technology, Surindra Rajabhat University. E-mail: suwat_gl@srru.ac.th*

บทนำ

ในปัจจุบันการเชื่อมต่อสื่อสารในระบบเครือข่ายจะใช้โพรโทคอล TCP/IP[1] ซึ่งมีรูปแบบการเชื่อมต่อแบบ Point-to-Point และการใช้เพียงโพรโทคอล TCP/IP จะไม่สัมฤทธิ์ผลเมื่อมีผู้ใช้บริการจำนวนมาก จึงทำให้มีบริการกระจายเนื้อความ (Content) ไปยังแหล่งต่าง ๆ เหมาะกับการบริการแบบ Best Effort Service เช่น การใช้ Content Distribution Network (CDN)[2] เป็นส่วนเสริมให้การเชื่อมต่อโดยใช้โพรโทคอล TCP/IP ให้มีประสิทธิภาพมากขึ้น แต่ CDN ต้องเสียค่าใช้จ่ายสูงและยังมีประเด็นปัญหาด้านความมั่นคงดัง Liang และคณะ[2] ได้กล่าวไว้

Named Data Networking (NDN)[3] จึงถูกพัฒนาขึ้นเพื่อปรับปรุงสถาปัตยกรรมเครือข่าย ให้เหมาะกับลักษณะการเชื่อมต่อเครือข่ายในปัจจุบัน ซึ่งเป็นแบบ Content-Distribution ที่มีการกำหนดให้แพ็กเก็ตข้อมูล (Data) มีชื่อนำหน้า (Prefix) กำกับ และผู้รับ Data ใช้แพ็กเก็ต Interest ส่งไปยังเครือข่ายเพื่อหา Data ที่ต้องการโดยไม่ต้องกังวลถึงแหล่งที่มา และสามารถสำรอง (Cache) ข้อมูลไว้ตามอุปกรณ์เร้าเตอร์ได้

ถึงแม้เครือข่าย NDN จะมีวิธีทำให้ข้อมูลมี ชื่อนำหน้า (Prefix) กำกับและถูกสำรองข้อมูลได้ แต่สถาปัตยกรรมเครือข่าย NDN ยังคงมีปัญหาหลายประการ และหนึ่งในปัญหาที่สำคัญ คือ การปรับปรุงเส้นทางการเชื่อมต่อ ที่ต้องมีการตอบสนอง (Responsiveness) ที่ยังไม่รวดเร็วต่อการเปลี่ยนแปลงของเครือข่าย ในงานวิจัยนี้จึงออกแบบและพัฒนาระบบการปรับปรุงเส้นทางและการเลือกเส้นทาง Longest Prefix Reduction (LPR) สำหรับเครือข่าย NDN ด้วยการปรับปรุง Routing Cost ของเส้นทางการเชื่อมต่อ ให้ตอบสนองต่อเส้นทางเชื่อมต่อล้มเหลวได้รวดเร็วขึ้น

ซอฟต์แวร์ที่ใช้สำหรับพัฒนาต้นแบบจากแนวคิดของงานวิจัยนี้ มีหลายกลุ่มซอฟต์แวร์ที่สนับสนุนการส่งต่อแพ็กเก็ตในเครือข่าย NDN เช่น Named Forwarding Daemon (NFD)[4] และ Content Centric Network (CCN)[5] ซึ่งงานวิจัยนี้ได้เลือกใช้ NFD เป็นเครื่องมือพัฒนาต่อยอด เพราะ NFD เป็นซอฟต์แวร์โอเพ่นซอร์ส และเป็นต้นแบบจากกลุ่มนักวิจัยเครือข่าย NDN แล้วทำการทดลองซอฟต์แวร์ต้นแบบบนระบบเครือข่าย Testbed จากผลการทดลองพบว่า NFD จากงานวิจัยนี้ช่วยปรับปรุงเส้นทางได้รวดเร็วขึ้นอย่างมีนัยสำคัญ ช่วยเพิ่มอัตรการส่งได้รับข้อมูลสูงขึ้น ซึ่งช่วยให้การเชื่อมต่อในเครือข่าย NDN มีประสิทธิภาพดีขึ้น

วัตถุประสงค์

1. เพื่อปรับปรุงขั้นตอนการปรับปรุงเส้นทางสำหรับเครือข่าย NDN

2. พัฒนาต่อยอดซอฟต์แวร์ NFD สำหรับการเลือกเส้นทางส่งต่อแพ็กเก็ตในเครือข่าย NDN

ทฤษฎีที่เกี่ยวข้อง

เครือข่ายเนมดาต้า

เครือข่าย Named Data Networking (NDN)[3] เป็นสถาปัตยกรรมเครือข่ายที่เกิดจากแนวคิดของ Jacobson[6] หลังจากเล็งเห็นถึงรูปแบบการเชื่อมต่อสื่อสารบนอินเทอร์เน็ตที่มีปริมาณ (Volume) สูงขึ้น และรูปแบบการใช้งานที่เปลี่ยนแปลงจาก Point-to-point เป็น Content Distribution หน่วยงาน National Science Foundation (NFS)[7] จึงเสนอที่จะกำหนดสถาปัตยกรรมเครือข่ายที่สนับสนุนการใช้งานในอนาคตมากขึ้น NDN เป็นหนึ่งในสถาปัตยกรรมเครือข่ายที่เป็นที่ยอมรับในกลุ่มนักวิจัย

การปรับปรุงเครือข่าย NDN ดัง Figure 1 ด้านซ้ายเป็นสถาปัตยกรรม TCP/IP ซึ่งมี IP เป็นเอว (Waist) ของระบบเครือข่าย ซึ่งเป็นตัวประสานระหว่างโพรโทคอลใน Layer 2 ที่ทำงานกับฮาร์ดแวร์เครือข่ายและประสิทธิภาพสูงขึ้น และชั้น TCP, UDP และแอพพลิเคชัน ที่ควบคุมการส่งแพ็กเก็ตและขยายตัวอย่างมากในระดับแอพพลิเคชัน การใช้ IP จำเป็นต้องระบุต้นทางและปลายทาง จึงทำให้ IP เป็นการส่งข้อมูลแบบ Stateless ที่ประกอบด้วยต้นทาง (Source) และปลายทาง (Destination) ในสถาปัตยกรรมเครือข่าย NDN จะให้ความสำคัญกับข้อมูล มากกว่าแหล่งของข้อมูล เช่น ต้นทาง (Source) และปลายทาง (Destination) โดยข้อมูลจะถูกแบ่งเป็นก้อน Content-Chunk ซึ่ง Content-Chunk สามารถถูกกระจายนำไปเก็บสำรองไว้ได้บนอุปกรณ์เครือข่ายต่างๆ และเป็นส่วน Waist ของระบบเครือข่าย NDN ในชั้นที่อยู่สูงขึ้นจะมีชั้น Security ซึ่งใช้สำหรับจัดการความมั่นคงให้กับข้อมูล ด้วยวิธีเข้ารหัส และชั้น Strategy เครือข่าย NDN ใช้สำหรับการวางแผนการส่งต่อแพ็กเก็ต โดย NDN ยังคงสนับสนุนฟังก์ชันของเครือข่ายหลายประเภท เช่น TCP/IP, Ethernet และอื่น ๆ เป็นต้น ทำให้ NDN ถูกนำไปประยุกต์ใช้ได้ง่าย

ชั้น Strategy เป็นหนึ่งในส่วนสำคัญของเครือข่าย NDN ซึ่งปัจจุบันในชั้นนี้ใช้อัลกอริธึม Best-Route(BR)[8] ในการปรับปรุงเส้นทางเพื่อค้นหา Content โดยใช้ค่า Routing Cost ด้วยการใช้อัลกอริธึม Dijksra และการกำหนดเอง จึงทำให้การปรับปรุงเส้นทางทำได้ช้า เป็นที่มาของงานวิจัยนี้ที่จะพัฒนากลไกการปรับปรุงเส้นทางที่ช่วยให้ประสิทธิภาพการเชื่อมต่อของเครือข่าย NDN สูงขึ้น

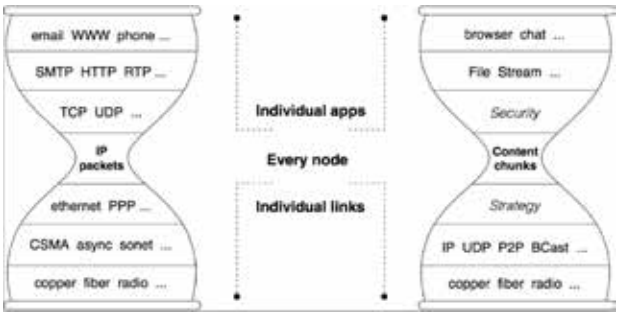

Figure 1 NDN Architecture[3]

การส่งต่อแพ็กเก็ตในเครือข่าย NDN

ในเครือข่าย NDN มีการพัฒนาซอฟต์แวร์ที่หลากหลาย เช่น CCN[5], NFD[4] และการพัฒนาเป็นระบบปฏิบัติการ เช่น RIOS เป็นต้น โดยทุกซอฟต์แวร์จะมีกลไกการทำงานตามกลไกใน Figure 2 ซึ่งในเครือข่าย NDN จะมีแพ็กเก็ต 2 ประเภท คือ Interest และ Data ในแพ็กเก็ต Interest ทำหน้าที่ค้นหา Data ที่ต้องการและเร้าเตอร์มีขั้นตอนการทำงานตามลำดับต่อไปนี้

1. เมื่อได้รับแพ็กเก็ต Interest ที่เร้าเตอร์จะค้นหาแพ็กเก็ต Data ที่ถูกสำรองไว้ก่อนหน้านี้ใน Content Store (CS) และถ้าหากพบ Data อยู่ใน CS ก็จะตอบกลับไปยังผู้ส่ง Interest และสิ้นสุดการทำงาน

2. หากไม่พบแพ็กเก็ต Data ใน CS ระบบจะนำ Interest ไปตรวจสอบในตาราง Pending Interest Table (PIT) ซึ่งเป็นตารางในการตรวจสอบสถานการณ์ส่ง Interest ภายใต้ Data เดียวกันก่อนหน้า ซึ่งถ้าหากมี Interest ถูกส่งออกไปก่อนหน้านี้แล้ว เร้าเตอร์จะเก็บ Face (Network Interface ในสถาปัตยกรรมเครือข่าย TCP/IP) ไว้ใช้ส่งแพ็กเก็ต Data กลับทาง Face ต่าง ๆ ที่มีการร้องขอ จึงทำให้เร้าเตอร์ไม่ส่งแพ็กเก็ต Interest ในเส้นทาง Upstream ซ้ำซ้อน

3. ในกรณีที่แพ็กเก็ต Interest ถูกนำไปตรวจสอบใน PIT แล้วไม่พบการส่ง Interest ก่อนหน้า ระบบจะตรวจสอบตาราง Forwarding Information Base (FIB) โดยค้นหา Face ที่จะส่ง Interest ออกไปยังปลายทางด้วยอัลกอริธึม Longest Prefix Match (LPM)[8] และส่งต่อแพ็กเก็ตออกจากเร้าเตอร์เพื่อหา Data ต่อไป แต่ถ้าหากใช้ LPM ค้นหา Face ที่จะส่งแพ็กเก็ตไม่ได้ เร้าเตอร์จะแจ้งผู้ส่ง Interest ในเส้นทาง Downstream โดยใช้แพ็กเก็ต Negative ACK (NACK) เพื่อแจ้งเร้าเตอร์ก่อนหน้า ไม่มีเส้นทางเชื่อมต่อไปยังปลายทางได้

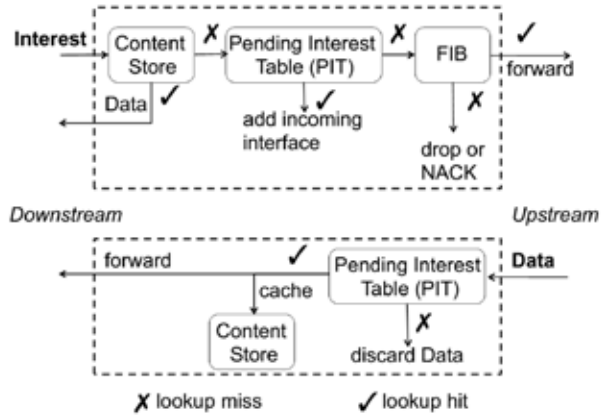

Figure 2 NDN Forwarding[3]

เร้าเตอร์ที่ได้รับแพ็กเก็ต Data แล้ว เร้าเตอร์จะเริ่มตรวจสอบชื่อของข้อมูล (Prefix) ที่อยู่ในแพ็กเก็ตประเภท Data ซึ่งถ้าหากยังคงมีอยู่ใน PIT ก็จะส่งแพ็กเก็ต Data ออกทางรายการ Face ที่มีการร้องขอก่อนหน้า และทำการสำรอง (Cache) แพ็กเก็ต Data นั้นไว้เพื่อใช้กับร้องขอในครั้งถัดไป แต่ถ้าหากไม่พบ Prefix ของ Data ในตาราง PIT เร้าเตอร์จะละทิ้ง (Drop) แพ็กเก็ต Data ดังกล่าว เพราะเป็น Data แปลกปลอมในระบบกลไกการปรับปรุงเส้นทางในเครือข่าย NDN ก่อนหน้า

อัลกอริธึมที่ใช้จัดการเส้นทางการส่งแพ็กเก็ต Interest มีหลายอัลกอรธึม เช่น BR[8,9], NCC, Broadcast และ Client Control โดย BR คือ อัลกอริธึมที่ถูกนำมาใช้ใน NFD สำหรับการส่งต่อแพ็กเก็ต Interest ซึ่ง BR ทำงานหลังจากได้รับแพ็กเก็ต Interest แล้วใช้ Prefix ของ Interest ตรวจสอบใน PIT และถ้าหากพบ Prefix ใน PIT เร้าเตอร์จะส่งแพ็กเก็ตออกไปยัง Face ที่มีค่า Routing Cost ต่ำที่สุด แต่ถ้าหากไม่พบ แพ็กเก็ตจะถูกนำไปประมวลผลกับ Face ที่แพ็กเก็ตถูกส่งครั้งล่าสุดไม่เกิน 100 ms และมี Prefix คล้ายกันมากที่สุด

จากการใช้ BR จะเห็นว่า Routing Cost ถูกกำหนดโดยการใช้อัลกอริธึม Dijksra คำนวณระยะทางไปยังปลายทางแบบ Multipath ตามกลไกของโพรโท-คอล Named Link-State Routing (NLSR)[10] แต่การกำหนด Routing Cost ด้วยวิธีดังกล่าวถูกกำหนดแบบถาวร ทำให้การใช้อัลกอริธึม BR เลือกส่งแพ็กเก็ตผ่านทาง Face ที่ไม่สามารถเชื่อมต่อได้ โดยไม่คำนึงถึงปัจจัยอื่น ๆ เช่น เส้นทางการเชื่อมต่อล้มเหลว เป็นต้น อัลกอริธึม BR จึงยังไม่มีประสิทธิภาพเพียงพอต่อการนำไปใช้กับเครือข่าย NDN ที่หลากหลาย จึงเป็นที่มาของงานวิจัยนี้ที่จะออกแบบและพัฒนาการปรับปรุงการปรับปรุงเส้นทางแบบอัตโนมัติ และทันต่อสถานการณ์การเปลี่ยนทางเครือข่ายได้ ซอฟต์แวร์ Named Forwarding Daemon

ซอฟต์แวร์ Named Forwarding Daemon (NFD)[4]

ถูกพัฒนาเป็นเครื่องมือสำหรับประมวลผลแพ็กเก็ต Interest และ Data บนอุปกรณ์เร้าเตอร์ในเครือข่าย NDN โดย NFD มีโครงสร้างดัง Figure 3 ทำหน้าที่เป็น Multiplexer ระหว่างเร้าเตอร์ ประกอบด้วยส่วน Forwarder ทำหน้าที่จัดการวางแผนการส่งต่อแพ็กเก็ต โดยใช้โมดูล Strategies และเป็นส่วนติดต่อกับระหว่างระบบ Face (Face System) ระบบ CS, PIT และ FIB เป็นต้น ในส่วนของ Face System นั้น จะจัดการวิธีการ Multiplex ซึ่งปัจจุบันสามารถทำงานได้บนเครือข่าย Ethernet และการวางซ้อน (Overlay) อยู่บน TCP และ UDP ได้ เมื่อ Forwarder ได้รับ Prefix ของแพ็กเก็ตที่จะส่งไปยังปลายทางแล้ว Forwarder จะส่งแพ็กเก็ตที่มี Prefix ดังกล่าวสู่ระบบ Face System และส่งไปยังปลายทางผ่านช่องทางต่าง ๆ ต่อไป

ใน Forwarder ยังเชื่อมกับระบบ CS, PIT และ FIB ตามสถาปัตยกรรมเครือข่าย NDN ที่กล่าวไปแล้วข้างต้น ซึ่งจัดการโดยใช้ไลบรารี ndn-cxx และเป็นไลบรารีสำคัญสำหรับ NFD ในการประมวลผลแพ็กเก็ต Interest และ Data

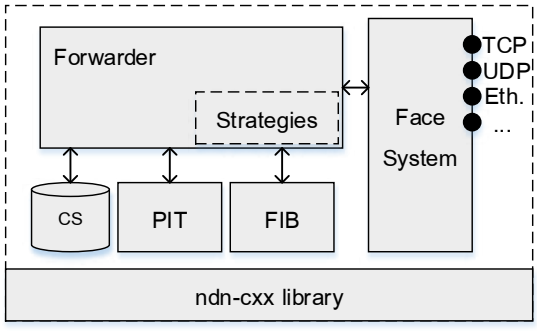

Figure 3 NFD Components

การออกแบบและพัฒนา LPR

อัลกอริธึม Longest Prefix Reduction

อัลกอริธึม Longest Prefix Reduction (LPR) ของงานวิจัยนี้ มีแนวคิดที่จะปรับปรุงวิธีจัดการ Routing Cost ร่วมกับการจัดการ Prefix ของแพ็กเก็ต Interest ในการเลือกเส้นทางการเชื่อมต่อในเครือข่าย NDN โดยได้เพิ่มขั้นตอนการประมวลผลแพ็กเก็ต Interest ในเร้าเตอร์ดังนี้

1. เมื่อ LPR ได้รับแพ็กเก็ต Interest จะเลือก Face ที่ใช้ส่งต่อแพ็กเก็ตไปที่ Upstream จาก Face (F_{out}) ที่มี C ต่ำที่สุด และ Prefix ตรงกันยาวที่สุด ด้วยการใช้อัลกอริธึม Longest PrefixMatch ที่อยู่ใน NFD

2. เมื่อ F_{out} เกิด Timeout LPR จะเลือก Face ถัดไปที่มี C ต่ำสุดรองลงมาจนครบทุก Face ที่ใช้งานอยู่

3. หากทุก Face ไม่สามารถเชื่อมต่อได้ LPR จะลด Prefix ของชื่อ Data ในตาราง FIB ลดลงครั้งละ 1 เช่น Prefix ในแพ็กเก็ต Interest ชื่อ /ndn/th/ac/srru/data/01 และใน FIB ของเร้าเตอร์มี Prefix ชื่อ /ndn/th/ac/srru/data ซึ่งตรงกันใน Face ที่ i (F_i) และเชื่อมต่อไม่ได้ LPR จะลดชื่อ Prefix ในแพ็กเก็ต Interest ให้เหลือ /ndn/th/ac/srru/data แล้วค้นหา Face ที่สนับสนุนการส่ง Interest ใหม่ตามขั้นตอนที่ 1 และ 2

การพัฒนาและทดลองโปรแกรมต้นแบบ

การพัฒนาส่วนประมวลผลแพ็กเก็ต Interest

การพัฒนาต้นแบบจากแนวคิดของ LPR ได้พัฒนาและนำมารวมกับซอฟต์แวร์ NFD โดยเพิ่มฟังก์ชันการประมวลผลแพ็กเก็ต Interest ดังแสดงใน Figure 4 ตามขั้นตอนดังนี้

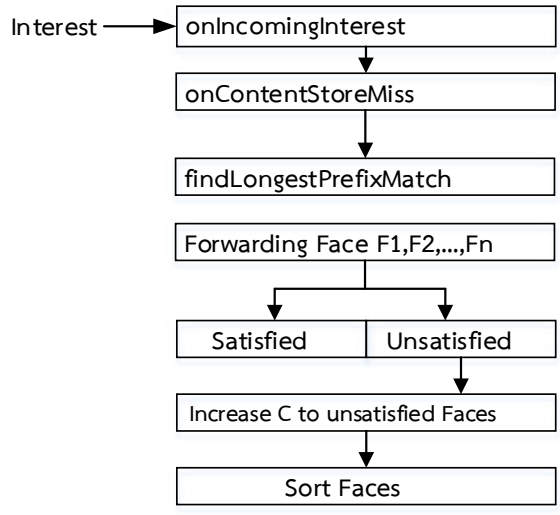

Figure 4 Interest processing for LPR algorithm

1. เมื่อเร้าเตอร์ได้รับแพ็กเก็ต Interest ผ่านทาง Face ที่ F_1, F_2, F_3, ... F_n หลังจาก De-multiplex แล้วแพ็กเก็ตถูกส่งเข้าไปยังฟังก์ชั่น onIncomingIntest ซึ่งภายใน onIncomingIntest จะมีการตรวจสอบตามขั้นตอนของสถาปัตยกรรม NDN แต่ถ้าหากไม่ได้รับ Data ระบบจะแจ้งเตือนไปยังฟังก์ชัน onContentStoreMiss จากนั้นจะเข้าสู่การเลือกเส้นทางเลือก Face ที่จะส่งแพ็กเก็ต Interest สู่เส้นทาง Upstream

2. ค้นหา Face ที่สามารถส่งแพ็กเก็ต Interest ได้จาก Face ที่มี Prefix ตรงกับ Prefix ในแพ็กเก็ต Intest มากที่สุดด้วยฟังก์ชัน findLongestPrefixMatch หลังจากได้รายการ F_1, F_2, F_3, ... F_n ที่สนับสนุน การส่งแพ็กเก็ตแล้ว จากนั้นเลือก Face ที่มี Routing Cost (C) ต่ำที่สุด (F_{low}) แล้วส่งแพ็กเก็ต Interest ผ่าน F_{low} ดังกล่าว

3. เมื่อเกิด Timeout ทาง F_{low} ที่มีการส่ง Interest ออกไปแล้ว NFD จะแจ้งผลไปยังส่วนของ Unsatisfied และ

เพิ่ม C ขึ้นครั้งละ 1 ทาง F_{low} ก่อนที่จะทำการเรียง Face ทั้งหมดใหม่ โดยให้เรียงจากน้อยไปหามาก แล้วลด Prefix สุดท้ายลงทีละ 1 ตามอัลกอริธึม LPR ก่อนที่จะส่งแพ็กเก็ต Interest ใหม่ (Retransmission) ทาง Face ใหม่ที่มี C ต่ำที่สุด การพัฒนาส่วนประมวลผลแพ็กเก็ต Data

เมื่อเร้าเตอร์ได้รับแพ็กเก็ต Data หมายถึง Face ดังกล่าวสามารถเชื่อมต่อได้ ดังนั้นใน Face ที่มีสัดส่วนการส่ง Data จาก Downstream สูงจึงมีโอกาสที่การส่ง Interest จะได้รับ Data สูง ในอัลกอริธึม LPR จึงมีการปรับปรุง Routing Cost (C) ตามขั้นตอนใน Figure 5

เมื่อเร้าเตอร์ได้รับแพ็กเก็ต Data ระบบจะนำเข้าสู่ฟังก์ชัน onIncomingData แล้วจึงลด C ให้กับFace ที่ได้รับ Data จากนั้นใช้ฟังก์ชัน getNextHops จะได้รับ Face F_1, F_2, F_3, ... F_i ที่มี Prefix เดียวกับ Data ที่ได้รับก่อนที่จะลด C ตั้งแต่ C_1, C_2, C_3, ... C_i ให้กับทุก Face ที่มี Prefix เดียวกัน เพื่อบำรุงระยะห่างของ C ในแต่ละ Face ไม่ให้มีความต่างกันสูงเกินไป และไม่ทำให้ C สูงขึ้นแบบไม่สิ้นสุดจากขั้นตอนการจัดการ C ในส่วนของการประมวลผลแพ็กเก็ต Interest

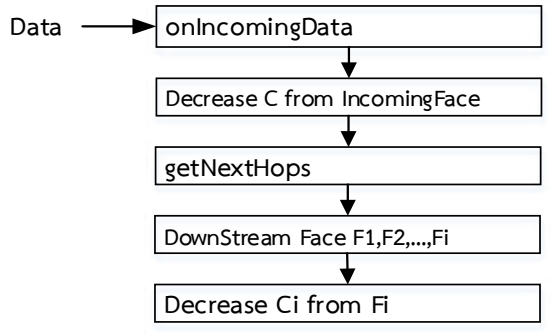

Figure 5 Data processing for LPR algorithm

วิธีการทดลอง

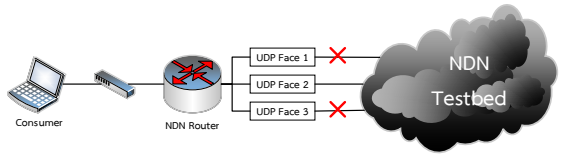

Figure 6 an evaluation scenario

Figure 6 คือ แผนผังการทดลองเพื่อทดสอบอัลกริธึม BR กับ LPR โดยที่การทดลองครั้งนี้ติดตั้ง NFD รุ่น 0.4.1 เชื่อมต่อกันโดยใช้สับเน็ต 192.168.6.0/24 และใช้ Face ประเภท UDP ทั้งหมด และใช้ซอฟต์แวร์ VirtualBox สร้าง Virtual Machine (VM) ติดตั้งระบบปฏิบัติการลินุกส์ CentOS รุ่น 7 มีหน่วยประมวลผล 1 แกน RAM 1024Gb ฮาร์ดดิสก์ 20Gb และกำหนดเครือข่ายแบบ Bridge

เครื่อง Consumer กำหนดให้เชื่อมต่อไปยัง NDN Router ผ่านทาง UDP Face โดยกำหนด Prefix ชื่อ /ndn ให้กับ FIB ซึ่งจะทำให้ทุกแพ็กเก็ตถูกส่งไปยัง NDN Router ขณะเดียวกันเครื่อง NDN Router ก็ใช้ UDP Face เชื่อมต่อเข้าสู่ระบบ NDN Testbed ตาม Figure 7 ผ่านทางโหนด SRRU และทั้ง 3 Face ของ NDN Router มีค่า Routing Cost เท่ากับ 0 และกำหนด Prefix ที่เชื่อมเข้าสู่โหนด SRRU คือ /ndn เมื่อเชื่อมเข้าสู่โหนด SRRU แล้ว ทำการตัดการเชื่อมต่อของ Face ที่ 1 และ Face ที่ 3 เพื่อประเมินประสิทธิภาพของ Convergence Time เมื่อมีเส้นทางการเชื่อมต่อมีความข้อขัดข้อง

Figure 7 NDN Testbed

ที่เครื่อง Consumer ใช้ซอฟต์แวร์ ndnping ทดสอบเชื่อมต่อไปยังโหนดต่าง ๆ ตาม Prefix ใน Table 1 แล้วประเมินค่า Convergence Time และ Lost Rate ของการใช้อัลกอริธึม BR และ LPR ก่อนที่จะนำมาวิเคราะห์ผล โดยในงานวิจัยนี้กำหนดให้การประเมินประสิทธิภาพและประสิทธิผลใช้ช่วงความเชื่อมั่น 95%

แต่ละ Prefix จาก Table 1 ในการทดสอบ Convergence Time แต่ละโหนดถูกส่งแพ็กเก็ตจำนวน 30 ครั้ง สำหรับการตรวจสอบ Loss Rate การเชื่อมต่อไปยังปลายทางจะใช้จำนวนครั้งต่างกัน 3 แบบ คือ 10, 50 และ 100 ครั้งตามลำดับ เพื่อสังเกตประสิทธิภาพการใช้ทรัพยากรของเครื่องทดสอบ และการส่งแพ็กเก็ตแต่ละครั้งจะมีการควบคุมค่าพารามิเตอร์ในระดับเครือข่ายท้องถิ่น ที่ประกอบด้วย ช่องแบนด์วิดท์ 100Mbps และไม่ให้แพ็กเก็ตภายนอกรบกวนระบบ Testbed ในเครือข่ายท้องถิ่นของงานวิจัยนี้

Table 1 Testbed Prefixes

/ndn/br/ufpa	/ndn/ch/unibas
/ndn/de/uni-goettingen	/ndn/edu/arizona
/ndn/edu/byu	/ndn/edu/colostate
/ndn/edu/illinois	/ndn/edu/memphis
/ndn/edu/neu	/ndn/edu/uci
/ndn/edu/ucla	/ndn/edu/ucla/remap
/ndn/edu/wustl	/ndn/es/urjc
/ndn/fr/lip6	/ndn/gov/nist
/ndn/it/unipd	/ndn/jp/ac/osaka-u
/ndn/kr/anyang	/ndn/kr/re/kisti
/ndn/no/ntnu	/ndn/pt/ulusofona/copelabs
/ndn/pt/uminho	/ndn/cn/edu/tongji
/ndn/id/ac/ui	/ndn/nl/tno
/ndn/org/caida	/ndn/com/orange
/ndn/edu/umich	/ndn/th/ac/srru

ผลการประเมิน NFD

ผลการประเมิน Convergence Time

ในการประเมินผลค่า Convergence Time ระหว่างการใช้อัลกอริธึม BR และ LPR โดยผลการทดลองแสดงให้เห็นถึงประสิทธิภาพของ LPR ที่ช่วยให้ Convergence Time ต่ำกว่าการใช้ BR ซึ่งผลการทดลองใน Figure 8 เป็น Convergence Time ของอัลกอริธึม LPR โดยในแกน x คือ โหนดทดสอบมีทั้งสิ้น 30 โหนด และแกน y คือ Convergence Time ที่เกิดขึ้นหลังจากมีการปรับปรุงเส้นทางในหน่วยมิลลิวินาที ซึ่งงานวิจัยนี้ใช้ค่าเฉลี่ยที่ได้จากซอฟต์แวร์ ndnping และ LPR มี Convergence Time โดยรวมอยู่ระหว่าง 5095.5±9.04 ms แตกต่างจากการใช้ BR ซึ่งไม่สามารถเชื่อมต่อกับปลายทางได้ จึงเกิด Timeout และไม่มีผลการปรับปรุงเส้นทางแต่อย่างใด

ผลการประเมิน Loss Rate

Loss Rate เป็นปัจจัยสำคัญที่จะทำให้ประสิทธิภาพและประสิทธิผลให้การส่งต่อแพ็กเก็ตได้ในปริมาณที่สูงขึ้น ผลที่เกิดจากการประเมินระหว่างการใช้อัลกอริธึม BR และ LPR แสดงใน Figure 9 ในแกน x คือ จำนวนแพ็กเก็ตที่ส่งไปยังปลายทาง และแกน y คือสัดส่วนร้อยละของ Loss Rate จากผลการทดลองพบว่าการใช้อัลกอริธึม BR เกิดการสูญหายของแพ็กเก็ตร้อยละ 100 เพราะการตอบสนองต่อปัญหาการเชื่อมต่อมีความล่าช้า และอัลกอริธึม BR ไม่เลือกเส้นทางใหม่จนกว่าจะได้รับแพ็กเก็ตประเภท NACK และลบเส้นทางดังกล่าวออกจาก FIB แต่ในการทดลองบนระบบ Testbed ในงานวิจัยนี้ 1) แพ็กเก็ต NACK ไม่ถูกส่งกลับมายัง Consumer เพราะ NDN Router มีตาราง FIB และตาราง PIT ซึ่งรอการเกิด Timeout และถูกอัพเดตด้วยแพ็กเก็ต Interest อย่างต่อเนื่อง 2) การปรับค่า Cost ของอัลกอริธึม BR ถูกกำหนดเองโดยผู้ดูแลระบบทำให้ BR ต้องเลือกเส้นทางเดิมอยู่ตลอดเวลาถึงแม้จะไม่สามารถเชื่อมต่อได้ จึงทำให้การส่งแพ็กเก็ตจำนวน 10, 50 และ 100 ไปยัง Testbed ไม่สามารถทำได้ แต่เมื่อปรับเปลี่ยนมาใช้อัลกอริธึม LPR จะเห็นว่าเมื่อส่งแพ็กเก็ตจำนวน 10 แพ็กเก็ตจะมี Loss Rate อยู่ที่ร้อยละ 2.08±4.0 เมื่อ

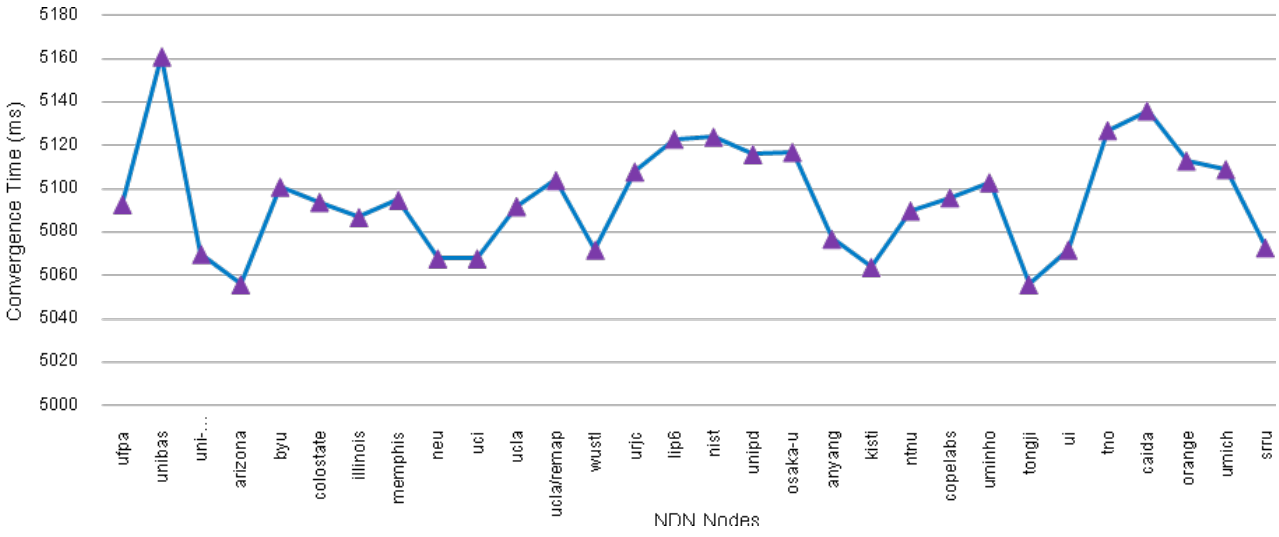

Figure 8 LPR Convergence Time

เพิ่มจำนวนแพ็กเก็ตเป็น 50 แพ็กเก็ต Loss Rate มีสัดส่วนลดลงเหลือร้อยละ 1.67±1.78 และในกรณีที่ส่งแพ็กเก็ตจำนวน 100 แพ็กเก็ต Loss Rate ต่ำลงเหลือร้อยละ 0.9±1.10 จากผลการทดลองนี้แสดงถึงอัลกอริธึม LPR สามารถที่จะปรับปรุง Routing Cost และช่วยให้ NFD สามารถเลือกเส้นทางส่งแพ็กเก็ต Interest ทาง Face ที่ใช้งานได้จริงได้อย่างรวดเร็ว และช่วยเพิ่มทั้งประสิทธิภาพและประสิทธิผลการเชื่อมต่อสื่อสารในเครือข่าย NDN ได้

ผลการประเมินประสิทธิภาพ

การประเมินประสิทธภาพของการใช้ทรัยพยากรณ์จากงานวิจัยนี้ได้ทดสอบการใช้ CPU และ RAM โดยทำการเก็บข้อมูลทุก ๆ 5 นาทีระหว่างการทดลอง จำนวน 30 ครั้ง พบว่า อัตราการจัดการ Memory ใน

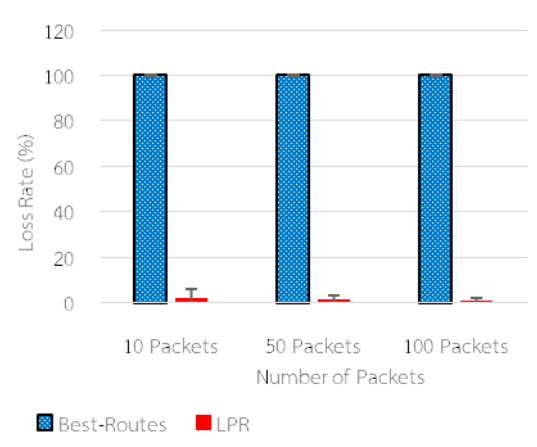

Figure 9 Loss Rate of Best-Route and LPR

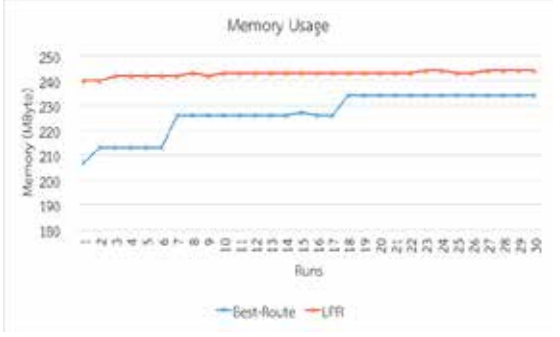

Figure 10 Memory Load

Figure 10 จะเห็นว่าอัลกอริธึม LPR มีอัตราการใช้ Memory สูงกว่าการใช้อัลกอริธึม BR ซึ่งเกิดจากการการเพิ่มฟังก์ชันการปรับปรุง Routing Cost และการลด Prefix เพื่อค้นหาเส้นทางในเครือข่าย NDN แต่การใช้ Memory ของอัลกริธึม LPR ค่อนข้างอยู่ในระดับคงที่ และสูงกว่าการใช้อัลกอริธึม BR เพียงเล็กน้อย

สำหรับในเป็นร้อยละ CPU คงเหลือหลังจากใช้ NFD ที่เปิดใช้งานอัลกอริธึม BR และ LPR ซึ่งในแกน x แสดงอัลกอริธึมที่ใช้ และในแกน y คือ ร้อยละของ CPU คงเหลือ โดยผลจากการทดลองเห็นได้ชัดว่าการใช้ NFD ที่มีอัลกอริธึม BR และ LPR มีอัตรา CPU คงเหลืออยู่ในระดับเดียวกันคือมากกว่าร้อยละ 99 ดังนั้น BR และ LPR ไม่แตกต่างกันอย่างมีนัยสำคัญของการใช้ CPU

สรุปและอภิปรายผล

สถาปัตยกรรมเครือข่าย NDN ถูกสร้างเพื่อให้เหมาะกับการนำไปใช้ในยุคที่ Content สำคัญสำหรับแอพพลิเคชันในเครือข่าย NDN จึงใช้ Named Prefix เป็นเครื่องมือในการรับ Content โดยใช้แพ็กเก็ต Interest และ Data การใช้ Named Prefix เป็นเครื่องมือในการเร้าแพ็กเก็ตช่วยให้ NDN เชื่อมต่อเครือข่ายได้แบบ Multipath โดยในปัจจุบัน NDN ใช้อัลกอริธึม BR และมีการตัดสินใจด้วยการใช้ Routing Cost ซึ่ง ยังมีปัญหาหลายด้าน เช่น ตอบสนองต่อการเลือกเส้นทางที่เชื่อมต่อล้มเหลวล่าช้า งานวิจัยนี้จึงพัฒนาอัลกอริธึม LPR ช่วยปรับปรุงเส้นทางเลือกเส้นทางส่งแพ็กเก็ตให้สนับสนุนการส่งแพ็กเก็ตแบบ Multipath ได้ดีขึ้น โดยได้จัดการวิธีส่งแพ็กเก็ต Interest จากการปรับปรุงวิธีกำหนด Routing Cost แบบยืดหยุ่น

งานวิจัยนี้ได้นำแนวคิดพัฒนาต่อยอดในซอฟต์แวร์ NFD ทดสอบบนระบบ NDN Testbed และผลการทดลองแสดงถึงประสิทธิภาพและประสิทธิผลการปรับปรุงเส้นทางส่งแพ็กเก็ต Interest ซึ่งช่วยให้ LPR มี Lost Rate ต่ำว่าการใช้ BR มากกว่าร้อยละ 90 ได้

เอกสารอ้างอิง

1. Socolofsky T, Kale C. A TCP/IP Tutorial. IETF; 1991 Jan. Report No.: 1991.
2. Liang J, Jiang J, Duan H, Li K, Wan T, Wu J. When HTTPS Meets CDN: A Case of Authentication in Delegated Service. In Washington, DC, USA; 2014. p. 67–82.
3. Zhang L, Afanasyev A, Burke J, Jacobson V, claffy kc, Crowley P, et al. Named Data Networking. ACM SIGCOMM Computer Communication Review. 2014 Jul;44(3):66–73.
4. Afanasyev A, Shi J. Named Data Networking Forwarding Daemon [Internet]. 2016 [cited 2016 Apr 9]. Available from: http://redmine.named-data.net/projects/nfd

5. Jacobson V, Smetters D, Thornton J, Plass M, Briggs N, Braynard R. Networking Named Content. Communication of the ACM. 2012 Jan;55(1):117–24.
6. Jacobson V. A New Way to look at Networking [Internet]. 2006 [cited 2014 May 1]. Available from: https://www.youtube.com/watch?v=8Z685OF-PS8
7. Cruikshank D. NSF Future Internet Architecture Project [Internet]. 2010 [cited 2015 Jun 30]. Available from: http://www.nets-fia.net/
8. Afanasyev et al A. NFD Developer's Guide. University of California, Los Angeles; 2014 Aug. Report No.: NDN-0021. p. 1-60.
9. Shi J. Best Route Strategy version 2: recognize consumer retransmission [Internet]. 2014 [cited 2015 Aug 1]. Available from: https://redmine.named-data.net/issues/1871
10. Hoque M, Syed A, Alyyan A, Zhang B, Zhang L, Wang L. NLSR: Named-data Link State Routing Protocol. In Hong Kong, China; 2013. p. 15–20.