

ต้นแบบการติดตั้งซอฟต์แวร์คอมพิวเตอร์โดยใช้ตรรกศาสตร์คลุมเครือช่วยค้นหา

เส้นทางการดาวน์โหลดบนเครือข่ายเพียร์ทูเพียร์

Framework for Installation Software System Based on P2P By Using Fuzzy logic Download Path Selection

ยุรนันท์ มุขอ¹ และ ดร.ธงชัย แก้วกิริยา²

¹อาจารย์ ประจำคณะบริหารธุรกิจ

สถาบันเทคโนโลยีไทย-ญี่ปุ่น กรุงเทพมหานคร

²อาจารย์ ประจำคณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีไทย-ญี่ปุ่น กรุงเทพมหานคร

ABSTRACT – Operating system installation on multiple computers is usually done by creating an image file from the server, then clients in the network download the image file for installation. This research proposes a framework for installing operating systems on multiple computers by Peer-to-Peer (P2P) connection and a Fuzzy logic algorithm. The Fuzzy logic is used for route calculations to reduce network congestion. The results show the proposed framework can reduce network congestion by 4.32% for 5 computer clients and 2.55% for 10 computer clients when compared with normal P2P downloads.

KEY WORDS – Peer to Peer; Fuzzy logic; Congestion;

บทคัดย่อ – การติดตั้งระบบปฏิบัติการบนเครื่องคอมพิวเตอร์ครั้งละหลายเครื่องส่วนใหญ่จะใช้วิธีสร้างอิมเมจไฟล์จากเครื่องคอมพิวเตอร์ต้นฉบับ จากนั้นเครื่องคอมพิวเตอร์ในเครือข่ายจะเข้ามาดาวน์โหลดเพื่อนำไฟล์ไปติดตั้ง งานวิจัยฉบับนี้นำเสนอต้นแบบการดาวน์โหลดไฟล์บนเครือข่ายเพียร์ทูเพียร์ และใช้ตรรกศาสตร์คลุมเครือ (Fuzzy logic) เข้ามาช่วยคำนวณเลือกเส้นทางการดาวน์โหลดเพื่อช่วยลด Congestion ในเครือข่ายลง โดยเปรียบเทียบผลระหว่างการดาวน์โหลดไฟล์แบบเพียร์ทูเพียร์ กับการดาวน์โหลดไฟล์แบบเพียร์ทูเพียร์ แบบมีฟัซซี่ลอจิก ผลการทดลองฟัซซี่ลอจิก สามารถช่วยลดความหนาแน่นในเครือข่ายลงดังนี้ เครือข่ายเครื่องคอมพิวเตอร์จำนวน 5 เครื่อง สามารถลดความหนาแน่นได้ร้อยละ 4.32 และเครือข่ายเครื่องคอมพิวเตอร์จำนวน 10 เครื่อง สามารถลดความหนาแน่นได้ร้อยละ 2.55

คำสำคัญ – เพียร์ทูเพียร์; ตรรกศาสตร์คลุมเครือ; ความหนาแน่นในเครือข่าย

1. บทนำ

ปัจจุบันคอมพิวเตอร์ถือได้ว่าเป็นปัจจัยอย่างหนึ่งที่มิขาดขาดสำคัญทางด้านการเรียนการสอน ไม่ว่าจะเป็นระดับมัธยมศึกษา หรือระดับอุดมศึกษา ล้วนแล้วแต่มีห้องปฏิบัติการคอมพิวเตอร์ ซึ่งเครื่องคอมพิวเตอร์ในแต่ละห้องปฏิบัติการจะมีจำนวนมากกว่า 40 เครื่องขึ้นไป และมีแนวโน้มสูงขึ้นในทุกๆปี เมื่อมีการใช้คอมพิวเตอร์ในการเรียนการสอนประกอบกับผู้ใช้งานที่สลับเปลี่ยนกันใช้งาน ปัญหาหลายอย่างที่เกี่ยวกับคอมพิวเตอร์ก็ตามมาเช่น คอมพิวเตอร์บูทเข้าใช้งานไม่ได้, คอมพิวเตอร์ติด

ไวรัส และเครื่องคอมพิวเตอร์ช้า การแก้ปัญหาที่ดีที่สุดคือการ Format เครื่องและติดตั้งระบบปฏิบัติการใหม่ แต่ในการติดตั้งระบบปฏิบัติการใหม่จำเป็นจะต้องใช้เวลาหลายชั่วโมงต่อเครื่อง

การติดตั้งระบบปฏิบัติการบนเครื่องคอมพิวเตอร์ที่หลายเครื่องในเวลาเดียวกัน ปัจจุบันมีซอฟต์แวร์ที่นิยมนำมาใช้งานคือ GhostCast Server [1] ซึ่งเป็นการทำไฟล์ต้นฉบับ (Image file) จากเครื่องคอมพิวเตอร์ต้นฉบับ จากนั้นนำไฟล์ที่ได้ไปเก็บไว้ที่เครื่อง Server เพื่อให้เครื่องคอมพิวเตอร์ Client ได้เข้ามา

ดาวน์โหลดคอมพิวเตอร์ไปเก็บไว้ที่เครื่อง จากนั้นทำการแตกอิมเมจไฟล์ ที่เครื่องปลายทางที่ละเครื่อง ดังกล่าวนี้นี้เป็นวิธีการติดตั้งระบบปฏิบัติการแบบ Client - Server ซึ่งมีข้อจำกัดคือหากเครื่อง Server เกิดขัดข้องระหว่างที่ Client กำลังดาวน์โหลดไฟล์อยู่จะทำให้การติดตั้งระบบปฏิบัติการที่ดำเนินการอยู่ทั้งหมดไม่สามารถดำเนินการติดตั้งต่อไปได้

จากข้อจำกัดของการรับ - ส่งข้อมูลแบบ Client - Server ที่กล่าวมาโปรโตคอล Peer-to-Peer (P2P) มีความสามารถในการสนับสนุนการรับ - ส่งไฟล์แบบแยกส่วนและมีช่องทางการสื่อสารมากกว่าหนึ่งช่องทางในการรับ - ส่งไฟล์มาใช้งานร่วมกับการติดตั้งระบบปฏิบัติการบนเครื่องคอมพิวเตอร์โดยการนำอิมเมจไฟล์ เก็บไว้ที่เครื่องแม่ข่าย (Server) เพื่อการกระจายอิมเมจไฟล์ ให้แก่เครื่องลูกข่าย (Client) ผ่านทางระบบเครือข่ายที่เชื่อมต่อระหว่างกันโดยเป็นการเชื่อมต่อกันแบบ P2P อย่างไรก็ตามเครือข่าย P2P ยังคงมีข้อจำกัดอยู่คือ การเข้าลำดับคิวการให้บริการ ดังนั้นผู้วิจัยจึงมีแนวคิดที่จะนำฟัซซี่ลอจิก มาช่วยค้นหาเส้นทางที่ดีที่สุดในการดาวน์โหลดอิมเมจไฟล์ เพื่อช่วยลดความหนาแน่นของข้อมูลในเครือข่ายการติดตั้งระบบปฏิบัติการให้ยิ่งขึ้น

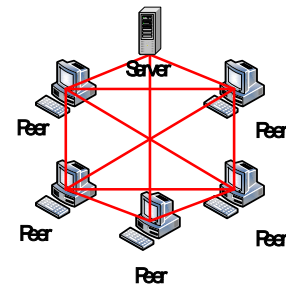
2. พื้นฐานและงานวิจัยที่เกี่ยวข้อง

2.1 ระบบเครือข่ายแบบ Peer-to-Peer (P2P)

Peer-to-Peer (P2P) [2] คือเครือข่ายการเชื่อมต่อกันระหว่างเครื่องคอมพิวเตอร์กับเครื่องคอมพิวเตอร์ตั้งแต่ 2 เครื่องขึ้นไปผ่านอุปกรณ์เน็ตเวิร์คสวิตช์ โดยไม่จำเป็นต้องมีเครื่องเซิร์ฟเวอร์ควบคุมการทำงานของเครื่องคอมพิวเตอร์ภายในเครือข่าย เนื่องจากการส่งข้อมูลในเครือข่ายเพียร์ทูเพียร์นั้นเครื่องคอมพิวเตอร์สามารถเป็นได้ทั้ง Server และ Client ในเครื่องเดียวกันได้ ซึ่งในการเชื่อมต่อกันของเครือข่ายเพียร์ทูเพียร์ นั้นยังสามารถแบ่งออกได้เป็น 3 รูปแบบคือ

- 1) รูปแบบ Pure Peer-to-peer เป็นรูปแบบการเชื่อมต่อกันระหว่างเครื่องคอมพิวเตอร์กับเครื่องคอมพิวเตอร์ในเครือข่าย เพื่อแลกเปลี่ยนข้อมูลซึ่งกันและกันโดยไม่มีเครื่อง Server คอยควบคุมดูแล
- 2) รูปแบบ Hybrid Peer-to-Peer เป็นรูปแบบการเชื่อมต่อกันระหว่างเครื่องคอมพิวเตอร์กับเครื่องคอมพิวเตอร์ในเครือข่าย เพื่อแลกเปลี่ยนข้อมูลซึ่งกันและกันแต่จะมี

มีเครื่อง Server ทำหน้าที่ควบคุมการรับ - ส่งข้อมูลกันในเครือข่าย



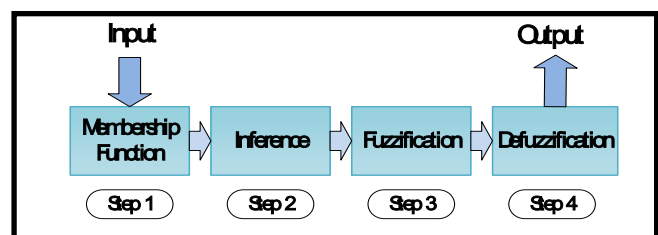
รูปที่ 1. แสดง Hybrid Peer-to-Peer Model

- 3) รูปแบบ Super-Peer เป็นรูปแบบการเชื่อมต่อกันระหว่างเครื่องคอมพิวเตอร์กับเครื่องคอมพิวเตอร์ในเครือข่าย เพื่อแลกเปลี่ยนข้อมูลซึ่งกันและกันแต่จะเป็นการนำทั้ง 2 รูปแบบการเชื่อมต่อแบบเพียร์ทูเพียร์มารวมกันคือ Pure P2P และ Hybrid P2P Super Peer

งานวิจัยฉบับนี้ได้เลือกใช้รูปแบบ Hybrid Peer-to-Peer มาใช้ในการส่งอิมเมจไฟล์ ภายในเครือข่ายเครื่องคอมพิวเตอร์จำนวน 5 เครื่อง และ 10 เครื่อง

2.2 ฟัซซี่ลอจิก (Fuzzy logic)

Fuzzy logic คือตรรกศาสตร์การคำนวณแบบคลุมเคลือ [3] เป็นผลงานวิทยานิพนธ์ระดับปริญญาเอกในปีค.ศ. 1965 ซึ่งผู้คิดค้นคือ L. A. Zadeh เป็นตรรกะในการคิดคำนวณประเภทหนึ่งที่ไม่ได้มีแค่คำตอบแค่เพียง ถูก หรือ ผิด แต่จะมีคำตอบที่คลุมเครือ Fuzzy เพิ่มขึ้นมาคือ อาจจะถูก หรือ อาจจะผิด ซึ่งในการคำนวณหาคำตอบของฟัซซี่ลอจิกนั้น จำเป็นจะต้องใช้ผู้เชี่ยวชาญในการกำหนดตัวแปร เพื่อนำมาเปรียบเทียบกับกฎพื้นฐาน Rule base ในการตีความหาเหตุผลของคำตอบให้ตรงกับความต้องการของระบบฟัซซี่ลอจิกที่ออกแบบขึ้น หลักการคำนวณของฟัซซี่ลอจิก อธิบายได้ตามรูปที่ 2



รูปที่ 2. โครงสร้างพื้นฐานของการประมวลผลแบบฟัซซี่

จากรูปที่ 2 สามารถอธิบายหลักการทำงานของระบบฟัซซี่โดยแบ่งออกเป็น 4 ขั้นตอนคือ

ขั้นตอนที่ 1 Membership Function เป็นขั้นตอนการกำหนดค่าความเป็นสมาชิกให้กับตัวแปรที่ถูกกำหนดขึ้น เพื่อนำตัวแปรที่เป็นคำพูดแบบทวินัยมาแปลงให้เป็นภาษาของฟัซซี่ลอจิก หรือที่เรียกว่ากระบวนการ Fuzzification เป็นขั้นตอนการแปลงข้อมูลอินพุต ของตัวแปรที่กำหนดขึ้นซึ่งตัวแปรที่กำหนดขึ้นนั้น จะอยู่ในรูปแบบของเซตฟัซซี่หรือเรียกว่า คำตัวแปรทางภาษา (Linguistic Variable) และมีค่าความเป็นสมาชิกในตัวแปร

ขั้นตอนที่ 2 Inference เป็นขั้นตอนการสร้างความสัมพันธ์ระหว่างตัวแปรอินพุตและเอาต์พุตของฟัซซี่ลอจิกด้วยการทำฐานกฎ (Rule base) ในรูปแบบ ถ้า (If) และ (And) หรือ (Or) จากนั้น (Then) และเก็บไว้ในฐานข้อมูล (Database)

ขั้นตอนที่ 3 Fuzzification เป็นขั้นตอนการตัดสินใจเลือกการกระทำตามเอาต์พุตของระบบฟัซซี่ลอจิกที่ได้ออกแบบไว้ โดยการเปรียบเทียบกับฐานกฎของระบบฟัซซี่

ขั้นตอนที่ 4 Defuzzification เป็นขั้นตอนการคำนวณแปลงค่าเอาต์พุตของฟัซซี่ลอจิกที่ได้จากการสรุปผลการตัดสินใจในขั้นตอนที่ 3 ให้อยู่ในรูปแบบทวินัย

2.3 งานวิจัยที่เกี่ยวข้อง

กิตติศักดิ์ วัฒนกุล [4] ได้นำเสนอวิธีการนำเทคนิคการกระจายไฟล์แบบเพียร์ทูเพียร์ (Peer-to-Peer) โดยใช้โปรโตคอลบิททอเรนท มาประยุกต์ใช้งานร่วมกับการดาวน์โหลดไฟล์ วิดีโอโอเอิร์นนิง ช่วยลดความคับคั่งในการรับ - ส่งข้อมูลจากเครื่อง Server และช่วยให้การดาวน์โหลดไฟล์วิดีโอสามารถทำได้เร็วยิ่งขึ้น จากนั้นทำการทดสอบประสิทธิภาพทางด้านเวลาในการกระจายไฟล์ ด้วยโปรโตคอล HTTP เปรียบเทียบกับการกระจายไฟล์แบบเพียร์ทูเพียร์

วีระชัย เข้มวจิ [5] นำเสนอการค้นหาเส้นทางโดยใช้ฟัซซี่ลอจิก เพื่อลดปัญหาความหนาแน่นของการจราจรในเครือข่าย โดยจะทำการเปรียบเทียบผลการคำนวณเลือกเส้นทางด้วยฟัซซี่ลอจิก มาเปรียบเทียบกับโปรโตคอลค้นหาเส้นทางที่มีอยู่ได้แก่ โปรโตคอล DSDV (Destination Sequenced Distance Vector), โปรโตคอล DSR (Dynamic Source Routing) และ โปรโตคอล AODV (Ad-hoc On-demand Distance Vector) ด้วยการทดลองบนระบบจำลองการทำงานของเครือข่ายเฉพาะกิจแบบเคลื่อนที่

โดยใช้โปรแกรมเน็ตเวิร์กซิมูเลเตอร์ 2 (Network Simulator 2: NS-2) และตั้งชื่อโปรโตคอลใหม่ที่น่าสนใจว่า FzAODV จากนั้นทำการเปรียบเทียบผลการทดลองประสิทธิภาพการทำงานของโปรโตคอล FzAODV, DSR และ AODV ซึ่งผลการทดลองพบว่าโปรโตคอล FzAODV ที่น่าสนใจมีประสิทธิภาพดีกว่าโปรโตคอลอื่น

A. Mahapatra et al. [6] ได้นำเสนอการนำ Fuzzy Inference System เข้ามาช่วยจัดการระหว่าง Peer ที่ทำการดาวน์โหลดข้อมูลอยู่ในเครือข่าย Bittorrent P2P โดยใช้ฟัซซี่ลอจิกแบบ Mamdani-Type มาใช้งาน กำหนดอินพุต จำนวน 3 ตัวแปร คือ Number of File Share, Upload Bandwidth และ Seeding Time เพื่อให้ได้ผลเอาต์พุต คือ Reputation จากนั้นนำไปเก็บที่ฐานข้อมูลส่วนกลางเพื่อเขียน Algorithm ช่วยแนะนำให้ Peer ที่อยู่ในเครือข่ายเรียกใช้งาน

A. Mahapatra et al. [7] ได้สร้างโมเดลจัดการ Peer ในเครือข่าย Bittorrent P2P เพื่อคัดแยกและช่วยลดจำนวน Peer ที่อาจก่อให้เกิดอันตรายกับเครือข่าย Peer ที่มีคุณภาพต่ำ และ Peer ที่มีการปล่อยไฟล์เสีย โดยการนำฟัซซี่ลอจิก เข้ามาเพิ่มความน่าเชื่อถือระหว่างผู้ใช้งานกับเครือข่ายโดยกำหนดตัวแปรค่าอินพุต จำนวน 3 ตัวแปร คือ Time Spent in Forum, No. of Post Count และ Bandwidth Donation และกำหนดแบ่งค่า เอาต์พุต จำนวน 5 Linguistic คือ Very Low, Low, Medium, High, Very High เพื่อนำค่าที่ได้ไปสร้าง Algorithm ในการเรียกใช้งาน

A. Mahapatra and N. Tarasia [8] ได้นำเสนอการจัดการเครือข่าย Bittorrent P2P เพื่อให้ปลอดภัยจาก Peer ที่ไม่เป็นประโยชน์ซึ่งอาจเป็นอันตรายกับเครือข่ายได้ เช่น Fake file ด้วยการจัดลำดับ Peer ด้วยการอนุมานค่าโดยใช้ฟัซซี่ลอจิก กำหนดอินพุตเป็น 3 ค่าคือ Quality, Popularity และ Size of the File Share จากนั้นนำค่าที่ได้ไปสร้าง Algorithm ในการเรียกใช้งาน

S. N. Ingole and M. Nashipudi [9] ได้นำเสนอแนวทางแก้ปัญหาความแออัด และปัญหาคอขวด Bottleneck ภายในเครือข่ายคอมพิวเตอร์ โดยการนำ Fuzzy Logic Controller (FLC) และ Fuzzy Inference System (FIS) เข้ามาช่วยจัดการการจราจรเพื่อป้องกันไม่ให้แพ็กเก็ตเกิดการตกค้างและเพิ่มประสิทธิภาพการควบคุมความแออัดผลที่จะได้รับการกำหนดระดับของการใช้คิวและให้เพียงพอรดับของการควบคุมความแออัด

B. S. Kiruthika Devi et al. [10] นำเสนอการใช้ Fuzzy Method จัดการความหนาแน่นบน Core Router Buffer ด้วยวิธีจัดการคิวในการรับส่งไฟล์ Voice Over IP บน Router เพื่อควบคุมบัฟเฟอร์ของแพ็คเก็ตในการส่งข้อมูล โดยใช้เทคนิค Adaptive Drop Tail Fuzzy Logic (ADT-FL) โดยใช้ตัวแปรอินพุต คือ Traffic Conditions และ Available Link Bandwidth ผลที่ได้คือคิวที่จัดการลำดับคิวการรับ – ส่งข้อมูล

R. Lal and A. Chiou [11] นำเสนอการใช้ฟัซซี่ลอจิกจัดการความหนาแน่นที่เกิดขึ้นบน Network Edge and Bottleneck Route โดยใช้ Computational Intelligence (CI) คู่กับ Fuzzy Logic เพื่อจัดการปัญหา Congestion

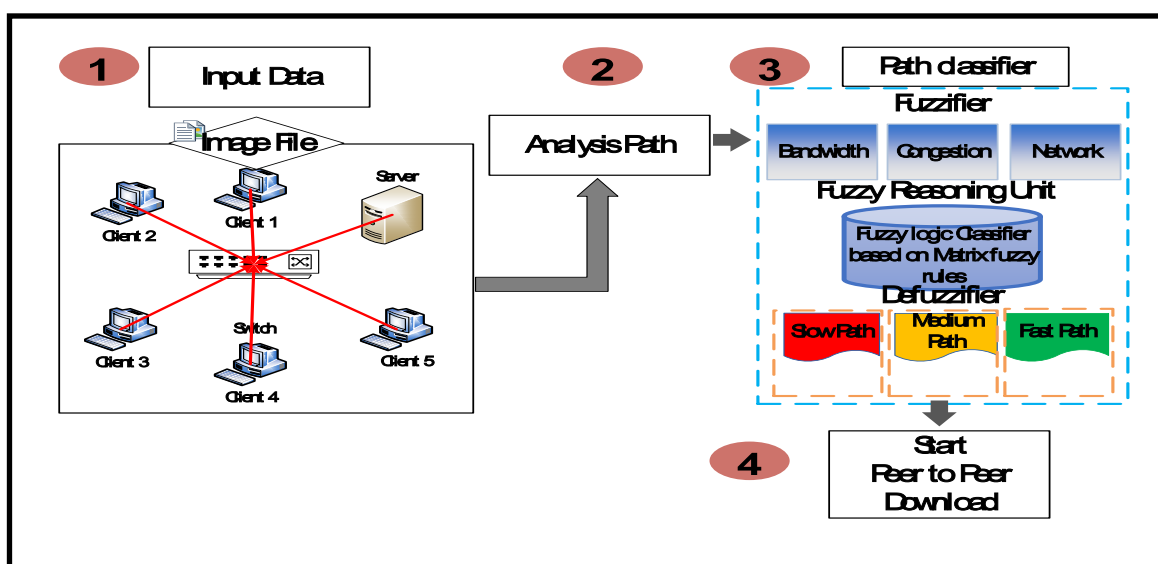
B. Safaeizadeh et al. [12] นำเสนอ Adaptive Drop Tail Using Fuzzy Logic (ADT-FL) โดยใช้ DT Algorithm ในการหาค่าฟัซซี่และใช้อินพุต คือ Traffic Intensity และ Available Link Bandwidth ส่วนของเอาต์พุตที่ได้คือ การจัดการคิว Queue Limit แบ่งออกเป็น 5 ประเภท คือ (VL, L, M, H, VH) จากนั้นทำการเปรียบเทียบ Algorithms ระหว่าง DT ,RED และ ADT-FL Algorithms และเปรียบเทียบค่า 3 ค่าคือ Packet Loss Rate, Throughput และ Delay

จากตัวอย่างงานวิจัยที่กล่าวมาสามารถแบ่งเป็นกลุ่มได้ 3 กลุ่ม กลุ่มที่หนึ่งคืองานวิจัยที่เกี่ยวข้องกับการนำ P2P เข้ามาประยุกต์ใช้กับงานวิจัย กลุ่มที่สองคืองานวิจัยที่นำฟัซซี่ลอจิกเข้ามาปรับปรุงการทำงานของ P2P ซึ่งในแต่ละงานวิจัยจะมีการใช้กฎพื้นฐานอินพุตของฟัซซี่ลอจิกที่แตกต่างกันไปขึ้นอยู่กับความเหมาะสมและจุดประสงค์ของแต่ละงานวิจัย กลุ่มสุดท้ายคือกลุ่มงานวิจัยที่ช่วยลดปัญหาความหนาแน่น Congestion ในเครือข่าย

งานวิจัยนี้จึงมีแนวคิดในการออกแบบต้นแบบการติดตั้งซอฟต์แวร์คอมพิวเตอร์โดยการนำเอา P2P เข้ามาช่วยในการดาวน์โหลดไฟล์ และนำเอาฟัซซี่ลอจิก มาใช้ในการคำนวณเลือกเส้นทางการดาวน์โหลดโหนดคอมพิวเตอร์ไฟล์ใน 4 ช่วงเวลาการดาวน์โหลด คือช่วงเวลาที่เริ่มต้นการดาวน์โหลด, ช่วงเวลาการดาวน์โหลด 25%, ช่วงเวลาการดาวน์โหลด 50%, และช่วงเวลาการดาวน์โหลด 75% โดยพิจารณาจากตัวแปรอินพุตที่สำคัญคือ Bandwidth, Congestion และ Network ซึ่งเป็นปัจจัยที่มีผลต่อการเกิดความหนาแน่น เข้ามาช่วยเลือกเส้นทางการดาวน์โหลดไฟล์บนเครือข่ายเพียร์ทูเพียร์

3. กรอบแนวคิดในการวิจัย

ออกแบบโมเดลต้นแบบการติดตั้งซอฟต์แวร์คอมพิวเตอร์แบบเพียร์ทูเพียร์ โดยใช้ Fuzzy logic ช่วยคำนวณเลือกเส้นทางการดาวน์โหลด ประกอบไปด้วย 4 โมดูลดังรูปที่ 3



รูปที่ 3. โมเดลตัวต้นแบบการติดตั้งซอฟต์แวร์คอมพิวเตอร์แบบ P2P โดยใช้ตรรกศาสตร์คลุมเครือ

จากรูปที่ 3 สามารถอธิบายหลักการทำงานของโมเดลโดยแบ่งออกเป็น 4 ขั้นตอนคือ

ขั้นตอนที่ 1 Input Data คือกลุ่มคอมพิวเตอร์ที่เชื่อมต่อกันอยู่ภายในเครือข่าย P2P และมีอินเทอร์เน็ตของเครื่องคอมพิวเตอร์ต้นฉบับที่จะทำการส่ง

ขั้นตอนที่ 2 Analysis Path เป็นขั้นตอนการระบุเส้นทาง Path ทั้งหมดที่มีการเชื่อมต่อกันของกลุ่มเครื่องคอมพิวเตอร์แบบ Complete Graph จะได้เส้นทางทั้งหมดที่เครื่องเซิร์ฟเวอร์ต้นทางสามารถเดินทางไปกลุ่มเครื่องคอมพิวเตอร์ทั้งหมด

ขั้นตอนที่ 3 Path Classifier ขั้นตอนนี้เป็นการเสนอแนวคิดการนำตรรกศาสตร์คลุมเครือ (Fuzzy Logic) มาใช้ในการเลือกเส้นทางในการคำนวณโหลดคอมพิวเตอร์บนเครือข่ายเพียร์ทูเพียร์ เนื่องจากการคำนวณโหลดไฟล์แบบเพียร์ทูเพียร์ จะมีการเลือกเส้นทางแบบสุ่ม ดังนั้นผู้วิจัยจึงเสนอแนวทางการเพิ่ม Fuzzy Logic โดยการกำหนด Rule Base จากนั้นกำหนดรูปแบบและทำการเลือก Membership Function ใน Fuzzy logic ซึ่งจะเลือกตามความเหมาะสมและครอบคลุมกับการเลือกเส้นทางที่ดีที่สุดในการคำนวณโหลด ที่รับเข้ามาผ่านทางอินเทอร์เน็ตและกำหนดความเป็นสมาชิก ให้ค่าอินพุตและฟังก์ชันความเป็นสมาชิกให้เหมาะสมตามกฎที่กำหนดไว้ ด้วยวิธีการฟัซซีลอจิกโดยการทำการออกแบบฟัซซีลอจิก (Fuzzy Logic Controller) ประกอบด้วยขั้นตอน 3 ขั้นตอนดังนี้

3.1) การทำฟัซซีฟิเคชันของข้อมูลนำเข้า (Fuzzyfication) เป็นขั้นตอนการกำหนดข้อมูลนำเข้าข้อมูลให้ระบบ โดยการแปลงค่าอินพุต แบบทวินัยให้เป็นการอินพุตแบบตัวแปรฟัซซีและมีค่าลำดับชั้น (Linguistic Variable) ประกอบด้วยอินพุตจำนวน 3 ตัวแปร ได้แก่ ตัวแปร Bandwidth ตัวแปร Congestion และตัวแปร Network ส่วนตัวแปรเอาต์พุตจะประกอบไปด้วย 1 ตัวแปรคือ Path

ตัวแปรอินพุต Bandwidth คือขนาดของช่องสัญญาณกำหนดค่าความกว้างของช่องสัญญาณ 1000 Mbps ซึ่งในแต่ละตัวแปรจะมีค่า Linguistic อยู่ 3 ระดับ คือ ต่ำ (Low) จะอยู่ในช่วง 0 Mbps ถึง 500 Mbps ค่าปานกลาง (Medium) จะอยู่ที่ช่วง 250 Mbps ถึง 750 Mbps และค่าสูง (High) จะอยู่ที่ช่วง 500 Mb ถึง 1000 Mbps

ตัวแปรอินพุตที่ Congestion คือค่าความหนาแน่นของข้อมูลในเครือข่าย กำหนดค่าความหนาแน่นของช่องสัญญาณ 100 เปอร์เซ็นต์ ซึ่งในแต่ละตัวแปรจะมีค่า Linguistic อยู่ 3

ระดับ คือ ต่ำ (Low) จะอยู่ในช่วงระหว่าง 0 ถึง 50 เปอร์เซ็นต์ ค่าปานกลาง (Medium) จะอยู่ที่ช่วงระหว่าง 25 ถึง 75 เปอร์เซ็นต์ และค่าสูง (High) จะอยู่ที่ช่วงระหว่าง 50 ถึง 100 เปอร์เซ็นต์

ตัวแปรอินพุต Network คือ ความเร็วของลิงก์เน็ตเวิร์ก กำหนดค่าขนาดของลิงค์ค่าต่ำที่ 1000 Mbps ซึ่งในแต่ละตัวแปรจะมีค่า Linguistic อยู่ 3 ระดับ คือ ต่ำ (Ethernet) จะอยู่ในช่วง 0 Mbps ถึง 10 Mbps ค่าปานกลาง (Fast Ethernet) จะอยู่ที่ช่วง 11 Mbps ถึง 100 Mbps และค่าสูง (Gigabit) จะอยู่ที่ช่วง 101 Mbps ถึง 1000 Mbps

ตัวแปรเอาต์พุต Path คือผลจากการคำนวณหาเส้นทางทั้งหมดด้วยทฤษฎี Complete Graph และเลือกเส้นทางที่ดีที่สุดภายในเครือข่ายเพียร์ทูเพียร์ ซึ่งจากตัวอย่างมีจำนวนโหนดอยู่ 5 โหนดด้วยกันจึงได้เส้นทางทั้งหมด 25 เส้นทาง จากนั้นทำการกำหนดค่าขนาดของเส้นทางออกเป็น 3 เส้นทาง ซึ่งในแต่ละตัวแปรจะมีค่า Linguistic อยู่ 3 ระดับ คือ เส้นทางที่มีความเร็วต่ำ (Slow) จะอยู่ในช่วงระหว่าง 0 ถึง 12.5 เส้นทางที่มีความเร็วปานกลาง (Medium) จะอยู่ที่ช่วงระหว่าง 6.25 ถึง 18.5 และเส้นทางที่มีความเร็วสูง (Fast) จะอยู่ที่ช่วงระหว่าง 18.5 ถึง 25

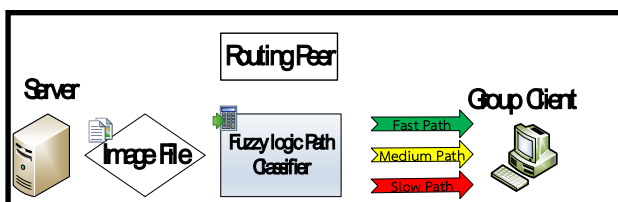
3.2) การสร้างเงื่อนไขและกฎควบคุม Fuzzy Reasoning Unit เป็นขั้นตอนการสร้างกฎพื้นฐานและเงื่อนไขความสัมพันธ์ในการคำนวณ ระหว่างข้อมูลอินพุตทั้ง 3 ค่า ด้วยกฎพื้นฐาน Fuzzy Rule จำนวน 27 กฎ เพื่อคำนวณหาเอาต์พุตเส้นทางจำนวน 3 เส้นทาง คือ Fast Path, Medium Path and Slow Path ในลักษณะ If และ And เพื่อให้ระบบเลือกเส้นทางในการคำนวณโหลดที่ดีที่สุดในแต่ละช่วงเวลา ซึ่งสามารถเขียนให้อยู่ในรูปแบบ If..... Then ยกตัวอย่างกฎที่ 1 คือ If (Bandwidth is Low) and (Congestion is Low) and (Network is Ethernet) then (Fuzzy Output is Slow)

ในการคำนวณของฟัซซีลอจิก จะคำนวณจากความสัมพันธ์ระหว่างอินพุตทั้งหมดที่มีความเกี่ยวข้องกับเอาต์พุตของเอาต์พุต โดยนำค่าอินพุตทั้ง 3 ค่าคือ Bandwidth, Congestion และ Network มากำหนดเงื่อนไขการตีความเพื่อให้ได้มาซึ่งผลลัพธ์เอาต์พุตในฟัซซีลอจิกซึ่งในงานวิจัยนี้จะสร้างกฎจากตัวแปรอินพุตทั้งหมดออกมาได้จำนวน 27 กฎ

3.3) การแปลงค่าฟัซซีลอจิกของข้อมูลนำเข้า Defuzzifier เป็นขั้นตอนการกำหนดวิธีการรวมผลของฟัซซีเพื่อให้ได้ผลลัพธ์ที่เป็นจริงในรูปแบบคริสป์เซต (Crisp Set) จากเงื่อนไขในขั้นตอนที่ 3.2 หากค่าที่แม่นยำตรงเพียงค่าเดียวเพื่อให้ได้ค่าของผลลัพธ์ที่แท้จริงของตัวควบคุมฟัซซีลอจิก ซึ่งจะใช้

วิธีการหาจุดศูนย์กลาง (Central of Gravity : COG) เป็นวิธีการเฉลี่ยผลที่ได้จากการตีความหาเหตุ เพื่อหาค่าเอาต์พุตเส้นทางการดาวน์โหลดจำนวน 3 เส้นทาง คือ Fast Path, Medium Path and Slow Path

ขั้นตอนที่ 4 Start Peer to Peer Download เป็นขั้นตอนสุดท้ายของโมเดลตัวต้นแบบการเลือกเส้นทางการดาวน์โหลดอิมเมจไฟล์บนเครือข่ายเพียร์ทูเพียร์ ด้วยตรรกศาสตร์คลุมเครือ ซึ่งการคำนวณจะแบ่งออกเป็น 4 ช่วงเวลาคือ ช่วงระยะเวลาเริ่มต้นการดาวน์โหลดไฟล์ 0 เปอร์เซ็นต์, ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 25 เปอร์เซ็นต์, ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 50 เปอร์เซ็นต์ และ ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 75 เปอร์เซ็นต์ ดังรูปที่ 4



รูปที่ 4. การส่งอิมเมจไฟล์บนเครือข่ายเพียร์ทูเพียร์ โดยใช้ตรรกศาสตร์คลุมเครือ

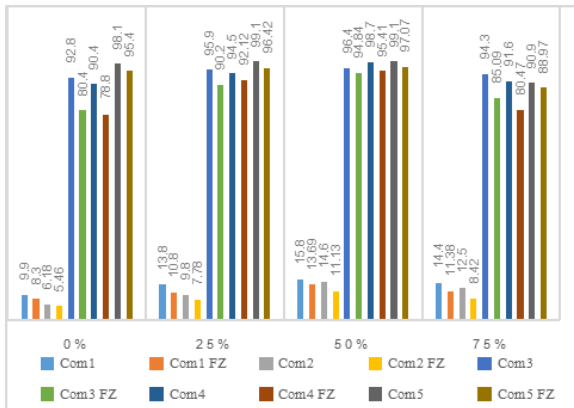
จากรูปที่ 4 แสดงขั้นตอนการส่งอิมเมจไฟล์จากต้นทางไปยังปลายทางเริ่มต้นจากเครื่องเซิร์ฟเวอร์แบ่งอิมเมจไฟล์ออกเป็นชิ้นๆ แล้วส่งไปยังกลุ่มเครื่องคอมพิวเตอร์ทั้งหมดในเครือข่ายเพียร์ทูเพียร์ โดยแบ่งออกเป็น 4 ช่วงเวลาคือ ช่วงระยะเวลาเริ่มต้นการดาวน์โหลดไฟล์ 0 เปอร์เซ็นต์, ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 25 เปอร์เซ็นต์, ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 50 เปอร์เซ็นต์ และ ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 75 เปอร์เซ็นต์ตามเส้นทางหลัก 3 เส้นทางคือ Fast Path, Medium Path และ Slow Path ที่ได้จากการคำนวณของฟัซซี่ลอจิก ซึ่งในระบบฟัซซี่ลอจิกที่นำเสนอจะคำนวณค่าเส้นทางครั้งแรกให้ส่งไฟล์ไปยัง Fast Path และจะทำการคำนวณอยู่ตลอดเวลา จากนั้นเมื่อเส้นทาง Fast Path เกิดความหนาแน่นของเส้นทาง ฟัซซี่ก็จะใช้เส้นทาง Medium Path อีกเส้นทางสุดท้ายเมื่อเส้นทางที่เป็น Medium Path มีปริมาณความหนาแน่นสูง ฟัซซี่ก็จะทำการเริ่มดาวน์โหลดเส้นทาง Slow Path เพิ่มอีกหนึ่งเส้นทาง ซึ่งกระบวนการดังกล่าวจะช่วยลดปัญหาความหนาแน่นในเครือข่ายได้

ออกแบบการทดลองการติดตั้งระบบปฏิบัติการบนเครื่องคอมพิวเตอร์แบบเพียร์ทูเพียร์

งานวิจัยนี้ได้ออกแบบการทดลองเริ่มจากการสร้างอิมเมจไฟล์ โดยจะติดตั้งระบบปฏิบัติการ Windows 7 Professional 64 Bit และโปรแกรม Microsoft Office 2007 ไว้ที่เครื่องคอมพิวเตอร์ต้นฉบับ จากนั้นทำการตรวจสอบพื้นที่ของฮาร์ดดิสก์ที่ถูกใช้งานไปจำนวน 15.5 GB โดยเช็คจาก Properties ของ My Computer จากนั้นจึงใช้โปรแกรม Ghostcast Server สร้างอิมเมจไฟล์ จากเครื่องคอมพิวเตอร์ต้นฉบับจะได้อิมเมจไฟล์ชื่อ Imagefile-p2p.GHO ขนาด 5.2 GB เพื่อนำอิมเมจไฟล์ ที่ได้จากเครื่องคอมพิวเตอร์ต้นฉบับ ส่งให้กับเครื่องคอมพิวเตอร์ลูกข่ายจำนวน 5 เครื่อง และ 10 เครื่อง ซึ่งในการทดลองดาวน์โหลดไฟล์ในครั้งแรกจะใช้การดาวน์โหลดไฟล์แบบ Normal P2P จากนั้นทำการบันทึกค่า Congestion ที่เกิดขึ้นใน 4 ช่วงเวลา คือ ช่วงระยะเวลาเริ่มต้นการดาวน์โหลดไฟล์ 0 เปอร์เซ็นต์, ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 25 เปอร์เซ็นต์, ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 50 เปอร์เซ็นต์ และ ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 75 เปอร์เซ็นต์ จากนั้นนำค่า Congestion ที่บันทึกได้มาคำนวณด้วยกฎพื้นฐานของฟัซซี่ลอจิก ที่ได้กำหนดไว้ เพื่อเริ่มการทดลองดาวน์โหลดไฟล์แบบเพียร์ทูเพียร์ แบบมีฟัซซี่ลอจิก ช่วยคำนวณจัดลำดับเส้นทางการดาวน์โหลด จากนั้นทำการบันทึกค่า Congestion ที่เกิดขึ้นใน 4 ช่วงเวลา อีกครั้ง เพื่อนำมาเปรียบเทียบผลการทดลองระหว่าง 2 รูปแบบการดาวน์โหลด

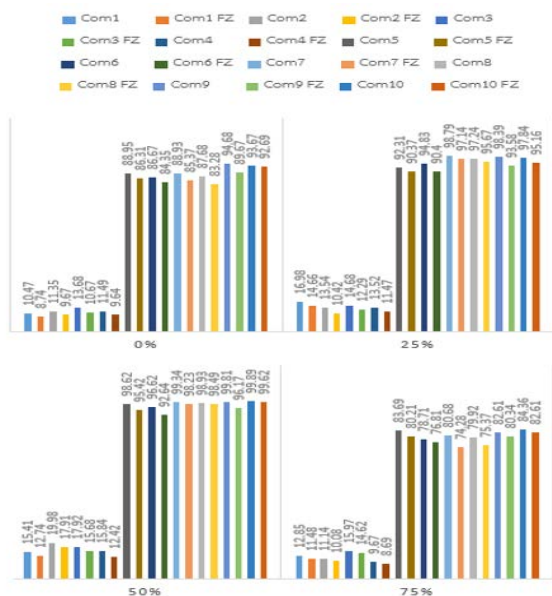
4. ผลการทดลอง

การประเมินผลการทดลองการดาวน์โหลดอิมเมจไฟล์ บนเครือข่ายเครื่องคอมพิวเตอร์จำนวน 5 เครื่อง และ 10 เครื่อง จะเปรียบเทียบค่า Congestion ระหว่างการดาวน์โหลดไฟล์แบบ Normal P2P และการดาวน์โหลดไฟล์แบบเพียร์ทูเพียร์ แบบมีฟัซซี่ลอจิก ช่วยคำนวณจัดลำดับเส้นทางการดาวน์โหลดที่เหมาะสมใน 4 ช่วงเวลา คือ ช่วงระยะเวลาเริ่มต้นการดาวน์โหลดไฟล์ 0 เปอร์เซ็นต์, ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 25 เปอร์เซ็นต์, ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 50 เปอร์เซ็นต์ และ ช่วงระยะเวลาการดาวน์โหลดไฟล์ที่ 75 เปอร์เซ็นต์



รูปที่ 5. เปรียบเทียบความหนาแน่นของเครื่องคอมพิวเตอร์ 5 เครื่อง

จากรูปที่ 5 แสดงการเปรียบเทียบความหนาแน่นของข้อมูลในเครือข่ายที่เกิดขึ้นในระหว่างการทดลองการดาวน์โหลดคอมพิวเตอร์ 5 เครื่องที่เชื่อมต่อแบบ Normal P2P แสดงด้วยสัญลักษณ์ Com ของกราฟในรูปที่ 5 และการดาวน์โหลดไฟล์แบบ P2P แบบมีพีชชีลอจิก ช่วยเลือกเส้นทาง แสดงด้วยสัญลักษณ์ Com FZ ของกราฟในรูปที่ 5 ซึ่งค่าความหนาแน่น Congestion ของข้อมูลที่เกิดขึ้นในเครือข่าย เปรียบเทียบผลการทดลองแบบเครื่องต่อเครื่อง ซึ่งผลการทดลองแสดงให้เห็นว่าการนำพีชชีลอจิกเข้ามาช่วยคำนวณค้นหาเส้นทางในการดาวน์โหลดที่ดีที่สุดจะช่วยลดความหนาแน่นของข้อมูลลงได้ในทุกช่วงเวลาการดาวน์โหลดไฟล์ นอกจากนี้การดาวน์โหลดไฟล์แบบเพียร์ทูเพียร์ แบบมีพีชชีลอจิก



รูปที่ 6. เปรียบเทียบความหนาแน่นของเครื่องคอมพิวเตอร์ 10 เครื่อง

จากรูปที่ 6 แสดงการเปรียบเทียบความหนาแน่นของข้อมูลในเครือข่ายที่เกิดขึ้นในระหว่างการทดลองการดาวน์โหลดคอมพิวเตอร์ 10 เครื่องที่เชื่อมต่อแบบ Normal P2P แสดงด้วยสัญลักษณ์ Com ของกราฟในรูปที่ 6 และการดาวน์โหลดไฟล์แบบเพียร์ทูเพียร์ แบบมีพีชชีลอจิก ช่วยเลือกเส้นทาง แสดงด้วยสัญลักษณ์ Com FZ ของกราฟในรูปที่ 6 ซึ่งค่าความหนาแน่น Congestion ของข้อมูลที่เกิดขึ้นในเครือข่าย เปรียบเทียบผลการทดลองแบบเครื่องต่อเครื่อง ซึ่งผลการทดลองแสดงให้เห็นว่าการนำพีชชีลอจิกเข้ามาช่วยคำนวณค้นหาเส้นทางในการดาวน์โหลดที่ดีที่สุดจะช่วยลดความหนาแน่นของข้อมูลลงได้ในทุกช่วงเวลาการดาวน์โหลดไฟล์

5. สรุปผลและข้อเสนอแนะ

งานวิจัยนี้มีวัตถุประสงค์ที่จะนำการ Transfer file แบบเพียร์ทูเพียร์ เข้ามาช่วยลดภาระการทำงานของเครื่อง Server และนำพีชชีลอจิก เข้ามาช่วยคำนวณเลือกเส้นทางในการดาวน์โหลดไฟล์ของกลุ่มเครื่องคอมพิวเตอร์จำนวน 5 เครื่อง และ 10 เครื่อง เพื่อช่วยลด Congestion ของข้อมูลบนเครือข่ายลง โดยการทดลองจะเปรียบเทียบระหว่างการดาวน์โหลดแบบ Normal P2P กับ การดาวน์โหลดแบบเพียร์ทูเพียร์แบบมีพีชชีลอจิก เลือกเส้นทาง ผลที่ได้คือการดาวน์โหลดแบบเพียร์ทูเพียร์ แบบมีพีชชีลอจิกเลือกเส้นทาง สามารถช่วยลดความหนาแน่นของข้อมูลในระบบเครือข่ายใน 4 ช่วงเวลาของการดาวน์โหลดไฟล์ของเครื่องคอมพิวเตอร์บนเครือข่ายจำนวน 5 เครื่อง ได้ดังนี้ ช่วงที่ 1 ลดความหนาแน่นของข้อมูลได้ร้อยละ 5.80, ช่วงที่ 2 ลดความหนาแน่นของข้อมูลได้ร้อยละ 3.15, ช่วงที่ 3 ลดความหนาแน่นของข้อมูลได้ร้อยละ 2.49 และช่วงที่ 4 ลดความหนาแน่นของข้อมูลได้ร้อยละ 5.87 บนเครือข่ายเครื่องคอมพิวเตอร์จำนวน 10 เครื่อง ตัวค้นแบบสามารถลดความหนาแน่นและเวลาการดาวน์โหลดได้ดังนี้ ช่วงที่ 1 ลดความหนาแน่นของข้อมูลได้ร้อยละ 2.71, ช่วงที่ 2 ลดความหนาแน่นของข้อมูลได้ร้อยละ 2.69, ช่วงที่ 3 ลดความหนาแน่นของข้อมูลได้ร้อยละ 2.30 และช่วงที่ 4 ลดความหนาแน่นของข้อมูลได้ร้อยละ 2.51 จากผลการทดลองโมเดลต้นแบบที่นำเสนอใน

งานวิจัยฉบับนี้ สามารถช่วยลดความหนาแน่นของข้อมูลในเครือข่ายเครื่องคอมพิวเตอร์จำนวน 5 เครื่องสามารถลดความหนาแน่นได้ร้อยละ 4.32 และเครือข่ายเครื่องคอมพิวเตอร์จำนวน 10 เครื่อง สามารถลดความหนาแน่นได้ร้อยละ 2.55 ข้อสังเกตงานวิจัยหากมีการเพิ่มจำนวนเครื่องคอมพิวเตอร์มากขึ้น ความหนาแน่นของข้อมูลในเครือข่ายจะเพิ่มมากขึ้น เป็นเหตุทำให้ผลการทดลองที่เครื่องคอมพิวเตอร์จำนวน 10 เครื่องมีค่าลดลง ดังนั้นหากมีจำนวนเครื่องคอมพิวเตอร์มากขึ้นผลการลดความหนาแน่นในเครือข่ายก็จะลดน้อยลง ในอนาคตผู้วิจัยมีแนวคิดที่จะนำ Algorithm ตัวอื่นเช่น Neural Networks มาทำการทดลองเปรียบเทียบประสิทธิภาพกับฟัซซี่ลอจิก และเพิ่มการพิจารณาในเรื่องของ Download time ควบคู่กันกับ Congestion control

เอกสารอ้างอิง

[1] Symantec, "Symantec Ghost Implementation Guide," [Online] Available www.symantec.com/public/english_us_canada/products/symantec_ghost_solution_suite/2.0/manuals/Ghost_imp_guide.pdf [Accessed : July 4, 2016]

[2] เจษฎา วินโรจน์, "Peer-to-Peer Network," [Online]. Available : <https://sites.google.com/site/jesadawin/pheiyr-thu-pheiyr-peer-to-peer-network>. [Accessed : 2 สิงหาคม 2559]

[3] พยุง มีสัง, ระบบฟัซซี่และโครงข่ายประสาทเทียม, กรุงเทพฯ : คณะ เทคโนโลยีสารสนเทศ (มจพ), 2559

[4] กิตติศักดิ์ วัฒนกุล, "การกระจายไฟล์ข้อมูลสำหรับการเรียนการสอนด้วยสื่ออิเล็กทรอนิกส์ภายในเครือข่ายคอมพิวเตอร์แบบเพียร์ทูเพียร์," วิทยานิพนธ์ วศ.ม. (วิศวกรรมคอมพิวเตอร์), มหาวิทยาลัยสงขลานครินทร์, สงขลา, 2553

[5] วีรชัย เข้มขจี, "การศึกษาโปรโตคอลค้นหาเส้นทางบนเครือข่ายไร้สายเฉพาะกิจแบบเคลื่อนที่ได้ด้วยตรรกศาสตร์คลุมเครือ," วิทยานิพนธ์ วศ.ม. (วิศวกรรมไฟฟ้า), มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี, ปทุมธานี, 2555

[6] A. Mahapatra et al., "A fuzzy approach for reputation management in bittorrent p2p network," *International Conference on Electronics Computer Technology, ICECT 2011, Kanyakumari, India*, April 8-10 2011. pp. 360-365

[7] A. Mahapatra et al., "A fuzzy approach for reputation management in online communities for bittorrent p2p network," *International Journal of Computer Science and Information Technologies*, Vol. 2, No. 3, July 2011. pp. 1564-1568

[8] A. Mahapatra and N. Tarasia, "A fuzzy approach for reputation management for reputation management using voting scheme in bittorrent p2p network," *International Journal of Computer Science and Information Technologies*, Vol. 2, No. 2, July 2011. pp. 735-740

[9] S. N. Ingokey and M. Nashipudi, "A review: fuzzy logic in congestion control of computer network," *International Conference in Recent Trends in Information Technology and Computer Science, ICRITTCS 2012, Chennai, India*, April 19-21 2012. pp. 975-981

[10] B. S. Kiruthika Devi et al., "A novel fuzzy congestion control algorithm for router buffers," *IEEE-International Conference on Recent Trends in Information Technology, ICRITIT 2011, Chennai, India*, June 3-5 2011. pp. 423-427

[11] R. Lal and A. Chiou, "Computational intelligence utilisation in simulation of congestion control in TCP/IP edge network," *Fourth International Conference on Network and System Security, NSS 2010, Melbourne, Australia*, September 1-3 2010. pp. 191-198

[12] B. Safaiezadeh et al., "A new fuzzy congestion control algorithm in computer networks," *International Conference Computer and Communication, ICCCC 2009, San Francisco, USA*, August 3-6 2009. pp. 314-318