

การศึกษาประสิทธิภาพในการออกแบบเครือข่ายไอพีด้วยวิธีมีมติก

A Study on IP Network Design by Memetic Algorithm

กายรัฐ เจริญราษฎร์¹ และณัฐชามณูห์ ศรีจำเริญรัตน์^{2*}

Kairat Jaroenrat¹ and Natchamol Srichumroenrattana^{2*}

¹ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ กำแพงแสน มหาวิทยาลัยเกษตรศาสตร์

¹Department of Computer Engineering, Faculty of Engineering at Kamphaeng Saen, Kasetsart University

²สาขาคอมพิวเตอร์ธุรกิจ คณะวิทยาการจัดการ มหาวิทยาลัยราชภัฏนครปฐม

²Department of Business Computer, Faculty of Management Science, NakhonPathom Rajabhat University

*Corresponding author e-mail: fengkrj@ku.ac.th

Received: February 23, 2023

Revised: March 28, 2023

Accepted: April 5, 2023

บทคัดย่อ

การออกแบบเครือข่ายแพ็กเก็ตสวิตชิงเป็นปัญหาที่ยากและซับซ้อนจนไม่เหมาะที่จะแก้ปัญหาดังกล่าวด้วยวิธีการทางคณิตศาสตร์ โดยเฉพาะโพรโทคอลโอเอสพีเอฟบนเครือข่ายเราเตอร์หรือเครือข่ายไอพีที่สามารถหาเส้นทางได้ด้วยตัวเอง ยิ่งทำให้การออกแบบและจัดเส้นทางให้ข้อมูลนั้นไม่สอดคล้องกับความต้องการใช้งานเครือข่าย ผู้วิจัยจึงประยุกต์ใช้วิธีมีมติกในการเลือกติดตั้งสายสัญญาณในเครือข่ายอย่างมีประสิทธิภาพและคุ้มค่า โดยทำการทดลองใช้วิธีมีมติก เปรียบเทียบกับวิธีชิมเพล็กซ์ และวิธีเมนเทอร์-II ที่เครือข่ายขนาด 4 ถึง 10 โหนด ได้ผลปรากฏว่า วิธีมีมติกที่มีการแลกเปลี่ยนมีมสามารถออกแบบเครือข่ายที่มีประสิทธิภาพด้านเวลาหน่วงสายสัญญาณที่ดีกว่าวิธีเมนเทอร์-II และเกือบเท่ากับวิธีชิมเพล็กซ์ที่นัยสำคัญ .05 แต่มีความซับซ้อนของกระบวนการที่น้อยกว่าวิธีชิมเพล็กซ์

คำสำคัญ: มีมติก โอเอสพีเอฟ การออกแบบเครือข่ายแพ็กเก็ตสวิตชิง

Abstract

Designing packet switch networks is a difficult and complex problem. The mathematical methods were improperly used to solve this problem. Especially the router network (or IP network) protocols that find the route by themselves, such as the OSPF protocol, may not make the traffic flow routing get together with the traffic requirement. We apply the Memetic algorithm for a worthwhile and effective network design. Then, the performance of 4–10 node networks was compared to the Simplex and MENTOR-II methods. The results of the comparison between the Memetic method and the Simplex method showed that the delay efficiency of the Memetic with crossover method is close to the Simplex method with a significance level of .05 but with less complexity.

Keywords: Memetic, OSPF, Packet-Switching Network Design



บทนำ

ในปัจจุบันระบบคอมพิวเตอร์นับเป็นสิ่งจำเป็นกับชีวิตประจำวันในหลาย ๆ ด้าน ส่งผลให้เกิดความต้องการที่จะเชื่อมต่อให้เป็นระบบเครือข่ายคอมพิวเตอร์ ซึ่งจะเพิ่มขีดความสามารถของระบบให้มีประสิทธิภาพสูงขึ้น ซึ่งสิ่งสำคัญที่ต้องมีในเครือข่ายคอมพิวเตอร์ โดยเฉพาะเครือข่ายอินเทอร์เน็ตนั้น ก็คือ โพรโทคอลชี้เส้นทางของโครงข่ายเราเตอร์ โดยโพรโทคอลชี้เส้นทางที่เป็นที่นิยมมากที่สุดตัวหนึ่ง คือ โพรโทคอลโอเอสพีเอฟ (OSPF) (Duvedi, Ashraf & Gairola, 2022) ที่มีคุณสมบัติที่ดีในหลายด้าน อาทิ ความสามารถในการหาเส้นทางที่ดีที่สุดด้วยตัวเอง และการอัปเดตตารางเส้นทางที่ไม่ต้องรับส่งข้อมูลตลอดเวลา แต่การกำหนดเส้นทางด้วยวิธีโอเอสพีเอฟนี้อาจทำให้เกิดปัญหาความคับคั่งของข้อมูล จากการที่ข้อมูลหลายกระแสไหลไปยังเส้นทางเดียวกันทำให้เวลาที่ใช้ในการเดินทางของข้อมูล (delay) บางเส้นทางในโครงข่ายสูงและไม่ตรงกับความต้องการของผู้ใช้ ซึ่งปัญหาเรื่องเวลาในการเดินทางของข้อมูลนี้ถือเป็นปัญหาหลักในเครือข่ายคอมพิวเตอร์ (Charnkeitkong, 2007; Dehestani & Hajipour, 2010)

ทั้งนี้ การออกแบบเครือข่ายแบบตาข่ายบางส่วน (partial mesh network) เป็นรูปแบบการเชื่อมต่อที่เน้นจำนวนของช่องสัญญาณที่เชื่อมต่อระหว่างกันหลายช่องทาง

โดยจะเลือกกว่าจะใช้ช่องสัญญาณใดเชื่อมต่อกันบ้างเพื่อให้เหมาะสมและคุ้มค่ากับการทำงานอย่างมีประสิทธิภาพ ดังนั้นงานวิจัยนี้จึงนำเสนอวิธีการประยุกต์ใช้ขั้นตอนวิธีมีติกสำหรับแก้ปัญหาการออกแบบติดตั้งสายสัญญาณในเครือข่าย เพื่อเลือกติดตั้งช่องสัญญาณให้มีประสิทธิภาพและมีความคุ้มค่าด้านเวลามากที่สุด และเพื่อที่จะศึกษาถึงประสิทธิภาพของขั้นตอนวิธีมีติกนี้ ผู้วิจัยทำการพิจารณาเปรียบเทียบประสิทธิภาพกับวิธีซิมพล็กซ์ (Lanthong & Jaroenrat, 2012) และวิธีเมนเทอร์-II (Monsakul, 2015) เนื่องจากวิธีซิมพล็กซ์เป็นวิธีการที่มีประสิทธิภาพในทางทฤษฎี โดยวิธีนี้พัฒนามาจากวิธีทางพีชคณิตที่อาศัยทฤษฎีของเมทริกซ์เข้าร่วมจัดรูปแบบปัญหาให้มีระบบยิ่งขึ้น โดยเริ่มต้นจากสมการกำหนดเป้าหมายให้ได้ผลแนวโน้มนำเป้าหมายด้วยการจัดรูปสมการเข้าเป็นตารางแล้วดำเนินการตามขั้นตอนที่ถูกต้อง ทำให้ได้ผลลัพธ์ตามเป้าหมายด้วยผลลัพธ์ที่ดีที่สุด ส่วนวิธีเมนเทอร์-II เป็นวิธีฮิวริสติกในการออกแบบติดตั้งช่องทางสื่อสารในเครือข่ายไปพร้อมกับการกำหนดค่านำหนักของช่องสัญญาณที่จะทำให้ข้อมูลไหลไปในเส้นทางที่ต้องการด้วยวิธีที่เรียกว่า Incremental Shortest Path-ISP โครงข่ายที่ได้จากการออกแบบด้วยวิธีเมนเทอร์-II แม้ไม่ได้เป็นโครงข่ายที่ดีที่สุด แต่ก็เป็นโครงข่ายที่มีคุณสมบัติต่าง ๆ รวมถึงสมรรถนะที่ดีพอสมควร

วัตถุประสงค์การวิจัย

ศึกษาประสิทธิภาพของการใช้วิธีมีติกเปรียบเทียบ กับวิธีอื่น ๆ ที่แตกต่างกัน สำหรับออกแบบเครือข่ายไอพี โดยประเมินประสิทธิภาพจากผลรวมของเวลาที่ใช้ในการ ส่งข้อมูลผ่านช่องสัญญาณทั้งเครือข่าย

กรอบแนวคิดการวิจัย

ในการวิจัยนี้ ผู้วิจัยได้ใช้เครื่องมือการทดลองเป็น โปรแกรมออกแบบโครงข่ายที่ได้พัฒนาขึ้นเอง ซึ่งมีภาพรวม ของระบบดังแสดงในภาพ 1 โดยเริ่มต้นทำงานด้วยการรับ ค่าความต้องการในการออกแบบเครือข่าย อันประกอบด้วย จำนวนโหนดของเครือข่าย ค่าความต้องการไหลระหว่างคู่ โหนด (traffic requirement) อัตราส่วนระหว่างจำนวน สายสัญญาณกับจำนวนโหนด (node degree) และจำนวน จุดแลกเปลี่ยนมีม จากนั้นจึงใช้วิธีการมีติกคำนวณการติด ตั้ง การไหล และค่าใช้จ่ายในการติดตั้งสายสัญญาณ และ สุดท้ายจึงเป็นการแสดงผลค่าผลลัพธ์ที่เป็นรูปแบบการติด ตั้งสายสัญญาณและค่าใช้จ่ายที่ต้องใช้ในการติดตั้ง

ทั้งนี้ ในส่วนของขั้นตอนการทำงานของวิธีมีติกที่ คำนวณการติดตั้ง การไหล และค่าใช้จ่ายในการติดตั้งสาย สัญญาณนั้น ประกอบไปด้วยกระบวนการต่าง ๆ ที่แสดง ในภาพ 2 ดังนี้

ขั้นตอนแรกเริ่มจากการรับค่าจำนวนโหนด โหนดคิกรี ความต้องการการไหลของข้อมูล และจำนวนจุดแลกเปลี่ยน มีม เพื่อทำการสุ่มสร้างเครือข่ายจำนวน 1,000 เครือข่าย ด้วยค่าโหนดคิกรีตามที่กำหนด แล้วทำการสุ่มค่าน้ำหนัก ให้สายสัญญาณแต่ละเส้นในเครือข่าย

ต่อมาในขั้นตอนที่สองจะเป็นการคำนวณค่าการะ การไหลของข้อมูลในแต่ละสายสัญญาณ โดยค่าการะการ ไหลของข้อมูลในสายสัญญาณนั้นได้จากผลรวมของปริมาณ ข้อมูลที่ไหลในสายสัญญาณตั้งแต่ต้นทางไปปลายทาง และนำค่าการะการไหลนี้มาคำนวณหาค่าความกว้างช่อง สัญญาณที่เหมาะสม แล้วจึงคำนวณหาประสิทธิภาพด้าน ค่าเวลาหน่วงของเครือข่าย

ในส่วนของขั้นตอนที่สามเป็นการคัดเลือกชุด โครโมโซมของค่าการติดตั้งสายสัญญาณระหว่างคูโหนด

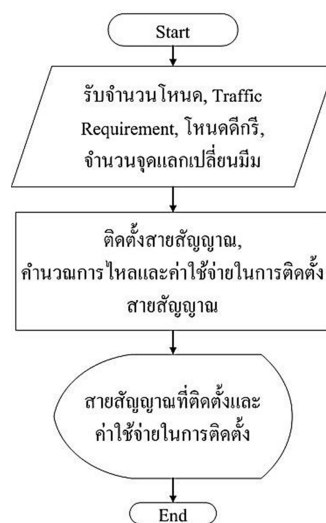
200 ชุด ด้วยการสุ่มแบบวงล้อรูเล็ต ซึ่งค่าเวลาหน่วงของ สายสัญญาณที่น้อยจะมีโอกาสถูกเลือกมากกว่า และจับคู่ ชุดโครโมโซมของค่าการติดตั้งสายสัญญาณระหว่างคูโหนด มาทำการแลกเปลี่ยนมีมกัน

จากนั้นในขั้นตอนที่สี่ เป็นการแลกเปลี่ยนมีม โดย จุดที่ทำการแลกเปลี่ยนมีมอยู่ในช่วงอัตรา 0.2-0.8 จาก การสุ่มตำแหน่ง ซึ่งทุกคู่ของโครโมโซมที่ถูกเลือกยกเว้น ชุดโครโมโซมของตัวเองจะแลกเปลี่ยนมีมกัน และจำนวน ของตำแหน่งของการแลกเปลี่ยนมีมที่ 1-10 จุด

ในขั้นตอนที่ห้าเป็นกระบวนการการกลายพันธุ์ ซึ่ง เป็นการสุ่ม 2 ครั้ง โดยครั้งแรกจะสุ่มตำแหน่งมีมที่มีโอกาส เกิดการกลายพันธุ์ โดยอัตราส่วนในแต่ละตำแหน่งของมีม ที่มีโอกาสเกิด เท่ากับ 0-1% และสุ่มครั้งที่สอง เป็นการสุ่ม ค่ามีมค่าใหม่

ต่อมาในขั้นตอนที่หก เมื่อได้ชุดของโครโมโซม ของค่าการติดตั้งสายสัญญาณระหว่างคูโหนด จึงทำการค้นหา แบบเฉพาะที่ โดยใช้วิธีการปีนเขาแบบง่ายที่ใช้การสลับค่า ภายในชุดโครโมโซม ดังภาพ 3

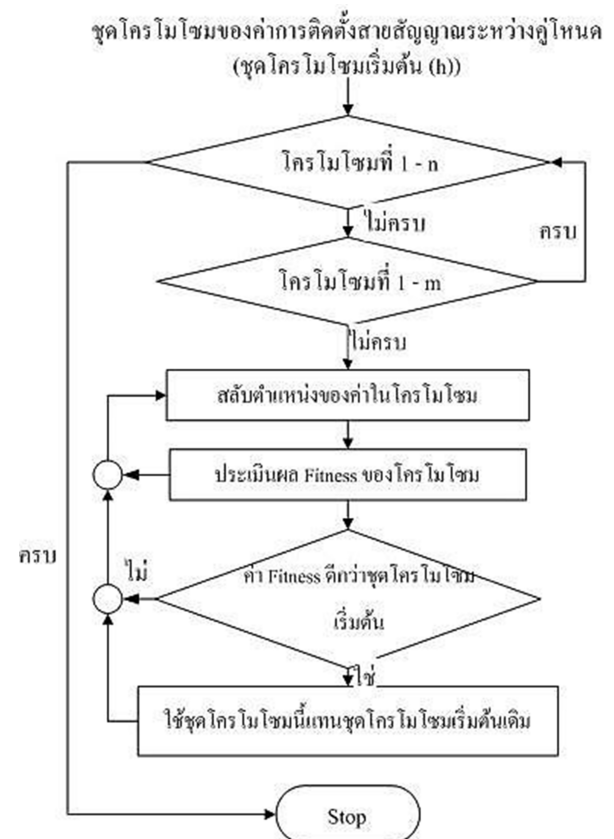
จากนั้นจึงทำการทำซ้ำกระบวนการคัดเลือก การ แลกเปลี่ยนมีม การกลายพันธุ์ และการทำค้นหาแบบเฉพาะ ที่ จำนวน 100 รอบ เพื่อนำผลมาเปรียบเทียบหาเครือข่าย ที่ให้ค่าเวลาหน่วงของสายสัญญาณน้อยที่สุด ทั้งนี้การค้นหา เฉพาะที่แบบ Simple Hill climbing มีขั้นตอนเป็นไปดัง ภาพ 3



ภาพ 1 ภาพรวมของระบบ



ภาพ 2 ขั้นตอนการทำงานของวิธีการมีมติก



ภาพ 3 ภาพรวมขั้นตอน Simple Hill climbing

วิธีดำเนินการวิจัย

ผู้วิจัยได้ทำการทดสอบการเปรียบเทียบสมรรถนะของการจัดการไหลของข้อมูลในเครือข่ายที่ออกแบบด้วยวิธีมีมติกเปรียบเทียบกับเครือข่ายที่ออกแบบด้วยวิธีซิมเพิลคลimbing และวิธีเมทาเทอร์-II

โดยในงานวิจัยนี้จะแบ่งประชากรการวิจัยออกเป็น 3 กลุ่ม กลุ่มแรก คือ กลุ่มเครือข่ายที่ออกแบบด้วยวิธีมีมติก มีจำนวนประชากรเท่ากับ 504 เครือข่าย ประกอบไปด้วยชุดข้อมูลเครือข่ายขนาด 4-10 โหนด ที่โหนดดีกรี 1.0 1.2 1.4 1.6 1.8 และ 2.0 ที่สุ่มความต้องการการไหลของข้อมูลที่แตกต่างกัน 3 ชุด แต่ละชุดมีเครือข่ายที่ใช้พารามิเตอร์จำนวนจุดที่ใช้ในการแลกเปลี่ยนมีมในการออกแบบที่ต่างกัน คือ 0 1 3 และ 6 จุด และพารามิเตอร์การวนซ้ำจำนวน 100 รอบ

กลุ่มที่สอง คือ กลุ่มเครือข่ายที่ออกแบบด้วยวิธีซิมเพิลคลimbingจะมีจำนวนประชากรเท่ากับ 108 เครือข่าย ประกอบไปด้วยชุดข้อมูลเครือข่ายขนาด 4-9 โหนด ที่โหนด

ดีกรี 1.0 1.2 1.4 1.6 1.8 และ 2.0 ที่สะท้อนความต้องการการไหลของข้อมูลที่แตกต่างกัน 3 ชุด

กลุ่มที่สาม คือ กลุ่มเครือข่ายที่ออกแบบด้วยวิธีเมนเทอร์-II มีจำนวนประชากรเท่ากับ 210 เครือข่าย โดยเครือข่ายถูกกำหนดให้ค่าผลรวมของข้อมูลขาเข้าและขาออกในแต่ละโหนด เท่ากับ 150 200 และ 250 Mbps รวมจำนวนเครือข่ายทั้งสิ้น 210 เครือข่าย

ทั้งนี้ ผู้วิจัยได้ใช้โปรแกรม DELITE (Cahn, 1998) สร้างระบบโหนดพร้อมค่าความต้องการการไหลของข้อมูล แล้วจึงใช้โปรแกรมที่พัฒนาขึ้นเองด้วยภาษา C++ ร่วมกับ GLPK Library (Makhorn, 2000) ออกแบบเครือข่ายตามแต่ละวิธีและพารามิเตอร์ที่กำหนด จากนั้นจึงทำการเปรียบเทียบประสิทธิภาพของแต่ละวิธีการออกแบบดังแสดงในหัวข้อถัดไป

การประเมินประสิทธิภาพของเครือข่าย

การวิจัยนี้เป็นการเปรียบเทียบประสิทธิภาพของโครงข่ายโดยพิจารณาจากผลรวมของเวลาที่ใช้ในการส่งข้อมูลผ่านช่องสัญญาณทั้งโครงข่าย (Charnkeitkong, 2007; Dehestani & Hajipour, 2010) โดยเวลาที่ใช้ในการเดินทางของข้อมูลผ่านช่องสัญญาณนี้ ประกอบด้วย 2 ส่วนดังสมการ (1)

$$\text{Link Delay} = \text{Fixed Delay} + \text{Variable Delay} \quad (1)$$

ค่าในส่วน Fixed Delay คือ เวลาที่สัญญาณเดินทางในสายสัญญาณ โดยทั่วไปจะประมาณค่าอยู่ที่ประมาณ 5 μ s ต่อกิโลเมตร ส่วน Variable Delay นั้นประกอบไปด้วย เวลาที่เข้าแถวรอในบัฟเฟอร์ และเวลาที่ใช้ในการส่งท่อนข้อมูลจนเสร็จ ซึ่งถ้ารู้ขนาดเฉลี่ยของท่อนข้อมูล (average packet size) และความเร็วของสายสัญญาณ (link capacity) ก็จะสามารถคำนวณค่าเวลาเฉลี่ย (T_p) ที่ใช้ในการส่งท่อนข้อมูล ได้จากสมการ (2)

$$T_p = \frac{\text{Average Packet Size}}{\text{Link Capacity}} \quad (2)$$

สำหรับเวลารอในแถวรอของท่อนข้อมูล สามารถคำนวณโดยประมาณด้วยทฤษฎีแถวรอ (queuing theory) ด้วยตัวแบบ M/M/1 (Charnkeitkong, 2007; Dehestani & Hajipour, 2010; Jaroenrat & Samrongsub, 2014) และเวลาเฉลี่ยที่รอในบัฟเฟอร์ (average queuing delay) (T_q) จะสามารถคำนวณได้จากสมการ (3)

$$T_q = T_p \frac{U}{1 - U} \quad (3)$$

เมื่อ U คือ อัตราการใช้ช่องสัญญาณ (link utilization) ทำให้เวลาที่ใช้ในการส่งข้อมูลทั้งหมดผ่านช่องสัญญาณ (T_r) หรือก็คือ Variable Delay จะเป็นไปตามสมการ (4)

$$T_r = T_p + T_q = \frac{U}{1 - U} T_p \quad (4)$$

โดย U คือ Utilization ($U=l/c$)

l คือ Load ในสายสัญญาณ

c คือ Capacity หรือ ค่าปริมาณการไหลของข้อมูล

ทั้งนี้ค่า T_p สามารถตัดออกจากการคำนวณได้เพราะเป็นค่าคงที่ (Charnkeitkong, 2007) ซึ่งจะได้สมการใหม่-ของค่าเวลาหน่วง Delay Cost ($\emptyset$) เป็นดังสมการ (5)

$$\text{Delay Cost } (\emptyset) = \frac{U}{1 - U} \quad (5)$$

เมื่อนำค่า U แทนในสมการที่ (6) จะได้เป็นสมการใหม่ ดังสมการ (6)

$$\emptyset_a(l_a, c_a) = \frac{l_a}{c_a - l_a} \quad (6)$$

แต่สมการนี้ไม่สามารถนำมาใช้ได้เนื่องจากเมื่อค่า $\emptyset$ เข้าใกล้ $\emptyset$ ผลลัพธ์ค่าเวลาหน่วงจะลู่เข้าสู่การเป็น

อนันต์จึงต้องมีสมการใหม่มาแก้ไขปัญหานี้ เพื่อแก้ไขปัญหานี้ (Fortz & Thorup, 2000) ประมาณค่า $\frac{l_a}{c_a - l_a}$ ด้วยสมการเส้นตรงแบ่งช่วง (piecewise linear) โดยแบ่งสมการออกเป็นหลายเงื่อนไขเพื่อหาค่าเวลาหน่วยของสายสัญญาณที่ให้เงื่อนไขที่มีค่าสูงที่สุด ดังสมการที่ (7) และเมื่อนำค่าเวลาหน่วยของทุกสายสัญญาณมารวมกันก็จะได้ค่าเวลาหน่วยรวมทั้งเครือข่ายนั่นเอง

ϕ_a (Delay Cost) =

$$\text{Max} \left\{ \begin{array}{l} \phi_a \geq l_a \\ \phi_a \geq 3l_a - \frac{2}{3}C_a \\ \phi_a \geq 10l_a - \frac{16}{3}C_a \\ \phi_a \geq 70l_a - \frac{178}{3}C_a \\ \phi_a \geq 500l_a - \frac{1468}{3}C_a \\ \phi_a \geq 5000l_a - \frac{19468}{3}C_a \end{array} \right. \quad (7)$$

เมื่อ ϕ_a คือ ค่าเวลาหน่วย (delay cost) ที่ใช้ในสายสัญญาณของคูโหนดที่พิจารณา l_a คือ ค่าภาระการไหลของข้อมูล (Load) ของสายสัญญาณของคูโหนดที่พิจารณา C คือ ค่าความกว้างช่องสัญญาณ (bandwidth) ของคูโหนดที่พิจารณา และ a คือ ลิงก์ของคูโหนดที่พิจารณา

อัลกอริทึมที่ใช้ในการเปรียบเทียบประสิทธิภาพ

การวิจัยนี้เป็นการเปรียบเทียบสมรรถนะของการจัดการไหลของข้อมูลในเครือข่ายที่ออกแบบด้วยวิธีมีมิติกเปรียบเทียบกับเครือข่ายที่ออกแบบด้วยวิธีซิมเพล็กซ์และวิธีเมนเทอร์-II โดยแต่ละอัลกอริทึมมีรายละเอียดการทำงานดังต่อไปนี้

1) Mesh Network Topology Optimization and Routing-II (MENTOR-II)

เมนเทอร์เป็นฮิวริสติกอัลกอริทึมออกแบบเครือข่ายแบบตาข่ายที่มีความซับซ้อนต่ำระดับ $O(N^2)$ (Charnkeitkong, 2007) ประมวลผลได้เร็ว แต่กำหนดเส้นทางการไหลของข้อมูลโดยไม่คำนึงถึงการทำงานของ Routing Protocol เช่น OSPF ดังนั้นหากจะออกแบบเครือข่ายไอพี จึงต้องใช้วิธีเมนเทอร์-II (Monsakul, 2015)

ผู้วิจัยจึงได้นำวิธีเมนเทอร์-II นี้มาเป็นตัวเปรียบเทียบประสิทธิภาพกับวิธีมีมิติก ซึ่งการทำงานของวิธีเมนเทอร์-II นี้ คือ ในขณะที่ย่อแบบติดตั้งช่องทางสื่อสารในเครือข่ายก็จะกำหนดค่าน้ำหนักของช่องสัญญาณที่จะทำให้ข้อมูลไหลไปในเส้นทางที่ต้องการไปพร้อม ๆ กัน ด้วยวิธีที่เรียกว่า Incremental Shortest Path--ISP

2) Simplex Method

วิธีซิมเพล็กซ์ (Simplex Method) (Lanthong & Jaroenrat, 2012) เป็นวิธีหนึ่งที่นิยมของโปรแกรมเชิงเส้นที่มีประสิทธิภาพในทางปฏิบัติเพื่อหาผลลัพธ์ที่ดีที่สุด ผู้วิจัยจึงได้นำวิธีการซิมเพล็กซ์นี้มาเป็นตัวเปรียบเทียบประสิทธิภาพกับวิธีมีมิติก วิธีนี้พัฒนามากจากวิธีทางพีชคณิตที่อาศัยทฤษฎีของเมทริกซ์เข้าร่วมจัดรูปแบบปัญหาให้มีระบบช่วยให้สังเกตความเปลี่ยนแปลงตัวแปรได้ง่ายและสามารถเข้าใจแนวทางที่ตัวแปรแต่ละตัวจะเปลี่ยนไปอย่างมีเหตุมีผล วิธีดังกล่าวจะเริ่มต้นการเปลี่ยนตัวแปรต่าง ๆ ให้มีผลต่อสมการกำหนดเป้าหมายโดยมีผลแนวโน้มสู่เป้าหมายในทางที่เร็วที่สุด การจัดรูปสมการเข้าเป็นตารางแล้วดำเนินการตามขั้นตอนที่ถูกต้อง และทำให้ได้ผลลัพธ์ตามเป้าหมายทั้งนี้การแก้ปัญหการออกแบบเครือข่ายนี้มีค่าตัวแปรที่เป็นจำนวนเต็ม ทำให้การหาค่าขอบเขตของคำตอบทำได้ยากและมักเป็นปัญหาแบบ NP-Hard (Charnkeitkong, 2007)

3) MEMETIC Algorithm

ขั้นตอนวิธีมีมิติก Ammaruekarat and Meesad, (2012); Wongthatsanekorn (2009); Chaimanee and Supithak (2018) เป็นอัลกอริทึมแบบเมตาฮิวริสติกที่คล้ายกับขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithm) Deb, Pratap, Agarwal and Meyarivan (2002); Srikaew, (2002); Nuipian and Phimpha (2019) ซึ่งใช้เทคนิคการค้นหาคำตอบด้วยหลักการของทฤษฎีวิวัฒนาการทางชีววิทยา และมีเทคนิคการค้นหาแบบเฉพาะที่เพิ่มเข้ามาเพื่อให้ได้คำตอบที่ดีขึ้น โดยในส่วนประกอบของโครโมโซมจะถูกเรียกว่า มีม (Meme) แทนคำว่ายีน (Gene) ทั้งนี้ ขั้นตอนวิธีมีมิติก ประกอบไปด้วยกระบวนการต่าง ๆ ดังนี้

3.1) กระบวนการคัดเลือก (selection) เป็น

กระบวนการเลือกโครโมโซมที่ดีที่สุด ซึ่งก็คือการทำให้เกิดการค้นหาคำตอบที่ดีที่สุดของปัญหาในแต่ละรุ่นของจำนวน

โครโมโซมทั้งหมด โดยเทคนิคที่ใช้ในงานวิจัยนี้ คือ การสุ่มแบบ วงล้อรูเล็ต (Roulette Wheel Selection) ซึ่งเทคนิคนี้ สามารถนำมาเปรียบเทียบได้กับวงล้อรูเล็ตที่มีช่อง Slot ขนาดไม่เท่ากันโดยช่อง Slot ที่มีขนาดใหญ่จะเทียบได้กับ โอกาสของโครโมโซมที่มีค่าความเหมาะสมมากและสามารถ ถูกเลือกไปสู่รุ่นถัดไปได้มากในขณะที่ช่อง Slot ที่มีขนาดเล็กจะเทียบได้กับโอกาสของโครโมโซมที่มีค่าความเหมาะสม น้อยและจะมีโอกาสถูกเลือกไปสู่รุ่นถัดไปได้น้อย ซึ่งขนาด Slot ของวงล้อรูเล็ตแต่ละช่องจะหาได้จากอัตราส่วนของค่า ความเหมาะสมของโครโมโซมแต่ละตัวกับค่าความเหมาะสม รวมของโครโมโซมทุกตัว ในการคัดเลือกจะทำการกำหนด จุดคงที่หนึ่งจุดแล้วหมุนวงล้อรูเล็ตโดยการสุ่ม เมื่อวงล้อ รูเล็ตหยุดที่จุดใด โครโมโซมที่ถูกแทนโดยช่อง Slot ช่อง นั้นจะถูกคัดเลือกไปเป็นโครโมโซมของประชากรรุ่นต่อไป จากนั้นจะทำการหมุนวงล้อเพื่อคัดเลือกโครโมโซมจนครบ ตามจำนวนประชากรที่ต้องการ

3.2) การแลกเปลี่ยนยีน (crossover) เป็นขั้นตอน วิธีเชิงพันธุกรรมเพื่อเปลี่ยนแปลงโครโมโซมที่ผ่าน กระบวนการคัดเลือกมาแล้ว ซึ่งกระบวนการนี้จะเริ่มต้น ด้วยการสุ่มเลือกโครโมโซมมาคู่หนึ่งจากโครโมโซมที่ผ่าน กระบวนการคัดเลือกมาแล้ว โดยโครโมโซมคู่นี้จะเรียกว่า โครโมโซมพ่อแม่ (parents) และจากนั้นจะทำการแลกเปลี่ยนยีนกันระหว่างโครโมโซมพ่อแม่เกิดเป็นโครโมโซม ลูกมาสองตัว แต่โดยทั่วไปแล้วการแลกเปลี่ยนยีนจะไม่ได้เกิดขึ้นกับทุกโครโมโซมพ่อแม่ที่ถูกเลือกมา ขึ้นอยู่กับ ความน่าจะเป็นในการเกิดการแลกเปลี่ยนยีน ที่มักจะมีค่า อยู่ในช่วง 0.7 ถึง 0.9 โดยหากไม่เกิดการแลกเปลี่ยนยีน จะทำให้โครโมโซมลูกที่ได้มีลักษณะเหมือนโครโมโซมพ่อแม่ทุกประการ เทคนิคของการแลกเปลี่ยนยีนที่ใช้ในงาน วิจัยนี้ คือ N-Point Crossing-over ที่เป็นการแลกเปลี่ยน ยีนระหว่างโครโมโซมที่จับคู่กัน โดยที่การแลกเปลี่ยนยีนนี้ จะเกิดขึ้นทางด้านใดด้านหนึ่งของจุดที่กำหนด แล้วสลับกัน ไปตามจุดของการแลกเปลี่ยนยีน โดยจำนวนจุดของการ แลกเปลี่ยนยีน n จุดนี้จะต้องมีค่ามากกว่าหรือเท่ากับ 1

3.3) การกลายพันธุ์ (mutation) เป็นการกลาย พันธุ์ของโครโมโซมโดย Allele ของยีนที่ตำแหน่งนี้จะถูก เปลี่ยนจาก 1 เป็น 0 ซึ่งการเกิดการกลายพันธุ์ นี้จะไม่ได้ เกิดขึ้นในทุกตำแหน่งของยีนบนโครโมโซม แต่จะเกิดได้

จากความน่าจะเป็นของการกลายพันธุ์ ซึ่งโดยทั่วไปแล้วจะมี ค่าอยู่ในช่วง 0 ถึง 0.1 ซึ่งน้อยมากเมื่อเทียบความน่าจะเป็น ในการแลกเปลี่ยนยีน ดังนั้นการแลกเปลี่ยนยีนจะเป็น ตัวดำเนินการหลักที่มีผลอย่างมากต่อขั้นตอนวิธีนี้มีติด ส่วน การกลายพันธุ์แม้จะมีโอกาสในการเกิดน้อยกว่าจึงทำให้มี ผลน้อย แต่ก็เป็นตัวดำเนินการหนึ่งที่สามารถสร้างโครโมโซม ที่มีความเหมาะสมกว่าประชากรรุ่นพ่อแม่ได้

3.4) การค้นหาแบบเฉพาะที่ (local search) (Sirinaovakul, 2008) มีความแตกต่างจากการค้นหาข้อมูล แบบธรรมดาที่ทำการค้นหาข้อมูลเพื่อจะตรวจสอบข้อมูลที ละตัวจนครบ แต่การค้นหาคำตอบเฉพาะที่นั้นจะไม่ตรวจสอบ ข้อมูลทุกตัว จะเลือกได้คำตอบที่เหมาะสมให้กับการ ค้นหา ซึ่งมีข้อดี คือ สามารถทำการค้นหาคำตอบจากข้อมูล ที่มีขนาดใหญ่มา ๆ ได้ แต่มีข้อเสีย คือ คำตอบที่ได้เป็น เพียงคำตอบที่ดีเท่านั้น ไม่แน่ว่าจะเป็นที่ดีที่สุด ตัวอย่างขั้นตอน วิธีที่ใช้ในการค้นหาคำตอบเฉพาะที่เช่น วิธีการค้นหาแบบไต่ เขา (hill climbing) การค้นหาแบบทาบ (Tabu search) และ Simulated annealing เป็นต้น ซึ่งงานวิจัยนี้เลือก ใช้วิธีการไต่เขาแบบง่าย (simple hill climbing) ที่เป็น วิธีการค้นหาข้อมูลที่มีลักษณะคล้ายกับการปีนภูเขา การที่ นักปีนภูเขาจะเลือกเดินทางไปถึงยอดภูเขาโดยเร็วที่สุด ซึ่ง จะมองทิศทางที่เมื่อปีนแล้วจะไถลยถอยเขาและหลีกเลี่ยง ทิศทางที่เมื่อไปแล้วจะทำให้ตัวเองห่างจากยอดเขา โดยมีข้อเสีย คือ ปีนขึ้นอย่างเดียวไม่มีการปีนลง ทำให้บางที เส้นทางที่ดีที่สุดอาจจะไม่สามารถทำให้ถึงเป้าหมายได้

ผลการวิจัย

ผู้วิจัยทำการทดลองเพื่อหาประสิทธิภาพด้านต่าง ๆ ของระบบเครือข่ายที่ถูกสร้างขึ้นด้วยวิธีนี้มีติดแยกตาม จำนวนจุด cross over 0 1 3 และ 6 จุด เปรียบเทียบกับวิธีซิมเพล็กซ์ และวิธีเมทาเทอร์-II โดยเปรียบเทียบ ประสิทธิภาพด้านค่าเวลาหน่วยของสายสัญญาณ เวลาที่ ใช้ในการประมวลผลรวมและค่าใช้จ่ายในการออกแบบ ทั้งระบบเครือข่าย ได้ผลค่าเฉลี่ยดังแสดงในตาราง 1 – 3 และผลค่า t-Test ดังตารางที่ 4

ตาราง 1

ค่าเฉลี่ยเวลาหน่วยในสายสัญญาณของเครือข่ายที่ออกแบบด้วยวิธีต่าง ๆ

จำนวนโหนด ในเครือข่าย	ค่าเฉลี่ยเวลาหน่วยในสายสัญญาณของเครือข่าย					
	Memetic-0	Memetic-1	Memetic-3	Memetic-6	Simplex	MENTOR-II
4	148.00	100.50	100.75	116.00	84.50	186.00
5	377.50	211.50	211.50	211.67	253.00	456.00
6	507.50	430.17	443.83	485.17	383.33	758.00
7	711.83	541.00	559.83	588.17	517.67	1162.00
8	1499.67	1211.17	1215.83	1314.50	1095.67	1895.67
9	2075.67	1815.17	1643.50	1801.00	1238.06	2485.00
10	2859.98	2722.58	2086.49	2549.32	-	3897.90

ตาราง 2

ค่าเฉลี่ยเวลาที่ใช้ในการประมวลผลออกแบบเครือข่ายด้วยวิธีต่าง ๆ

จำนวนโหนด ในเครือข่าย	ค่าเฉลี่ยเวลาที่ใช้ในการประมวลผลออกแบบเครือข่าย					
	Memetic-0	Memetic-1	Memetic-3	Memetic-6	Simplex	MENTOR-II
4	1224.72	8960.00	9050.00	9304.62	0.027	0.005
5	1357.92	12865.55	12817.77	13171.36	0.210	0.005
6	2005.50	24466.59	24614.63	27228.80	1.233	0.005
7	2727.69	36285.04	38294.29	42256.64	217.32	0.005
8	5396.80	59118.63	63284.85	75327.86	5558.03	0.020
9	9635.00	75512.02	81672.85	93797.50	335284.00	0.023
10	17222.77	93062.52	97911.08	129881.40	-	0.023

ตาราง 3

ค่าเฉลี่ยค่าใช้จ่ายในการติดตั้งเครือข่ายที่ออกแบบด้วยวิธีต่าง ๆ

จำนวนโหนด ในเครือข่าย	ค่าเฉลี่ยค่าใช้จ่ายในการติดตั้งเครือข่าย					
	Memetic-0	Memetic-1	Memetic-3	Memetic-6	Simplex	MENTOR-II
4	326.50	316.00	310.50	318.00	253.50	270.00
5	717.33	785.67	790.67	785.67	545.44	540.00
6	1521.17	1435.00	1412.00	1540.00	1110.00	930.00
7	1711.00	1775.83	1792.00	1959.33	1586.67	1190.00
8	3820.00	4334.33	4613.67	3977.67	3288.67	2243.60
9	4235.00	4643.67	5460.33	4797.00	3714.67	2902.00
10	4532.51	4595.64	6376.03	4618.67	-	4478.45

ตาราง 4

ค่า t -value เวลาหน่วยของวิธีมีติดที่พารามิเตอร์ต่าง ๆ เมื่อเทียบกับวิธีชิมเพล็กซ์และวิธีเมนเทอร์-II

t-Test เวลาหน่วยของวิธีมีติดที่พารามิเตอร์ต่าง ๆ				
เทียบกับ	Memetic-0	Memetic-1	Memetic-3	Memetic-6
Simplex	2.43833002	1.3186357	1.553	1.78114324
MENTOR-II	-3.02374247	-3.97528598	-3.00904127	-3.39907716

*นัยสำคัญทางสถิติที่ .05

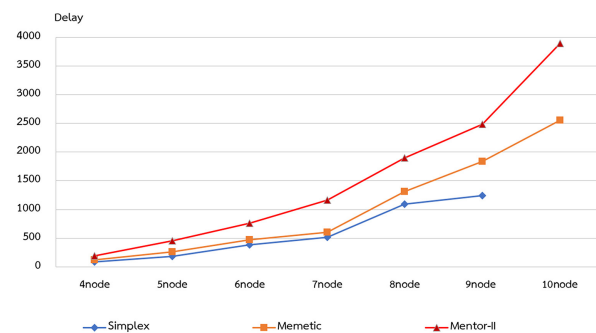
1) ผลการทดลองในด้านเวลาหน่วยของสายสัญญาณรวมที่เครือข่ายขนาด 4 5 6 8 9 และ 10 โหนด เป็นไปดังภาพ 4 โดยการเปรียบเทียบวิธีมีติด (เฉลี่ย) กับวิธีชิมเพล็กซ์และวิธีเมนเทอร์-II แสดงให้เห็นว่า วิธีชิมเพล็กซ์ให้เครือข่ายที่มีเวลาความหน่วงของสายสัญญาณน้อยกว่าวิธีมีติดและวิธีเมนเทอร์-II โดยวิธีมีติดมีประสิทธิภาพด้านเวลาหน่วยที่ด้อยกว่าวิธีชิมเพล็กซ์เล็กน้อย แต่ให้ประสิทธิภาพที่ดีกว่าวิธีเมนเทอร์-II

2) ผลการทดลองในด้านเวลาที่ใช้ออกแบบเครือข่าย (ในหน่วยวินาที) ที่เครือข่ายขนาด 4 5 6 8 9 และ 10 โหนด เป็นไปดังภาพ 5 โดยเมื่อทำการเปรียบเทียบวิธีมีติด (เฉลี่ย) กับวิธีชิมเพล็กซ์และวิธีเมนเทอร์-II พบว่า วิธีเมนเทอร์-II ใช้เวลาในการประมวลผลออกแบบน้อยกว่าวิธีชิมเพล็กซ์และวิธีมีติด แต่มีข้อสังเกตว่า วิธีการชิมเพล็กซ์นั้นจะใช้เวลาน้อยกว่าวิธีมีติดสำหรับเครือข่ายที่มีขนาดไม่เกิน 8 โหนด แต่สำหรับเครือข่ายขนาดใหญ่ที่มี 9 โหนดขึ้นไป วิธีมีติดจะใช้เวลาออกแบบน้อยกว่าวิธีชิมเพล็กซ์ และยังมีแนวโน้มที่จะใช้เวลาน้อยกว่ายิ่งขึ้นเมื่อเครือข่ายที่ออกแบบมีขนาดใหญ่ขึ้น

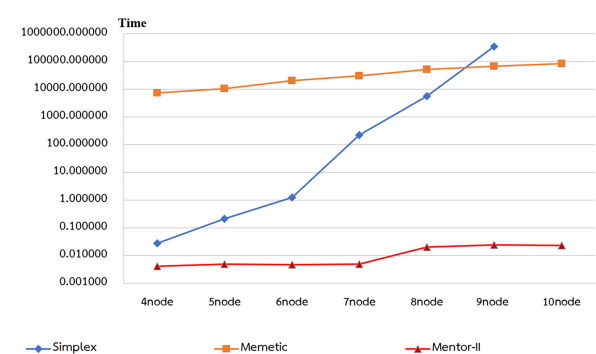
3) ผลการทดลองในด้านค่าใช้จ่ายในการออกแบบระบบเครือข่ายที่ขนาด 4 5 6 8 9 และ 10 โหนด เป็นไปดังภาพ 6 โดยเมื่อทำการเปรียบเทียบวิธีมีติด (เฉลี่ย) กับวิธีชิมเพล็กซ์และวิธีเมนเทอร์-II พบว่า วิธีมีติดให้เครือข่ายที่มีค่าใช้จ่ายในการติดตั้งที่มากกว่าวิธีชิมเพล็กซ์และวิธีเมนเทอร์-II ตามลำดับ

ทั้งนี้ ผู้วิจัยได้ทดลองเปรียบเทียบเวลาที่ใช้ในการออกแบบเครือข่ายด้วยวิธีมีติดที่มีจำนวนจุดแลกเปลี่ยน

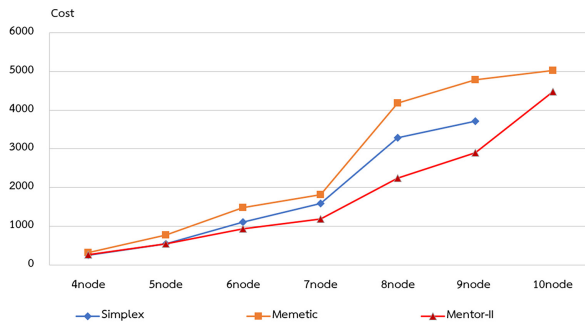
มีที่แตกต่างกัน ได้ผลดังแสดงในภาพ 7 โดยพบว่า การมีกระบวนการแลกเปลี่ยนมีจะทำให้ใช้เวลาในการประมวลผลออกแบบมากขึ้น และหากมีจำนวนจุดแลกเปลี่ยนมีมากขึ้นก็จะใช้เวลาในการออกแบบมากขึ้นเล็กน้อย



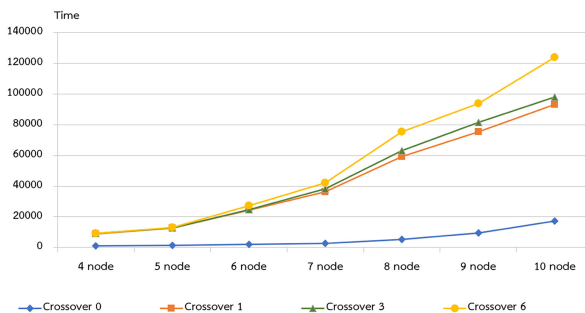
ภาพ 4 ความต่างของค่าเวลาหน่วยของสายสัญญาณรวมของเครือข่ายขนาด 4-10 โหนด



ภาพ 5 เวลาในการประมวลผลออกแบบเครือข่ายขนาด 4-10 โหนด



ภาพ 6 ค่าใช้จ่ายในการติดตั้งเครือข่ายขนาด 4-10 โหนด



ภาพ 7 เวลาที่ใช้การออกแบบเครือข่าย 4-10 โหนด ด้วยวิธีมีติก ที่จำนวนจุดแลกเปลี่ยนมี เท่ากับ 0 1 3 และ 6

การอภิปรายผล

ผู้วิจัยได้ทำการทดสอบประสิทธิภาพการออกแบบโครงข่ายด้วยวิธีมีติกเปรียบเทียบกับวิธีซิมเพล็กซ์และวิธีเมนเทอร์-II โดยเริ่มทดลองด้วยวิธีมีติกเปรียบเทียบกับวิธีซิมเพล็กซ์ ที่แปรผันค่าจำนวนโหนดของโครงข่าย ค่าโหนดดีกรี และความต้องการการไหลของข้อมูลในโครงข่ายรวมเป็นจำนวนทั้งสิ้น 432 เครือข่าย ส่วนการทดลองเปรียบเทียบระหว่างวิธีมีติกกับวิธีเมนเทอร์-II นั้น ได้ทำการทดลองเป็นจำนวนทั้งสิ้น 210 เครือข่าย

จากตาราง 4 แสดงให้เห็นว่า ค่าเวลาหน่วยของเครือข่ายที่ใช้วิธีมีติกแบบ 1 3 และ 6 จุดแลกเปลี่ยนมีเมื่อนั้นเมื่อเปรียบเทียบกับวิธีซิมเพล็กซ์จะมีค่าใกล้เคียงกัน ส่วนค่าเวลาหน่วยของเครือข่ายที่ใช้วิธีมีติกแบบไม่มีจุดแลกเปลี่ยนมีเมื่อนั้นเมื่อเปรียบเทียบกับวิธีซิมเพล็กซ์จะมีค่ามากกว่าวิธีซิมเพล็กซ์ที่ระดับนัยสำคัญทางสถิติ .05 ส่วนค่าเวลาหน่วยของเครือข่ายที่ใช้วิธีมีติกทุกแบบเมื่อเปรียบเทียบกับวิธีเมนเทอร์-II จะพบว่า มีค่าน้อยกว่าวิธีเมนเทอร์-II ที่ระดับนัยสำคัญทางสถิติ .05 จึงสรุปได้ว่า ในด้านของเวลาหน่วยของสายสัญญาณในเครือข่าย วิธีมีติกที่มีการแลกเปลี่ยนมีมีประสิทธิภาพด้านเวลาหน่วยที่ใกล้เคียงหรือดีกว่าวิธีซิมเพล็กซ์เล็กน้อย แต่ให้ประสิทธิภาพที่ดีกว่าวิธีเมนเทอร์-II

ส่วนในด้านของเวลาที่ใช้ในการประมวลผลเพื่อออกแบบเครือข่าย พบว่า วิธีเมนเทอร์-II ใช้เวลาในการประมวลผลออกแบบน้อยกว่าวิธีซิมเพล็กซ์และวิธีมีติก ทั้งนี้มีข้อสังเกตว่าวิธีการซิมเพล็กซ์นั้นจะใช้เวลาน้อยกว่าวิธีมีติกสำหรับเครือข่ายที่มีขนาดไม่เกิน 8 โหนด แต่สำหรับเครือข่ายขนาดใหญ่ตั้งแต่ 9 โหนดขึ้นไป วิธีมีติกจะใช้เวลาออกแบบน้อยกว่าวิธีซิมเพล็กซ์ และยังมีแนวโน้มที่จะใช้เวลาน้อยกว่ายิ่งขึ้นเมื่อเครือข่ายที่ออกแบบมีขนาดใหญ่ขึ้น ในด้านของค่าใช้จ่ายของการติดตั้งเครือข่ายพบว่า วิธีมีติกให้เครือข่ายที่มีค่าใช้จ่ายในการติดตั้งมากกว่าวิธีซิมเพล็กซ์และวิธีเมนเทอร์-II ตามลำดับ

จึงสามารถสรุปได้ว่า วิธีมีติกเป็นวิธีการออกแบบโครงข่ายที่มีประสิทธิภาพดีในแง่ของเวลาหน่วยและเวลาในการประมวลผลออกแบบ ซึ่งออกแบบได้เครือข่ายที่มีเวลาหน่วยน้อยใกล้เคียงกับวิธีซิมเพล็กซ์ แต่กระบวนการมีความซับซ้อนน้อยกว่าทำให้ออกแบบได้เร็วและเหมาะสมสำหรับใช้ในการออกแบบเครือข่ายขนาดใหญ่กว่าวิธีซิมเพล็กซ์



References

- Ammaruekarat, P., & Meesad, P. (2012). Optimization by Multi-Objective Chaos-Memetic Algorithm. *Information Technology Journal*, 8(1), 54-61. (in Thai)
- Cahn, R. (1998). *The design tool: Delite (software)*. Retrieved from <http://www.mkp.com/wand.htm>.

- Chaimanee, A., & Supithak, W. (2018). A memetic algorithm to minimize the total sum of earliness tardiness and sequence dependent setup costs for flow shop scheduling problems with job distinct due windows. *Songklanakarin Journal of Science and Technology*, 40(5), 1203-1218. (in Thai)
- Charnkeitkong, P. (2007). *Network analysis and design*. Pathumthani: Rangsit University. (in Thai)
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evolutionary Computation*, 6(2), 182-197. doi: 10.1109/4235.996017.
- Dehestani, A., & Hajipour, P. (2010). Comparative study of M/Er/1 and M/M/1 queuing delay models of the two IP-PBXs. *Journal of Convergence Information Technology*, 5(2), 36-42.
doi:10.4156/JCIT.VOL5.ISSUE2.4
- Duvedi, A., Ashraf, A., & Gairola, S. (2022). A Comparative study on routing protocols: RIP, OSPF and EIGRP. *Proceeding of International Conference on Cyber Resilience, Vol.2* (pp.519–528). Dubai UAE: Skyline University College.
- Fortz, B., & Thorup, M. (2000). Internet traffic engineering by optimizing OSPF Weights, *Proceeding of IEEE Computer and Communications Societies Vol.2* (pp.519–528). Tel Aviv: IEEE Computer Society.
- Jaroenrat, K., & Samrongsub, T. (2014). OSPF network traffic engineering by branch exchange method. *The Journal of King Mongkut's University of Technology North Bangkok*, 24(3), 503-511. (in Thai)
- Lanthong, K., & Jaroenrat, K. (2012). OSPF weight optimization by linear programming method. *Proceeding of 9th National Kasetsart University Kamphaeng Saen Conference* (pp. 180-186). Nakhon Pathom: Kasetsart University. (in Thai)
- Makhorin, A. (2000). *GLPK (GNU linear programming kit)*. Retrieved from <http://www.gnu.org/software/glpk>.
- Monsakul, A. (2015). Performance of design with MENTOR-II Algorithm for IP Backbone network of Internet Service Provider. *Information Technology Journal*, 11(2), 62-69. (in Thai)
- Nuipian, V., & Phimpha, A. (2019). Comparison of selection and mutation in genetic algorithms for examination schedule problem. *The Journal of Industrial Technology*, 15(2), 112-124. (in Thai)
- Sirinaovakul, B. (2008). *Artificial Intelligence: Swarm Intelligence*. Bangkok: Top. (in Thai)
- Srikaew, A. (2002). Genetic Algorithm–Part I. *Suranaree Journal of Science and Technology*, 9, 69-83. (in Thai)
- Wongthatsanekorn, W. (2009). Reviewing and comparisons of five evolutionary-based algorithms. *The Journal of King Mongkut's University of Technology North Bangkok*, 19(2), 285-290. (in Thai)

