

ประสิทธิภาพของ MENTOR อัลกอริทึมในการออกแบบที่มีต่อการใช้เส้นทางโครงข่าย

On Routing Efficiency of a Network Design Algorithm

นายรังษี เจริญรายภูร์, อัจฉริยะ แบกซอง, พศ.ดร.พิสิษฐ์ ชาญเกียรติกุล

คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยรังสิต

บทคัดย่อ

Mesh Network Topological Optimization and Routing (MENTOR) เป็นอัลกอริทึมที่ใช้ในการออกแบบโครงข่าย Partial Mesh ที่มีความซับซ้อนต่ำและมีประสิทธิภาพดี โดยการศึกษานี้แสดงให้เห็นความสัมพันธ์ระหว่างค่าตัวแปรที่กำหนดในการออกแบบและประสิทธิภาพในการจัดการการไฟล์ข้อมูลที่ใช้ MENTOR อัลกอริทึม ซึ่งผู้วิจัยได้ทำการวิเคราะห์ โครงข่ายจำนวน 432 โครงข่ายที่ออกแบบด้วยอัลกอริทึม MENTOR พบว่า ประสิทธิภาพในการจัดการการไฟล์ข้อมูลนั้นขึ้นอยู่กับชนิดของ Tree ที่ MENTOR เลือกใช้ในช่วงต้น และการกำหนดค่าต่ำสุดและสูงสุดของ Traffic ที่ผ่านสายสัญญาณผลการวิจัยพบว่าโครงข่ายที่เกิดจาก MENTOR ที่เริ่มต้นด้วย Shortest Path Tree จะมีประสิทธิภาพดีกว่าที่เริ่มต้นด้วยแบบ Minimum Spanning Tree และยังพบว่าประสิทธิภาพในการจัดการการไฟล์จะใกล้เคียงค่าสูงสุด เมื่อความแตกต่างระหว่างค่าต่ำสุดและสูงสุดของอัตราการใช้งานสายสัญญาณที่ยอมรับได้มีค่าน้อย และประสิทธิภาพจะลดลงเมื่อความแตกต่างนี้มีค่านักขั้น นอกจากนี้ยังพบว่าประสิทธิภาพในการจัดการการไฟล์ลดลงเมื่อตัวกรีดของ node เพิ่มขึ้น และค่าสูงสุดของอัตราการใช้งานสายสัญญาณที่ยอมรับได้มีค่าน้อยลง

คำสำคัญ: MENTOR Algorithm, ออกแบบโครงข่าย, การทำเส้นทางที่เหมาะสม, Mesh network, Traffic routing

Abstract

Mesh Network Topological Optimization and Routing (MENTOR) algorithm is a low complexity and efficient partial mesh networks design algorithm. This study explores the relation between design parameters and performance of traffic assignment of MENTOR algorithm. We analyze 432 networks designed by MENTOR for 4 sets of 50 nodes each with equivalently distributed demand and randomly generated locations. For each of these networks, the performances at normal load and at congestion threshold of MENTOR flow assignment are calculated and compared with the optimum solution obtained by solving the linear programming. It is found that the routing performances depend on the initial tree used in the MENTOR algorithm, as well as the allowable minimum and maximum link utilization. MENTOR networks start with star topology give much better performance than that start with minimum spanning tree. In term of utilization, routing performances keep very close to that of the optimal when the gap between maximum and minimum utilization is small and get as worse the gap increase. The impacts of node degree on routing performances are also investigated. We observed that the performances decrease as node degree increase, and get worse when maximum utilization increase.

Key-Words: - MENTOR Algorithm, network design, Optimum routing, Mesh network, traffic routing

1. บทนำ

เมื่อ Internet กลายเป็นโครงสร้างพื้นฐานสำคัญทางธุรกิจ การออกแบบระบบสื่อสารข้อมูลขนาดใหญ่จำเป็นต้องคำนึงถึงประเด็นใหม่ๆ มากขึ้น หนึ่งในประเด็นที่สำคัญที่สุดคือ IP Network เป็น datagram network ซึ่ง routing protocol ต้อง route traffic ในระหว่างที่สั้นที่สุด และเพื่อแก้ปัญหาดังกล่าว backbone ของ ISPs ส่วนใหญ่จะใช้วิธี Overlay ซึ่ง route traffic บน Permanent Virtual Connections (PVC) ของ ATM หรือบังก์ใช้ Label Switch Paths (LSP) Multi Protocol Label Switching (MPLS) [3]

ประเด็นสำคัญอีกประการหนึ่งที่ ISPs ต้องคำนึงถึงคือ Traffic Demand ที่เดินโดยย่างราวดเร็ว ทำให้ ISPs ต้องจัดสรร Capacity ของโครงข่ายไว้รองรับการขยายตัวดังกล่าวในอนาคต ซึ่งสามารถควบคุมด้วยได้ด้วยค่าของ parameters ที่ใช้ในการออกแบบโครงข่าย ซึ่งที่ชัดเจนที่สุดคือค่าที่ยอมรับได้ของ maximum link utilization และ minimum link utilization โดยการตั้งค่าที่ยอมรับได้ของ maximum link utilization และ minimum link utilization ไว้ต่ำ มักจะนำไปสู่ระบบโครงข่ายที่มีค่าใช้จ่ายสูงแต่สามารถสำรอง Capacity ไว้ได้มากขึ้น

Kershenbaum และคณะ [1] ได้เสนอ Heuristic อัลกอริทึม ที่มีความซับซ้อนต่ำ (ระดับ $O(N^2)$) ที่เรียกว่า MENTOR (Mesh Network Topological Optimization and Routing) ซึ่งโครงข่ายที่ได้จาก MENTOR นั้นมีสมรรถนะใกล้เคียงกับโครงข่ายที่ได้จากอัลกอริทึมมีความซับซ้อนสูง และ MENTOR ยังมีความยืดหยุ่นมากพอที่จะนำไปใช้เป็นกระบวนการในการออกแบบโครงข่าย Virtual Circuit อย่างเช่น ATM หรือ MPLS ซึ่งใช้เป็นระบบ Overlay สำหรับ Backbone ของ ISP เลยทีเดียว อย่างไรก็ตามการ routing traffic ของ MENTOR ไม่ได้เป็นวิธีการที่ดีที่สุดเสมอไป ทั้งนี้ขึ้นอยู่กับ parameters ต่างๆ ในการออกแบบด้วย

งานวิจัยนี้แสดงให้เห็นความสัมพันธ์ระหว่างค่าตัวแปรสำคัญในการออกแบบและประสิทธิภาพในการ

จัดการ traffic ของ อัลกอริทึม MENTOR โดยผู้เขียนได้วิเคราะห์ โครงข่ายจำนวน 432 โครงข่าย ที่ออกแบบโดย MENTOR ซึ่งแต่ละ โครงข่ายมีจำนวน Node 50 nodes ส่วนการกำหนดค่า Traffic demand และตำแหน่งของ Node ในแต่ละกลุ่มนี้จะทำโดยวิธีสุ่ม แล้วคำนวณประสิทธิภาพในการจัดสรร Traffic ของแต่ละ โครงข่าย ที่ปริมาณการใช้งานปกติ และในภาวะที่เริ่มแออัด และนำมาเปรียบเทียบกับการจัดสรร Traffic ที่ดีที่สุด (Optimum solution) ซึ่งมาจากการแก้ปัญหาโดยใช้ linear programming

2. ปัญหาในการวิจัย

2.1 อัลกอริทึม MENTOR

MENTOR อัลกอริทึม [1] เป็นอัลกอริทึมในการออกแบบระบบโครงข่ายแบบ Heuristic ที่มีความซับซ้อนต่ำ ซึ่งเกิดจากพิจารณาการ Route ไปด้วยในขณะที่พิจารณาการตัดตั้ง link โดย traffic demand จะถูกกำหนดให้ใช้เส้นทางที่ตรงมากที่สุด หากที่มี Utilization ที่เหมาะสม

กระบวนการของ MENTOR เริ่มด้วยการจัดกลุ่มซึ่งในขั้นตอนนี้ โหนดจะแยกออกเป็นโหนดต้นทาง และ backbone ทั้งนี้ในงานวิจัยนี้ผู้เขียนพิจารณาเฉพาะในกรณีที่แต่ละ โหนดเป็น Backbone Node

จากนั้น Tree จะถูกสร้างขึ้นมาเพื่อเชื่อมระหว่าง Backbone Node ทั้งหมด โดยใช้วิธีที่พัฒนาจากอัลกอริทึม ของ Prim และ Dijkstra ซึ่งในการสร้าง tree แต่จะมีค่าตัวแปร

$\alpha, 0 \leq \alpha \leq 1$ เป็นตัวกำหนดชนิดของ tree โดยเมื่อ $\alpha=0$ จะเป็น MST(Minimum Spanning Tree) และเมื่อ $\alpha=1$ จะเป็น SPT(Shortest Path Tree)

หลังจากได้ Tree แล้ว MENTOR จะตัดตั้ง Direct Link ระหว่างคู่ Node หากมีปริมาณ traffic ที่เหมาะสม กำหนดให้ค่าสูงสุดของ utilization เป็น ρ ส่วนค่าต่ำสุดของ utilization จะอยู่ในรูปของ ρ และ slack(s) ดังสมการ $(1-s)\rho$ เมื่อ $s, 0 \leq s \leq 1$ หากพิจารณาระหว่างคู่ Node A และ B โดยให้ C_{AB} เป็น

Link Capacity และ l_{AB} เป็น load ทั้งหมดที่ไหลระหว่าง A กับ B โดยอัตราที่มีนี้จะพิจารณาการติดตัว Link โดยถ้า traffic ระหว่างจุด A และ B น้อยมากๆ ซึ่งคือ $l_{AB} < \rho C_{AB}(1-s)$ Link นี้จะไม่ถูกติดตัวและ traffic ทั้งหมดของ l_{AB} ก็จะลับไปยังเส้นทางต่อตัวที่สั้นที่สุดเป็นลำดับรองลงไป แต่หาก traffic อยู่ระหว่างค่าสูงสุดและต่ำสุด $\rho C_{AB}(1-s) \geq l_{AB} \leq \rho C_{AB}$ Link ก็จะถูกติดตัว อย่างไรก็ได้ในกรณีที่ $l_{AB} > \rho C_{AB}$ ทางเขื่อมต่อตัวจะถูกเพิ่มเข้าไป โดยทางเขื่อมใหม่ C_{AB} จะถูกติดตัวเข้าไปเพื่อรับ traffic ρC_{AB} และ traffic ที่เหลือ จะลับไปยังเส้นทางที่ใกล้ที่สุดเป็นลำดับรองลงไป

MENTOR นี้ให้ผลลัพธ์ที่ค่อนข้างดี อย่างไรก็ตาม การศึกษาเกี่ยวกับผลกระทบในการออกแบบค่าต่างๆ อาทิ ρ , s และ α ที่มีต่อประสิทธิภาพของการกำหนดเส้นทางการขนถ่ายข้อมูล นั้นยังไม่ได้มีการศึกษามาก่อน (5)

2.2 Objective Function

เมื่อพิจารณาไฟของระบบโครงข่าย $G = (N, A)$ ความเร็ว C_a สำหรับแต่ละ $a \in A$ และตามที่อธิบายไว้ใน ก่อนหน้านี้ กำหนดให้ d_{st} แทนจำนวนของ traffic flow ระหว่าง s และ t ให้ f_a^{st} แสดงปริมาณของ traffic flow จาก s ถึง t บนส่วนโถง a

ปริมาณ traffic l_a บน link $a \in A$ เป็นผลรวมของ f_a^{st} ใน [4] นี้ได้เสนอวิธีการวัดประสิทธิภาพของระบบโครงข่ายไว้โดยใช้ฟังก์ชันต้นทุน (cost function)

$$\Phi = \sum_{a \in A} \Phi_a(l, C_a), \quad (1)$$

ในขณะที่ $\phi_a(l_a, C_a)$ อิงกับค่า Delay ในตัวแบบ M/M/1 ทฤษฎี Queueing ซึ่งได้จาก

$$\phi_a(l_a, C_a) = l_a / (C_a - l_a) \quad (2)$$

2.3 Optimum Solution

อย่างไรก็ได้สมการนี้ใช้ไม่เหมาะสมในการคำนวนหาค่า optimum ในกรณีที่ l_a มีค่ามาก เช่น $l_a \geq C_a$ เพื่อแก้ปัญหานี้ $l_a / (C_a - l_a)$ จะถูกประมาณค่าโดยใช้ฟังก์ชันเชิงเส้นแบบ piecewise ดังนี้

$$\phi'_a(l_a, C_a) = \begin{cases} 1 & \text{for } 0 \leq l_a/C_a \leq 1/3, \\ 3 & \text{for } 1/3 \leq l_a/C_a < 2/3, \\ 10 & \text{for } 2/3 \leq l_a/C_a \leq 9/10, \\ 70 & \text{for } 9/10 \leq l_a/C_a < 1, \\ 500 & \text{for } 1 \leq l_a/C_a < 11/10, \\ 5000 & \text{for } 11/10 \leq l_a/C_a < \infty. \end{cases} \quad (3)$$

ด้วยฟังก์ชันเชิงเส้นที่เป็น piecewise ตามสมการ (3) ปัญหาการหาเส้นทางเดินของข้อมูลสามารถแสดงในรูปของโปรแกรมเชิงเส้นดังสมการ [4] [5]

$$\text{Min } \Phi = \sum_{a \in A} \phi_a \quad (4)$$

$$\sum_{u:(u,v) \in A} f_{u,v}^{s,t} - \sum_{u:(u,v) \in A} f_{v,u}^{s,t} = \begin{cases} d_{st} & \text{if } v = t \\ -d_{st} & \text{if } v = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$v, s, t \in N,$$

$$\phi_a \geq l_a \quad a \in A, \quad (6)$$

$$\phi_a \geq 3l_a - 2/3c_a \quad a \in A, \quad (7)$$

$$\phi_a \geq 10l_a - 16/3c_a \quad a \in A, \quad (8)$$

$$\phi_a \geq 70l_a - 178/3c_a \quad a \in A, \quad (9)$$

$$\phi_a \geq 500l_a - 1468/3c_a \quad a \in A, \quad (10)$$

$$\phi_a \geq 5000l_a - 19468/3c_a \quad a \in A, \quad (11)$$

$$l_a = \sum_{t \in N} f_a^{s,t} \quad a \in A, \quad (12)$$

$$f_a^{s,t} \geq 0 \quad a \in A; t \in N. \quad (13)$$

ข้อกำหนด (5) แทนเงื่อนไข Flow conservation ข้อกำหนด (6)–(11) แทนเงื่อนไขของ (3) และข้อกำหนด (12) แทนเงื่อนไขของ load บนแต่ละ Arc และเพื่อให้สามารถเปรียบเทียบผลกับโครงข่ายที่ได้จาก MENTOR ซึ่งมีการกำหนดค่าสูงสุดของ Utilization อย่างเช่น อัตราที่มี ผู้วิจัยจึงได้เพิ่มเงื่อนไขดังสมการ (14) ลงไปด้วย

$$l_a / C_a \leq \rho \quad (14)$$

2.4 Normalizing Routing Cost

Fortz และ Thorup [3] ได้เสนอ Normalize scaling Factor สำหรับทำให้การเปรียบเทียบระหว่างโครงข่ายที่มีขนาดและ topology ต่างกัน มีความเป็นไปได้ดังสมการ (15)

$$\Phi_{UNCAP} = \sum_{s,t \in NxN} d_{s,t} h_{s,t} \quad (15)$$

เมื่อ h_{st} = จำนวน hop น้อยสุดระหว่าง s และ t สำหรับ routing cost Φ โดยนั้น จะมี scaling routing cost หรือ normalizing routing cost ดังสมการ

$$\Phi^* = \Phi / \Phi_{UNCAP} \quad (16)$$

สมการต่างๆข้างต้นนี้เป็นสมการ Linear Programming ที่สมบูรณ์แบบในการแก้ปัญหาทั่วไปของ การส่งถ่ายข้อมูล ซึ่งเราใช้ Φ แทน Routing cost general routing cost ที่ดีที่สุด

3. วิธีการวิจัย

3.1 ระบบวิธีวิจัย

ผู้จัดได้ใช้โปรแกรม DELITE [6] ในการ สังเคราะห์ระบบขนาด 50 Node จำนวน 4 ชุด โดยแต่ละ ชุดแตกต่างกันที่การกระจายของ Node ที่กำหนด ซึ่งผู้ทดสอบจะแทนโครงข่ายแต่ละชุดด้วย N1, N2, N3 และ N4 ส่วนค่า traffic demand สำหรับโครงข่ายแต่ละชุดได้ ถูกกำหนดจากโดยให้ค่าผลรวมของ traffic ขาเข้า และ traffic ขาออกจากแต่ละ Node เท่ากับ 100 Mbps

โครงข่ายที่ใช้ในการทดลองนี้เป็นโครงข่ายที่ สร้างด้วยอัลกอริทึม MENTOR ทั้งหมด 432 โครงข่ายซึ่ง Link เป็นแบบ full-duplex ความเร็ว 45 Mbps โดย 4 ชุด กีอ และแต่ชุด N1-N4 จะถูกแบ่งเป็นสองกลุ่ม กลุ่มแรก เป็นโครงข่ายที่เริ่มด้วย MST ($\alpha=0$) ส่วนกลุ่มที่สองกีอ โครงข่ายที่เริ่มด้วย SPT ($\alpha=1$) และสำหรับแต่ละกลุ่มจะ ประกอบไปด้วย 54 โครงข่าย ที่ทำการเปลี่ยนแปลงตัว

แปรดังนี้ กีอให้ ρ , $\rho \in (0.4, 0.5, 0.6, 0.7, 0.8, 1.0)$
และ s, $s \in (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8)$

3.2 Routing Performances

การหาเส้นทางที่เหมาะสมที่สุดในแต่ละ โครงข่ายจะทำโดยกำหนดค่า ρ ตามที่อธิบายไว้ใน section 2.3 และหาเส้นทางด้วยโปรแกรม GLPK[2] ซึ่ง ใน Intel Pentium IV Xenon 3.3 GHz จะใช้เวลาประมาณ 2 ชั่วโมง เพื่อจะหาเส้นทางที่เหมาะสมดังกล่าว ด้วย demand ของโครงข่ายที่กำหนด ประสิทธิภาพในการ กำหนดการไฟลของ MENTOR ณ โหลดปกติ จะถูก ประเมินค่าโดยผลต่างเป็น % ของ Cost ของโครงข่ายที่ สร้างด้วยอัลกอริทึม MENTOR กับ โครงข่ายที่ดีที่สุดที่ ค้นพบด้วยวิธี Linear Programming

$$\Delta C = \frac{\Phi_M^* - \Phi_O^*}{\Phi_O^*} \times 100, \quad (17)$$

เมื่อ Φ_M^* เป็นค่า Normalized Cost ของการ กำหนดการไฟลด้วยวิธี MENTOR และ Φ_O^* เป็นค่าที่ได้จากวิธีแก้ปัญหาที่ดีที่สุด

จากหัวข้อที่ 2 จะเห็นว่าค่า Cost function จะ เพิ่มขึ้นอย่างรวดเร็ว หลังจาก $\phi_a = 10/3$ ประสิทธิภาพ การไฟลของข้อมูลในโครงข่ายที่ออกแบบด้วยอัลกอริทึม MENTOR จะเข้าไปสู่จุดวิกฤตจนถึงจุดคับกั่ง ซึ่งวัดจาก ผลต่างของ Cost เป็น % จากค่าที่ได้จากโครงข่ายที่ดีที่สุด

$$\Delta D = \frac{D_M - D_O}{D_O} \times 100 \quad (18)$$

เมื่อ D_M คือ Scaling traffic demand ของ โครงข่ายที่สร้างจาก MENTOR และ D_O คือค่าที่ได้จาก โครงข่ายที่ดีที่สุด โดยวัดเมื่อค่า $\phi_a = 10/3$ โดยผลการ ทดลองเป็นดังรูปที่ 1-16

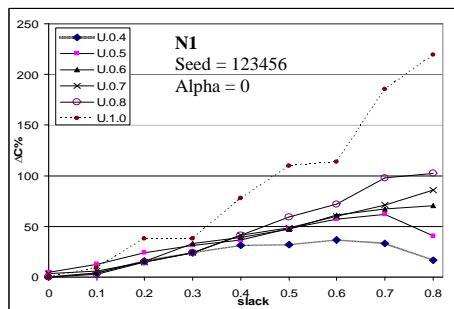
3.3 วิเคราะห์ผลการทดลอง

จากรูป 1-4 แสดงค่า ΔC ของระบบโครงข่ายที่ออกแบบด้วยอัลกอริทึม MENTOR ด้วย $\alpha = 0$ สำหรับ N1-N4 ส่วนและรูปที่ 5-8 แสดงค่า ΔC ของโครงข่ายที่ออกแบบด้วยอัลกอริทึม MENTOR ด้วย $\alpha = 1$ สำหรับ N1-N4 พบว่า ในโครงข่ายที่ $\alpha = 0$ นั้น ΔC มีค่าเฉลี่ยอยู่ที่ 5.5244% ในขณะที่ ในโครงข่ายที่ $\alpha = 1$ นั้น ΔC จะมีค่าเฉลี่ยอยู่ที่ 0.0037% ซึ่งต่างก็เป็นค่า ΔC ที่จัดว่าน้อยมาก

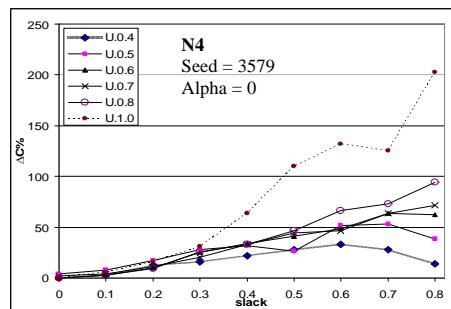
จากรูป 9-12 แสดงให้เห็น ΔD ของระบบโครงข่ายที่ออกแบบด้วยอัลกอริทึม MENTOR ด้วย $\alpha =$

0 สำหรับ N1-N4 ตามลำดับ และภาพที่ 13-16 แสดงให้เห็น ΔD ของระบบโครงข่ายที่ออกแบบด้วยอัลกอริทึม MENTOR ด้วย $\alpha = 1$ สำหรับ N1-N4 ตามลำดับ พบว่า ในโครงข่ายที่ $\alpha = 0$ นั้น ΔD จะมีค่าเฉลี่ยอยู่ที่ 0.9629% ในขณะที่ ในโครงข่ายที่ $\alpha = 1$ นั้น ΔD จะมีค่าเฉลี่ยอยู่ที่ 0.0077% ซึ่งต่างก็เป็นค่า ΔD ที่จัดว่าน้อยมาก

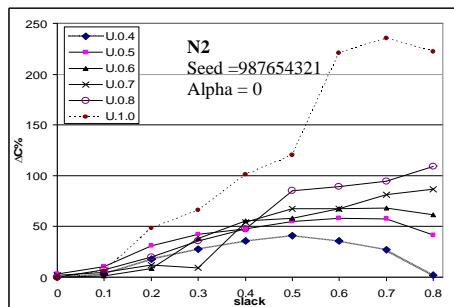
เมื่อเปรียบเทียบค่า ΔC และ ΔD แล้วจะเห็นได้ว่า ประสิทธิภาพของระบบที่มี $\alpha = 1$ มีประสิทธิภาพมากกว่าระบบที่มี $\alpha = 0$



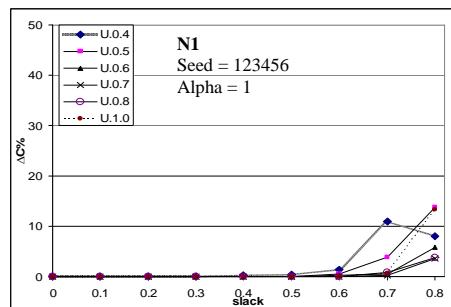
รูปที่ 1 แสดง ΔC ของ Network ที่มี ค่า $\alpha = 0$ สำหรับ N1.



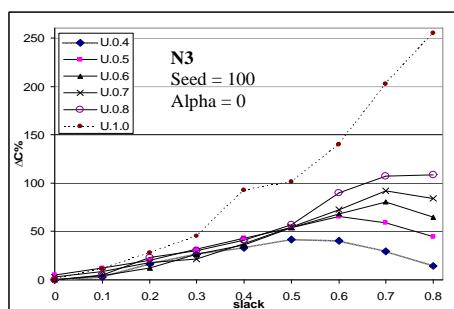
รูปที่ 4 แสดง ΔC ของ Network ที่มี ค่า $\alpha = 0$ สำหรับ N4.



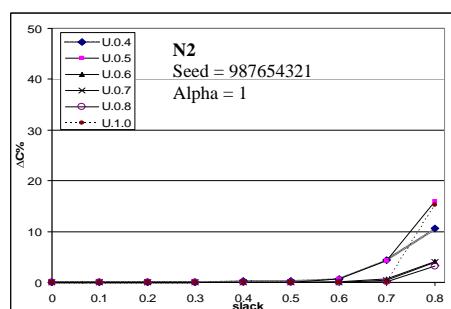
รูปที่ 2 แสดง ΔC ของ Network ที่มี ค่า $\alpha = 0$ สำหรับ N2.



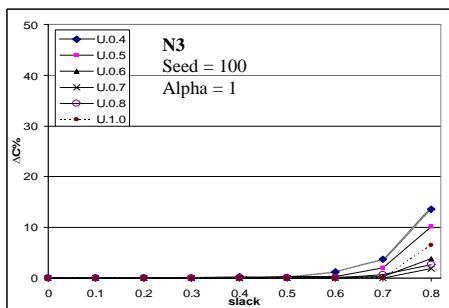
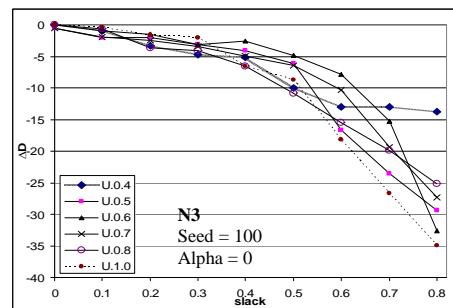
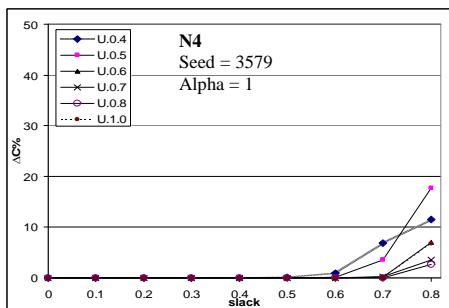
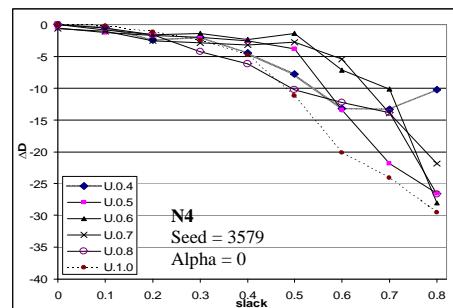
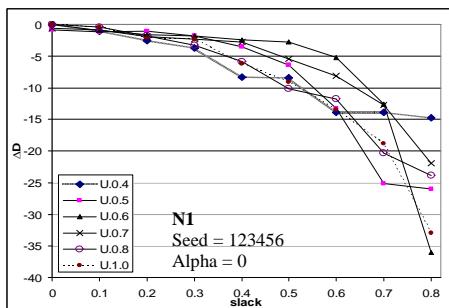
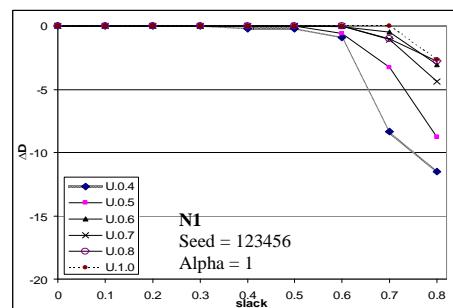
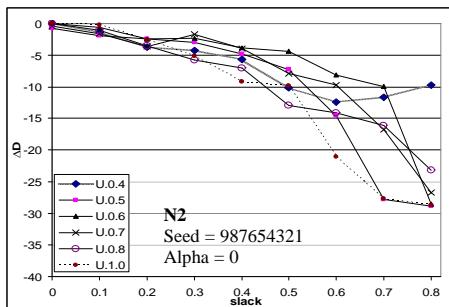
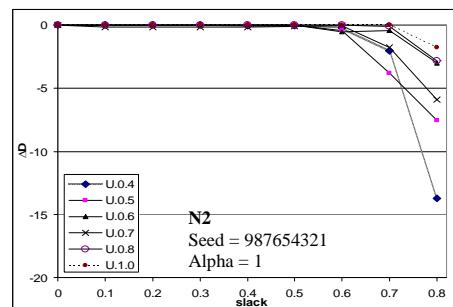
รูปที่ 5 แสดง ΔC ของ Network ที่มี ค่า $\alpha = 1$ สำหรับ N1.

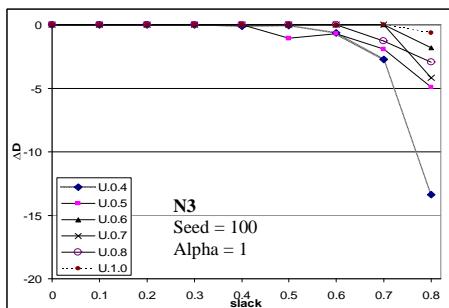
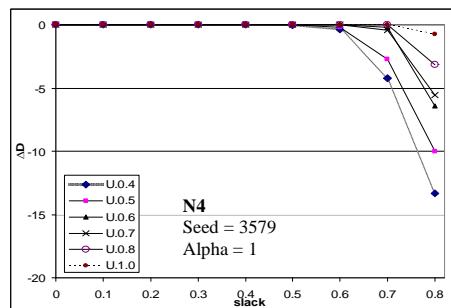


รูปที่ 3 แสดง ΔC ของ Network ที่มี ค่า $\alpha = 0$ สำหรับ N3.



รูปที่ 6 แสดง ΔC ของ Network ที่มี ค่า $\alpha = 1$ สำหรับ N2.

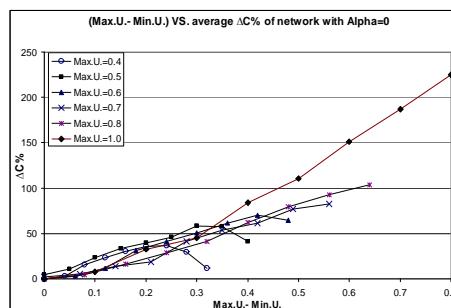
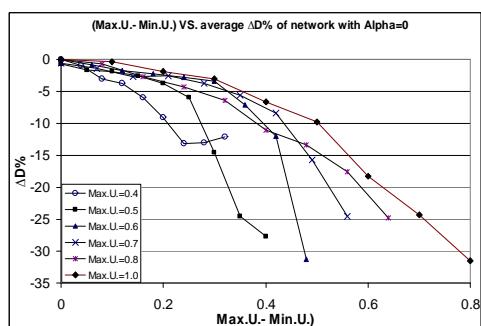
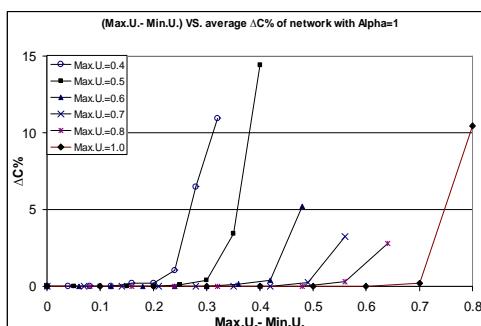
รูปที่ 7 แสดง ΔC ของ Network ที่มีค่า $\alpha = 1$ สำหรับ N3.รูปที่ 11 แสดง ΔD ของ Network ที่มีค่า $\alpha = 0$ สำหรับ N3.รูปที่ 8 แสดง ΔC ของ Network ที่มีค่า $\alpha = 1$ สำหรับ N4.รูปที่ 12 แสดง ΔD ของ Network ที่มีค่า $\alpha = 0$ สำหรับ N4.รูปที่ 9 แสดง ΔD ของ Network ที่มีค่า $\alpha = 0$ สำหรับ N1.รูปที่ 13 แสดง ΔD ของ Network ที่มีค่า $\alpha = 1$ สำหรับ N1.รูปที่ 10 แสดง ΔD ของ Network ที่มีค่า $\alpha = 0$ สำหรับ N2.รูปที่ 14 แสดง ΔD ของ Network ที่มีค่า $\alpha = 1$ สำหรับ N2.

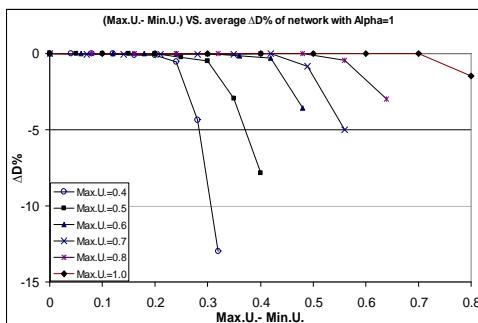
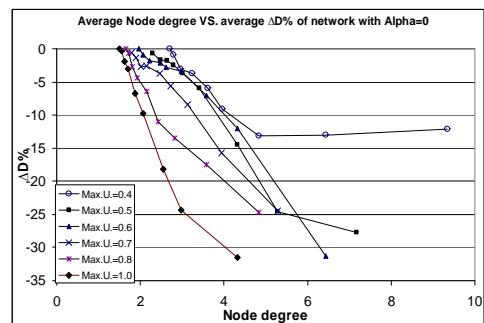
รูปที่ 15 แสดง ΔD ของ Network ที่มีค่า $\alpha = 1$ สำหรับ N3.รูปที่ 16 แสดง ΔD ของ Network ที่มีค่า $\alpha = 1$ สำหรับ N4.

เพื่อแสดงให้เห็นความสัมพันธ์ระหว่างประสิทธิภาพในการ routing กับ utilization ให้ขัดเจนขึ้น ผู้วิจัยได้นำค่าเฉลี่ยของ ΔC และ ΔD ที่มี α , ρ และ s เท่ากัน และ plot กราฟเทียบกับ $\Delta U = s\rho$ ซึ่งคือความแตกต่างระหว่างค่าต่ำสุดและสูงสุดของ utilization โดยรูปที่ 17-18 แสดงให้เห็นว่า สำหรับ $\alpha = 0$ ทั้ง ΔC และ ΔD จะมากขึ้น เมื่อ ΔU เพิ่มขึ้น และในรูปที่ 18 ซึ่งให้ ρ ก่ออยู่เพิ่มขึ้นตามค่า ΔU ที่เพิ่มขึ้นตาม ρ ส่วนรูปที่ 19-20 แสดงให้เห็นเมื่อค่า $\alpha = 1$ ความสัมพันธ์จะเป็นลักษณะเส้นโค้งลง และจากค่า ρ ที่กำหนดทั้ง ΔC และ ΔD จะมีค่าน้อยเข้าใกล้ค่า Optimal และจะเพิ่มอย่างรวดเร็วเมื่อ ΔU เข้าใกล้ ρ

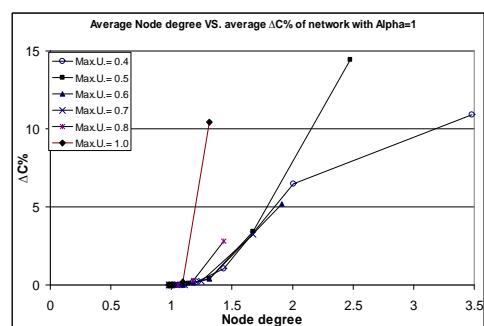
เนื่องจากใน [3] และ [7] พบว่า node degree (δ) ของโครงข่ายที่ได้จากการอัลกอริทึม MENTOR ดังนั้นผู้วิจัยจึงทำการวิเคราะห์ความสัมพันธ์ระหว่าง δ และสมรรถนะของโครงข่าย ค่าเฉลี่ย δ ของโครงข่ายที่สร้างขึ้นใน section 3.2 ที่มี α , ρ และ s เท่ากันแสดงได้ตามตารางที่ 1 และตารางที่ 2

ในรูปที่ 21-22 ที่ค่า $\alpha = 0$ ผลลัพธ์ที่ได้แสดงให้เห็นว่าเมื่อทั้ง ΔC และ ΔD เพิ่มขึ้นอย่างรวดเร็วจนกระทั่งค่าของ δ ขึ้นอยู่กับ ρ หลังจากนั้น ΔC และ ΔD ก็จะเริ่มคงที่ หรือลดลงอย่างช้าๆ ในรูปที่ 23-24 ที่ค่า $\alpha = 1$ ได้แสดงทั้ง ΔC และ ΔD ที่เพิ่มขึ้นเมื่อ δ เพิ่ม โดยเส้นกราฟคาดขึ้นตามค่า δ ที่เพิ่มขึ้น

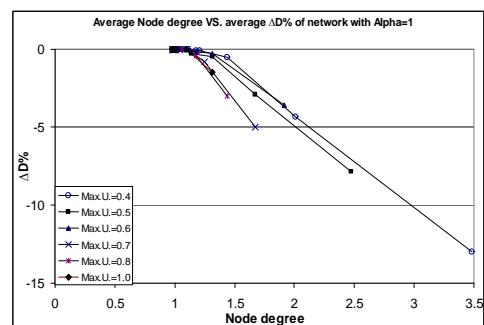
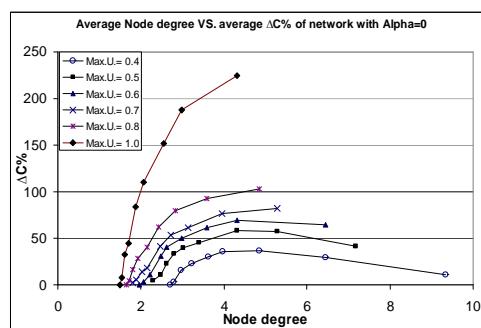
รูปที่ 17 แสดงค่าเฉลี่ยของ ΔC เทียบกับ ΔU สำหรับ $\alpha = 0$ รูปที่ 18 แสดงค่าเฉลี่ยของ ΔD เทียบกับ ΔU สำหรับ $\alpha = 0$ รูปที่ 19 แสดงค่าเฉลี่ยของ ΔC เทียบกับ ΔU สำหรับ $\alpha = 1$

รูปที่ 20 แสดงค่าเฉลี่ยของ ΔD เทียบกับ ΔU สำหรับ $\text{Alpha}=1$ รูปที่ 22 แสดงค่าเฉลี่ยของ ΔD เทียบกับ Δ สำหรับ $\text{Alpha}=0$

Node Degree of Network with Alpha=0						
slack	max U=0.4	max U=0.5	max U=0.6	max U=0.7	max U=0.8	max U=1.0
0	2.705	2.295	1.985	1.795	1.655	1.49
0.1	2.795	2.485	2.06	1.895	1.715	1.545
0.2	2.96	2.625	2.22	2.02	1.8	1.615
0.3	3.23	2.785	2.48	2.145	1.925	1.7
0.4	3.615	3.01	2.62	2.46	2.145	1.875
0.5	3.965	3.4	2.98	2.715	2.435	2.07
0.6	4.845	4.315	3.585	3.14	2.825	2.54
0.7	6.44	5.29	4.315	3.95	3.585	2.98
0.8	9.345	7.18	6.445	5.29	4.845	4.315

ตารางที่ 1 แสดงระดับของ Node ใน Network ที่มีค่า $\text{Alpha}=0$ รูปที่ 23 แสดงค่าเฉลี่ยของ ΔC เทียบกับ Δ สำหรับ $\text{Alpha}=1$

Node Degree of Network with Alpha=1						
slack	max U=0.4	max U=0.5	max U=0.6	max U=0.7	max U=0.8	max U=1.0
0	1.005	0.99	0.985	0.98	0.98	0.98
0.1	1.03	1	0.985	0.985	0.98	0.98
0.2	1.06	1.01	0.995	0.985	0.985	0.98
0.3	1.11	1.045	1.005	0.995	0.985	0.98
0.4	1.18	1.095	1.03	1.005	0.995	0.985
0.5	1.2075	1.14	1.095	1.045	1.005	0.99
0.6	1.435	1.315	1.18	1.11	1.06	1.01
0.7	2.01	1.675	1.315	1.25	1.18	1.095
0.8	3.485	2.475	1.915	1.675	1.435	1.315

ตารางที่ 2 แสดงระดับของ Node ใน Network ที่มีค่า $\text{Alpha}=1$ รูปที่ 24 แสดงค่าเฉลี่ยของ ΔD เทียบกับ Δ สำหรับ $\text{Alpha}=1$ รูปที่ 21 แสดงค่าเฉลี่ยของ ΔC เทียบกับ Δ สำหรับ $\text{Alpha}=0$

4. สรุปผลการวิจัย

งานวิจัยนี้แสดงให้เห็นความสัมพันธ์ระหว่างประสิทธิภาพของการกำหนดเส้นทางการไฟฟ้ากับตัวแปรในการออกแบบของ MENTOR โดยการวิเคราะห์ข้อมูลใน 432 โครงข่าย ที่สร้างด้วย MENTOR ประสิทธิภาพของ network ถูกประเมินด้วย ΔC หรือค่าร้อยละของความเบี่ยงเบนของ Cost ในการ routing จากค่าที่ดีที่สุดเมื่อการปักดิ้น และ ΔD หรือร้อยละของความ

เบื้องบนของ Traffic Demand จากค่าที่ดีที่สุดการ เมื่อต้นทุนในการ routing ρ เท่ากับจุดที่เก็บจะแอกัดผลการวิจัยพบว่าโครงข่าย MENTOR ที่เริ่มต้นด้วย SPT จะมีประสิทธิภาพกว่าที่เริ่มต้นด้วยแบบ MST และยังพบว่าประสิทธิภาพในการจัดการการไหลจะใกล้เคียงค่าที่ดีที่สุดเมื่อ Δ_U หรือความแตกต่างระหว่างค่าต่ำสุดและสูงสุดของอัตราการใช้งานที่ยอมรับได้ของสายสัญญาณมีค่าน้อย และประสิทธิภาพจะลดลงเมื่อความแตกต่างนี้มีค่าเพิ่ม นอกจากนี้ยังพบว่า Node degree (δ) มีผลกระทบต่อประสิทธิภาพของการ routing โดยที่ Δ_C และ Δ_D จะลดลงเมื่อ δ เพิ่มขึ้น และจะแยกออกเมื่อ ρ เพิ่มขึ้น

References:

- [1] Aaron Kershenbaum, Parviz Kermani, and George A. Grover, MENTOR: An Algorithm for Mesh Network Topological Optimization and Routing, *IEEE Transaction on Communications*, Vol.39, April 1991, pp.503-513.
- [2] Andrew Makhorin. *GLPK (GNU Linear Programming Kit)*, 2000, <http://www.gnu.org/software/glpk/>
- [3] Bernard Fortz, Jennifer Rexford, and Mikkel Thorup, Traffic engineering with traditional IP routing protocols, *IEEE Commun. Mag.*, vol. 40, Oct. 2002, pp. 118–124.
- [4] Bernard Fortz and Mikkel Thorup, Internet traffic engineering by optimizing OSPF weights, in *Proc. IEEE INFOCOM*, vol. 2, Mar. 2000, pp.519–528.
- [5] Kairat Jaroenrat, Posit CharnKeitkong, On Routing Performance of MENTOR Algorithm. *WSEAS TRANSACTION ON COMMUNICATIONS*, Issue 9, Vol.5, September 2006, pp. 1769-1776.
- [6] Robert Cahn, *Wide Area Network Design*. Morgan Kaufmann Publisher, San Francisco, CA, 1998.
- [7] Walid Ben-Ameur, Eric Gourdin, Bernard Liau, Nicholas Michel, “Designing Internet Networks”; in *Proceeding of DRCN2000*, pages 56-61, Munich, April 2000