

การหาเส้นทางมัลติแคสต์แบบไดนามิกด้วยการสงวนทรัพยากรล่วงหน้า

DYNAMIC MULTICAST ROUTING ALGORITHM WITH ADVANCE
RESOURCE RESERVATIONS

อภิพล คุณาภิบาล
APIPOL GUNABHIBAL

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของงานศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2546

ISBN 974-324-872-2

การหาเส้นทางมัลติแคสต์แบบไดนามิกด้วยการสงวนทรัพยากรล่วงหน้า

DYNAMIC MULTICAST ROUTING ALGORITHM WITH ADVANCE
RESOURCE RESERVATIONS



อภิพล คุณาภิบาล

APIPOL GUNABHIBAL

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า
บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2546

ISBN 974-324-872-2

มี.....
ทะเบียน..... 48943 ✓
เดือน, ปี 13 ส.ค. 2547

.b.....
.i.....

**DYNAMIC MULTICAST ROUTING ALGORITHM WITH ADVANCE
RESOURCE RESERVATIONS**

APIPOL GUNABHIBAL

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2003

ISBN 974-324-872-2

COPYRIGHT 2003

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การหาเส้นทางมัลติแคสต์แบบไดนามิกด้วยการสงวนทรัพยากรล่วงหน้า
DYNAMIC MULTICAST ROUTING ALGORITHM WITH ADVANCE
RESOURCE RESERVATIONS

ชื่อนักศึกษา นายอภิพล คุณาภิบาล

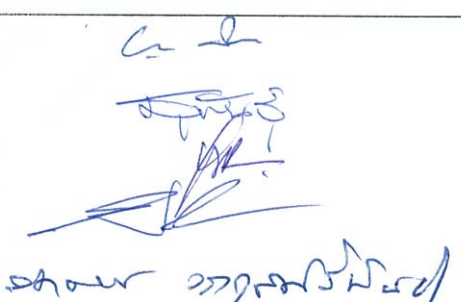
รหัสประจำตัว 42061039

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมไฟฟ้า

อาจารย์ผู้ควบคุมวิทยานิพนธ์ รศ.ดร.รัตติกกร วรากุลศิริพันธุ์

อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม ผศ.ดร.โชติพัชร์ ภรณ์วลัย

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
ดร.บุษยามา	คิดใจเดี๋ยว	
ดร.สุพันธ์	ตั้งจิตกุศลมัน	
รศ.ดร.บุษยามา	รังสรรค์เสรี	
ผศ.ดร. โชติพัชร์	ภรณ์วลัย	
รศ.ดร.รัตติกกร	วรากุลศิริพันธุ์	

วัน/เดือน/ปี ที่สอบ 25 กันยายน 2546 เวลา 10.30-12.30 น.
สถานที่สอบ ณ อาคาร 12 ชั้น ชั้น 4 (ห้อง E12-404)


บัณฑิตวิทยาลัยรับรองแล้ว
(รศ.ดร.บุญวัฒน์ อดิษฐ์)
คณบดีบัณฑิตวิทยาลัย

วันที่.....19.....เดือน.....พฤษภาคม.....พ.ศ. 2546

หัวข้อวิทยานิพนธ์	การหาเส้นทางมัลติแคสต์แบบไดนามิกด้วยการสงวนทรัพยากร ล่วงหน้า
นักศึกษา	นาย อภิล คุณาภิบาล
รหัสประจำตัว	42061039
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
พ.ศ.	2546
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ. ดร. รัตติกร วรากุลศิริพันธุ์
อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม	ผศ. ดร. โชติพัชร ภรณวลัย

บทคัดย่อ

รูปแบบการสงวนทรัพยากรสำหรับโปรแกรมประยุกต์ที่ถูกร้องขอใช้บริการนั้น เป็นหนึ่งในแนวคิดที่มีประสิทธิภาพที่ถูกเสนอขึ้นมา เพื่อรองรับ โปรแกรมประยุกต์แบบหลายสื่อ ที่มีส่วนเกี่ยวข้องกับเวลาอย่างสูง เนื่องด้วยทรัพยากรในระบบเครือข่ายนั้นมีอยู่อย่างจำกัด ความสามารถที่จะสงวนทรัพยากรล่วงหน้าในโครงข่ายแบบนั้นจึงมีความจำเป็นอย่างมากสำหรับ โปรแกรมประยุกต์ที่สามารถรองรับผู้ใช้บริการได้จำนวนหลายคน และมีการเข้าถึงและออกจากกลุ่มของการสื่อสารอย่างอิสระ เช่น ระบบการสื่อสารร่วมระยะไกลในยุคใหม่ เป็นต้น อัลกอริทึมโมดิฟายด์กรีดี เป็นอัลกอริทึมที่มีประสิทธิภาพสูงสำหรับการสร้างเส้นทางมัลติแคสต์แบบไดนามิก ภายใต้สภาวะการสงวนทรัพยากรแบบล่วงหน้า แต่อย่างไรก็ตามประสิทธิภาพของอัลกอริทึมนี้ยังมีความอ่อนไหวต่อความถูกต้องแม่นยำของข้อมูลที่ถูกแจ้งมาโดยผู้ขอใช้บริการอยู่พอสมควร ในงานวิทยานิพนธ์นี้ เราจึงได้นำเสนออัลกอริทึมโมดิฟายด์กรีดี ในรูปแบบพัฒนา ซึ่งมีความยืดหยุ่นมากขึ้นต่อความผิดพลาดของข้อมูลที่แจ้งเข้ามา กับสถานการณ์ที่เกิดขึ้นจริงได้ รวมถึงเราจักได้ แสดงให้เห็นถึงความมีประสิทธิภาพของการรวมแนวคิดของการสงวนทรัพยากรแบบล่วงหน้าเข้ากับการหาเส้นทางมัลติแคสต์แบบไดนามิกเพื่อการลดความสูญเสียของการใช้ทรัพยากรในระบบเครือข่าย ดังแสดงในผลการทดลอง

Thesis Title	Dynamic Multicast Routing Algorithm with Advance Resource Reservations
Student	Mr. Apipol Gunabhibal
Student ID.	42061039
Degree	Master of Electrical Engineering
Programme	Electrical Engineering
Year	2001
Thesis Advisor	Assoc. Prof. Dr. Ruttikorn Varakulsiripunth
Thesis Assoc. Advisor	Asst. Prof. Dr. Chotipat Pornavalai

ABSTRACT

Reserving the resources for the requested applications is one of the most effective schemes that were proposed to offer the time-critical multimedia applications recently. Because the resource is limited, an ability to provide resource reservations in advance, is essential in multi-party applications with dynamic accessing and leaving of user such as modern integrated (voice, video, and data) collaboration system. Modified Greedy algorithm was proposed for multicast routing in advance reservations environment. However, the performance of this algorithm was quite sensitive to the precision of the information given by user. In this work, we improved this algorithm in order to obtain a good performance, even when the inaccuracy of information given by user occurred. We also show the efficiency of network resources reduction by combining reservation concepts with dynamic multicast routing as shown in simulation results.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี โดยได้รับความช่วยเหลือและสนับสนุน จากบุคคลหลายๆ ฝ่าย ซึ่งผู้เขียนขอขอบพระคุณทุกท่านอย่างสูง มา ณ โอกาสนี้

คุณพ่อคุณแม่ ผู้เล็งเห็นความสำคัญของการศึกษา คอยสนับสนุนเป็นกำลังใจให้ด้วยดีเสมอมา

บริษัท ทศท. คอร์ปอเรชั่น จำกัด มหาชน ซึ่งได้สนับสนุนทุนการศึกษาตลอดหลักสูตร

รศ. ดร. รัตติกอ วรากุลศิริพันธุ์ อาจารย์ที่ปรึกษา และ ผศ. ดร. โชติพัชร ภรณ์วลัย อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม ที่กรุณาให้คำปรึกษา ชี้แนะแนวทางที่เป็นประโยชน์ในการพัฒนางานวิจัย รวมทั้งเสียสละเวลาในตรวจทานแก้ไข วิทยานิพนธ์นี้จึงสำเร็จลุล่วงด้วยดี

รศ. ดร. มนต์ สัจวรศิลป์ และ รศ. ประภากร สุวรรณะ ที่ได้อบรมและสั่งสอนวิชาความรู้ซึ่งเป็นประโยชน์ในการประกอบอาชีพ

คุณกานต์มณี กุลวานิช ที่คอยให้กำลังใจ ช่วยเหลือในการจัดทำต้นฉบับ และตรวจทานแก้ไขวิทยานิพนธ์ให้มีความสมบูรณ์ยิ่งขึ้น

คุณพันธ์ศักดิ์ อาภาขจร คุณประมาณ อุสาหะการี และเพื่อนๆ พี่ๆ ที่ สำนักงานมาตรฐานเชื่อมต่อโครงข่าย บริษัท ทศท. คอร์ปอเรชั่น จำกัด มหาชน ที่ให้ความเอื้อเฟื้อและอำนวยความสะดวกให้เสมอมา

ห้องปฏิบัติการเครือข่ายการสื่อสาร (Communication Network Lab.) สำนักวิจัยการสื่อสารและเทคโนโลยีสารสนเทศ (ReCCIT : Research Center for Communications and Information Technology) ที่สนับสนุนอุปกรณ์และสถานที่การทำวิจัย

และทุกท่านที่ไม่ได้กล่าวถึงในที่นี้ ที่ให้ความช่วยเหลือและกำลังใจ ในการทำวิทยานิพนธ์ฉบับนี้จนสำเร็จ

คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ผู้เขียนขอขอบแต่ผู้มีพระคุณทุกท่าน

อภิพล คุณาภิบาล

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	1
1.3 สมมติฐานของการศึกษา.....	1
1.4 ทฤษฎีและแนวความคิดที่ใช้ในงานวิจัย.....	2
1.4.1 อัลกอริทึมการคำนวณหาเส้นทางที่สั้นที่สุด ระหว่างจุดในกราฟ.....	2
1.4.2 อัลกอริทึมการคำนวณเส้นทางต้นไม้ที่สั้นที่สุด.....	5
1.5 ขอบเขตการวิจัย.....	8
1.6 ขั้นตอนการศึกษา.....	8
บทที่ 2 การสร้างเส้นทางมัลติแคสต์.....	9
2.1 การส่งข้อมูลในระบบเครือข่ายสื่อสาร.....	9
2.2 การให้บริการมัลติแคสต์ในระบบเครือข่าย.....	10
2.2.1 การสร้างเส้นทางต้นไม้เพื่อการให้บริการมัลติแคสต์.....	10
2.2.2 ปัญหาการสร้างเส้นทางต้นไม้เพื่อการส่งข้อมูลมัลติแคสต์.....	12
2.2.3 ประเภทของปัญหาการสร้างเส้นทางมัลติแคสต์.....	13
2.3 การสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบสแตติก.....	14
2.4 การสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิก.....	16
2.4.1 อัลกอริทึมการหาเส้นทางแบบเนฟ.....	17
2.4.2 อัลกอริทึมการหาเส้นทางแบบกรีดี.....	18
2.4.3 อัลกอริทึมแบบทังก์เสมือน.....	18

สารบัญ (ต่อ)

	หน้า
2.5 บทสรุป.....	21
บทที่ 3 เส้นทางมัลติแคสต์แบบสงวนทรัพยากรล่วงหน้า.....	22
3.1 การสงวนทรัพยากรล่วงหน้าในระบบเครือข่าย.....	22
3.2 เจือไนซ์การสร้างเส้นทางมัลติแคสต์แบบสงวนทรัพยากรล่วงหน้า.....	24
3.3 อัลกอริทึมการสร้างเส้นทางมัลติแคสต์โมดิฟายด์กรีดี.....	25
3.4 ประโยชน์ของการสร้างเส้นทางมัลติแคสต์ด้วยการสงวนทรัพยากรล่วงหน้า.....	28
3.5 ประสิทธิภาพของเส้นทางมัลติแคสต์แบบสงวนทรัพยากรล่วงหน้า.....	30
3.5.1 สภาพแวดล้อมการหาเส้นทางบนระบบเครือข่ายจำลอง.....	31
3.5.2 ค่าที่ใช้ในการวัดประสิทธิภาพของอัลกอริทึม การสร้างเส้นทางมัลติแคสต์.....	32
3.5.3 นโยบายการกระจายค่าสถานะลิงค์.....	33
3.5.4 ค่าปริมาณความปลอดภัยของลิงค์ในระบบเครือข่าย.....	34
3.5.5 อัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบรับประกันแบนด์วิธ.....	36
3.5.6 ผลการทดลองบนระบบเครือข่ายจำลอง.....	38
3.6 บทสรุป.....	39
บทที่ 4 การพัฒนาอัลกอริทึมการหาเส้นทางมัลติแคสต์.....	41
4.1 การเปลี่ยนแปลงเงื่อนไขของการสร้างเส้นทางมัลติแคสต์.....	41
4.2 วิเคราะห์ผลกระทบของค่า $\Delta t_{dep}(z)$ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์.....	42
4.2.1 ผลต่อค่าคอสต์ของเส้นทางมัลติแคสต์เมื่อ $\Delta t_{dep}(z) > 0$	43
4.2.2 ผลต่อค่าคอสต์ของเส้นทางมัลติแคสต์เมื่อ $\Delta t_{dep}(z) < 0$	44
4.3 แนวทางการพัฒนาอัลกอริทึม โมดิฟายด์กรีดีแบบใหม่.....	45
4.4 อัลกอริทึม โมดิฟายด์กรีดีแบบใหม่.....	47
4.4.1 อัลกอริทึมการเพิ่ม โหนดสมาชิก.....	47
4.4.2 อัลกอริทึมการลบ โหนดสมาชิก.....	51
4.5 บทสรุป.....	54

สารบัญ (ต่อ)

	หน้า
บทที่ 5 รายงานผลการทดลอง.....	55
5.1 ขั้นตอนของการทำซิมูเลชัน.....	55
5.1.1 การสร้างระบบเครือข่ายแบบสุ่ม.....	55
5.1.2 การกำหนดลำดับการเข้าร่วมกลุ่มมัลติแคสต์ของ โหนดสมาชิก.....	57
5.2 การซิมูเลชันเพื่อวัดประสิทธิภาพของอัลกอริทึม โมดิฟายด์กรีดีแบบใหม่.....	59
5.2.1 ผลกระทบของพารามิเตอร์ γ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์.....	60
5.2.2 ผลกระทบของพารามิเตอร์ $P(\Delta t_{dep}(z))$ ต่อค่าคอสต์ ของเส้นทางมัลติแคสต์.....	62
5.2.3 ผลกระทบของพารามิเตอร์ $\Delta t_{dep}(z)$ ต่อค่าคอสต์ ของเส้นทางมัลติแคสต์.....	63
5.3 บทสรุป.....	65
บทที่ 6 รายงานสรุปและข้อเสนอแนะ.....	66
6.1 รายงานสรุป.....	66
6.2 ข้อเสนอแนะเพื่องานวิจัยในอนาคต.....	67
เอกสารอ้างอิง.....	68
ภาคผนวก ก รหัสเทียมของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ โมดิฟายด์กรีดีแบบใหม่.....	70
ภาคผนวก ข ผลงานวิจัยที่ได้รับการตีพิมพ์.....	74
ประวัติผู้เขียน.....	80

สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางการทำงานของอัลกอริทึมแบบไดจิสต์รา	5
1.2 การปรับปรุงค่าผลตกในแต่ละรอบการทำงานของอัลกอริทึมของพริม	7
2.1 แสดงเปรียบเทียบจำนวนฮอปของการส่งข้อมูลมัลติแคสต์	12
3.1 ตารางเวลาการเข้าและออกจากกลุ่มมัลติแคสต์ของโหนดสมาชิก	29
3.2 ค่าคอสต์เสมือนของแต่ละเส้นทาง	30
4.1 เวลาการเข้าร่วมและออกจากกลุ่มมัลติแคสต์ของโหนดสมาชิก	49
5.1 ค่าพารามิเตอร์ที่ใช้ในการสร้างเครือข่ายแบบสุ่ม	57
5.2 เวลาของการส่งข้อมูลมัลติแคสต์ T เมื่อพารามิเตอร์ γ มีค่ามากขึ้น	59
5.3 ผลกระทบของค่า γ ต่อคอสต์ของเส้นทางมัลติแคสต์	60
5.4 ผลกระทบของค่า $P(\Delta_{t_{dep}}(z))$ ต่อคอสต์ของเส้นทางมัลติแคสต์	62
5.5 ผลกระทบของค่า $\Delta_{t_{dep}}(z)$ ต่อคอสต์ของเส้นทางมัลติแคสต์	64

สารบัญรูป

รูปที่	หน้า
1.1 กราฟ G ซึ่งเปรียบเทียบกับได้กับระบบเครือข่ายจริง	3
1.2 กราฟประกอบตัวอย่างการคำนวณหาเส้นทางที่สั้นที่สุด	4
1.3 ตัวอย่างของเส้นทาง	5
1.4 กราฟประกอบการแสดงตัวอย่างการหาเส้นทางต้นไม้ที่สั้นที่สุด	7
2.1 การติดต่อสื่อสารระหว่างผู้ใช้บริการผ่านระบบเครือข่ายสื่อสาร	9
2.2 ตัวอย่างการส่งข้อมูลมัลติแคสต์ในระบบเครือข่าย	11
2.3 ตัวอย่างแสดงลำดับการทำงานของอัลกอริทึมเคเอ็มบี	16
2.4 ลำดับขั้นตอนการทำงานของการสร้างทรีค์เสมือน	20
3.1 ค่าคอสต์เสมือนของเส้นทางมัลติแคสต์	26
3.2 การคำนวณค่าคอสต์เสมือน $W(s, x_i)$	27
3.3 การคำนวณค่าคอสต์เสมือน $W(x_i, z)$	28
3.4 ตัวอย่างการคำนวณเส้นทางมัลติแคสต์	29
3.5 เส้นทางมัลติแคสต์เมื่อรวม โหนด e	30
3.6 ค่าปริมาณความปลอดภัยของลิงค์	35
3.7 ตัวอย่างของการสร้างเส้นทางการรับประกันแบนด์วิธแบบกรีดี	37
3.8 การเปรียบเทียบประสิทธิภาพของอัลกอริทึมเมื่อ b_{req} เท่ากับ 0.34	38
3.9 การเปรียบเทียบประสิทธิภาพของอัลกอริทึมเมื่อ b_{req} เท่ากับ 0.45	39
4.1 ผลกระทบของค่า $\Delta_{dep}(z)$ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์	43
4.2 โหนดสมาชิกออกจากกลุ่มมัลติแคสต์ล่าช้ากว่ากำหนดหรือ $\Delta_{dep}(z) > 0$	44
4.3 โหนดสมาชิกออกจากกลุ่มมัลติแคสต์เร็วกว่ากำหนดหรือ $\Delta_{dep}(z) < 0$	45
4.4 ตัวอย่างการคำนวณหาเส้นทางมัลติแคสต์ด้วยอัลกอริทึมแบบใหม่	49
4.5 ตัวอย่างการคำนวณค่ากรอบเวลา	50
4.6 เส้นทาง $abcd$, abd , และ $abgfd$	50
4.7 เส้นทางมัลติแคสต์หลังจากเชื่อมต่อกับโหนดสมาชิกใหม่	51
4.8 ตัวอย่างการลบโหนดสมาชิกออกจากเส้นทางมัลติแคสต์	52
4.9 แผนผังการไหลของอัลกอริทึมการลบโหนดสมาชิก	53
4.10 เส้นทางมัลติแคสต์หลังจากทำการลบโหนดสมาชิกออก	53
5.1 ตัวอย่างของระบบเครือข่ายแบบสุ่ม	56

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.2 การเข้าและออกจากกลุ่มมัลติแอสท์ของโนนครสมาชิกภายในช่วงเวลา T	58
5.3 ผลกระทบของค่า γ ต่อคอสท์ของเส้นทางมัลติแอสท์	61
5.4 ผลกระทบของค่า $P(\Delta_{\text{dep}}(z))$ ต่อคอสท์ของเส้นทางมัลติแอสท์	63
5.5 ผลกระทบของค่า $\Delta_{\text{dep}}(z)$ ต่อคอสท์ของเส้นทางมัลติแอสท์	65

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในยุคปัจจุบันนั้น ผู้ใช้บริการภายในระบบเครือข่ายไม่ได้ต้องการเพียงแต่รูปแบบการบริการขั้นพื้นฐานเช่นการโอนถ่ายเพิ่มข้อมูลจากระยะไกลหรือบริการการเข้าถึงระบบเครือข่ายจากระยะไกลเท่านั้น รูปแบบการให้บริการใหม่ๆ บนระบบเครือข่ายมีความต้องการที่จะส่งข้อมูลที่เหมือนกันไปยังผู้ให้บริการจำนวนมาก ด้วยเหตุนี้การสื่อสารข้อมูลในลักษณะมัลติแคสต์ (Multicast) จึงถูกนำมาใช้อย่างแพร่หลายในอินเทอร์เน็ต (Internet) และเนื่องจากการใช้ทรัพยากรต่างๆ ในอินเทอร์เน็ตมีการเจริญเติบโตอย่างรวดเร็ว ดังนั้นการสร้างเส้นทางมัลติแคสต์ที่สามารถใช้งานทรัพยากรที่มีอยู่อย่างจำกัดได้อย่างมีประสิทธิภาพ จึงมีความสำคัญเป็นอย่างยิ่ง

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

เนื่องจากทรัพยากรในระบบเครือข่ายมีอยู่อย่างจำกัด และความต้องการในการสื่อสารแบบมัลติแคสต์มีอัตราที่เพิ่มสูงขึ้นอย่างต่อเนื่อง ดังนั้นวิทยานิพนธ์นี้จึงมุ่งที่จะศึกษาวิจัยและพัฒนาอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ที่สามารถใช้ทรัพยากรบนระบบเครือข่ายได้อย่างมีประสิทธิภาพ โดยมีวัตถุประสงค์หลักของการวิจัยคือศึกษาหลักการพื้นฐานต่างๆ สำหรับการสร้างเส้นทางมัลติแคสต์ทั้งสำหรับกลุ่มสมาชิกแบบไดนามิก (Dynamic) และแบบสแตติก (Static) และศึกษาปัจจัยต่างๆ ของการสื่อสารแบบมัลติแคสต์ที่มีผลต่อการใช้ทรัพยากรในระบบเครือข่ายพัฒนาอัลกอริทึมสำหรับการสร้างเส้นทางมัลติแคสต์ที่สามารถใช้ทรัพยากรในระบบเครือข่ายได้อย่างมีประสิทธิภาพมากขึ้น

1.3 สมมติฐานของการศึกษา

อัลกอริทึมโมดิฟายด์กรีดี้ (Modified Greedy Algorithm) เป็นอัลกอริทึมที่ใช้ในการสร้างเส้นทางมัลติแคสต์แบบไดนามิกภายใต้สภาวะการสงวนทรัพยากรล่วงหน้าซึ่งถูกนำเสนอในงานวิจัย [14] โดยการทำงานของอัลกอริทึมโมดิฟายด์กรีดี้มีสมมติฐานว่าโหนดสมาชิกกลุ่มมัลติแคสต์มีการใช้ทรัพยากรในระบบเครือข่ายภายในช่วงเวลาที่แจ้งไว้ แต่เนื่องจากในสถานการณ์จริงของ

ระบบเครือข่ายมีโอกาสที่โหนดสมาชิกจะใช้ทรัพยากรของระบบเครือข่ายในช่วงเวลาที่สั้นกว่าหรือนานกว่าช่วงเวลาที่แจ้งไว้ส่งผลให้เส้นทางมัลติแคสต์ที่สร้างจากสมมติฐานดังกล่าวมีการใช้ทรัพยากรในระบบเครือข่ายเพิ่มสูงขึ้นดังนั้นในวิทยานิพนธ์นี้จึงจะทำการวิจัยและพัฒนาอัลกอริทึมโมติฟายด์กรีตแบบใหม่ที่มีการสงวนทรัพยากรในช่วงเวลาที่ยาวนานขึ้นสำหรับโหนดสมาชิกเพื่อให้เส้นทางมัลติแคสต์มีความยืดหยุ่นมากขึ้นในการรองรับสถานการณ์ดังกล่าว

อย่างไรก็ตามในทางปฏิบัติ โพรโตคอลการคำนวณเส้นทางทั้งแบบยูนิแคสต์ (Unicast) และมัลติแคสต์จะประกอบด้วยขั้นตอนการทำงานที่สำคัญนอกเหนือจากการคำนวณหาเส้นทางนั้นคือขั้นตอนคือการทำให้โหนด (Node) รู้ข้อมูลต่างๆ ภายในระบบเครือข่ายก่อนที่จะคำนวณตารางเส้นทางเพื่อใช้ในการส่งแพคเกจ (Packet) ข้อมูลไปยังปลายทางโดยผ่านทางเส้นทางที่ดีที่สุดเท่าที่เป็นไปได้ เนื่องจากวิทยานิพนธ์ฉบับนี้มุ่งเน้นการศึกษาวิจัยเฉพาะการคำนวณเส้นทางมัลติแคสต์เท่านั้น ดังนั้นจึงตั้งสมมติฐานว่าโหนดต้นทางของกลุ่มสมาชิกมัลติแคสต์ที่รู้ข้อมูลทั้งหมดของระบบเครือข่ายอย่างสมบูรณ์แล้วจึงใช้ข้อมูลเหล่านี้คำนวณเส้นทางมัลติแคสต์ออกมา

ถึงแม้ว่าการตั้งสมมติฐานดังกล่าวจะทำให้อัลกอริทึมการคำนวณเส้นทางที่ทำการศึกษาวินิจฉัยเข้าข่ายลักษณะการหาเส้นทางในลักษณะรวมศูนย์ซึ่งเมื่อนำไปใช้งานจริงอาจเกิดปัญหาในกรณีที่ระบบเครือข่ายมีจำนวนโหนดมากขึ้น แต่เนื่องจากทางแก้ปัญหามีความซับซ้อนและอยู่นอกเหนือขอบเขตของงานวิทยานิพนธ์ฉบับนี้ดังนั้นจึงจะทำการศึกษาปัญหานี้ต่อไปในอนาคต

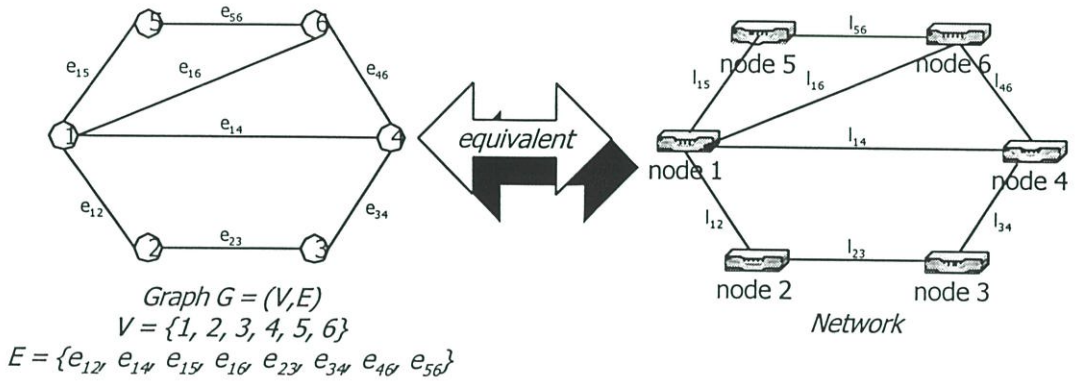
1.4 ทฤษฎีและแนวความคิดที่ใช้ในงานวิจัย

ในงานวิจัยนี้จะอ้างอิงทฤษฎีหลักที่เกี่ยวข้อง 2 ทฤษฎีเพื่อเป็นพื้นฐานของการศึกษาวิจัยเกี่ยวกับการสร้างเส้นทางมัลติแคสต์ โดยประกอบด้วย อัลกอริทึมการคำนวณหาเส้นทางที่สั้นที่สุดและอัลกอริทึมการคำนวณเส้นทางต้นไม้ที่สั้นที่สุด

1.4.1 อัลกอริทึมการคำนวณหาเส้นทางที่สั้นที่สุดระหว่างจุดในกราฟ

เป็นอัลกอริทึมพื้นฐานที่พัฒนามาจากหลักการของเบลล์แมน (Bellman 's Minimality Principle) เพื่อใช้ในการคำนวณเส้นทางที่สั้นที่สุดระหว่างโหนดในระบบเครือข่ายและได้ถูกนำไปประยุกต์ใช้อย่างแพร่หลายในโพรโตคอลการคำนวณเส้นทางทั้งแบบยูนิแคสต์และแบบมัลติแคสต์ โดยอัลกอริทึมดังกล่าวจะมีอยู่หลายประเภท แต่สำหรับงานวิทยานิพนธ์นี้จะใช้อัลกอริทึมของไดจ์สตรา (Dijkstra 's Shortest Path Algorithm) เป็นอัลกอริทึมอ้างอิงเช่นเดียวกับงานวิจัย [14] เนื่องจากมีขั้นตอนการทำงานที่ไม่ซับซ้อนและสะดวกต่อการจำลองระบบเครือข่ายเพื่อทำการทดลอง อัลกอริทึมของไดจ์สตรามีขั้นตอนการคำนวณเส้นทางดังนี้

กำหนดให้กราฟ G เป็นเซตของ V และ E ซึ่งเขียนแทนได้โดย $G = (V, E)$ เมื่อ V เป็นเซตของจุดใดๆ บนกราฟ G เซต E เป็นเซตของการเชื่อมต่อระหว่างจุดใดๆ บนกราฟ G และสมาชิกของเซต E เขียนแทนได้โดย e_{ij} โดยที่ i และ j เป็นสมาชิกของเซต V ซึ่งกราฟ G แทนระบบเครือข่ายรวมถึง โหนดและลิงค์ (Link) ระหว่างโหนดบนระบบเครือข่ายที่สมมูลกันดังแสดงในรูปที่ 1.1



รูปที่ 1.1 กราฟ G ซึ่งเปรียบเทียบกับระบบเครือข่ายจริง

หลังจากนิยามกราฟ G จุดต่างๆ ในเซต V ซึ่งสมมูลกับโหนดในระบบเครือข่ายจะถูกกำหนดค่าฉลากซึ่งแบ่งออกได้เป็น 2 ประเภทคือ ค่าฉลากแบบถาวรและค่าฉลากแบบชั่วคราวและมีค่าจำกัดความของค่าฉลากทั้ง 2 ประเภทคือ

- ค่าฉลากแบบถาวรกำหนดได้โดย ความยาว L_v ของเส้นทางที่สั้นที่สุดจากจุดเริ่มต้นไปยังจุดปลายทางใดๆ ในกราฟ G
- ค่าฉลากแบบชั่วคราวกำหนดได้โดยค่าขอบเขตบนของความยาว L_v ของเส้นทางที่สั้นที่สุดจากจุดเริ่มต้นไปยังจุดปลายทางใดๆ ในกราฟ G

เมื่อเซต P และ T เป็นเซตของจุดใดๆ บนกราฟ G ที่มีค่าฉลากแบบถาวรและชั่วคราวตามลำดับซึ่งการคำนวณเส้นทางจะเป็นไปตามขั้นตอนดังนี้

ขั้นตอนที่ 1 การกำหนดค่าเริ่มต้น

- จุดกำเนิดหรือจุด 1 จะมีค่าฉลากถาวรเท่ากับ $L_1 = 0$
- จุด j ซึ่งเป็นจุดปลายทางอื่นๆ ในกราฟ G ($2, 3, \dots, n$) และกำหนดค่าฉลากชั่วคราว $L_j^* = 1_j$ ให้กับจุดเหล่านี้และให้มีค่าเป็นอนันต์หากไม่มี ส่วนของเส้นตรงที่เชื่อมระหว่างจุด 1 และ j และกำหนดให้เซต $P = \{1\}$ และเซต $T = \{2, 3, \dots, n\}$

ขั้นตอนที่ 2 ทำการเลือกค่าสมาชิกใหม่ของเซต P

- โดยการเลือกจุด k ใดๆ ที่อยู่ในเซต T ที่มีค่าฉลากชั่วคราว L_k^* ที่น้อยที่สุด แล้วทำการเปลี่ยนให้เป็นค่าฉลากถาวร L_k ทำการเลือกค่า k ที่น้อยที่สุดถ้ามีจุด k

หลายจุดและทำการลบจุด k ออกจากเซต T และเพิ่มจุด k เข้าไปในเซต P และทำการตรวจสอบตามเงื่อนไขดังนี้

ถ้า $T = \emptyset$ แล้ว

คำตอบจะเท่ากับ $L_2, L_3, L_4, \dots, L_n$

จบการทำงาน

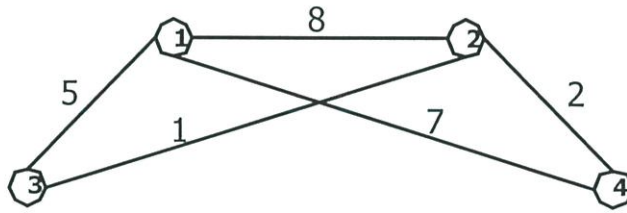
ถ้า $T \neq \emptyset$ แล้ว

ให้กระทำต่อไปใน ขั้นตอนที่ 3

ขั้นตอนที่ 3 ทำการปรับปรุงค่าฉลากชั่วคราว

- สำหรับค่า j ทุกๆ ค่าในเซต T ค่า $L_j^* = \min_k(L_k^*, L_k + l_{kj})$ และกลับไปขั้นตอนที่ 2

จากกราฟดังในรูปที่ 1.2 แสดงตัวอย่างการคำนวณหาเส้นทางที่สั้นที่สุดตามอัลกอริทึมของไดจ์คัสตราได้ดังนี้ เมื่อกำหนดให้จุดเริ่มต้นคือจุด 1 และทำการหาเส้นทางที่สั้นที่สุดจากจุดเริ่มต้นไปยังจุดอื่นๆ ที่เหลือบนกราฟคือจุด 2, 3 และ 4



รูปที่ 1.2 กราฟประกอบตัวอย่างการคำนวณหาเส้นทางที่สั้นที่สุด

จากขั้นตอนที่ 1 ของอัลกอริทึมทำการกำหนดค่าเริ่มต้นก่อนการทำงาน ได้ดังนี้

- ค่าฉลากถาวร $L_1 = 0$
- ค่าฉลากชั่วคราว $L_2^* = 8, L_3^* = 5, L_4^* = 7$
- เซต $P = \{1\}$ และเซต $T = \{2,3,4\}$

ในตารางที่ 1.1 จะแสดงถึงการทำงานของอัลกอริทึมของไดจ์คัสตราตามขั้นตอนที่ 2 และ 3 โดยแต่ละรอบการทำงานจะเป็นการเพิ่มสมาชิกใหม่ของเซต P พร้อมทั้งทำการปรับปรุงค่าฉลากชั่วคราวเพื่อใช้ในรอบการทำงานถัดไป

จากในแต่ละรอบการทำงานจะได้เส้นทางที่สั้นที่สุดจากจุด 1 ไปยังจุดที่เหลือดังนี้คือ

- รอบการทำงานที่ 1 จากจุด 1 ไปยังจุด 3 เส้นทางคือ (1, 3) โดยมีระยะทางเท่ากับ 5 หน่วย

- รอบการทำงานที่ 2 จากจุด 1 ไปยังจุด 2 เส้นทางคือ (1, 3, 2) โดยมีระยะทางเท่ากับ 6 หน่วย
- รอบการทำงานที่ 3 จากจุด 1 ไปยังจุด 4 เส้นทางคือ (1, 4) โดยมีระยะทางเท่ากับ 7 หน่วย

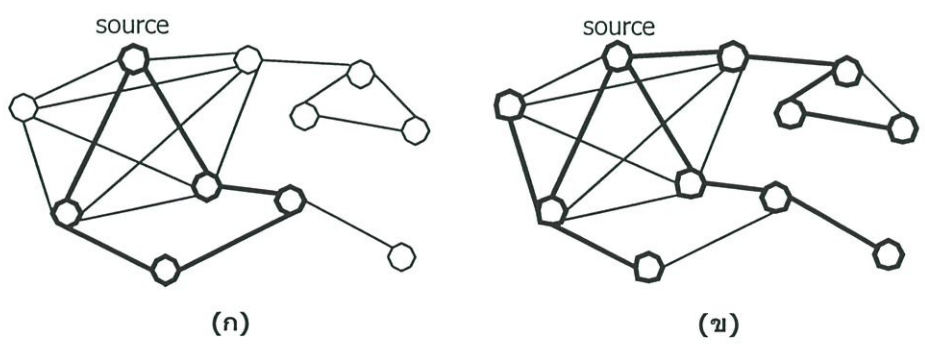
ตารางที่ 1.1 ตารางการทำงานของอัลกอริทึมแบบไดเจกซ์สตรา

รอบที่	L_2	L_3	L_4	k	สมาชิกของเซต P	สมาชิกของเซต T
1	8	5	7	3	{1,3}	{2,4}
2	$\min(8, L_3 + l_{12}) = 6$	-	$\min(7, L_3 + l_{14}) = 7$	2	{1,2,3}	{4}
3	-	-	$\min(7, L_2 + l_{24}) = 7$	4	{1,2,3,4}	ϕ

1.4.2 อัลกอริทึมการคำนวณเส้นทางต้นไม้ที่สั้นที่สุด

การคำนวณหาเส้นทางต้นไม้ที่สั้นที่สุดเป็นอีกทฤษฎีพื้นฐานที่มีความสำคัญเป็นอย่างมากในการสร้างเส้นทางมัลติแคสต์ทั้งกรณีกลุ่มสมาชิกแบบสแตติกและแบบไดนามิก โดยอัลกอริทึมการสร้างเส้นทางต้นไม้ที่สั้นที่สุดมีอยู่หลายประเภท แต่ในวิทยานิพนธ์นี้จะใช้อัลกอริทึมของพริม (Prim's Minimum Spanning Tree Algorithm) เป็นอัลกอริทึมอ้างอิง เนื่องจากเป็นอัลกอริทึมที่ไม่ซับซ้อนมากกล่าวคือมีความซับซ้อนอยู่ในระดับเดียวกับอัลกอริทึมของไดเจกซ์สตรา [3] จึงทำให้สะดวกต่อการจำลองระบบ เครือข่ายเพื่อทำการทดลอง

เส้นทางต้นไม้ที่สั้นที่สุด T ของกราฟ G จะหมายถึงกราฟย่อยซึ่งเชื่อมต่อทุกๆ จุดในกราฟ G เข้าด้วยกันโดยไม่มีการบรรจบหรือมีลูป (Loop) เกิดขึ้นในเส้นทางและมีค่า $\sum l_{ij}$ น้อยที่สุดเมื่อ l_{ij} คือความยาวของส่วนของเส้นตรงที่เชื่อมระหว่างจุด i และ j ใดๆ ภายในกราฟย่อย T ตัวอย่างของการบรรจบและเส้นทางต้นไม้สามารถแสดงได้ดังรูปที่ 1.3



รูปที่ 1.3 ตัวอย่างของเส้นทาง

- (ก) เกิดการบรรจบ
- (ข) เส้นทางต้นไม้ที่ไม่เกิดการบรรจบ

การหาเส้นทางต้นไม้ที่สั้นที่สุดมีประโยชน์อย่างมากในการนำมาประยุกต์ใช้สำหรับการคำนวณเส้นทางมัลติแคสต์ โดยเฉพาะในกรณีของกลุ่มสมาชิกแบบสแตติกและยังสามารถนำมาใช้ประโยชน์ในแง่ของการเป็นตัวเทียบวัดประสิทธิภาพการคำนวณเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิก เพื่อความสะดวกในการอธิบายอัลกอริทึมการหาเส้นทางต้นไม้ที่สั้นที่สุดของพริมจะทำการแทนชื่อจุดต้นทางของเส้นทางว่า จุดเริ่มต้น สำหรับวิธีการคำนวณหาเส้นทางต้นไม้ที่สั้นที่สุดของพริมสามารถกระทำได้ตามขั้นตอนดังนี้

ขั้นตอนที่ 1 กำหนดค่าเริ่มต้น

- กำหนดค่า $i(k) = 1, U = \{1\}, S = \emptyset$
- กำหนดค่าฉลากให้แก่จุดปลายทาง k ที่เหลือทุกจุด ($= 2, 3, \dots, n$) ด้วยค่า $\lambda_k = I_{1k}$ (และให้มีค่าเป็นอนันต์ถ้าไม่มีส่วนของเส้นตรงที่เชื่อมระหว่างจุด 1 และจุด k)

ขั้นตอนที่ 2 การเพิ่มส่วนของเส้นตรงเข้ากับเส้นทางต้นไม้ T

- กำหนดให้ λ_j เป็นค่าที่น้อยที่สุดของ λ_k สำหรับจุด k ใดๆ ที่ไม่ได้เป็นสมาชิกของเซต U
- ให้ j เป็นสมาชิกใหม่ของเซต U และให้ส่วนของเส้นตรง $(i(j), j)$ เป็นสมาชิกใหม่ของเซต S ด้วยและทำการตรวจสอบเงื่อนไขดังนี้
ถ้า $U = V$ แล้ว

$$\text{ค่าเส้นทางต้นไม้ที่สั้นที่สุด} = \sum I_{ij}$$

ส่วนของเส้นตรงที่อยู่ในกราฟย่อย T จะเท่ากับเซต S

จบการทำงาน

ถ้า $U \neq V$ แล้ว

ให้กระทำต่อไปในขั้นตอนที่ 3

ขั้นตอนที่ 3 ทำการปรับปรุงค่าฉลาก

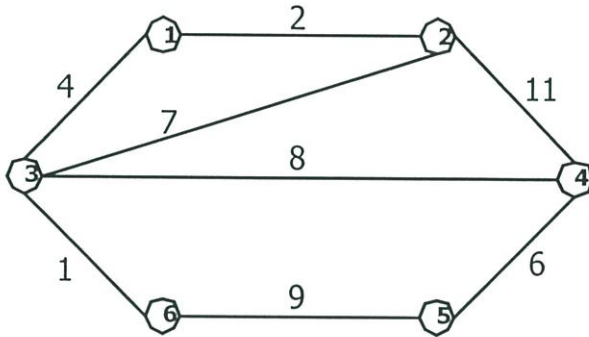
- สำหรับทุกๆ ค่า k ที่ไม่ได้เป็นสมาชิกของเซต U ให้ทำการตรวจสอบเงื่อนไขนี้
ถ้า $I_{jk} < \lambda_k$ แล้ว
กำหนดให้ $\lambda_k = I_{jk}$ และ $i(k) = j$ และกลับไปยังขั้นตอนที่ 2

จากกราฟในรูปที่ 1.4 จะแสดงตัวอย่างของการหาเส้นทางต้นไม้ที่สั้นที่สุดตามอัลกอริทึมของพริม ได้ดังนี้ กำหนดให้จุดเริ่มต้นคือจุด 1

กำหนดค่าเริ่มต้นก่อนการทำงานตามขั้นตอนที่ 1 ของอัลกอริทึมได้ดังนี้

- กำหนดค่า $i(k) = 1$
- กำหนดให้ เซต $U = \{1\}$ และเซต $S = \phi$

ในตารางที่ 1.2 จะแสดงถึงการปรับปรุงค่าผลลตกในรอบการทำงานแต่ละรอบเพื่อประกอบการอธิบายการทำงานของอัลกอริทึม



รูปที่ 1.4 กราฟประกอบการแสดงตัวอย่างการหาเส้นทางต้นไม้ที่สั้นที่สุด

ตารางที่ 1.2 การปรับปรุงค่าผลลตกในแต่ละรอบการทำงานของอัลกอริทึมของพริม

จุด	รอบของการทำงาน				
	1	2	3	4	5
2	$l_{12} = 2$	-	-	-	-
3	$l_{13} = 4$	$l_{13} = 4$	-	-	-
4	∞	$l_{24} = 11$	$l_{34} = 8$	$l_{34} = 8$	-
5	∞	∞	∞	$l_{65} = 9$	$l_{45} = 6$
6	∞	∞	$l_{36} = 1$	-	-

ในแต่ละรอบของการทำงานจะทำการเลือกจุดที่มีค่าผลลตกที่น้อยที่สุดเพื่อใช้ในการสร้างเส้นทางต้นไม้ โดยจุดดังกล่าวจะถูกเพิ่มเข้าไปในเซต U และส่วนของเส้นตรงที่เชื่อมไปยังจุดนั้นจะถูกเพิ่มเข้าไปเป็นส่วนหนึ่งของเส้นทางต้นไม้ซึ่งหมายถึงสมาชิกของเซต S นั้นเอง

- รอบที่ 1 ของการทำงานจุด 2 มีค่าผลลตกน้อยที่สุด เซต $U = \{1, 2\}$ และเซต $S = \{(1,2)\}$
- รอบที่ 2 ของการทำงานจุด 3 มีค่าผลลตกน้อยที่สุด เซต $U = \{1, 2, 3\}$ และเซต $S = \{(1,2), (1,3)\}$
- รอบที่ 3 ของการทำงานจุด 6 มีค่าผลลตกน้อยที่สุด เซต $U = \{1, 2, 3, 6\}$ ดังนั้น เซต $S = \{(1,2), (1,3), (3,6)\}$

- รอบที่ 4 ของการทำงานจุด 4 มีค่าฉลากน้อยที่สุด เซต $U = \{1, 2, 3, 6, 4\}$ และ เซต $S = \{(1,2), (1,3), (3,6), (3,4)\}$
- ในรอบสุดท้ายของการทำงานเหลือเพียงจุด 5 จุดเดียวและเซตของ S ซึ่งหมายถึงเซตของส่วนของเส้นตรงบนเส้นทางต้นไม้ที่สั้นที่สุดจะเท่ากับ $\{(1,2), (1,3), (3,6), (3,4), (4,5)\}$

และขนาดของเส้นทางต้นไม้ที่เล็กที่สุดจะเท่ากับระยะทางของส่วนของเส้นตรงทั้งหมดในเซต S รวมกันซึ่งเท่ากับ 21 หน่วย

1.5 ขอบเขตการวิจัย

ในงานวิจัยนี้จะมุ่งเน้นการวิจัยและพัฒนาการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกภายใต้สภาวะแวดล้อมการสงวนทรัพยากรล่วงหน้า โดยที่กลุ่มสมาชิกแบบไดนามิกหมายถึงการที่โหนดสมาชิกสามารถที่จะเข้าหรือออกจากกลุ่มมัลติแคสต์ได้อย่างอิสระ และสภาวะแวดล้อมแบบการสงวนทรัพยากรล่วงหน้าหมายถึงการที่โหนดในระบบเครือข่ายที่ต้องการเข้าเป็นสมาชิกในระบบมัลติแคสต์จะต้องแจ้งเวลาที่เข้าร่วมและออกจากกลุ่มมัลติแคสต์ให้โหนดต้นทางของกลุ่มมัลติแคสต์ทราบล่วงหน้าเพื่อประโยชน์ในการจัดสรรทรัพยากรของระบบเครือข่าย

1.6 ขั้นตอนการศึกษา

ในบทที่ 2 จะทำการศึกษาอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ในรูปแบบต่างๆ ทั้งสำหรับกลุ่มสมาชิกแบบสแตติกและแบบไดนามิกเพื่อเป็นรากฐานในการศึกษาและทำการวิจัยเพื่อพัฒนาอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ต่อไป บทที่ 3 จะกล่าวถึงรายละเอียด รูปแบบการให้บริการแบบสงวนทรัพยากรล่วงหน้ารวมทั้งแสดงให้เห็นถึงประโยชน์ของการรวมแนวคิดการสงวนทรัพยากรล่วงหน้าเข้ากับการหาเส้นทางมัลติแคสต์แบบไดนามิก โดยการทำการศึกษาเปรียบเทียบประสิทธิภาพกับอัลกอริทึมการหาเส้นทางมัลติแคสต์แบบดั้งเดิมภายใต้สภาวะแวดล้อมการคำนวณเส้นทางแบบรับประกันคุณภาพ และในบทที่ 4 จะทำการวิเคราะห์และพัฒนาอัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบไดนามิกภายใต้สภาพแวดล้อมการสงวนทรัพยากรล่วงหน้าให้มีประสิทธิภาพดียิ่งขึ้นในการจัดสรรทรัพยากรให้แก่โหนดสมาชิกกลุ่มมัลติแคสต์ และในบทที่ 5 จะเป็นการทดลองเพื่อทดสอบและวัดประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบใหม่ที่ได้นำเสนอในวิทยานิพนธ์นี้และในบทที่ 6 จะเป็นการสรุปผลการทดลอง

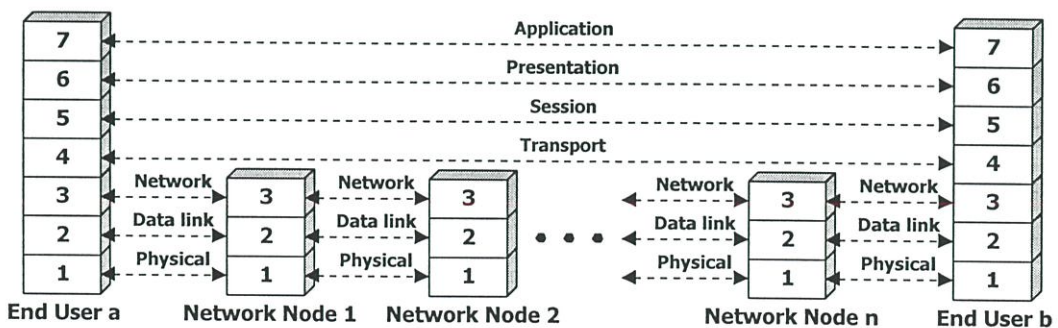
บทที่ 2

การสร้างเส้นทางมัลติแคสต์

ในบทนี้จะเริ่มด้วยการกล่าวถึงกลไกซึ่งเป็นมาตรฐานในการส่งข้อมูลผ่านระบบเครือข่ายสื่อสาร จากนั้นจะกล่าวถึงการส่งข้อมูลในรูปแบบมัลติแคสต์ในระบบเครือข่ายสื่อสาร ลำดับต่อมาจะเป็นการวิเคราะห์ปัญหาของการสร้างเส้นทางต้นไม้เพื่อใช้ในการส่งข้อมูลมัลติแคสต์ แนวทางการแก้ปัญหการสร้างเส้นทางต้นไม้เพื่อใช้ในการส่งข้อมูลมัลติแคสต์รวมทั้งอธิบายถึงอัลกอริทึมสำหรับการสร้างเส้นทางต้นไม้แบบต่างๆ เพื่อใช้ในการอ้างอิงกับอัลกอริทึมการสร้างเส้นทางต้นไม้แบบใหม่ที่ได้นำเสนอในวิทยานิพนธ์นี้

2.1 การส่งข้อมูลในระบบเครือข่ายสื่อสาร

อ้างอิงถึงแบบจำลองของสถาปัตยกรรมการเชื่อมต่อในระบบเปิด (Open System Interconnection) ซึ่งแบ่งกลไกการทำงานเพื่อการสื่อสารข้อมูลระหว่างผู้ใช้บริการใดๆ ที่เชื่อมต่อกับระบบเครือข่ายสื่อสารออกเป็น 7 ระดับชั้น การเชื่อมต่อระหว่างโหนดในระบบเครือข่ายเพื่อสร้างเส้นทางในการให้การส่งข้อมูลระหว่างผู้ใช้บริการใดๆ ผ่านระบบเครือข่ายสื่อสารได้ถูกจัดอยู่ในฟังก์ชันการทำงานของระดับชั้นที่ 3 ในสถาปัตยกรรมซึ่งถูกนิยามว่าเป็น “ระดับชั้นเครือข่าย” (Network Layer) และด้วยข้อดีของการแบ่งฟังก์ชันการทำงานออกเป็นระดับชั้นต่างๆ ทำให้เป็นการลดภาระของผู้ให้บริการระบบเครือข่ายที่อยู่ในระดับชั้นอื่นๆ ที่สูงกว่าในการสร้างเส้นทางในระบบเครือข่ายสื่อสารจากโหนดต้นทางผ่านโหนดต่างๆ ไปยังผู้ใช้บริการปลายทาง ในรูปที่ 2.1 แสดงถึงรูปแบบการเชื่อมต่อจากคอมพิวเตอร์ของผู้ใช้บริการต้นทางผ่านโหนดต่างๆ ในระบบเครือข่ายสื่อสารไปยังคอมพิวเตอร์ของผู้ใช้บริการปลายทาง



รูปที่ 2.1 การติดต่อสื่อสารระหว่างผู้ใช้บริการผ่านระบบเครือข่ายสื่อสาร

ภายในระดับชั้นเครือข่าย คอมพิวเตอร์ของผู้ใช้บริการทั้งต้นทางและปลายทางที่เชื่อมต่อกับโหนดในระบบเครือข่ายจะถูกกำหนดแอดเดรส (Address) ให้เพื่อใช้ในการระบุผู้ใช้บริการในระบบเครือข่ายและจากรูปที่ 2.1 จะพบว่า โหนดในระบบเครือข่ายจะมีหน้าที่ความรับผิดชอบอยู่เพียงระดับชั้นเครือข่ายเท่านั้น และสำหรับการให้บริการในระดับชั้นที่สูงกว่าระดับชั้นเครือข่ายขึ้นไปนั้นจะถูกรับผิดชอบโดยคอมพิวเตอร์ของผู้บริการต้นทางและปลายทาง

2.2 การให้บริการมัลติแคสต์ในระบบเครือข่าย

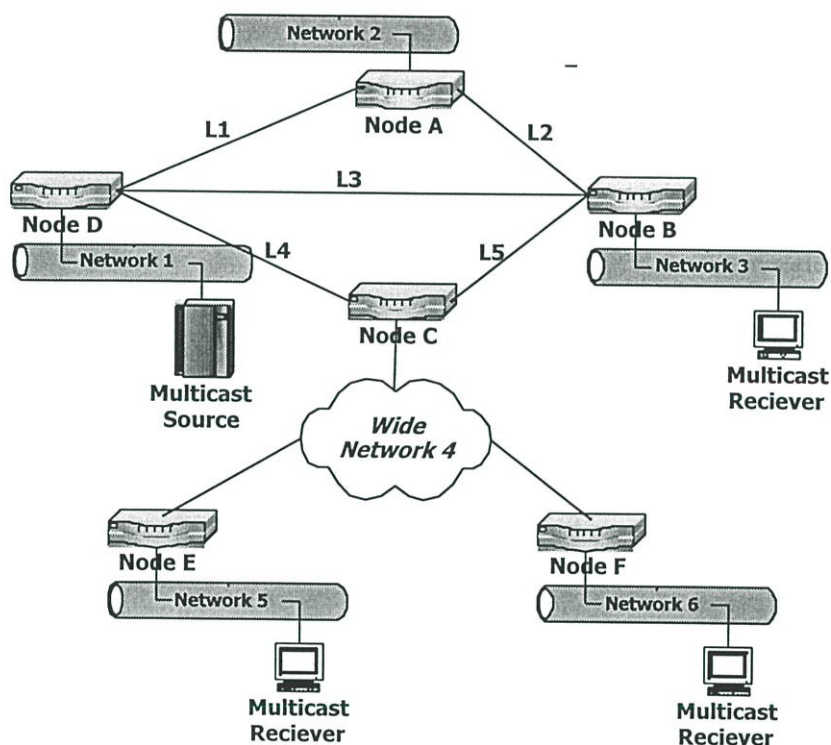
บริการมัลติแคสต์คือการส่งข้อมูลข่าวสารที่เหมือนกันจากผู้ใช้บริการซึ่งเป็นต้นทางของการส่งข้อมูลไปยังผู้ใช้บริการอื่นๆ ซึ่งเป็นผู้รับข่าวสารปลายทางจำนวนหลายคนในเวลาเดียวกัน ในทางปฏิบัติ การให้บริการมัลติแคสต์สามารถให้ทำได้ทั้งในระดับชั้นที่ 2 และระดับชั้นที่ 3 ในกรณีของการให้บริการมัลติแคสต์บนระดับชั้นที่ 2 ของสถาปัตยกรรมการเชื่อมต่อแบบระบบเปิด ซึ่งระบบเครือข่ายสื่อสารแบบท้องถิ่นหรือแลน (Local Area Network, LAN) เนื่องจากข้อดีของธรรมชาติของสื่อกลางที่เป็นลักษณะบรอดคาสต์ (Broadcast) จึงทำให้การให้บริการมัลติแคสต์เป็นไปได้โดยสะดวก

ในยุคปัจจุบันซึ่งการเจริญเติบโตของเทคโนโลยีและอุตสาหกรรมด้านคอมพิวเตอร์และระบบเครือข่ายสื่อสารเป็นไปอย่างรวดเร็วเป็นผลต่อเนื้อให้แอปพลิเคชัน (Application) เพื่อใช้ในการติดต่อสื่อสารมีความหลากหลายและทันสมัยมากยิ่งขึ้นดังนั้นความต้องการบริการมัลติแคสต์บนระบบเครือข่ายสื่อสารจึงมีมากขึ้นตามไปด้วย แต่ในระบบเครือข่ายสื่อสารซึ่งมีขนาดใหญ่และครอบคลุมพื้นที่ที่กว้างรวมถึงจำนวนผู้ใช้บริการจำนวนมากเช่น เครือข่ายอินเทอร์เน็ตหรือแวน (Wide Area Network, WAN) ซึ่งไม่มีธรรมชาติของสื่อกลางในแบบบรอดคาสต์ในลักษณะเดียวกับระบบเครือข่ายสื่อสารในระดับท้องถิ่นทำให้การให้บริการมัลติแคสต์บนระบบเครือข่ายดังกล่าวจึงจำเป็นที่จะต้องเกี่ยวข้องกับการเลือกเส้นทางรวมถึงการสวิตช์ (Switch) ข้อมูลซึ่งเป็นฟังก์ชันการทำงานที่อยู่ในระดับชั้นที่ 3 ทำให้การให้บริการมัลติแคสต์แบบนี้มีความซับซ้อนมากยิ่งขึ้นกว่าในระดับชั้นที่ 2 แต่ด้วยข้อดีของระบบแวนที่ครอบคลุมผู้ใช้บริการในวงกว้างรวมถึงการรองรับแอปพลิเคชันด้านการสื่อสารที่หลากหลายกว่า ดังนั้นในวิทยานิพนธ์นี้จึงทำการศึกษาการสร้างเส้นทางมัลติแคสต์เพื่อให้บริการแก่ผู้ใช้บริการในระบบเครือข่ายแบบแวนเป็นหลัก

2.2.1 การสร้างเส้นทางต้นไม้มือเพื่อการให้บริการมัลติแคสต์

เนื่องจากการสร้างเส้นทางมัลติแคสต์เกี่ยวข้องกับโหนดในระบบเครือข่ายเป็นหลัก ดังนั้นเพื่อความสะดวกในการอธิบายอัลกอริทึมต่างๆ ในภายหลังจึงเรียกโหนดในระบบเครือข่ายที่เชื่อมต่อกับคอมพิวเตอร์ของผู้บริการที่เป็นต้นทางการส่งข้อมูลมัลติแคสต์ว่า “โหนดต้นทาง” และ

เรียก โหนดที่ต่อกับคอมพิวเตอร์ของผู้ให้บริการซึ่งเป็นผู้รับข้อมูลมัลติแคสต์จากโหนดต้นทางว่า “โหนดปลายทาง” และรวมเรียกโหนดปลายทางทั้งหมดที่รับข้อมูลมัลติแคสต์ว่า “โหนดสมาชิก” โดยที่กลุ่มของโหนดทั้งหมดที่รับส่งข้อมูลมัลติแคสต์จะเรียกว่า “กลุ่มมัลติแคสต์” ในการส่งข้อมูลจากโหนดต้นทางไปยังโหนดปลายทางนั้นสามารถทำได้หลายวิธีเช่น การบรอดแคสต์ข้อมูลในระบบเครือข่ายโดยไม่คำนึงถึงจำนวนของโหนดสมาชิกของกลุ่มมัลติแคสต์ หรือใช้การส่งลักษณะยูนิแคสต์จากโหนดต้นทางไปยังแต่ละโหนดสมาชิกในระบบเครือข่าย หรือทำการสร้างเส้นทางต้นไม้ซึ่งโหนดต้นทางจะทำการส่งข้อมูลไปเพียง 1 แพกเกตต่อ 1 ลิงค์ระหว่างโหนดในระบบเครือข่าย และจะทำการสำเนาข้อมูลที่โหนดซึ่งเป็นโหนดสาขาในเส้นทางมัลติแคสต์เท่านั้น



รูปที่ 2.2 ตัวอย่างการส่งข้อมูลมัลติแคสต์ในระบบเครือข่าย

พิจารณาระบบเครือข่ายตัวอย่างดังในรูปที่ 2.2 เมื่อโหนด D เป็นโหนดต้นทางและโหนดสมาชิกที่เหลือคือโหนด E, B และ F เมื่อให้จำนวนฮอป (Hop Count) แทนปริมาณการใช้งานทรัพยากรในเครือข่าย ตารางที่ 2.1 แสดงค่าจำนวนฮอปสำหรับแต่ละวิธีการส่งข้อมูลมัลติแคสต์จากโหนดต้นทางไปยังโหนดสมาชิกที่เหลือทั้งหมด

จากตารางที่ 2.1 พบว่า วิธีการสร้างเส้นทางต้นไม้เพื่อการส่งข้อมูลมัลติแคสต์จะให้จำนวนฮอปรวมที่ต่ำที่สุดซึ่งแสดงให้เห็นถึงมีจำนวนแพกเกตของข้อมูลที่ถูกส่งผ่านระบบเครือข่ายสื่อสารน้อยที่สุดทำให้มีการใช้ปริมาณทรัพยากรในระบบเครือข่ายน้อยลงไปด้วย ด้วยเหตุนี้จึงทำ

ให้โปรโตคอลทั้งหมดที่ให้บริการมัลติแคสต์ในระบบเครือข่ายสื่อสารจึงใช้การเส้นทางต้นไม้เพื่อการลำเลียงข้อมูลจากโหนดต้นทางไปยังโหนดสมาชิกทุกโหนด

ตารางที่ 2.1 แสดงเปรียบเทียบจำนวนฮอปของการส่งข้อมูลมัลติแคสต์

Link	Broadcast				Multiple Unicast			Tree Shaped Route
	S to N2	S to N3	S to N5	S to N6	S to N3	S to N5	S to N6	
N1	1	1	1	1	1	1	1	1
N2								
N3		1			1			1
N4			1	1		1	1	2
N5			1			1		1
N6				1			1	1
L1	1							
L2								
L3		1			1			1
L4			1	1		1	1	1
L5								
Total		13			11			8

2.2.2 ปัญหาการสร้างเส้นทางต้นไม้เพื่อการส่งข้อมูลมัลติแคสต์

จากหัวข้อที่แล้วจะพบว่า การสร้างเส้นทางมัลติแคสต์ที่ดีจะทำให้การใช้ทรัพยากรในระบบเครือข่ายเป็นไปอย่างมีประสิทธิภาพมากขึ้น แม้ว่าในการสร้างเส้นทางต้นไม้สำหรับการให้บริการมัลติแคสต์จะคล้ายคลึงกับปัญหาการสร้างเส้นทางต้นไม้ที่เล็กที่สุดซึ่งเป็นการหาเส้นทางต้นไม้ที่ครอบคลุมทุกโหนดในระบบเครือข่าย ดังที่กล่าวไว้ในบทที่ 1 แต่ในทางปฏิบัติการสร้างเส้นทางต้นไม้ให้ครอบคลุมเฉพาะบางโหนดในระบบเครือข่ายมีความซับซ้อนในการแก้ปัญหาสูงกว่ามาก เนื่องจากในการสร้างเส้นทางต้นไม้ที่ครอบคลุมเฉพาะบางโหนดในระบบเครือข่าย นอกจากความต้องการที่เส้นทางดังกล่าวจะสามารถครอบคลุมโหนดสมาชิกของกลุ่มมัลติแคสต์ทุกโหนดในระบบเครือข่ายแล้วยังมีความต้องการที่จะลดจำนวนโหนดอื่นๆ ที่ไม่ใช่โหนดสมาชิกแต่อยู่ในเส้นทางให้มากที่สุดเท่าที่เป็นไปได้ซึ่งโหนดที่ไม่ได้เป็นโหนดสมาชิกของกลุ่มมัลติแคสต์แต่เป็นส่วนหนึ่งของเส้นทางมัลติแคสต์จะถูกนิยามว่าเป็น โหนดสไตน์เนอร์ (Steiner Node) ดังนั้นปัญหาการสร้างเส้นทางต้นไม้เพื่อครอบคลุมเฉพาะโหนดบางส่วนและมีความพยายามให้มีโหนดสไตน์เนอร์ที่อยู่ในเส้นทางที่น้อยที่สุดจึงถูกเรียกว่า ปัญหาเส้นทางต้นไม้สไตน์เนอร์ (Steiner Tree Problem) และปัญหานี้ได้ถูกพิสูจน์ว่าเป็นปัญหาในลักษณะ เอ็นพีสัมบูรณ์ (NP complete) [3] ซึ่งทำให้การหาคำตอบที่ดีที่สุดไม่สามารถกระทำได้ในช่วงเวลาโพลีโนเมียล (Polynomial)

ในทางปฏิบัติ การแก้ปัญหาในลักษณะเอ็นพีสัมบูรณ์สามารถกระทำได้หลายวิธีแต่สำหรับกรณีของงานวิจัยที่เกี่ยวข้องกับการหาเส้นทางมัลติแคสต์รวมถึงโปรโตคอลที่มีการใช้งานเพื่อให้บริการมัลติแคสต์แก่ผู้ใช้บริการในระบบเครือข่ายมีวิธีการในการแก้ปัญหาดังกล่าวอยู่ 2 วิธี วิธีแรก

คือการใช้อัลกอริทึมแบบฮิวริสติก (Heuristic Algorithm) ซึ่งเป็นอัลกอริทึมที่ใช้เพื่อการประมาณค่าคำตอบของปัญหาโดยคำตอบที่ได้จากอัลกอริทึมฮิวริสติกจะมีค่าไม่แตกต่างไปจากคำตอบที่ดีที่สุดมากอย่างมีนัยสำคัญหรือมีค่าที่ใกล้เคียงมากสำหรับบางกรณี และวิธีที่ 2 คือการใช้อัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบเนฟ (Naive Algorithm) ซึ่งได้ถูกนำเสนอในงานวิจัย [6] และจะได้ทำการกล่าวถึงรายละเอียดในหัวข้อที่ 2.4.1 ต่อไป

ในวิทยานิพนธ์นี้ได้นำเสนอการแก้ปัญหาการสร้างเส้นทางมัลติแคสต์ด้วยอัลกอริทึมในลักษณะฮิวริสติกเช่นเดียวกับงานวิจัยส่วนมากที่เกี่ยวข้องกับปัญหาการสร้างเส้นทางมัลติแคสต์เช่นงานวิจัย [5], [7], [8], [14] และ [15] เนื่องจากแม้ว่าจะมีความซับซ้อนของอัลกอริทึมมากกว่าแต่สามารถสร้างเส้นทางมัลติแคสต์ที่มีการใช้ทรัพยากรของระบบเครือข่ายในปริมาณที่น้อยกว่ามากเมื่อเทียบกับอัลกอริทึมแบบเนฟ ดังจะเห็นได้จากการทดลองในช่วงท้ายของวิทยานิพนธ์นี้

2.2.3 ประเภทของปัญหาการสร้างเส้นทางมัลติแคสต์

กรณีปัญหาการหาเส้นทางมัลติแคสต์ เมื่อนำพฤติกรรมของโหนดสมาชิกมาพิจารณาจะพบว่าสามารถแบ่งออกได้เป็น 2 ประเภทคือ ปัญหาการสร้างเส้นทางมัลติแคสต์แบบสแตติกซึ่งหมายถึงโหนดสมาชิกของกลุ่มมัลติแคสต์ไม่มีการเปลี่ยนแปลงตลอดเวลาที่โหนดต้นทางทำการส่งข้อมูลให้กับโหนดสมาชิก และประเภทไดนามิกซึ่งเป็นกรณีที่โหนดสมาชิกมีการเปลี่ยนแปลงตลอดเวลาโดยสามารถเข้าร่วมหรือแยกออกจากกลุ่มมัลติแคสต์ได้ตลอดเวลาการส่งข้อมูลจากโหนดต้นทางไปยังโหนดสมาชิกทุกโหนด

โดยงานวิทยานิพนธ์นี้จะทำการศึกษาวิจัยและพัฒนาการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกเป็นหลัก เนื่องจากพฤติกรรมที่โหนดสมาชิกสามารถเข้าและออกจากกลุ่มมัลติแคสต์ได้อย่างอิสระนั้นสอดคล้องกับลักษณะการให้บริการจริงของมัลติแคสต์ และในการสร้างเส้นทางมัลติแคสต์จะไม่อนุญาตให้ทำการเรียบเรียงเส้นทางใหม่ทุกครั้งที่มีการเปลี่ยนแปลงของโหนดสมาชิกกลุ่มมัลติแคสต์ เนื่องจากเมื่อมีการอนุญาตให้ทำการเรียบเรียงเส้นทางใหม่นั้นจะเป็นการสูญเสียทรัพยากรต่างๆ เพิ่มเติมเช่น แบนด์วิธ (Bandwidth) หน่วยความจำบัฟเฟอร์ (Buffer) และความสามารถในการประมวลผลเพื่อเรียบเรียงแพ็คเกจข้อมูลใหม่ที่โหนดหรือที่อุปกรณ์ปลายทางเป็นต้น รวมถึงอาจเป็นการเพิ่มค่าใช้จ่ายและความซับซ้อนให้แก่ตัวอุปกรณ์ด้วยเช่นกัน นอกจากนี้ยังต้องการกลไกที่จะทำให้การส่งแพ็คเกจข้อมูลในช่วงระหว่างที่ทำการเรียบเรียงเส้นทางใหม่ไม่มีความราบรื่นไม่เกิดการสะดุดหรือขาดตอนอีกด้วย ด้วยเหตุนี้จึงพบว่างานวิจัยส่วนใหญ่ที่เกี่ยวกับการสร้างเส้นทางมัลติแคสต์แบบไดนามิกจึงไม่กล่าวถึงการเรียบเรียงเส้นทางใหม่ระหว่างที่มีการเปลี่ยนแปลงโหนดสมาชิกของกลุ่มมัลติแคสต์

ในหัวข้อต่อไปของบทนี้จะทำการกล่าวถึงเงื่อนไขของการสร้างเส้นทางมัลติแคสต์รวมถึงทำการวิเคราะห์อัลกอริทึมที่ใช้ในการสร้างเส้นทางมัลติแคสต์ที่ปรากฏอยู่ในงานวิจัยหลายๆ ชิ้น

โดยเงื่อนไขและอัลกอริทึมที่ใช้ในการสร้างเส้นทางมัลติแคสต์เหล่านี้จะถูกอธิบายโดยอ้างอิงตามทฤษฎีกราฟพื้นฐานดังที่กล่าวไปแล้วในบทที่ 1

2.3 การสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบสแตติก

ถึงแม้ว่าในวิทยานิพนธ์ฉบับนี้จะมุ่งเน้นการศึกษาไปที่การสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกเป็นหลักก็ตาม แต่ในบางส่วนของทฤษฎีประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มไดนามิกจะใช้การสร้างเส้นทางสำหรับกลุ่มแบบสแตติกมาเป็นมาตรฐานในการอ้างอิงเช่นเดียวกับงานวิจัยอื่นๆ เช่นงานวิจัย [5], [6] และ [7] ที่เกี่ยวข้องกับอัลกอริทึมการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิก

ในกลุ่มมัลติแคสต์แบบไดนามิก โหนดสมาชิกของกลุ่มจะมีการเปลี่ยนแปลงตลอดเวลาเมื่อทำการพิจารณาในช่วงเวลาหนึ่งๆ ของกลุ่มมัลติแคสต์แบบไดนามิกซึ่งเป็นช่วงเวลาที่ไม่มีการเปลี่ยนแปลงของกลุ่มสมาชิกมัลติแคสต์ เมื่อแทนกลุ่มมัลติแคสต์ในขณะนั้นด้วยเซต S^* และในช่วงเวลานั้นทำการใช้อัลกอริทึมสำหรับกลุ่มแบบสแตติกกับกลุ่มสมาชิกมัลติแคสต์ S^* อัลกอริทึมสำหรับกลุ่มสมาชิกแบบสแตติกที่ดีจะใช้ค่าคอสต์ (Cost) เพื่อการสร้างเส้นทางมัลติแคสต์น้อยกว่าอัลกอริทึมการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มแบบไดนามิก เนื่องจากเมื่อไม่อนุญาตให้ทำการเรียบเรียงเส้นทางใหม่แล้ว อัลกอริทึมสำหรับการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มไดนามิกจะทำการเพิ่มโหนดสมาชิกเข้ามายังกลุ่มหรือทำการย้ายสมาชิกออกจากกลุ่มมัลติแคสต์โดยเปลี่ยนแปลงเส้นทางมัลติแคสต์จากเส้นทางที่มีอยู่เดิมเป็นหลัก ดังนั้นจึงสามารถใช้อัลกอริทึมสำหรับกลุ่มสมาชิกแบบสแตติกที่มีประสิทธิภาพดีมาเป็นมาตรฐานการอ้างอิงประสิทธิภาพของอัลกอริทึมสำหรับกลุ่มสมาชิกแบบไดนามิกได้

สำหรับอัลกอริทึมการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบสแตติกที่นำมาใช้เป็นอัลกอริทึมอ้างอิงในวิทยานิพนธ์นี้จะเป็นอัลกอริทึมเคเอ็มบี (KMB algorithm) ซึ่งเคยถูกนำเสนอในงานวิจัย [4] และเป็นอัลกอริทึมที่ได้รับการยอมรับว่าให้ประสิทธิภาพการหาเส้นทางที่ดีเมื่อเทียบกับค่าคอสต์ที่ดีที่สุด ดังนั้นจึงถูกนำมาเป็นมาตรฐานอ้างอิงในงานวิจัยอื่นๆ เพื่อเป็นการทดสอบประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มแบบไดนามิก อีกทั้งยังมีกลไกการทำงานที่ไม่ซับซ้อนเพื่อความสะดวกในการทำซิมูเลชัน (Simulation) อีกด้วย เงื่อนไขของการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบสแตติกเป็นดังนี้

กำหนดให้

กราฟ $G = (V, E)$ เมื่อ V เป็นเซตของโหนดใดๆ บนกราฟ G และ E เป็นเซตของลิงค์ ซึ่งมีลักษณะสมมาตรที่เชื่อมระหว่างโหนดใดๆ บนกราฟ G โดยสมาชิกของเซต E

สามารถเขียนแทนได้ด้วย c_{ij} เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V

เซต C ซึ่งมีสมาชิกเป็นคอสต์ฟังก์ชันของสมาชิกของเซต E โดย $C \subset \mathbb{R}^+$ (เซตของจำนวนจริงบวกรวมถึงค่า 0) และสมาชิกของเซต C สามารถเขียนแทนด้วย $c(e_{ij})$ เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V และเนื่องจากฟังก์ชันในระบบเครือข่ายมีลักษณะที่สมมาตร ดังนั้นจึงทำให้ $c(e_{ij}) = c(e_{ji})$

เซตของโหนดซึ่งเป็นสมาชิกของกลุ่มมัลติแอสท์เท่ากับ D โดยที่ $D \subseteq V$

ในวิทยานิพนธ์ฉบับนี้จะใช้ค่าระยะทางบนพิคตการ์ที่เขียน (Cartesian) ของกราฟมาเป็นค่าคอสต์เช่นเดียวกับงานวิจัยอื่นๆ ที่เกี่ยวข้องกับการสร้างเส้นทางมัลติแอสท์ นอกจากนี้ค่าระยะทางดังกล่าวยังสะท้อนให้เห็นถึงค่าหน่วยเวลาของการส่งหน่วยข้อมูลไปบนเส้นทางนั้นอีกด้วย

จากเงื่อนไขข้างต้น สร้างเส้นทางมัลติแอสท์ T คือกราฟ $T = (V_T, E_T)$ ซึ่งค่าคอสต์รวมของเส้นทางต้นไม้ T ซึ่งมีค่าเท่ากับ $\sum_{e \in E_T} c(e)$ มีค่าน้อยที่สุด โดยที่กราฟ T เป็นกราฟย่อยของกราฟ G และเซต V_T เป็นเซตของโหนดต่างๆ บนกราฟ T โดยอยู่ภายใต้เงื่อนไข $D \subseteq V_T$ และ $V_T \subseteq V$ เซต E_T ซึ่งเป็นเซตของฟังก์ชันที่เชื่อมระหว่างจุดบนกราฟ T โดยที่ $E_T \subseteq E$ และเรียกกราฟ T นี้ว่าเส้นทางต้นไม้สโตนเนอร์สำหรับกลุ่มสมาชิก D บนกราฟ G เส้นทางต้นไม้สโตนเนอร์นี้ก็คือเส้นทางมัลติแอสท์ของกลุ่มมัลติแอสท์ D ในระบบเครือข่ายที่สมมูลกับกราฟ G นั่นเอง

และจากเงื่อนไขของการสร้างเส้นทางมัลติแอสท์ดังกล่าว อัลกอริทึมเคเอ็มบีสำหรับการสร้างเส้นทางมัลติแอสท์สำหรับกลุ่มแบบสแตติกมีขั้นตอนการทำงานดังนี้

- ขั้นตอนที่ 1 จากกราฟ $G = (V, E)$ และกลุ่มมัลติแอสท์ D สร้าง กราฟสมบูรณ์ $G^* = (D, E^*)$ บนกราฟ G โดยสมาชิกของเซต E^* คือคอสต์น้อยที่สุดบนฟังก์ชันที่เชื่อมต่อระหว่างโหนดใดๆ ในเซต D บนกราฟ G ซึ่งได้จากการคำนวณด้วย อัลกอริทึมของไดจค์สตรา
- ขั้นตอนที่ 2 ทำการคำนวณค่าเส้นทางต้นไม้ที่สั้นที่สุด T^* ที่อยู่บนกราฟ G^*
- ขั้นตอนที่ 3 ทำการสร้างกราฟ G^{**} บนกราฟ G จากการแปลงการฟังก์ชันในเส้นทางต้นไม้ T^* ให้กลับไปเป็นฟังก์ชันที่สมมูลกันซึ่งอยู่ในเซต E
- ขั้นตอนที่ 4 ทำการสร้างเส้นทางต้นไม้ที่สั้นที่สุด T^{**} บนกราฟ G^{**}
- ขั้นตอนที่ 5 จากเส้นทางต้นไม้ T^{**} ให้ทำการตัดทิ้งโหนดและฟังก์ชันที่ไม่จำเป็นออก หลังจากนั้นจะได้เส้นทางมัลติแอสท์ T สำหรับกลุ่มสมาชิกแบบสแตติก D

การหาเส้นทางมัลติแอสต์สำหรับกลุ่มสมาชิกแบบไดนามิกที่ได้พัฒนาขึ้นมาใหม่ในวิทยานิพนธ์ฉบับนี้ ซึ่งเงื่อนไขของการสร้างเส้นทางมัลติแอสต์แบบกลุ่มไดนามิกสามารถกำหนดได้ดังนี้

กำหนดให้

กราฟ $G = (V, E)$ เมื่อ V เป็นเซตของโหนดใดๆ บนกราฟ G และ E เป็นเซตของลิงค์ ซึ่งมีลักษณะสมมาตรที่เชื่อมระหว่างโหนดใดๆ บนกราฟ G ซึ่งสมาชิกของเซต E สามารถเขียนแทนได้ด้วย e_{ij} เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V

เซต C ซึ่งมีสมาชิกเป็นคอสต์ฟังก์ชันของสมาชิกของเซต E ซึ่ง $C \subset R^+$ (เซตของจำนวนจริงบวกรวมถึงค่า 0 ด้วย) และสมาชิกของเซต C สามารถเขียนแทนได้ด้วย $c(e_{ij})$ เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V และเนื่องจากลิงค์ในระบบเครือข่ายมีลักษณะที่สมมาตร ดังนั้นจึงทำให้ $c(e_{ij}) = c(e_{ji})$

เซตของลำดับเหตุการณ์ที่จะมีโหนดเข้าร่วมหรือออกจากการเป็นสมาชิกของกลุ่มมัลติแอสต์ในแต่ละช่วงเวลาเท่ากับ เซต $R = \{r_1, r_2, r_3, \dots, r_k\}$ เมื่อ r_i ใดๆ เป็นโหนด z_i ใดๆ ในระบบเครือข่ายและเป็นสมาชิกของเซต V โดยที่ในแต่ละเหตุการณ์จะมีโหนดเข้าหรือออกจากกลุ่มมัลติแอสต์ได้เพียงครั้งละ 1 โหนดเท่านั้น

เซตของโหนดสมาชิกของกลุ่มมัลติแอสต์ในแต่ละช่วงเวลาเท่ากับเซต $S = \{s_1, s_2, s_3, \dots, s_k\}$

จากข้อกำหนดข้างต้นทำการสร้างเส้นทางมัลติแอสต์ T_i ด้วยอัลกอริทึมการสร้างเส้นทางมัลติแอสต์หลังจากเกิดเหตุการณ์ r_i ขึ้นซึ่ง T_i คือเส้นทางมัลติแอสต์ที่มีค่าคอสต์รวมน้อยที่สุดโดย T_i เป็นกราฟย่อยของกราฟ G และครอบคลุมทุกโหนดในกลุ่มมัลติแอสต์ s_i

2.4.1 อัลกอริทึมการหาเส้นทางแบบเนฟ (Naïve algorithm)

เป็นอัลกอริทึมที่ถูกเสนอขึ้นมาในงานวิจัย [6] เป็นอัลกอริทึมที่มีความซับซ้อนน้อยที่สุดในการสร้างเส้นทางมัลติแอสต์สำหรับกลุ่มสมาชิกแบบไดนามิกทำให้มีการนำมาใช้กันอย่างแพร่หลายในโปรโตคอลสำหรับการสร้างเส้นทางมัลติแอสต์บนเครือข่ายอินเทอร์เน็ต โดยมีหลักการทำงานคือเมื่อมีโหนดใดๆ ในระบบเครือข่ายต้องการเข้าร่วมเป็นสมาชิกใหม่ของกลุ่มมัลติแอสต์ โหนดต้นทางจะทำการเชื่อมต่อโหนดดังกล่าวนั้นเข้าสู่กลุ่มมัลติแอสต์เดิมโดยใช้อัลกอริทึมการหาเส้นทางที่สั้นที่สุดทำการหาเส้นทางที่สั้นที่สุดจากโหนดต้นทางไปยังโหนดสมาชิกใหม่และทำการส่งแพคเกจข้อมูลผ่านเส้นทางนั้น

สำหรับข้อดีของอัลกอริทึมชนิดนี้คือง่ายต่อการสร้างสำหรับการใช้งานจริงในระบบเครือข่ายรวมถึงสามารถใช้งานเป็นโปรโตคอลการหาเส้นทางแบบกระจายได้ เนื่องจากความเป็นอิสระต่อกันระหว่างโหนดสมาชิกและสามารถสร้างเส้นทางมัลติแคสต์ได้โดยใช้โปรโตคอลการหาเส้นทางแบบยูนิแคสต์ที่มีการใช้งานอยู่แล้ว แต่เนื่องจากการมีระดับของความเป็นอิสระต่อกันสูงระหว่างโหนดสมาชิกของกลุ่มมัลติแคสต์ จึงทำให้มีการใช้งานทรัพยากรร่วมกันระหว่างโหนดสมาชิกกลุ่มมัลติแคสต์น้อยมากจึงทำให้ค่าคอสม์ของเส้นทางมัลติแคสต์มีค่าสูง และจะยิ่งก่อให้เกิดปัญหามากขึ้นเมื่อจำนวนโหนดสมาชิกของกลุ่มมัลติแคสต์มีปริมาณมากขึ้น

2.4.2 อัลกอริทึมการหาเส้นทางแบบกรี้ดี (Greedy algorithm)

เป็นอัลกอริทึมที่ถูกเสนอในงานวิจัย [5] เพื่อการสร้างเส้นทางมัลติแคสต์กลุ่มสมาชิกแบบไดนามิกโดยมีหลักการการทำงานคือเริ่มต้นด้วยการสร้างเส้นทางมัลติแคสต์ที่มีค่าคอสม์น้อยที่สุดเท่าที่เป็นไปได้หลังจากนั้นเมื่อมีการเพิ่มเติมโหนดสมาชิกใหม่เข้ามาจะทำการสร้างเส้นทางใหม่เพิ่มเติมจากเส้นทางที่มีอยู่เดิมโดยทำการเปลี่ยนแปลงเส้นทางมัลติแคสต์ที่มีอยู่เดิมให้น้อยที่สุด

ในอัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบกรี้ดี เมื่อมีโหนดใดๆ ในระบบเครือข่ายต้องการเข้าร่วมเป็นโหนดสมาชิกใหม่ของกลุ่มมัลติแคสต์ โหนดต้นทางจะทำการต่อเชื่อมโหนดดังกล่าวเข้ากับกลุ่มมัลติแคสต์เดิมโดยเลือกใช้เส้นทางที่มีค่าคอสม์น้อยที่สุดระหว่างโหนดสมาชิกใหม่กับโหนดใดๆ ที่อยู่ในเส้นทางมัลติแคสต์ในขณะนั้น

อัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบกรี้ดีมีข้อดีคือสามารถสร้างเส้นทางมัลติแคสต์ได้อย่างมีประสิทธิภาพดีในแง่ของค่าคอสม์รวมของเส้นทางมัลติแคสต์ แต่อัลกอริทึมนี้ยังมีข้อเสียเนื่องจากแต่ละโหนดสมาชิกในกลุ่มมัลติแคสต์จะมีระดับความสัมพันธ์ต่อกันค่อนข้างสูงซึ่งมีผลให้เกิดปัญหาด้านการขยายขนาดของระบบเครือข่ายเมื่อนำไปสร้างเป็นโปรโตคอลสำหรับการใช้งานจริงในระบบเครือข่าย ซึ่งปัญหานี้ได้มีงานวิจัยบางชิ้น [8] ให้ความสนใจและเริ่มต้นพัฒนากลไกเพื่อแก้ปัญหานี้ขึ้นมาอันเนื่องจากจุดเด่นในแง่ประสิทธิภาพของอัลกอริทึมนี้ อย่างไรก็ตามเนื่องจากปัญหานี้อยู่นอกเหนือขอบเขตการศึกษาวิจัยของวิทยานิพนธ์นี้ ดังนั้นจึงจะทำการศึกษาปัญหานี้ต่อไปสำหรับเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกภายใต้สภาวะแวดล้อมแบบสวิตช์ทรัพยากรล่วงหน้าต่อไปในอนาคต

2.4.3 อัลกอริทึมแบบทรัังก์เสมือน (Virtual Trunk Dynamic Multicast)

เป็นอัลกอริทึมการสร้างเส้นทางที่ถูกนำเสนอในงานวิจัย [7] โดยมีรากฐานจากแนวความคิดที่ว่าในการคำนวณหาเส้นทางมัลติแคสต์สำหรับกลุ่มแบบไดนามิก ถ้าอนุญาตให้ทำการเรียงเส้นทางใหม่ทุกครั้งที่มีการเปลี่ยนแปลงของโหนดสมาชิกกลุ่มมัลติแคสต์โดยใช้อัลกอริทึมการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบสแตติก จะพบว่ามีย่อยของกลุ่มของโหนดและลิงค์

จำนวนหนึ่งซึ่งถูกใช้เพื่อการสร้างเส้นทางอยู่เป็นประจำ จากข้อสังเกตนี้จึงเกิดสมมติฐานของการสร้างแม่แบบของการสร้างเส้นทางมัลติแอสต์จากกลุ่มของโหนดและลิงค์ดังกล่าวเพื่อใช้ประโยชน์ในการสร้างเส้นทางมัลติแอสต์สำหรับกลุ่มสมาชิกแบบไดนามิกที่สามารถให้ค่าคอสต์ที่ได้ใกล้เคียงกับการใช้อัลกอริทึมสำหรับกลุ่มสมาชิกแบบสแตติกโดยปราศจากการเรียงเรียงเส้นทางใหม่ทุกครั้งที่มีการเปลี่ยนแปลงของโหนดสมาชิกกลุ่มมัลติแอสต์ และนิยามแม่แบบที่เกิดจากกลุ่มของโหนดและลิงค์ที่พบอยู่เป็นประจำในการสร้างเส้นทางว่า ทรงค์เสมือน (Virtual trunk) ขั้นตอนสำหรับการสร้างทรงค์เสมือนเป็นดังนี้

ขั้นตอนที่ 1 ใช้อัลกอริทึมการหาเส้นทางที่สั้นที่สุดระหว่างโหนด u และ v ใดๆ ทุกคู่ในกราฟ G และทำการนับจำนวนของเส้นทางที่สั้นที่สุดที่ผ่านแต่ละโหนดเอาไว้

ขั้นตอนที่ 2 ทำการนิยามเซต F ซึ่งประกอบด้วยโหนดจำนวน ΘN โหนดที่มีจำนวนเส้นทางที่สั้นที่สุดผ่านโหนดนั้นมากที่สุด

ขั้นตอนที่ 3 ทำการสร้างกราฟสมบูรณ์ $G^* = (F, E^*)$ เมื่อ E^* คือเซตของเส้นทางที่มีค่าคอสต์น้อยที่สุดระหว่างโหนด u และ v ใดๆ ในเซต F บนกราฟ G ซึ่งได้จากการคำนวณด้วยอัลกอริทึมของไดจค์สตรา

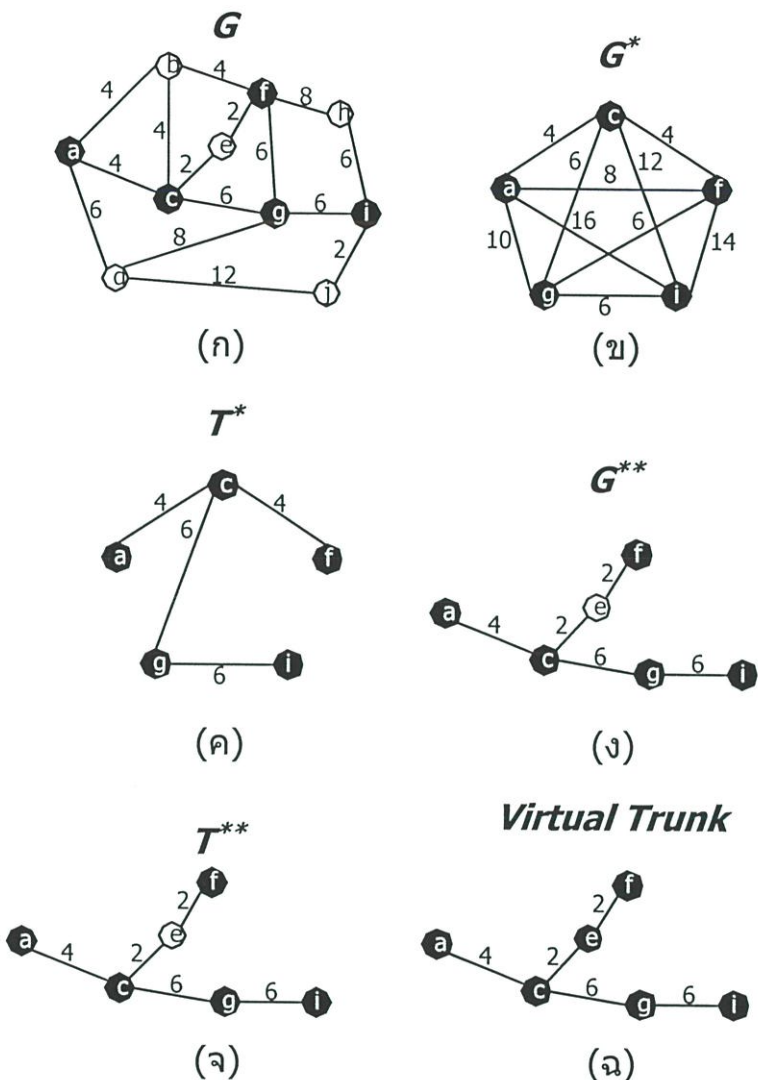
ขั้นตอนที่ 4 ทำการคำนวณเส้นทางต้นไม้ที่สั้นที่สุด T^* บนกราฟ G^*

ขั้นตอนที่ 5 ทำการสร้างกราฟ G^{**} บนกราฟ G จากการแปลงลิงค์ในเส้นทางต้นไม้ T^* ให้กลับไปเป็นลิงค์ที่สมมูลกันซึ่งอยู่ในเซต E

ขั้นตอนที่ 6 ทำการหาเส้นทางต้นไม้ที่สั้นที่สุด T^{**} บนกราฟ G^{**}

ขั้นตอนที่ 7 จากเส้นทางต้นไม้ T^{**} ให้ทำการตัดทิ้งจุดและส่วนของเส้นตรงที่เชื่อมจุดที่ไม่จำเป็นออก กราฟที่เหลือนี้จะเป็น ทรงค์เสมือน ของกราฟ G ซึ่งประกอบไปด้วยโหนดในทรงค์เสมือนและเส้นทางจากโหนดในทรงค์เสมือนไปยังโหนดต้นทาง

จากรูปที่ 2.4 กราฟ G หลังจากผ่านขั้นตอนที่ 1 และ 2 จะได้สมาชิกของเซต F คือ $\{a, c, f, g, i\}$ และจากการใช้อัลกอริทึมของไดจค์สตราจะสามารถสร้างกราฟสมบูรณ์ G^* ได้ดังในรูปที่ 2.4 (ข) ในขั้นตอนต่อไปทำการหาเส้นทางต้นไม้ที่สั้นที่สุด T^* จากกราฟสมบูรณ์ G^* และเมื่อทำการแปลงลิงค์ที่อยู่บนกราฟ T^* กลับเป็นลิงค์ที่สมมูลบนกราฟ G จะได้กราฟย่อย G^{**} บนกราฟ G ดังรูปที่ 2.4 (ง) เมื่อทำการหาเส้นทางต้นไม้ที่สั้นที่สุดอีกครั้งและทำการตัดโหนดและลิงค์ที่ไม่จำเป็นออกจะได้กราฟย่อยบนกราฟ G ดังรูปที่ 2.4 (จ) ซึ่งเป็นทรงค์เสมือนบนระบบเครือข่ายที่สมมูลกับกราฟ G



รูปที่ 2.4 ลำดับขั้นตอนการทำงานของกรสร้างทรีค้เสมือน

- (ก) กราฟ G (ข) กราฟสมบูรณ์ G^* (ค) เส้นทางต้นไม้ที่สั้นที่สุด T^*
- (ง) กราฟย่อย G^{**} (จ) เส้นทางต้นไม้ที่สั้นที่สุด T^{**} (ฉ) ทรีค้เสมือนของกราฟ G

ในการทำงานของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบทรีค้เสมือนนั้นเส้นทางจากโหนดซึ่งอยู่ในทรีค้เสมือนใดๆ ไปยังโหนดต้นทางจะไม่ถูกทำการต่อเชื่อมเอาไว้ล่วงหน้า เมื่อมีโหนดใดๆ ในระบบเครือข่ายต้องการเข้าร่วมเป็นสมาชิกของกลุ่มมัลติแคสต์ โหนดต้นทางจะทำการต่อเชื่อมโหนดสมาชิกใหม่เข้ากับกลุ่มมัลติแคสต์เดิมโดยเลือกเส้นทางที่มีค่าคอสต์น้อยที่สุดระหว่างโหนดสมาชิกใหม่กับโหนดใดๆ ในทรีค้เสมือน และเมื่อทำการต่อเชื่อมโหนดสมาชิกใหม่เข้ากับโหนดในทรีค้เสมือนแล้วเส้นทางจากโหนดในทรีค้เสมือนนี้ไปยังโหนดต้นทางก็จะถูก

เชื่อมต่อเช่นเดียวกันหากไม่ได้มีการเชื่อมต่อมาก่อนเนื่องจากการเชื่อมต่อของโหนดสมาชิกอื่นในกลุ่มมัลติแคสต์

อย่างไรก็ตาม อัลกอริทึมการสร้างเส้นทางแบบทรีคส์เสมือนนั้นจะสามารถให้ประสิทธิภาพการสร้างเส้นทางที่ดีได้ก็ต่อเมื่อการกำหนดค่า Θ สำหรับการคำนวณทรีคส์เสมือนมีความถูกต้องแม่นยำดังเห็นได้จากผลการทดลองในงานวิจัย [7] ซึ่งการกำหนดค่า Θ นั้นย่อมาหมายความว่า การคาดคะเนจำนวนผู้ใช้บริการของกลุ่มมัลติแคสต์ซึ่งการคาดคะเนดังกล่าวสามารถกระทำได้อย่างถูกต้องทั้งยังเป็นงานที่มีความซับซ้อนต่อระบบบริหารเครือข่ายอย่างมากรวมถึงยังส่งผลกระทบต่อเนื่องเมื่อมีการเปลี่ยนแปลงจำนวนของสมาชิกกลุ่มมัลติแคสต์หรือเมื่อมีการเปลี่ยนโหนดต้นทางในการส่งข้อมูล ซึ่งจะต้องทำการคำนวณทรีคส์เสมือนใหม่ทั้งหมด เพื่อรักษาประสิทธิภาพของการสร้างเส้นทางเอาไว้ให้อยู่ในระดับคงที่ ด้วยเหตุนี้อัลกอริทึมแบบนี้จะจึงขาดความคล่องตัวในการใช้งานเมื่อจำนวนโหนดสมาชิกของกลุ่มมัลติแคสต์ไม่สามารถกำหนดได้ล่วงหน้า

2.5 บทสรุป

ในบทนี้ได้ทำการศึกษาเกี่ยวกับเงื่อนไขรวมถึงอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ทั้งสำหรับกลุ่มสมาชิกแบบสแตติกและแบบไดนามิกจากงานวิจัยต่างๆ เพื่อเป็นแนวทางในงานวิจัยของวิทยานิพนธ์ฉบับนี้ โดยที่การสร้างเส้นทางสำหรับกลุ่มมัลติแคสต์ที่มีพฤติกรรมแบบไดนามิกมีความน่าสนใจมากกว่าเนื่องจากมีลักษณะที่สอดคล้องกับพฤติกรรมที่เกิดขึ้นของผู้ใช้บริการในระบบเครือข่ายมากกว่า ในบทต่อไปจะกล่าวถึงการนำรูปแบบการสงวนทรัพยากรล่วงหน้ามาผนวกเข้ากับอัลกอริทึมการสร้างเส้นทางมัลติแคสต์เพื่อให้การใช้ทรัพยากรในระบบเครือข่ายเป็นไปอย่างมีประสิทธิภาพมากขึ้นรวมถึงวิจัยและพัฒนาอัลกอริทึมการสร้างเส้นทางทางมัลติแคสต์แบบสงวนทรัพยากรล่วงหน้าให้มีประสิทธิภาพสูงยิ่งขึ้น

บทที่ 3

เส้นทางมัลติแคสต์แบบสงวนทรัพยากรล่วงหน้า

ในบทนี้จะกล่าวถึงการสงวนทรัพยากรล่วงหน้าในระบบเครือข่ายรวมถึงการนำแนวคิดการสงวนทรัพยากรล่วงหน้ามาประยุกต์ใช้กับการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกหรืออัลกอริทึมโมดิไฟด์กรีตซึ่งเป็นงานวิจัยที่นำเสนอโดย ผศ. ดร. โชติพัทธ์ ภรณวลัย และคณะ [14] และในช่วงท้ายของบทจะเป็นการทดลองเพื่อแสดงประสิทธิภาพที่เพิ่มขึ้นในการจัดสรรทรัพยากรในระบบเครือข่ายเมื่อนำข้อมูลการสงวนทรัพยากรล่วงหน้าของโหนดสมาชิกกลุ่มมัลติแคสต์มาประกอบการพิจารณาเพื่อการสร้างเส้นทางมัลติแคสต์

3.1 การสงวนทรัพยากรล่วงหน้าในระบบเครือข่าย

ในปัจจุบันระบบเครือข่ายสื่อสารได้ถูกพัฒนาให้มีขีดความสามารถมากขึ้นเช่นเดียวกับปริมาณของผู้ใช้บริการของระบบเครือข่ายที่เพิ่มสูงขึ้น ส่งผลให้มีแอปพลิเคชันจำนวนมากต้องการให้มีกลไกการสงวนทรัพยากรในระบบเครือข่ายเพื่อให้แอปพลิเคชันต่างๆ สามารถทำงานได้เต็มประสิทธิภาพ รายละเอียดเพิ่มเติมของโปรโตคอลรวมถึงสถาปัตยกรรมของการสงวนทรัพยากรและกำหนดทรัพยากรในระบบเครือข่ายที่มีใช้อยู่ในปัจจุบันสามารถพบได้ใน [17] แต่อย่างไรก็ตามรูปแบบการสงวนทรัพยากรที่พบอยู่ในปัจจุบันจะเป็นลักษณะการขอใช้ทรัพยากรในทันทีและไม่มีกำหนดช่วงเวลาที่น่านอนในการใช้ทรัพยากรในระบบเครือข่าย รูปแบบการสงวนทรัพยากรดังกล่าวอาจจะดูมีประโยชน์และสอดคล้องกับการให้บริการจริงในระบบเครือข่าย ถึงแม้ว่าการกำหนดให้มีกระบวนการช่วงเวลาในการใช้ทรัพยากรหรือการสงวนทรัพยากรล่วงหน้าจะเป็นการเพิ่มความซับซ้อนแก่ระบบการจัดการของเครือข่าย แต่สามารถก่อให้เกิดประสิทธิภาพมากขึ้นไม่เพียงในแง่ของประสิทธิภาพการทำงานของแอปพลิเคชันหากยังรวมถึงประสิทธิภาพการจัดการและบริหารทรัพยากรในระบบเครือข่ายที่สูงขึ้นด้วยเช่นกัน

เนื่องจากคุณประโยชน์ดังกล่าวจึงมีงานวิจัยจำนวนมากที่เกี่ยวข้องกับกลไกการสงวนทรัพยากรล่วงหน้าในระบบเครือข่าย โดยที่สามารถจำแนกได้เป็น 3 ประเภท [9] คือ การเพิ่มกลไกการทำงานให้กับโปรโตคอลหรือระบบซิกแนลลิง (Signaling) ที่มีการใช้งานอยู่แล้วในปัจจุบันให้เพิ่มขีดความสามารถในการรองรับการให้บริการทรัพยากรล่วงหน้าในระบบเครือข่ายได้ อีกประเภทคือการเพิ่มความสามารถในอัลกอริทึมการจัดสรรทรัพยากรในระบบเครือข่ายให้สามารถรองรับการสงวนทรัพยากรล่วงหน้าได้และในประเภทสุดท้ายซึ่งเกี่ยวกับการออกแบบอัลกอริทึมการเลือกเส้นทางส่งข้อมูลในระบบเครือข่ายที่มีการนำข้อมูลการสงวนทรัพยากรล่วงหน้าใน

ระบบเครือข่ายมาประกอบการพิจารณาในการเลือกเส้นทางและจากงานวิจัยทั้ง 3 ประเภทดังกล่าว พบว่างานวิจัยที่เกี่ยวข้องกับการเลือกเส้นทางในระบบเครือข่ายยังมีปริมาณน้อยเมื่อเทียบกับ 2 ประเภทที่เหลือ โดยเฉพาะอย่างยิ่งงานวิจัยที่เกี่ยวข้องกับการสร้างเส้นทางมัลติแคสต์ภายใต้สถานะแวดล้อมแบบสงวนทรัพยากรล่วงหน้า

ในสถานะแวดล้อมการสงวนทรัพยากรล่วงหน้าในระบบเครือข่าย ดังที่กล่าวมาแล้วว่านอก จากที่ผู้ใช้บริการในระบบเครือข่ายจะทำการระบุค่าพารามิเตอร์ (Parameter) ทางทราฟฟิค (Traffic) ที่สนับสนุนการส่งข้อมูลผ่านระบบเครือข่ายแล้วยังต้องทำการระบุช่วงเวลาที่ต้องการใช้บริการ ทรัพยากรในระบบเครือข่ายอีกด้วย โดยจะทำการระบุค่า t_{st} ซึ่งเป็นเวลาที่เริ่มต้นการให้บริการ ทรัพยากรในระบบเครือข่ายและช่วงเวลาในการใช้บริการทรัพยากรในระบบเครือข่ายโดยผ่านทาง ค่า t_{dep} ซึ่งเป็นเวลาที่จะยกเลิกการให้บริการทรัพยากรในระบบเครือข่าย (ช่วงเวลาของการใช้บริการ ทรัพยากรในระบบเครือข่ายจะเท่ากับ $t_{dep} - t_{st}$) ในกรณีของการระบุค่า t_{st} สามารถแบ่งได้เป็น 2 กรณีย่อย โดยกรณีแรกคือมีความต้องการใช้บริการทันทีเมื่อทำการสงวนทรัพยากรในระบบเครือ ข่าย และอีกกรณีคือทำการสงวนทรัพยากรล่วงหน้าก่อนที่จะมีการใช้บริการจริง ดังนั้นรูปแบบการ สงวนทรัพยากรล่วงหน้าจึงสามารถแบ่งได้ตามลักษณะดังนี้

รูปแบบที่ 1 เมื่อ $t_{cur} = t_{arr} = t_{st}$ และ

มีการแจ้งช่วงเวลาที่แน่นอนในการใช้บริการทรัพยากรเท่ากับ $t_{dep} - t_{st}$

รูปแบบที่ 2 เมื่อ $t_{cur} = t_{arr}$, $t_{arr} < t_{st}$ และ

มีการแจ้งช่วงเวลาที่แน่นอนในการใช้บริการทรัพยากรเท่ากับ $t_{dep} - t_{st}$

เมื่อ

t_{cur} เป็นเวลาปัจจุบันของระบบเครือข่าย

t_{arr} เป็นเวลาที่ผู้ใช้บริการทำการร้องขอการสงวนทรัพยากรในระบบเครือข่าย

ในงานวิทยานิพนธ์นี้จะเป็นการศึกษาวิจัยรูปแบบการสงวนทรัพยากรล่วงหน้าในรูปแบบที่ 1 รวมถึงส่วนขยายดังที่จะกล่าวถึงในบทต่อไป เนื่องจากเป็นรูปแบบการให้บริการสงวนทรัพยากร ล่วงหน้าซึ่งถือว่าเป็นจุดกึ่งกลางในการเปลี่ยนแปลงจากการสงวนทรัพยากรที่มีอยู่ในปัจจุบันซึ่ง ไม่มีการระบุช่วงเวลาที่แน่นอนในการใช้ทรัพยากรบนระบบเครือข่ายไปสู่รูปแบบการสงวนทรัพยากร ล่วงหน้าอย่างเต็มที่หรือรูปแบบที่ 2 และผลจากการศึกษาจะมีส่วนช่วยในการวิจัยและพัฒนาการให้ บริการในรูปแบบที่ 2 ให้มีประสิทธิภาพมากขึ้นในอนาคต

3.2 เงื่อนไขการสร้างเส้นทางมัลติแคสต์แบบสงวนทรัพยากรล่วงหน้า

ในหัวข้อนี้จะกล่าวถึงการเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ภายใต้สภาพแวดล้อมแบบสงวนทรัพยากรล่วงหน้า โดยจะเป็นการอ้างอิงกับทฤษฎีกราฟเช่นเดียวกับเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ในลักษณะทั่วไป เว้นแต่มีการนำข้อมูลด้านการสงวนทรัพยากรล่วงหน้าในระบบเครือข่ายสำหรับผู้ให้บริการมาประกอบการพิจารณาในการเลือกเส้นทางเพื่อใช้ในการส่งข้อมูล งานวิจัย [9] ได้เคยกล่าวถึงเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ในสภาพแวดล้อมการสงวนทรัพยากรล่วงหน้าสำหรับกลุ่มสมาชิกแบบสแตติก (ค่า t_{arr} และค่า t_{dep} มีค่าเท่ากันสำหรับแต่ละโหนดสมาชิกของกลุ่มมัลติแคสต์) ซึ่งการสร้างเส้นทางมัลติแคสต์ภายใต้เงื่อนไขนี้จะสมมูลกับปัญหาการสร้างเส้นทางต้นไม้สไตน์เนอร์ (Steiner Tree Problem) แต่ในวิทยานิพนธ์นี้จะให้ความสนใจที่เงื่อนไขของการสร้างเส้นทางมัลติแคสต์สำหรับกรณีกลุ่มสมาชิกแบบไดนามิก (ค่า t_{arr} และค่า t_{dep} มีค่าแตกต่างกันในแต่ละโหนดสมาชิกของกลุ่มมัลติแคสต์) เนื่องจากมีความสอดคล้องกับสถานการณ์จริงในระบบเครือข่ายมากกว่า โดยมีรูปแบบการสงวนทรัพยากรล่วงหน้าแบบที่ 1 เมื่อ $t_{cur} = t_{arr} = t_{st}$ และมีการแจ้งช่วงเวลาการใช้บริการทรัพยากรล่วงหน้าเท่ากับ $t_{dep} - t_{st}$ ดังที่ได้กล่าวถึงในหัวข้อที่ผ่านมา

ในการให้บริการสงวนทรัพยากรล่วงหน้า ทรัพยากรของลิงค์ระหว่างโหนดในระบบเครือข่ายจะถูกขยายออกไปในโดเมน (Domain) ของเวลาซึ่งประกอบด้วยสล็อตของเวลา (Time Slot) ที่มีขนาดเท่ากันหลายสล็อต $\{0, 1, 2, \dots, n\}$ ในช่วงเวลาของแต่ละสล็อตหมายถึงระยะเวลาที่น้อยที่สุดที่ระบบยอมให้ทำการสงวนทรัพยากรล่วงหน้าได้ ดังนั้นผู้ใช้บริการสงวนทรัพยากรล่วงหน้าจะใช้ตัวเลขจำนวนเต็มของจำนวนสล็อตในการบอกถึงช่วงเวลาที่ต้องการสงวนทรัพยากร สำหรับวิทยานิพนธ์นี้กำหนดให้ขนาดของสล็อตเท่ากับ 1 หน่วยเวลาใดๆ และช่วงเวลาทำการสงวนทรัพยากรล่วงหน้าถูกกำหนดให้เป็นจำนวนเท่าของ 1 หน่วยเวลา ดังนั้นเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ในสภาพแวดล้อมแบบสงวนทรัพยากรล่วงหน้าสามารถนิยามได้ดังนี้

กำหนดให้

กราฟ $G = (V, E)$ เมื่อ V เป็นเซตของโหนดใดๆ บนกราฟ G และ E เป็นเซตของลิงค์ซึ่งมีลักษณะสมมาตรที่เชื่อมระหว่างโหนดใดๆ บนกราฟ G ซึ่งสมาชิกของเซต E สามารถเขียนแทนได้ด้วย e_{ij} เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V

เซต C ซึ่งมีสมาชิกเป็นคอสต์ฟังก์ชันของสมาชิกของเซต E ซึ่ง $C \subset \mathbb{R}^+$ (เซตของจำนวนจริงบวกรวมถึงค่า 0) และสมาชิกของเซต C สามารถเขียนแทนได้ด้วย $c(e_{ij})$ เมื่อ i

และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V และเนื่องจากลิงค์ในระบบเครือข่ายมีลักษณะที่สมมาตร ดังนั้นจึงทำให้ $c(e_{ij}) = c(e_{ji})$

เซตของลำดับเหตุการณ์ที่จะมีโหนดเข้าร่วมการเป็นสมาชิกของกลุ่มมัลติแคสต์ในแต่ละช่วงเวลาเท่ากับ เซต $R = \{r(t_1), r(t_2), r(t_3), \dots, r(t_k)\}$ เมื่อ $r(t_i)$ เป็นเหตุการณ์ที่มีโหนดเข้าร่วมเป็นสมาชิกกลุ่มมัลติแคสต์ $r(t_i)$ จะประกอบด้วย $(z_i, t_{dep}(z_i))$ โดยที่ z_i เป็นโหนดที่ต้องการเข้าร่วมกลุ่มมัลติแคสต์ t_i เป็นเวลาที่โหนดเริ่มเข้าร่วมกลุ่มมัลติแคสต์ และ $t_{dep}(z_i)$ เป็นเวลาที่โหนด z_i จะออกจากกลุ่มมัลติแคสต์

เซตของโหนดสมาชิกของกลุ่มมัลติแคสต์ในแต่ละช่วงเวลาเท่ากับเซต $S = \{s_1, s_2, s_3, \dots, s_k\}$

จากข้อกำหนดข้างต้น สร้างทางมัลติแคสต์ T_i ด้วยอัลกอริทึมการสร้างเส้นทางมัลติแคสต์หลังจากเกิดเหตุการณ์ r_i ขึ้นซึ่ง T_i คือเส้นทางมัลติแคสต์ซึ่งเป็นกราฟย่อยของกราฟ G และครอบคลุมทุกโหนดในกลุ่มมัลติแคสต์ s_i โดยปราศจากการจัดเรียงเส้นทางใหม่และมีจุดประสงค์เพื่อให้ค่าคอสต์รวมตลอดช่วงเวลาของกลุ่มมัลติแคสต์มีค่าน้อยที่สุด ซึ่งเงื่อนไขนี้สามารถเขียนได้ตามสมการที่ 3.1

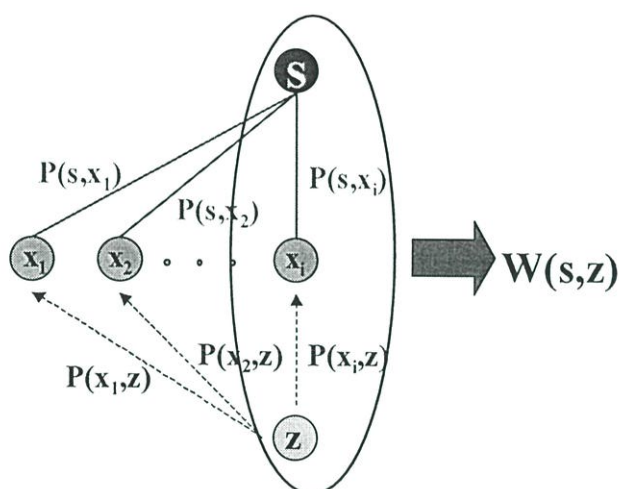
$$\{\min[\sum_{t_i=1}^{t_k} \text{Cost}(T_i)]\} \quad (3.1)$$

เมื่อ t_k เป็นระยะเวลาทั้งหมดในการส่งข้อมูลมัลติแคสต์

3.3 อัลกอริทึมการสร้างเส้นทางมัลติแคสต์โมดิฟายด์กรีดี

อัลกอริทึมโมดิฟายด์กรีดีได้ถูกนำเสนอโดย ผศ. ดร. โชติพัทธ์ ภรณ์วลัยและคณะ [14] เพื่อใช้ในการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกในสภาพแวดล้อมแบบสงวนทรัพยากรล่วงหน้าตามเงื่อนไขที่กล่าวถึงในหัวข้อที่ผ่านมา อัลกอริทึมนี้พัฒนามาจากอัลกอริทึมการหาเส้นทางแบบกรีดีแต่ได้นำข้อมูลของการสงวนทรัพยากรล่วงหน้าของโหนดสมาชิกมาใช้ในการจัดระเบียบเส้นทางมัลติแคสต์ส่งผลให้มีการใช้ทรัพยากรของระบบเครือข่ายร่วมกันมากขึ้นระหว่างโหนดสมาชิกกลุ่มมัลติแคสต์ เป็นการเพิ่มประสิทธิภาพของการสร้างเส้นทางมัลติแคสต์ในแง่ของการลดจำนวนของโหนดสโตนเนอร์ให้น้อยลงทำให้ค่าคอสต์รวมของเส้นทางมัลติแคสต์มีค่าลดน้อยลงเช่นกัน อัลกอริทึมโมดิฟายด์กรีดีมีขั้นตอนการทำงานดังนี้

เมื่อโหนด z ใดๆ ในระบบเครือข่ายต้องการเข้าร่วมเป็นสมาชิกของกลุ่มมัลติแคสต์ โหนดต้นทาง s ของกลุ่มมัลติแคสต์จะทำการเลือกโหนด x_i ใดๆ ในเส้นทางมัลติแคสต์ที่ทำให้ค่าคอสม์เหมือนของเส้นทางจากโหนด s ไปยังโหนด z ซึ่งเขียนแทนได้โดย $W(s,z)$ มีค่าน้อยที่สุด และโหนดต้นทาง s จะเลือกใช้เส้นทางดังกล่าวในการเชื่อมต่อโหนด z เข้ากับเส้นทางมัลติแคสต์ที่มีอยู่ในปัจจุบัน



รูปที่ 3.1 ค่าคอสม์เหมือนของเส้นทางมัลติแคสต์

จากรูปที่ 3.1 พบว่าค่าคอสม์เหมือนของเส้นทางจากโหนดต้นทางของกลุ่มมัลติแคสต์ s ไปยังโหนดสมาชิกใหม่ z ใดๆ หรือ $W(s,z)$ ในระบบเครือข่ายจะสามารถแบ่งออกได้เป็น 2 ช่วง ช่วงแรกคือเส้นทางจากโหนดต้นทาง s ไปยังโหนด x_i ใดๆ ในเส้นทางมัลติแคสต์ปัจจุบันโดยที่โหนด x_i สามารถเป็นได้ทั้งโหนดสมาชิกหรือโหนดสโตนเนอร์ใดๆ ที่อยู่ในเส้นทางมัลติแคสต์ และช่วงที่สองคือเส้นทางจากโหนด x_i นั้นไปยังโหนดสมาชิกใหม่ z ดังนั้นค่าคอสม์เหมือน $W(s,z)$ สามารถเขียนได้ดังสมการที่ 3.2

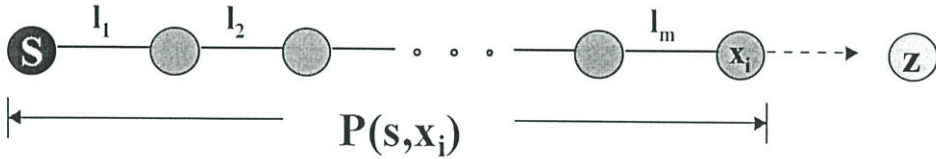
$$W(s,z) = W(s,x_i) + W(x_i,z) \quad (3.2)$$

เมื่อ

$W(s,x_i)$ เป็นค่าคอสม์เหมือนของเส้นทางจากโหนดต้นทาง s ไปยังโหนด x_i ใดๆ ที่อยู่ในเส้นทางมัลติแคสต์

$W(x_i,z)$ เป็นค่าคอสม์เหมือนของเส้นทางจากโหนด x_i ใดๆ ในเส้นทางมัลติแคสต์ โดยผ่านโหนดอื่นๆ ในระบบเครือข่ายซึ่งไม่ได้อยู่ในเส้นทางมัลติแคสต์ ไปยังโหนดสมาชิกใหม่ z

ค่าคอสม์เสมือน $W(s, x_i)$ สามารถคำนวณได้โดยนำค่าคอสม์เสมือนของแต่ละลิงค์บนเส้นทางจากโหนด s ถึงโหนด x_i หรือ $P(s, x_i)$ มารวมกันตามรูปที่ 3.2 โดยก่อนที่จะทำการคำนวณค่าคอสม์เสมือน โหนดต้นทางจะทำการกำหนดค่ารอบเวลาซึ่งเขียนแทนได้โดย t_{cr} เพื่อเป็นกรอบในการพิจารณาถึงค่าต่างๆ ที่อยู่ในเส้นทางจากโหนดต้นทาง s ไปยังโหนด x_i ใดๆ หรือ $P(s, x_i)$ ในเส้นทางมัลติแคสต์ สำหรับการอัลกอริทึม โมดิฟายด์ครีติจจะกำหนดให้กรอบเวลาหรือค่า t_{cr} มีค่าเท่ากับเวลาที่โหนดสมาชิกใหม่ z จะออกจากกลุ่มมัลติแคสต์ ($t_{dep}(z)$)



รูปที่ 3.2 การคำนวณค่าคอสม์เสมือน $W(s, x_i)$

กำหนดให้ l_m เป็นลิงค์ใดๆ บนเส้นทางจากโหนดต้นทาง s ไปยังโหนด x_i ใดๆ ซึ่งอยู่ในเส้นทางมัลติแคสต์และมีค่า $t_{out}(l_m) < t_{dep}(z)$ แล้ว ค่าคอสม์เสมือน $W(s, x_i)$ จะมีค่าตามสมการ 3.3 และค่าคอสม์ของลิงค์ซึ่งมีค่า $t_{out}(l_m) > t_{dep}(z)$ จะถูกพิจารณาให้มีค่าเป็นศูนย์

$$W(s, x_i) = \sum_{l_m \in P(s, x_i)} \{ [t_{dep}(z) - t_{out}(l_m)] \times C(l_m) \} \quad (3.3)$$

เมื่อ

$t_{out}(l_m)$ เป็นเวลาที่ลิงค์ l_m จะถูกยกเลิกการใช้งานในเส้นทางมัลติแคสต์

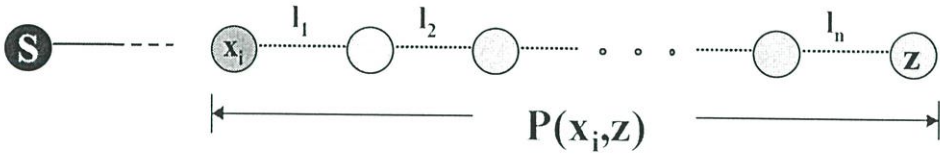
$C(l_m)$ เป็นค่าคอสม์ของลิงค์ l_m ใดๆ

จากสมการของค่าคอสม์เสมือน $W(s, x_i)$ พบว่าเส้นทางที่ประกอบด้วยลิงค์ l_m ส่วนใหญ่ที่มีค่า t_{out} ใกล้เคียงหรือมากกว่าค่ารอบเวลา $t_{dep}(z)$ จะมีค่าคอสม์เสมือนที่ต่ำซึ่งแสดงถึงปริมาณการใช้ทรัพยากรร่วมกันระหว่างโหนดสมาชิกที่มากขึ้นด้วยเช่นกัน กรณีของค่าคอสม์เสมือน $W(x_i, z)$ สามารถคำนวณได้ในลักษณะที่คล้ายกับ $W(s, x_i)$ โดยนำค่าคอสม์เสมือนของแต่ละลิงค์บนเส้นทางจากโหนด x_i ถึง z หรือ $P(x_i, z)$ มารวมกันดังรูปที่ 3.3 กำหนดให้ l_n เป็นลิงค์ใดๆ บนเส้นทางดังกล่าว ค่าคอสม์เสมือน $W(x_i, z)$ จะมีค่าตามสมการ 3.4

$$W(x_i, z) = \sum_{l_n \in P(x_i, z)} \{ [t_{dep}(z) - t_{st}(z)] \times C(l_n) \} \quad (3.4)$$

เมื่อ

$t_s(z)$ เป็นเวลาที่โหนดสมาชิกใหม่ z เริ่มต้นเข้าร่วมกลุ่มมัลติแคสต์
 $C(l_n)$ เป็นค่าคอสต์ของลิงค์ l_n ใดๆ



รูปที่ 3.3 การคำนวณค่าคอสต์เสมือน $W(x_i, z)$

หลังจากที่ทำการคำนวณหาค่าคอสต์เสมือนของแต่ละเส้นทางที่สามารถใช้ในการเชื่อมต่อไปยังโหนดสมาชิกใหม่ z ได้แล้ว โหนดต้นทางจะทำการแผ่ขยายเส้นทางต้นไม้มืออยู่เดิมให้สามารถครอบคลุมโหนดสมาชิกมัลติแคสต์เดิมรวมถึงโหนดสมาชิกใหม่ z โดยเลือกที่จะแผ่ขยายเส้นทางต้นไม้มืออยู่เดิมออกไปโดยใช้เส้นทางที่มีค่าคอสต์เสมือน $W(s, z)$ ที่ต่ำที่สุด

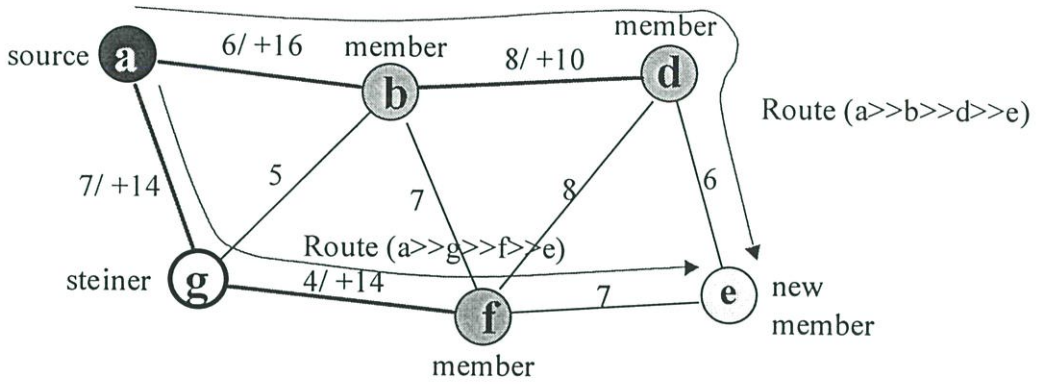
จากสมการที่ 3.2 ถึง 3.4 พบว่า เมื่อมีโหนดใดๆ ในระบบเครือข่ายต้องการเข้าเป็นสมาชิกของกลุ่มมัลติแคสต์นอกจากอัลกอริทึม โมดิฟายด์กรีดีจะสร้างเส้นทางมัลติแคสต์โดยอาศัยหลักการแผ่ขยายเส้นทางต้นไม้มืออยู่เดิมออกไปให้น้อยที่สุดเพื่อให้ครอบคลุมโหนดดังกล่าวโดยปราศจากการจัดเรียงเส้นทางใหม่เช่นเดียวกับอัลกอริทึมอื่นๆ แล้ว อัลกอริทึม โมดิฟายด์กรีดียังอาศัยประโยชน์จากข้อมูลการสงวนทรัพยากรล่วงหน้าของโหนดสมาชิกในการจัดระเบียบเส้นทางมัลติแคสต์ให้มีการใช้ทรัพยากรร่วมกันมากที่สุดระหว่างโหนดสมาชิก ซึ่งตัวอย่างและประโยชน์ของการสร้างเส้นทางมัลติแคสต์ด้วยอัลกอริทึม โมดิฟายด์กรีดีจะได้กล่าวถึงในหัวข้อต่อไป

3.4 ประโยชน์ของการสร้างเส้นทางมัลติแคสต์ด้วยการสงวนทรัพยากรล่วงหน้า

ในหัวข้อนี้จะแสดงถึงตัวอย่างการสร้างเส้นทางมัลติแคสต์ด้วยอัลกอริทึม โมดิฟายด์กรีดี และทำการเปรียบเทียบผลลัพธ์ที่ได้ในแง่ของคอสต์ของเส้นทางมัลติแคสต์กับอัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบกรีดีซึ่งไม่มีการนำข้อมูลการสงวนทรัพยากรล่วงหน้าของโหนดสมาชิกมาประกอบการพิจารณาในการสร้างเส้นทางมัลติแคสต์

จากรูปที่ 3.4 แสดงตัวอย่างการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกด้วยอัลกอริทึม โมดิฟายด์กรีดี เมื่อโหนด e ต้องการเข้าร่วมเป็นสมาชิกใหม่ของกลุ่มมัลติแคสต์ที่เวลา $t_s = +4$ สัมพันธ์กับเวลาที่โหนดต้นทางเริ่มต้นการส่งข้อมูลมัลติแคสต์และโหนด e จะออกจากกลุ่มมัลติแคสต์ที่เวลา $t_{dep} = +12$ สัมพันธ์กับเวลาที่โหนดต้นทางเริ่มต้นการส่งข้อมูลมัลติแคสต์เช่นเดียวกัน เมื่อกลุ่มมัลติแคสต์ปัจจุบันประกอบด้วยประกอบด้วยโหนด a, b, d และ f ตามลำดับ

และมีโหนด a เป็นโหนดต้นทางของกลุ่มมัลติแคสต์และเริ่มส่งข้อมูลมัลติแคสต์ตั้งแต่วันที่ $t = 0$ ค่าคอสต์ของแต่ละลิงค์ในระบบเครือข่ายจะถูกแสดงด้วยตัวเลขที่กำกับไว้ที่ลิงค์นั้น ตารางเวลาของการส่งวนทรัพยากรล่วงหน้าสำหรับแต่ละโหนดสมาชิกของกลุ่มมัลติแคสต์เป็นดังตารางที่ 3.1



รูปที่ 3.4 ตัวอย่างการคำนวณเส้นทางมัลติแคสต์

ตารางที่ 3.1 ตารางเวลาการเข้าและออกจากกลุ่มมัลติแคสต์ของโหนดสมาชิก

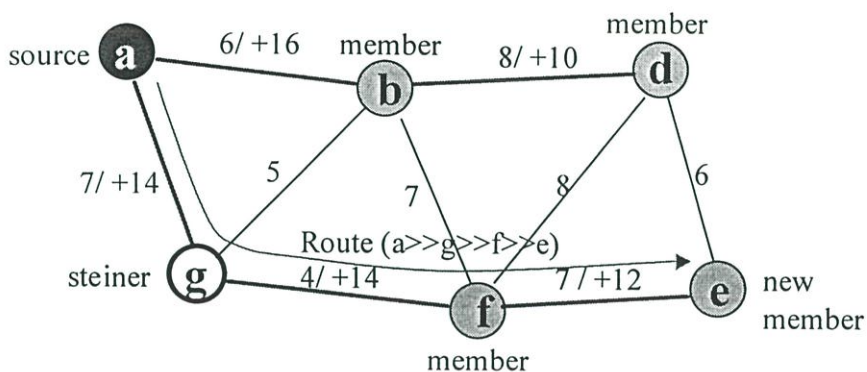
โหนด	เวลาที่เข้าร่วมกลุ่มมัลติแคสต์ (t_{st})	เวลาที่ออกจากกลุ่มมัลติแคสต์ (t_{dep})
b	0	+16
d	+1	+10
f	+2	+14
e	+4	+12

ในอัลกอริทึม โมดิฟายด์ครีติกก่อนที่จะเริ่มต้นคำนวณค่าคอสต์เสมือนสำหรับเส้นทางต่างๆ ที่สามารถเชื่อมต่อกับโหนดสมาชิกใหม่ได้ โหนดต้นทางของกลุ่มมัลติแคสต์จะทำการกำหนดค่ากรอบเวลาเพื่อเป็นเกณฑ์ในการพิจารณาลิงค์ต่างๆ ที่อยู่ในเส้นทางมัลติแคสต์โดยกำหนดให้มีค่าเท่ากับค่า t_{dep} ของโหนดสมาชิกใหม่ที่ต้องการเข้าร่วมกลุ่มมัลติแคสต์ จากรูปที่ 3.4 พบว่ามีเส้นทางที่สามารถใช้ในการเชื่อมต่อกับโหนดสมาชิกใหม่หรือโหนด e จำนวน 2 เส้นทางคือ เส้นทาง abde และเส้นทาง agfe โดยการใช้สมการการคำนวณค่าคอสต์เสมือน 3.2 ถึง 3.4 ค่าคอสต์เสมือนของทั้ง 2 เส้นทางเป็นไปดังตารางที่ 3.2

จากตารางที่ 3.2 พบว่า ในการขยายเส้นทางมัลติแคสต์เดิมออกไปให้ครอบคลุมโหนดสมาชิกใหม่หรือโหนด e โดยผ่านเส้นทาง agfe นั้นจะมีการเพิ่มค่าคอสต์ของการสร้างเส้นทางที่ต่ำกว่าภายในช่วงเวลาที่โหนด e เป็นสมาชิกของกลุ่มมัลติแคสต์ ดังนั้นโหนดต้นทางจะเลือกใช้เส้นทางดังกล่าวในการเชื่อมต่อโหนด e เข้ากับเส้นทางมัลติแคสต์ที่มีอยู่เดิมดังแสดงในรูปที่ 3.5

ตารางที่ 3.2 ค่าคอสต์เสมือนของแต่ละเส้นทาง

เส้นทาง	ค่าคอสต์เสมือน $W(s,x)$	ค่าคอสต์เสมือน $W(x,z)$	ค่าคอสต์เสมือน $W(s,z)$
เส้นทาง abde	$W(a,d) = [(12-10) \times 8] = 16$	$W(d,e) = [(12-4) \times 6] = 48$	64
เส้นทาง agfe	$W(a,f) = 0$	$W(f,e) = [(12-4) \times 7] = 28$	28



รูปที่ 3.5 เส้นทางมัลติเคสต์เมื่อรวมโหนด e

จากตัวอย่างเดียวกัน เมื่อใช้อัลกอริทึมการสร้างเส้นทางมัลติเคสต์แบบกริดซึ่งไม่มีการนำข้อมูลการสงวนทรัพยากรล่วงหน้ามาพิจารณาในการเลือกเส้นทาง โหนดต้นทางจะเลือกใช้เส้นทาง abde ในการเชื่อมต่อโหนด e เข้ากับเส้นทางมัลติเคสต์ เนื่องจากเมื่อพิจารณาการขยายเส้นทางมัลติเคสต์ออกไปเฉพาะในช่วงเวลาปัจจุบันเส้นทาง abde จะให้ค่าคอสต์ที่ต่ำกว่าเส้นทาง agfe แต่หากพิจารณาค่าคอสต์ของเส้นทางมัลติเคสต์ตลอดช่วงเวลาที่โหนด e เป็นสมาชิกกลุ่มมัลติเคสต์ จะพบว่าการใช้เส้นทาง abde จะให้ค่าคอสต์ที่เพิ่มสูงกว่าการเลือกใช้เส้นทาง agfe ในการเชื่อมต่อ ดังแสดงในตารางที่ 3.2 ดังนั้นการนำข้อมูลการสงวนทรัพยากรล่วงหน้าของโหนดสมาชิกมาประกอบการพิจารณาการสร้างเส้นทางมัลติเคสต์จะสามารถทำให้การใช้ทรัพยากรในระบบเครือข่ายเพื่อการสร้างเส้นทางมัลติเคสต์เป็นไปอย่างมีประสิทธิภาพมากขึ้น

3.5 ประสิทธิภาพของเส้นทางมัลติเคสต์แบบสงวนทรัพยากรล่วงหน้า

หลังจากที่ได้กล่าวถึงอัลกอริทึมการสร้างเส้นทางมัลติเคสต์แบบสงวนทรัพยากรล่วงหน้าสำหรับกลุ่มสมาชิกแบบไดนามิกในหัวข้อที่ผ่านมา ในหัวข้อนี้จะเป็นการจำลองระบบเครือข่ายเพื่อแสดงให้เห็นถึงการใช้ทรัพยากรในระบบเครือข่ายที่มีประสิทธิภาพมากขึ้นเมื่อนำข้อมูลการสงวนทรัพยากรล่วงหน้าจากโหนดสมาชิกมาประกอบการพิจารณาในการสร้างเส้นทางมัลติเคสต์ โดยเป็นการเปรียบเทียบระหว่างอัลกอริทึม โมดิไฟด์กริดและอัลกอริทึมแบบกริดดั้งเดิม [5]

3.5.1 สภาพแวดล้อมการหาเส้นทางบนระบบเครือข่ายจำลอง

ในการเปรียบเทียบประสิทธิภาพของอัลกอริทึมการหาเส้นทางมัลติแคสต์ในงานวิจัยโดยทั่วไป เพื่อความสะดวกในการศึกษาและการจำลองระบบเครือข่าย จึงนิยมใช้ค่าคอสต์ของเส้นทางที่คำนวณได้จากการแก้ปัญหาการสร้างเส้นทางมัลติแคสต์ที่นิยามไว้ในหัวข้อที่ 2.4 และ 3.2 เป็นเกณฑ์ในการพิจารณา ค่าคอสต์ดังกล่าวสามารถแสดงให้เห็นถึงปริมาณการใช้ทรัพยากรในระบบเครือข่ายในการสร้างเส้นทางมัลติแคสต์ซึ่งมีค่าแตกต่างกันไปในแต่ละอัลกอริทึม เงื่อนไขของการสร้างเส้นทางมัลติแคสต์ดังกล่าวมีสมมติฐานของโหนดต้นทางของกลุ่มมัลติแคสต์ว่า โหนดต้นทางจะรู้สถานะของลิงก์ทุกลิงก์ในระบบเครือข่ายโดยสมบูรณ์และทำการตัดลิงก์ที่มีทรัพยากรไม่เพียงพอออกจากการพิจารณา

ในสถานการณ์จริงของระบบเครือข่ายทั้งการเลือกเส้นทางแบบยูนิแคสต์และมัลติแคสต์ การทำให้โหนดรู้สถานะของลิงก์ที่ต้องการสำหรับทุกลิงก์ในระบบเครือข่ายโดยสมบูรณ์ต้องสูญเสียทรัพยากรในระบบเครือข่ายอย่างมาก [10] ดังนั้นในหัวข้อนี้จะทำการทดลองเพื่อเปรียบเทียบประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบโมดิฟายด์กรีดีและแบบกรีดีบนระบบเครือข่ายจำลองเมื่อข้อมูลสถานะลิงก์ในระบบเครือข่ายที่ใช้ในการคำนวณเส้นทางอาจจะมี ความไม่ถูกต้องเกิดขึ้น เนื่องจากความพยายามในการลดการสูญเสียทรัพยากรในระบบเครือข่าย โดยการลดความถี่ของการปรับปรุงข้อมูลสถานะลิงก์ เงื่อนไขของการสร้างเส้นทางมัลติแคสต์ซึ่ง สอดคล้องกับสถานการณ์ดังกล่าวเป็นดังนี้

กำหนดให้

กราฟ $G = (V, E)$ เมื่อ V เป็นเซตของโหนดใดๆ บนกราฟ G และ E เป็นเซตของลิงก์ ซึ่งมีลักษณะสมมาตรที่เชื่อมระหว่างโหนดใดๆ บนกราฟ G ซึ่งสมาชิกของเซต E สามารถเขียนแทนได้ด้วย e_{ij} เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V

เซต C ซึ่งมีสมาชิกเป็นคอสต์ฟังก์ชันของสมาชิกของเซต E ซึ่ง $C \subset \mathbb{R}^+$ (เซตของจำนวนจริงบวกรวมถึงค่า 0) และสมาชิกของเซต C สามารถเขียนแทนได้ด้วย $c(e_{ij})$ เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V และเนื่องจากลิงก์ในระบบเครือข่ายมีลักษณะที่สมมาตร ดังนั้นจึงทำให้ $c(e_{ij}) = c(e_{ji})$

เซต B ซึ่งมีสมาชิกเป็นแบนด์วิธที่สามารถให้บริการได้ของลิงก์ซึ่งเป็นสมาชิกของเซต E ซึ่งสมาชิกของเซต B สามารถเขียนแทนด้วย $b(e_{ij})$ เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V และ $0 \leq b(e_{ij}) \leq bc_{ij}$ เมื่อ bc_{ij} เป็นแบนด์วิธทั้งหมดของลิงก์ที่เชื่อมต่อระหว่างโหนด i และ j และเนื่องจากลิงก์ในระบบเครือข่ายมีลักษณะที่สมมาตร ดังนั้นจึงทำให้ $b(e_{ij}) = b(e_{ji})$

เซตของลำดับเหตุการณ์ที่จะมีโหนดเข้าร่วมหรือออกจากความเป็นสมาชิกของกลุ่มมัลติแคสต์เท่ากับ เซต $R = \{r_1, r_2, r_3, \dots, r_n\}$ เมื่อ $r(t_i)$ เป็นเหตุการณ์ที่มีโหนดเข้าร่วมเป็นสมาชิกกลุ่มมัลติแคสต์ $r(t_i)$ จะประกอบด้วย $(z_i, t_{dep}(z_i), b_{req})$ โดยที่ z_i เป็นโหนดที่ต้องการเข้าร่วมกลุ่มมัลติแคสต์ t_i เป็นเวลาที่โหนดเริ่มเข้าร่วมกลุ่มมัลติแคสต์ $t_{dep}(z_i)$ เป็นเวลาที่โหนด z_i จะออกจากกลุ่มมัลติแคสต์ และ b_{req} เป็นปริมาณของแบนด์วิธที่โหนดสมาชิกต้องการใช้ในการส่งข้อมูล โดยจะมีปริมาณที่เท่ากันสำหรับทุกโหนดสมาชิกเนื่องจากการสื่อสารแบบมัลติแคสต์

เซตของโหนดสมาชิกของกลุ่มมัลติแคสต์ในแต่ละช่วงเวลาเท่ากับเซต $S = \{s_1, s_2, s_3, \dots, s_k\}$

จากข้อกำหนดข้างต้น ทำการสร้างเส้นทางมัลติแคสต์ T_i ด้วยอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ หลังจากเกิดเหตุการณ์ r_i ให้สามารถครอบคลุมโหนดสมาชิกกลุ่มมัลติแคสต์ s_i ให้มากที่สุดเท่าที่เป็นไปได้โดยปราศจากการเรียงเรียงเส้นทางใหม่ เมื่อต้องรับประกันแบนด์วิธ b_{req} ให้กับโหนดสมาชิกในเส้นทางมัลติแคสต์ซึ่ง T_i เป็นกราฟย่อยของกราฟ G

3.5.2 ค่าที่ใช้ในการวัดประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์

เนื่องจากสมมติฐานว่าข้อมูลสถานะของลิงค์ในระบบเครือข่ายที่ใช้ในการคำนวณเส้นทางของโหนดอาจมีความไม่ถูกต้องเกิดขึ้น เนื่องจากความพยายามเพื่อลดการสูญเสียทรัพยากรของระบบเครือข่ายในการปรับปรุงข้อมูลสถานะลิงค์ ทำให้การสร้างเส้นทางมัลติแคสต์ต้องคำนึงถึงปัญหาการเลือกเส้นทางที่มีทรัพยากรเพียงพอ (b_{req}) ในการส่งข้อมูลไปยังโหนดสมาชิกซึ่งปัญหานี้มีความคล้ายคลึงกับปัญหาการหาเส้นทางยูนิแคสต์ในแบบรับประกันแบนด์วิธ ดังนั้นจึงเลือกใช้ค่าอัตราการใช้บริการแบนด์วิธซึ่งกล่าวถึงในงานวิจัย [10] เพื่อการเปรียบเทียบประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบโมดิไฟด์กรีดีและกรีดี ค่าดังกล่าวสามารถคำนวณได้จากอัตราส่วนของจำนวนของแบนด์วิธที่สามารถให้บริการได้ต่อจำนวนแบนด์วิธทั้งหมดที่ถูกขอใช้บริการ สามารถเขียนเป็นสมการได้ดังสมการที่ 3.5

$$\text{อัตราส่วนการให้แบนด์วิธ} = \frac{\text{จำนวนรวมแบนด์วิธที่ให้บริการได้}}{\text{จำนวนรวมแบนด์วิธที่ขอใช้บริการ}} \quad (3.5)$$

สำหรับการส่งข้อมูลแบบมัลติแคสต์ เนื่องจากโหนดต้นทางจะทำการส่งข้อมูลที่เหมือนกันให้กับทุกโหนดสมาชิกทำให้โหนดสมาชิกทุกโหนดจึงมีค่า b_{req} เท่ากัน ดังนั้นอัตราการใช้บริการ

แบนด์วิธจึงมีค่าเท่ากับอัตราส่วนจำนวนโหนดสมาชิกของกลุ่มมัลติแคสต์ที่ได้รับการประกันแบนด์วิธต่อจำนวนโหนดทั้งหมดที่ต้องการเข้าร่วมเป็นสมาชิกกลุ่มมัลติแคสต์

3.5.3 นโยบายการกระจายค่าสถานะลิงค์

ในการคำนวณหาเส้นทางที่เหมาะสมเพื่อใช้ในการส่งข้อมูลระหว่างโหนดต่างๆ ในระบบเครือข่ายไม่ว่าจะเป็นการส่งในแบบยูนิแคสต์หรือมัลติแคสต์ โหนดในระบบเครือข่ายจำเป็นที่จะต้องมีความรู้ข้อมูลสถานะของลิงค์ที่ถูกต้องสำหรับแต่ละลิงค์ในระบบเครือข่าย แต่การกระจายข้อมูลสถานะลิงค์เพื่อปรับปรุงข้อมูลที่ถูกรับที่กในฐานข้อมูลมีความถูกต้องและทันสมัยอยู่ตลอดเวลาจะทำให้สูญเสียทรัพยากรในระบบเครือข่ายอย่างมาก ดังนั้นนโยบายการกระจายค่าสถานะลิงค์จึงได้ถูกกำหนดขึ้นเพื่อควบคุมกราฟฟิคของการปรับปรุงฐานข้อมูลสถานะลิงค์ให้เกิดความเป็นระเบียบ รวมถึงมีปริมาณของกราฟฟิคที่เหมาะสมมากขึ้น

นโยบายการกระจายสถานะของลิงค์ที่มีการใช้งานกันโดยปกติมีอยู่หลายประเภท แต่เนื่องจากการทดลองนี้เกี่ยวข้องกับกรสร้างเส้นทางมัลติแคสต์เพื่อรับประกันแบนด์วิธซึ่งอยู่ในสภาพแวดล้อมการจำลองระบบเครือข่ายที่คล้ายกับงานวิจัยด้านการเลือกเส้นทางเพื่อการรับประกันแบนด์วิธ [10] ซึ่งเป็นกรณีสำหรับการส่งข้อมูลยูนิแคสต์ ในงานวิจัยดังกล่าวได้แสดงให้เห็นว่านโยบายการกระจายสถานะลิงค์แบบเทรชโฮลด์ (Threshold) สามารถให้ประสิทธิภาพที่ดีโดยพิจารณาจากอัตราการให้บริการแบนด์วิธรวมถึงทรัพยากรที่สูญเสียไปในการปรับปรุงฐานข้อมูลสถานะลิงค์ของโหนดเมื่อเปรียบเทียบกับนโยบายการกระจายข้อมูลแบบแบ่งระดับและแบบหน่วงเวลา เนื่องจากการสร้างเส้นทางมัลติแคสต์ได้พัฒนามาจากการหาเส้นทางยูนิแคสต์และการกระจายข้อมูลสถานะลิงค์ของการส่งข้อมูลทั้ง 2 แบบมีพื้นฐานเดียวกัน ดังนั้นจึงสามารถใช้นโยบายการกระจายสถานะลิงค์แบบเทรชโฮลด์เช่นเดียวกับงานวิจัย [10] ในการทดลองนี้ได้

นโยบายการกระจายค่าสถานะลิงค์แบบเทรชโฮลด์จะอนุญาตให้ทำการกระจายค่าใหม่ของสถานะลิงค์ออกไปยังโหนดทั้งหมดในระบบเครือข่ายต่อเมื่อเงื่อนไขความสัมพันธ์ตามสมการที่ 3.6 มีค่าเป็นจริง

$$\frac{|bw_1^o - bw_1^c|}{bw_1^o} \geq th \quad (3.6)$$

เมื่อ

bw_1^o คือค่าสถานะลิงค์แบนด์วิธที่สามารถให้บริการได้ของลิงค์ 1 ที่ถูกประกาศออกไปครั้งสุดท้าย

$b_{w_1}^c$ คือค่าสถานะลิงค์แบนด์วิธปัจจุบันของลิงค์ 1 ที่สามารถให้บริการได้
 th คือค่าเทรซโฮลด์มีค่าอยู่ในช่วง 0.0 – 1.0

จากสมการที่ 3.6 พบว่า นโยบายการกระจายแบบเทรซโฮลด์มีข้อดีในด้านการควบคุมปริมาณทราฟฟิกของการกระจายข้อมูลสถานะลิงค์คือในขณะที่ปริมาณทรัพยากรในลิงค์ที่สามารถให้บริการได้มีจำนวนค่อนข้างมากจะมีทราฟฟิกการกระจายค่าของสถานะลิงค์ในปริมาณน้อยและเมื่อปริมาณของทรัพยากรในลิงค์ที่สามารถให้บริการได้มีจำนวนลดน้อยลงระบบจะมีความถี่ในการกระจายค่าของสถานะลิงค์มากขึ้นส่งผลให้ค่าสถานะลิงค์ในฐานะข้อมูลจะถูกปรับปรุงให้ทันสมัยอยู่ตลอดเวลาเพื่อให้ประสิทธิภาพในการเลือกเส้นทางเป็นไปอย่างสม่ำเสมอ

3.5.4 ค่าปริมาณความปลอดภัยของลิงค์ในระบบเครือข่าย

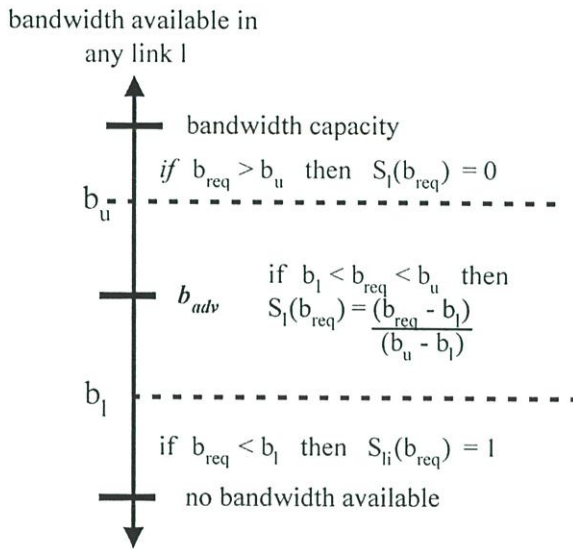
ในหัวข้อนี้จะกล่าวถึงรายละเอียดการนำข้อมูลสถานะลิงค์ไปคำนวณค่าปริมาณความปลอดภัย [10] เพื่อใช้ในการคำนวณหาเส้นทางที่สามารถให้บริการรับประกันแบนด์วิธโดยค่าปริมาณความปลอดภัยได้ถูกนำเสนอเพื่อใช้ในการคำนวณหาเส้นทางแบบรับประกันคุณภาพเมื่อความไม่ถูกต้องของสถานะลิงค์เนื่องจากการควบคุมปริมาณการกระจายข้อมูลสถานะลิงค์ในระบบเครือข่าย ค่าปริมาณความปลอดภัยในลิงค์มีคุณสมบัติเดียวกับค่าความน่าจะเป็น (มีพิสัยอยู่ในช่วง 0.0 - 1.0) โดยค่าความปลอดภัยสามารถบอกถึงความน่าจะเป็นที่จะมีปริมาณแบนด์วิธที่ต้องการใช้บริการบนลิงค์ใดๆ ในระบบเครือข่ายได้ ค่าปริมาณความปลอดภัยของลิงค์ 1 ใดๆ สามารถนิยามได้ด้วยฟังก์ชัน S_l ซึ่งเป็นฟังก์ชันของค่า b_{req} ตามสมการที่ 3.7

$$S_l(b_{req}) = \begin{cases} 1 & , \quad b_{req} \leq b_l \\ \frac{b_u - b_{req}}{b_u - b_l} & , \quad b_l < b_{req} \leq b_u \\ 0 & , \quad b_{req} > b_u \end{cases} \quad (3.7)$$

เมื่อ

b_{req} คือปริมาณแบนด์วิธที่โหนดสมาชิกต้องการการรับประกัน
 b_{adv} คือปริมาณแบนด์วิธที่สามารถให้บริการได้บนลิงค์ 1 ซึ่งถูกประกาศออกไปครั้งสุดท้าย
 b_l คือค่าขอบเขตต่ำสุดที่เป็นไปได้ของปริมาณแบนด์วิธที่สามารถให้บริการได้บนลิงค์ 1 หลังจากการประกาศค่า b_{adv}

b_u คือค่าขอบเขตสูงสุดที่เป็นไปได้ของปริมาณแบนด์วิธที่สามารถให้บริการได้บนลิงค์ 1 หลังจากการประกาศค่า b_{adv}



รูปที่ 3.6 ค่าปริมาณความปลอดภัยของลิงค์

สำหรับค่าของ b_u และ b_l นั้นจะมีวิธีในการคำนวณแตกต่างกันไปขึ้นอยู่กับนโยบายการกระจายค่าสถานะลิงค์ ในกรณีของนโยบายการกระจายค่าสถานะลิงค์แบบเทอร์ซโฮลด์ค่าทั้ง 2 สามารถนิยามได้ตามสมการที่ 3.8

$$\begin{aligned} b_l &= b_{adv} \times (1 - th) \\ b_u &= b_{adv} \times (1 + th) \end{aligned} \quad (3.8)$$

เนื่องจากค่าปริมาณความปลอดภัยเป็นค่าความน่าจะเป็นที่จะมีแบนด์วิธที่สามารถให้บริการได้อย่างน้อยที่สุดจำนวน b_{req} หน่วยบนลิงค์ใดๆ ในระบบเครือข่าย ดังนั้นค่าปริมาณความปลอดภัยจึงเข้าข่ายลักษณะของค่าปริมาณแบบมัลติพลิเคทีฟ (Multiplicative) [13] ดังนั้นค่าความปลอดภัยของเส้นทาง P ที่เกิดจากการรวมกันของกลุ่มลิงค์ใดๆ ในระบบเครือข่ายซึ่งเขียนแทนได้โดย $S(P)$ สามารถนิยามได้ดังสมการที่ 3.9

$$S(P) = \prod_{l \in P} S_l(b_{req}) \quad (3.9)$$

3.5.5 อัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบรับประกันแบนด์วิธ

ในการทดลองเปรียบเทียบประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ซึ่งสามารถรับประกันแบนด์วิธแก่ทุกโหนดโหนดสมาชิกของกลุ่มมัลติแคสต์ นอกจากค่าคอสต์เสมือนที่ใช้เป็นเกณฑ์ในการเลือกเส้นทางแล้วยังต้องมีการคำนึงถึงค่าปริมาณความปลอดภัยในแต่ละเส้นทางอีกด้วย ดังนั้นในหัวข้อนี้จะกล่าวถึงการดัดแปลงอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ให้สามารถทำการนำทั้งค่าคอสต์เสมือนและค่าปริมาณความปลอดภัยมาประกอบการพิจารณาการเลือกเส้นทางที่เหมาะสมสำหรับแต่ละโหนดสมาชิกโดยมีพื้นฐานมาจากการหาเส้นทางยูนิแคสต์แบบรับประกันแบนด์วิธ เมื่อค่าปริมาณความปลอดภัยเกี่ยวข้องกับความเสี่ยงจะเป็นที่เส้นทางที่ใช้ในการเชื่อมต่อระหว่างเส้นทางต้นไม่ในปัจจุบันและโหนดสมาชิกใหม่จะมีแบนด์วิธที่เพียงพอเพื่อใช้ในการส่งข้อมูลและค่าคอสต์เสมือนจะเกี่ยวข้องกับปริมาณทรัพยากรที่ถูกใช้ไปเพื่อการสร้างเส้นทางมัลติแคสต์

เมื่อมีโหนด z ใดๆ ในระบบเครือข่ายต้องการเข้าร่วมเป็นสมาชิกของกลุ่มมัลติแคสต์ โหนดต้นทางของกลุ่มมัลติแคสต์จะมีอัลกอริทึมการสร้างเส้นทางแบบรับประกันแบนด์วิธได้ดังนี้

ขั้นตอนที่ 1 คำนวณค่าความปลอดภัยของลิงค์ทั้งหมดในระบบเครือข่ายจากข้อมูลสถานะลิงค์แบนด์วิธและค่า b_{req}

ขั้นตอนที่ 2 ทำการคัดเลือกลิงค์ในระบบเครือข่ายโดยใช้เกณฑ์การคัดเลือกลิงค์ที่มีค่าความปลอดภัยเท่ากับ 0 ทิ้งไป [10]

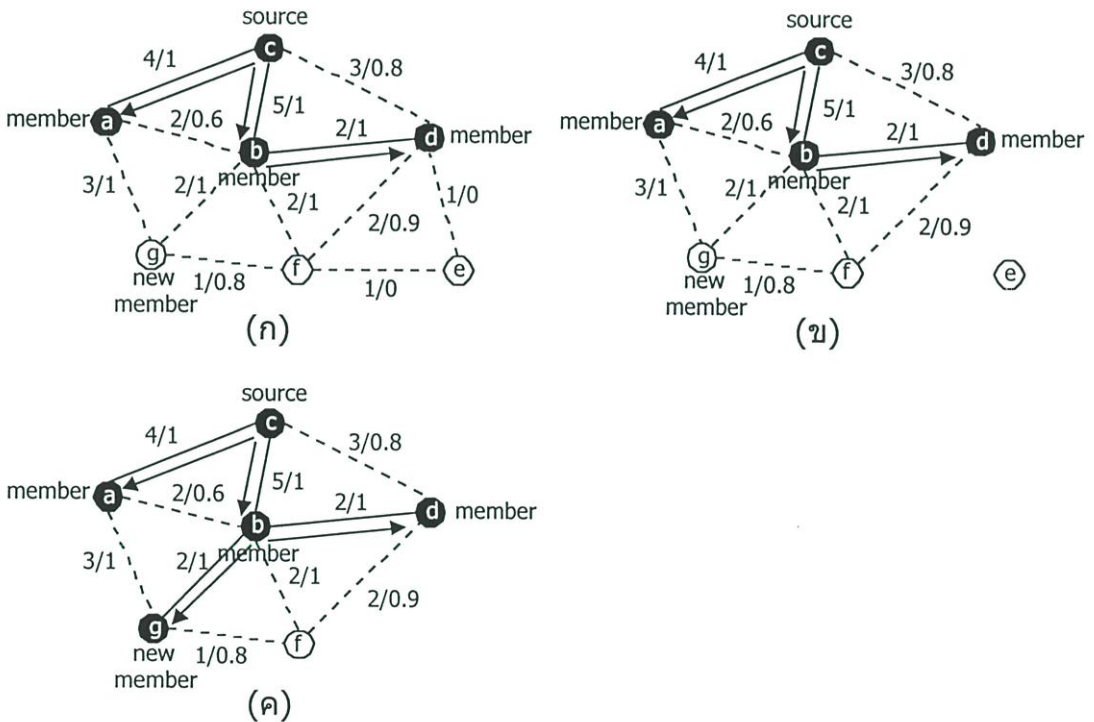
ขั้นตอนที่ 3 โหนดต้นทางจะทำการคำนวณเส้นทางที่ปลอดภัยที่สุดจากทุกๆ โหนดสมาชิกเดิมของกลุ่มมัลติแคสต์ไปยังโหนดสมาชิกใหม่ซึ่งในถ้ามีจำนวนเส้นทางที่ปลอดภัยมากที่สุดหลายเส้นทางก็จะเลือกเส้นทางที่มีค่าคอสต์น้อยที่สุด

ขั้นตอนที่ 4 ทำการเลือกเส้นทางจากโหนดสมาชิกเดิมในกลุ่มมัลติแคสต์ไปยังโหนดสมาชิกใหม่ที่คำนวณไว้ในขั้นตอนที่ 3 ที่มีค่าความปลอดภัยมากที่สุดซึ่งถ้ามีเส้นทางดังกล่าวหลายเส้นทางก็จะเลือกเส้นทางที่มีค่าคอสต์เสมือนน้อยที่สุด โดยที่การคำนวณค่าคอสต์เสมือนนั้นจะขึ้นอยู่กับอัลกอริทึมของการสร้างเส้นทางมัลติแคสต์

ในกรณีของการสร้างเส้นทางมัลติแคสต์แบบรับประกันแบนด์วิธ โดยอาศัยพื้นฐานการทำงานของอัลกอริทึมแบบกริดนั้น สามารถแสดงตัวอย่างของการสร้างเส้นทางได้ดังรูปที่ 3.7

จากรูปที่ 3.7 เป็นตัวอย่างของการสร้างเส้นทางมัลติแคสต์แบบรับประกันแบนด์วิธจากโหนด g ไปยังโหนดในเส้นทางมัลติแคสต์ ซึ่งประกอบไปด้วยโหนด a , b , c และ d โดยมีโหนด c เป็นโหนดต้นทางของกลุ่มมัลติแคสต์ โหนดและลิงค์ในระบบเครือข่ายที่เป็นเส้นที่บจะแสดงถึงโหนดและลิงค์ที่อยู่ในกลุ่มมัลติแคสต์ ในรูป 3.7 (ก) เป็นการทำงานในขั้นตอนที่ 1 และ 2 โดย

คำนวณค่าปริมาณความปลอดภัยในแต่ละลิงค์ในระบบเครือข่ายและตัดลิงค์ที่มีค่าความปลอดภัยเท่ากับ 0 ออก ต่อมาในขั้นตอนที่ 3 เลือกเส้นทางที่เหมาะสมไปยังแต่ละโหนดในเส้นทางมัลติแคสต์ โดยพิจารณาความปลอดภัยเป็นลำดับแรกและจากค่าคอสต์ของเส้นทางเป็นลำดับถัดไป ในขั้นตอนที่ 4 ดังรูป 3.7 (ข) พบว่าเส้นทางที่มีค่าความปลอดภัยมากที่สุดเท่ากับ 1 จากโหนด g ไปยังโหนดในเส้นทางมัลติแคสต์คือเส้นทางจากโหนด g ไปยังโหนด a ผ่านลิงค์ ga และจากโหนด g ไปยังโหนด b ผ่านลิงค์ gb แต่เนื่องจากเส้นทางไปยังโหนด b นั้นมีค่าคอสต์เสมือนที่ต่ำกว่า ดังนั้นจึงเลือกใช้โหนด b เพื่อต่อเชื่อมโหนด g เข้ากับกลุ่มมัลติแคสต์ ตามรูปที่ 3.7 (ค) สำหรับกรณีของอัลกอริทึมโมดิฟายด์ครีทีก็ก็สามารถทำได้ในลักษณะเดียวกัน เว้นแต่มีเงื่อนไขเพิ่มเติมว่าโหนดสมาชิกของกลุ่มมัลติแคสต์ทุกโหนดจะต้องทำการแจ้งเวลาเข้าและออกจากกลุ่มมัลติแคสต์ให้โหนดต้นทางทราบเพื่อใช้ในการคำนวณค่าคอสต์เสมือนของเส้นทางตามสมการที่ 3.2 ถึง 3.4



รูปที่ 3.7 ตัวอย่างของการสร้างเส้นทางการรับประกันแบนด์วิธแบบกรีด

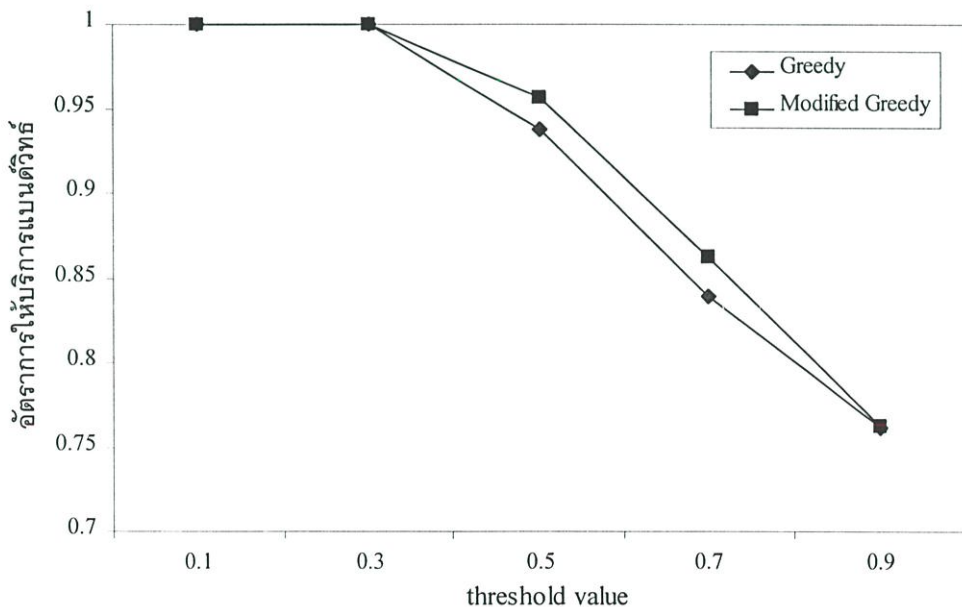
- (ก) เส้นทางมัลติแคสต์ก่อนที่โหนด g จะเข้าร่วมกลุ่ม
- (ข) ระบบเครือข่ายเมื่อตัดลิงค์ที่มีค่าความปลอดภัยเป็น 0 ออก
- (ค) เส้นทางมัลติแคสต์เมื่อ โหนด g เข้าเป็นสมาชิกของกลุ่ม

3.5.6 ผลการทดลองบนระบบเครือข่ายจำลอง

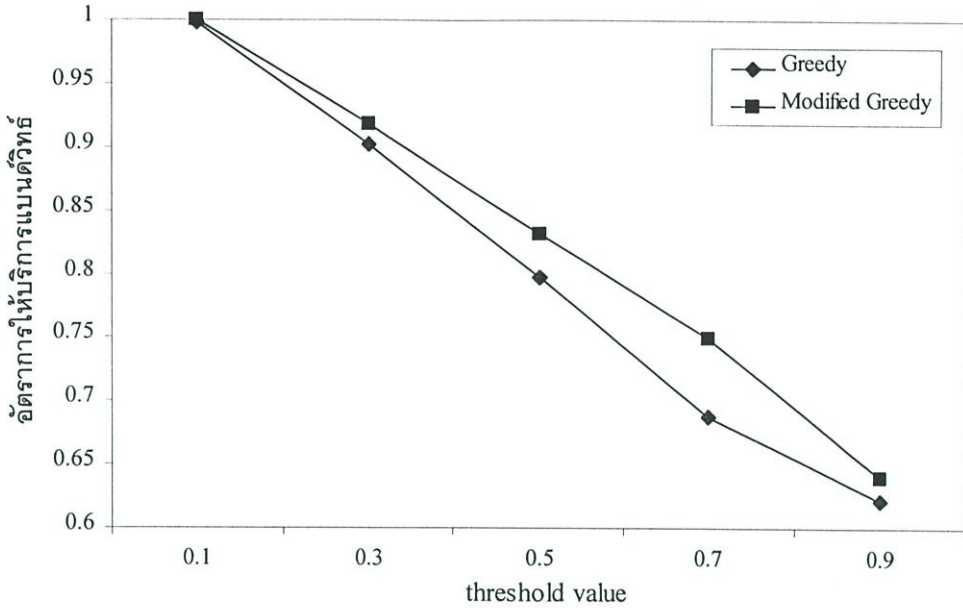
ในหัวข้อนี้จะกล่าวถึงการเปรียบเทียบประสิทธิภาพระหว่างการสร้างเส้นทางมัลติแคสต์แบบรับประกันแบนด์วิธที่ตัดแปลงมาจากอัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบธรรมดา โดยอ้างอิงถึงอัลกอริทึมแบบกรีดี และอัลกอริทึมที่มีการนำข้อมูลการสงวนทรัพยากรล่วงหน้ามาประกอบการพิจารณาในการสร้างเส้นทางมัลติแคสต์ซึ่งอ้างอิงถึงอัลกอริทึมโมดิไฟด์กรีดี โดยมีสมมติฐานว่าอัลกอริทึมที่มีประสิทธิภาพการใช้งานทรัพยากรของระบบเครือข่ายที่ดีกว่าย่อมส่งผลให้สามารถบริการจำนวนโหนดสมาชิกได้มากกว่า (อัตราการให้บริการแบนด์วิธมีค่าสูงกว่า)

ระบบเครือข่ายจำลองจะถูกสร้างขึ้น โดยอาศัยกราฟแบบสุ่ม [16] ซึ่งเป็นวิธีที่นิยมใช้ในการสร้างระบบเครือข่ายจำลองในงานวิจัยส่วนมากที่เกี่ยวข้องกับการหาเส้นทางมัลติแคสต์ สำหรับในการทดลองนี้กำหนดให้มีจำนวนโหนดในระบบเครือข่ายเท่ากับ 50 โหนดและเพื่อความสะดวกในการจำลองระบบกำหนดให้ความจุแบนด์วิธของแต่ละลิงค์ในระบบเครือข่ายมีค่าเท่ากัน โดยเท่ากับ 1 ค่า b_{req} จะอยู่ในช่วง $0 < b_{req} < 1$

เริ่มการทดลองโดยกำหนดให้ตลอดช่วงเวลาของการเก็บข้อมูลเท่ากับ 400 หน่วยเวลาจากเวลาอ้างอิง จำนวนกลุ่มมัลติแคสต์ที่ทำการส่งข้อมูลในช่วง 400 หน่วยเวลามีทั้งหมด 4 กลุ่มและมีค่า b_{req} เท่ากัน เวลาของการเริ่มส่งข้อมูลในแต่ละกลุ่มจะถูกสุ่มภายใต้เงื่อนไขว่าเวลาการเริ่มส่งข้อมูลของแต่ละกลุ่มต้องไม่เกินกว่า 200 หน่วยเวลานับจากเวลาอ้างอิงและมีความเป็นไปได้ที่จะมีปริมาณทราฟฟิกของแอปพลิเคชันประเภทอื่นบนลิงค์ต่างๆ ในระบบเครือข่ายอยู่ทำให้ปริมาณของแบนด์วิธที่สามารถให้บริการแก่โหนดสมาชิกของกลุ่มมัลติแคสต์อาจมีค่าไม่เท่ากับค่าความจุแบนด์วิธของลิงค์



รูปที่ 3.8 การเปรียบเทียบประสิทธิภาพของอัลกอริทึมเมื่อ b_{req} เท่ากับ 0.34



รูปที่ 3.9 การเปรียบเทียบประสิทธิภาพของอัลกอริทึมเมื่อ b_{req} เท่ากับ 0.45

จากรูปที่ 3.8 และ 3.9 ซึ่งเป็นผลการทดลองเมื่อ b_{req} เท่ากับ 0.34 และ 0.45 ตามลำดับ เมื่อแกน y แสดงถึงอัตราการใช้บริการแบนด์วิธและแกน x แสดงถึงความถี่ของนโยบายการกระจายค่าสถานะลิงค์ พบว่าอัลกอริทึมโมดิไฟด์กรีดีให้ประสิทธิภาพของการหาเส้นทางมัลติแคสต์แบบรับประกันแบนด์วิธได้ดีกว่าอัลกอริทึมแบบกรีดี เนื่องจากอัลกอริทึมโมดิไฟด์กรีดีมีการใช้ทรัพยากรในระบบเครือข่ายที่มีประสิทธิภาพมากกว่าโดยอาศัยข้อมูลการส่งวนทรัพยากรล่วงหน้ามาประกอบการพิจารณาในการสร้างเส้นทางมัลติแคสต์ทำให้มีการใช้ทรัพยากรร่วมกันระหว่างโหนดสมาชิกมากขึ้น ดังนั้นอัลกอริทึมโมดิไฟด์กรีดีจึงสามารถให้อัตราการใช้บริการแบนด์วิธที่สูงกว่าแม้ว่าจะมีความพยายามในการลดปริมาณทราฟฟิกเพื่อปรับปรุงฐานข้อมูลสถานะลิงค์ (เมื่อค่า th ของนโยบายการกระจายข้อมูลสถานะลิงค์มีค่ามากขึ้น) ซึ่งผลการทดลองนี้สอดคล้องกับผลการทดลองในงานวิจัย [14] เมื่อใช้ค่าคอสต์เป็นเกณฑ์ในการวัดประสิทธิภาพของอัลกอริทึม

3.6 บทสรุป

ในบทนี้ได้กล่าวถึงรายละเอียดของรูปแบบที่เป็นไปได้ของการส่งวนทรัพยากรล่วงหน้าบนระบบเครือข่ายซึ่งนำไปสู่เงื่อนไขของการสร้างเส้นทางมัลติแคสต์ในสภาพแวดล้อมแบบส่งวนทรัพยากรล่วงหน้ารวมถึงการนำเสนออัลกอริทึมโมดิไฟด์กรีดีในงานวิจัย [14] เพื่อใช้ในการสร้างเส้นทางมัลติแคสต์ในสภาพแวดล้อมดังกล่าว และในช่วงหลังได้กล่าวถึงการจำลองระบบเครือข่าย

เพื่อแสดงถึงประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ที่สูงขึ้นในแง่ของปริมาณการใช้ทรัพยากรในระบบเครือข่ายเพื่อสร้างเส้นทางมัลติแคสต์ เมื่อมีการนำข้อมูลการสงวนทรัพยากรล่วงหน้าของแต่ละโหนดสมาชิกมาประกอบการพิจารณาในการสร้างเส้นทางมัลติแคสต์เพื่อให้เกิดการใช้ทรัพยากรร่วมกันมากที่สุดระหว่างโหนดสมาชิกของกลุ่มมัลติแคสต์ โดยในบทความต่อไปจะกล่าวถึงการพัฒนาอัลกอริทึมโมดิไฟด์กรีดีให้มีประสิทธิภาพที่ดีขึ้นเมื่อข้อมูลการสงวนทรัพยากรล่วงหน้าของโหนดสมาชิกกลุ่มมัลติแคสต์อาจมีความคลาดเคลื่อนเกิดขึ้นได้

บทที่ 4

การพัฒนาอัลกอริทึมการหาเส้นทางมัลติแคสต์

ด้วยข้อดีของเทคนิคการสงวนทรัพยากรล่วงหน้าดังที่ได้กล่าวมาแล้วในเนื้อหาตอนต้นของวิทยานิพนธ์ฉบับนี้ ในบทนี้จะกล่าวถึงการพัฒนาการหาเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกภายใต้สภาวะแวดล้อมแบบสงวนทรัพยากรล่วงหน้าให้มีประสิทธิภาพมากขึ้น เมื่อทำการพิจารณาเงื่อนไขของการหาเส้นทางมัลติแคสต์ซึ่งมีลักษณะสอดคล้องกับสถานการณ์ที่เกิดขึ้นจริงในระบบเครือข่ายมากยิ่งขึ้น โดยที่โหนดสมาชิกของกลุ่มมัลติแคสต์ที่มีความเป็นไปได้ที่จะออกจากกลุ่มมัลติแคสต์ช้าหรือเร็วกว่าเวลาของการสงวนทรัพยากรล่วงหน้าที่ถูกแจ้งเข้ามาล่วงหน้ายังโหนดต้นทางช่วงเริ่มต้นการเข้าร่วมกลุ่มมัลติแคสต์

4.1 การเปลี่ยนแปลงเงื่อนไขของการสร้างเส้นทางมัลติแคสต์

ในหัวข้อนี้จะกล่าวถึงเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ภายใต้สภาวะแวดล้อมการสงวนทรัพยากรล่วงหน้าซึ่งจะมีความแตกต่างไปจากเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ที่ใช้ในงานวิจัย [14] ดังที่กล่าวถึงไปแล้วในบทที่ 3 เนื่องจากในงานวิจัย [14] นั้นมีสมมติฐานของเวลาของการเข้าร่วมกลุ่มมัลติแคสต์ที่ถูกแจ้งเข้ามาจากโหนดสมาชิกมีความถูกต้องแน่นอนและเชื่อถือได้ แต่เนื่องจากในสถานการณ์จริงของระบบเครือข่ายย่อมมีความน่าจะเป็นที่โหนดสมาชิกของกลุ่มมัลติแคสต์จะแยกตัวออกจากกลุ่มมัลติแคสต์เร็วกว่าหรือล่าช้ากว่าเวลาของการสงวนทรัพยากรที่แจ้งเข้ามาล่วงหน้ายังโหนดต้นทางของกลุ่มมัลติแคสต์ [15] ดังนั้นเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ที่สอดคล้องกับสถานการณ์ที่เกิดขึ้นจริงในระบบเครือข่ายจึงมีการเปลี่ยนแปลงเกิดขึ้นดังนี้

กำหนดให้

กราฟ $G = (V, E)$ เมื่อ V เป็นเซตของโหนดใดๆ บนกราฟ G และ E เป็นเซตของลิงค์ซึ่งมีลักษณะสมมาตรที่เชื่อมระหว่างโหนดใดๆ บนกราฟ G ซึ่งสมาชิกของเซต E สามารถเขียนแทนได้ด้วย e_{ij} เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V

เซต C ซึ่งมีสมาชิกเป็นคอสต์ที่ฟังก์ชันของสมาชิกของเซต E ซึ่ง $C \subset \mathbb{R}^+$ (เซตของจำนวนจริงบวกรวมถึงค่า 0 ด้วย) และสมาชิกของเซต C สามารถเขียนแทนได้ด้วย $c(e_{ij})$

เมื่อ i และ j เป็นโหนดสมาชิกใดๆ บนกราฟ G และเป็นสมาชิกของเซต V และเนื่องจาก
 ลิงค์ในระบบเครือข่ายมีลักษณะที่สมมาตร ดังนั้นจึงทำให้ $c(e_{ij}) = c(e_{ji})$

เซตของลำดับเหตุการณ์ที่จะมีโหนดเข้าร่วมการเป็นสมาชิกของกลุ่มมัลติแคสต์
 ในแต่ละช่วงเวลาเท่ากับ เซต $R = \{r(t_1), r(t_2), r(t_3), \dots, r(t_k)\}$ เมื่อ $r(t_i)$ เป็นเหตุการณ์ที่มี
 โหนดเข้าร่วมเป็นสมาชิกกลุ่มมัลติแคสต์ $r(t_i)$ จะประกอบด้วย $(z_i, t_{dep}(z_i), \Delta t_{dep}(z_i))$ โดย t_i
 เป็นเวลาที่โหนดเริ่มต้นเข้าร่วมกลุ่มมัลติแคสต์ z_i เป็นโหนดใดๆ ในระบบเครือข่ายที่
 ต้องการเข้าร่วมกลุ่มมัลติแคสต์ $t_{dep}(z_i)$ เป็นเวลาที่โหนด z_i จะออกจากกลุ่มมัลติแคสต์และ
 $\Delta t_{dep}(z_i)$ ซึ่งนิยามได้โดยสมการ 4.1

$$\Delta t_{dep}(z_i) = t'_{dep}(z_i) - t_{dep}(z_i) \quad (4.1)$$

เมื่อ

$t'_{dep}(z_i)$ เป็นเวลาจริงที่โหนดสมาชิก z_i ออกจากกลุ่มมัลติแคสต์

$t_{dep}(z_i)$ เป็นเวลาที่โหนดสมาชิก z_i แจ้งว่าจะออกจากกลุ่มมัลติแคสต์

ในกรณีที่ $\Delta t_{dep}(z_i) > 0$ หมายถึงโหนดสมาชิกจะออกจากกลุ่มมัลติแคสต์ล่าช้ากว่า
 เวลา $t_{dep}(z_i)$ และกรณีที่ $\Delta t_{dep}(z_i) < 0$ หมายถึงโหนดสมาชิกจะออกจากกลุ่มมัลติแคสต์เร็ว
 กว่าเวลา $t_{dep}(z_i)$

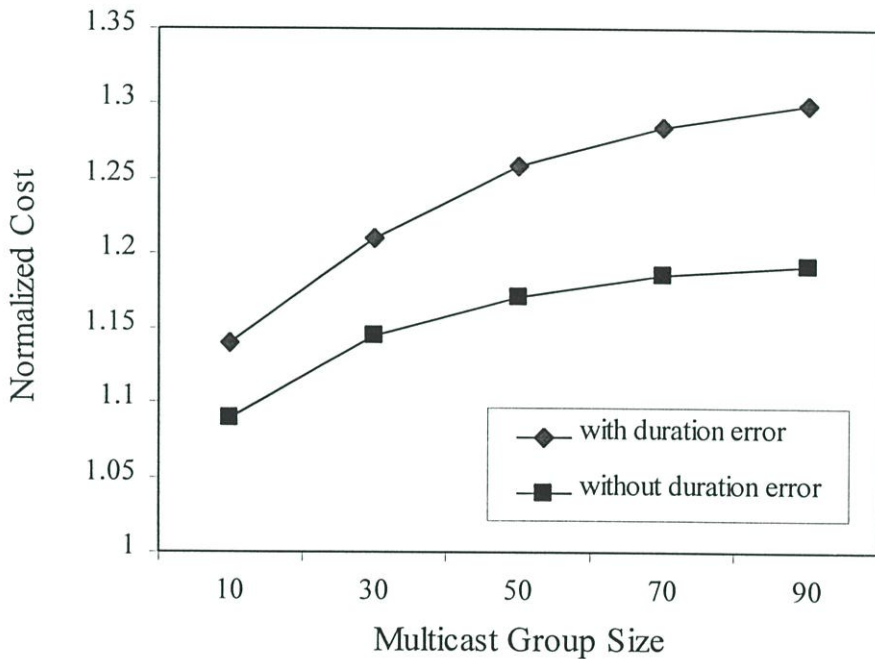
เซตของโหนดสมาชิกของกลุ่มมัลติแคสต์ในแต่ละช่วงเวลาเท่ากับเซต $S = \{s_1, s_2, s_3, \dots, s_k\}$

จากข้อกำหนดข้างต้น สร้างทางมัลติแคสต์ T_i ด้วยอัลกอริทึมการสร้างเส้นทางมัลติแคสต์
 หลังจากเกิดเหตุการณ์ r_i ขึ้นซึ่ง T_i คือเส้นทางมัลติแคสต์ซึ่งเป็นกราฟย่อยของกราฟ G และครอบคลุม
 ทุกโหนดในกลุ่มมัลติแคสต์ s_i โดยปราศจากการจัดเรียงเส้นทางใหม่และมีจุดประสงค์เพื่อทำ
 ให้ค่าคอสต์รวมตลอดช่วงเวลาของกลุ่มมัลติแคสต์มีค่าน้อยที่สุด ซึ่งเงื่อนไขนี้สามารถเขียนได้เช่น
 เดียวกับสมการที่ 3.1

4.2 วิเคราะห์ผลกระทบของค่า $\Delta t_{dep}(z)$ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์

ในหัวข้อนี้จะกล่าวถึงผลกระทบโดยรวมที่มีต่อเส้นทางมัลติแคสต์ เมื่อโหนดสมาชิกมี
 ความเป็นไปได้ที่จะออกจากกลุ่มมัลติแคสต์เร็วหรือช้ากว่าเวลาของการส่งวนทรัพยากรล่วงหน้า

แจ้งไว้ตอนต้นซึ่งเป็นกรณีที่ $\Delta t_{dep}(z)$ มีค่าไม่เท่ากับ 0 รูปที่ 4.1 แสดงถึงการเปลี่ยนแปลงที่เกิดขึ้นกับค่าคอสต์ของเส้นทางมัลติแคสต์ที่สร้างจากอัลกอริทึมแบบโมดิไฟด์กรีดี โดยค่าที่แสดงบนแกน y คือค่าคอสต์ที่ถูกละค่า (Normalized) ด้วยค่าคอสต์ของอัลกอริทึมเคเอ็มบีในเงื่อนไขที่โหนดสมาชิกทุกโหนดออกจากกลุ่มมัลติแคสต์ตรงตามเวลาของการส่งวนทรัพยากรล่วงหน้า เมื่อทำการเปลี่ยนแปลงขนาดของกลุ่มมัลติแคสต์ ภายใต้เงื่อนไขจำนวนของโหนดสมาชิกไม่เกินร้อยละ 50 มีโอกาสที่จะออกจากกลุ่มมัลติแคสต์เร็วหรือช้ากว่าค่า $t_{dep}(z)$ ที่แจ้งเข้ามา โดยความน่าจะเป็นที่ค่า $\Delta t_{dep}(z)$ จะมีค่ามากหรือน้อยกว่า 0 จะมีค่าเท่ากันคือ 0.5 และมีค่าเท่ากับร้อยละ 30 ของระยะเวลาในการเข้าร่วมกลุ่มมัลติแคสต์ของโหนดสมาชิก z ใดๆ



รูปที่ 4.1 ผลกระทบของค่า $\Delta t_{dep}(z)$ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์

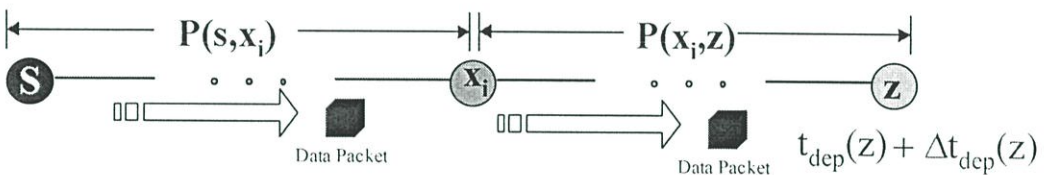
จากรูปที่ 4.1 พบว่าค่าคอสต์โดยรวมของเส้นทางมัลติแคสต์จะมีค่าสูงขึ้นเมื่อโหนดสมาชิกออกจากกลุ่มมัลติแคสต์ที่ไม่ตรงตามเวลา $t_{dep}(z)$ ที่แจ้งไว้ในช่วงเริ่มต้นเข้าร่วมกลุ่มมัลติแคสต์ ซึ่งการเปลี่ยนแปลงของค่าคอสต์เมื่อโหนดสมาชิกออกจากกลุ่มมัลติแคสต์ที่ไม่ตรงเวลาดังกล่าวสามารถวิเคราะห์แยกได้ตามกรณีดังนี้

4.2.1 ผลต่อค่าคอสต์ของเส้นทางมัลติแคสต์เมื่อ $\Delta t_{dep}(z) > 0$

เนื่องจากเส้นทางมัลติแคสต์จะมีลักษณะเป็นเส้นทางต้นไม้ โดยมีโหนดต้นทางเป็นรากของเส้นทางต้นไม้ ดังนั้นเส้นทางจากโหนดต้นทางไปยังโหนดสมาชิกใดๆ ของกลุ่มมัลติแคสต์จะประกอบด้วยเส้นทาง 2 ช่วงจากรูปที่ 4.2 ช่วงแรกหรือช่วงเส้นทาง $P(s, x_i)$ เป็นเส้นทางที่ลิงค์ต่างๆ

จะถูกใช้งานร่วมกับโหนดสมาชิกโหนดอื่นๆ ในเส้นทางมัลติแคสต์ และในช่วงหลังหรือช่วงเส้นทาง $P(x_i, z)$ ซึ่งประกอบด้วยลิงค์ที่ถูกใช้งานโดยโหนด z เพียงโหนดเดียว

กรณีที่โหนดสมาชิก z ออกจากกลุ่มมัลติแคสต์ช้ากว่าเวลาที่แจ้งไว้ล่วงหน้า $t_{dep}(z)$ ด้วยเวลา $\Delta t_{dep}(z)$ เมื่อไม่อนุญาตให้ทำการเรียงเรียงเส้นทางใหม่ ค่าคอสต์ของเส้นทางตั้งแต่โหนดต้นทาง s มายังโหนดสมาชิก z จึงมีค่าเพิ่มขึ้น โดยเมื่อพิจารณาในช่วง $P(s, x_i)$ ลิงค์ทุกลิงค์ในเส้นทาง $P(s, x_i)$ ที่มีค่า t_{out} น้อยกว่า $t_{dep}(z) + \Delta t_{dep}(z)$ จะต้องถูกขยายเวลาการใช้งานจนถึงเวลา $t_{dep}(z) + \Delta t_{dep}(z)$ และเมื่อพิจารณาเส้นทางในช่วง $P(x_i, z)$ ลิงค์ทุกลิงค์ในเส้นทาง $P(x_i, z)$ จะถูกขยายเวลาการใช้งานตั้งแต่เวลา t_{st} จนถึงเวลา $t_{dep}(z) + \Delta t_{dep}(z)$ เช่นเดียวกันดังรูปที่ 4.2 ด้วยสาเหตุดังกล่าว เมื่อโหนดสมาชิกออกจากกลุ่มมัลติแคสต์ช้ากว่าเวลาที่แจ้งไว้จึงส่งผลให้ค่าคอสต์รวมของเส้นทางมัลติแคสต์มีค่าเพิ่มมากขึ้น



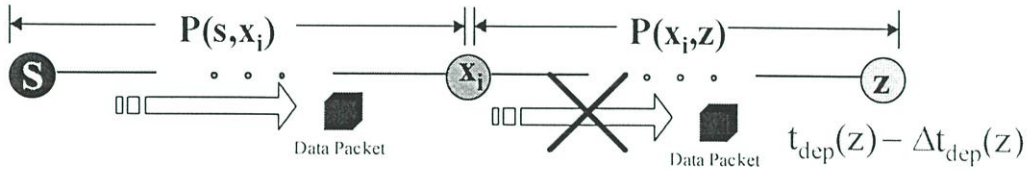
รูปที่ 4.2 โหนดสมาชิกออกจากกลุ่มมัลติแคสต์ล่าช้ากว่ากำหนดหรือ $\Delta t_{dep}(z) > 0$

4.2.2 ผลต่อค่าคอสต์ของเส้นทางมัลติแคสต์เมื่อ $\Delta t_{dep}(z) < 0$

กรณีที่โหนดสมาชิก z ออกจากกลุ่มมัลติแคสต์เร็วกว่าเวลาที่แจ้งไว้ล่วงหน้า $t_{dep}(z)$ ด้วยเวลา $\Delta t_{dep}(z)$ เมื่อไม่อนุญาตให้ทำการเรียงเรียงเส้นทางใหม่ ค่าคอสต์ของเส้นทางตั้งแต่โหนดต้นทาง s จนถึงโหนดสมาชิก z จะมีค่าลดลง เนื่องจากในช่วงของเส้นทาง $P(x_i, z)$ ลิงค์ทุกลิงค์ในเส้นทาง $P(x_i, z)$ จะถูกลดเวลาการใช้งานลง ตั้งแต่ เวลา t_{st} จนถึงเวลา $t_{dep}(z) - \Delta t_{dep}(z)$ แต่เมื่อพิจารณาเส้นทางในช่วง $P(s, x_i)$ ลิงค์ใดๆ ซึ่งอยู่ในเส้นทาง $P(s, x_i)$ ไม่จำเป็นจะต้องถูกลดเวลาการใช้งาน (t_{out}) ลงเหลือ $t_{dep}(z) - \Delta t_{dep}(z)$ เนื่องจากลิงค์เหล่านี้เป็นลิงค์ที่ถูกใช้งานร่วมกันระหว่างโหนดสมาชิกต่างๆ ในกลุ่มมัลติแคสต์ ดังแสดงในรูปที่ 4.3

จากการวิเคราะห์การเปลี่ยนแปลงของค่าคอสต์ของเส้นทางมัลติแคสต์ทั้งในหัวข้อที่ 4.2.1 และ 4.2.2 พบว่าค่าคอสต์โดยรวมของเส้นทางมัลติแคสต์จะมีค่าเพิ่มมากขึ้นเมื่อโหนดสมาชิกออกจากกลุ่มมัลติแคสต์ไม่ตรงตามเวลาที่แจ้งไว้ถึงแม้ว่าความน่าจะเป็นที่โหนดสมาชิกจะออกจากกลุ่มมัลติแคสต์ช้าและเร็วกว่าเวลาที่แจ้งไว้ล่วงหน้าจะมีค่าเท่ากัน เนื่องจากในกรณีที่ $\Delta t_{dep}(z)$ มีค่ามากกว่า 0 ค่าคอสต์ของเส้นทางจะเพิ่มขึ้นตลอดช่วงของเส้นทางตั้งแต่โหนดต้นทาง s มายังโหนดสมาชิก z เนื่องจากมีการขยายระยะเวลาการใช้งานเส้นทางจนถึงเวลา $t_{dep}(z) + \Delta t_{dep}(z)$ แต่ในกรณีที่

$\Delta t_{dep}(z)$ มีค่าน้อยกว่า 0 ค่าคอสม์ของเส้นทางจะลดลงเฉพาะในช่วง $P(x_i, z)$ เนื่องจากลิงค์ซึ่งอยู่ในเส้นทาง $P(s, x_i)$ ไม่จำเป็นจะต้องถูกลดเวลาการใช้งานลงเหลือ $t_{dep}(z) - \Delta t_{dep}(z)$ เนื่องจากเป็นลิงค์ที่ยังคงถูกใช้งานโดยโหนดสมาชิกอื่นๆ ในเส้นทางมัลติแคสต์



รูปที่ 4.3 โหนดสมาชิกออกจากกลุ่มมัลติแคสต์เร็วกว่ากำหนดหรือ $\Delta t_{dep}(z) < 0$

4.3 แนวทางการพัฒนาอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่

งานวิจัยในด้านการสร้างเส้นทางมัลติแคสต์ในสถานะแวดล้อมแบบสงวนทรัพยากรล่วงหน้าสำหรับกลุ่มสมาชิกแบบไดนามิกถูกนำเสนอเป็นครั้งแรกในงานวิจัย [14] โดยมีรากฐานมาจากงานวิจัยในด้านการหาเส้นทางยูนิแคสต์ภายใต้สถานะแวดล้อมแบบสงวนทรัพยากรล่วงหน้า [9] และตั้งสมมติฐานสำหรับแต่ละโหนดสมาชิกของกลุ่มมัลติแคสต์ว่าข้อมูลของการสงวนทรัพยากรล่วงหน้าที่ถูกส่งจากโหนดสมาชิกของกลุ่มมัลติแคสต์มายังโหนดต้นทางของกลุ่มมัลติแคสต์เพื่อใช้ในการสร้างเส้นทางมัลติแคสต์นั้นมีความถูกต้องทั้งหมด ดังนั้นเมื่อพิจารณาขั้นตอนการทำงานของอัลกอริทึมโมดิฟายด์กรีดีจากสมการที่ 3.2 ถึง 3.4 ในบทที่ 3 จะพบว่า โหนดต้นทางของกลุ่มมัลติแคสต์จะใช้ค่า t_{dep} ของโหนดสมาชิกใหม่ z ของกลุ่มมัลติแคสต์เป็นกรอบในการพิจารณาลิงค์อื่นๆ ที่อยู่ในเส้นทางมัลติแคสต์ เมื่อลิงค์ใดๆ ที่อยู่ในเส้นทางมัลติแคสต์มีค่า t_{out} มากกว่าหรือเท่ากับ t_{dep} ของโหนดสมาชิกใหม่ z ลิงค์เหล่านี้จะถูกพิจารณาว่ามีค่าคอสม์ที่เสมือนเท่ากับ 0 เนื่องจากสามารถใช้ในการสร้างเส้นทางเพื่อส่งข้อมูลมัลติแคสต์ให้กับโหนด z ได้โดยไม่เสียค่าคอสม์ของเส้นทางมัลติแคสต์เพิ่มเติม

เนื่องจากในเหตุการณ์จริงในระบบเครือข่ายย่อมมีความเป็นไปได้ที่โหนดสมาชิกของกลุ่มมัลติแคสต์อาจจะออกจากกลุ่มมัลติแคสต์เร็วหรือช้ากว่าเวลาของการสงวนทรัพยากรที่โหนดสมาชิกได้แจ้งเข้ามายังโหนดต้นทางของกลุ่มมัลติแคสต์ล่วงหน้า [15] ทำให้เส้นทางมัลติแคสต์ที่ถูกสร้างขึ้นมาโดยสมมติฐานที่ว่าข้อมูลการสงวนทรัพยากรของโหนดสมาชิกมีความถูกต้องทั้งหมดอาจมีความผิดพลาดเกิดขึ้นได้และส่งผลให้ค่าคอสม์รวมของเส้นทางมัลติแคสต์มีค่าสูงขึ้น ซึ่งหมายถึงปริมาณทรัพยากรในระบบเครือข่ายที่ถูกใช้ไปมากขึ้นเช่นกัน ดังนั้นในวิทยานิพนธ์ฉบับนี้จึงได้นำเสนออัลกอริทึมโมดิฟายด์กรีดีแบบใหม่ซึ่งให้ประสิทธิภาพที่ดีขึ้นเมื่อโหนดสมาชิกออกจากกลุ่มมัลติแคสต์ไม่ตรงตามเวลาที่แจ้งไว้ล่วงหน้า อัลกอริทึมโมดิฟายด์กรีดีแบบใหม่มีพื้นฐาน

การทำงานที่คล้ายกับอัลกอริทึม โมดิฟายด์กรีดีแบบเดิมคือเน้นให้เกิดการใช้ทรัพยากรในระบบเครือข่ายระหว่างโหนดสมาชิกของกลุ่มมัลติแคสต์ให้มากที่สุด โดยอาศัยประโยชน์จากข้อมูลการส่งวนทรัพยากรล่วงหน้าแต่ได้มีการพัฒนาแนวคิดด้านการกำหนดค่ากรอบเวลา t_{ref} ที่แตกต่างไปจากอัลกอริทึมแบบเดิมดังนี้

1. เนื่องจากโหนดสมาชิก z มีความเป็นไปได้ที่จะออกจากกลุ่มมัลติแคสต์ไม่ตรงตามเวลา $t_{dep}(z)$ ที่ได้แจ้งไว้ล่วงหน้า ความคลาดเคลื่อนนี้ส่งผลให้ค่าคอสต์ของเส้นทางมัลติแคสต์มีค่าเพิ่มขึ้น โดยเฉพาะอย่างยิ่งในกรณีที่ $\Delta t_{dep}(z) > 0$ ดังนั้นกรอบเวลา t_{ref} ที่ใช้ในการพิจารณาถึงค่างานๆ เพื่อการคำนวณเส้นทางมัลติแคสต์ควรมีความยืดหยุ่นมากขึ้นเพื่อให้ครอบคลุมความคลาดเคลื่อนที่อาจเกิดขึ้นดังกล่าวได้ ดังนั้นในสมมติฐานเบื้องต้นของการกำหนดกรอบเวลาสำหรับการพิจารณาถึงค่างานใดๆ ในเส้นทางมัลติแคสต์ควรมีค่าสูงกว่าค่า $t_{dep}(z)$ ซึ่งสามารถเขียนได้ตามสมการที่ 4.2

$$t_{ref} > t_{dep}(z) \quad (4.2)$$

2. ในการระบุค่าที่แน่นอนของกรอบเวลา t_{ref} ซึ่งต่อเนืองมาจากสมมติฐานเบื้องต้นในข้อที่ 1 เนื่องจากความคลาดเคลื่อนของข้อมูลด้านเวลาของโหนดสมาชิกหรือ $\Delta t_{dep}(z)$ ไม่สามารถถูกประเมินได้ล่วงหน้าโดยโหนดต้นทางของกลุ่มมัลติแคสต์ ดังนั้นการกำหนดค่ากรอบของเวลาจึงอาศัยข้อมูลทั้งหมดเท่าที่โหนดต้นทางของกลุ่มมัลติแคสต์มีอยู่เท่านั้น การกำหนดค่าที่แน่นอนของกรอบเวลา t_{ref} นั้นมีปัจจัยการพิจารณาที่สำคัญคือกรอบเวลาที่จะนำมาใช้ไม่ควรที่จะมีค่าที่สูงกว่าค่า $t_{dep}(z)$ จนเกินไป ถึงแม้ว่าการกำหนดกรอบเวลาที่มีช่วงกว้างจะทำให้สามารถครอบคลุมค่า $t_{dep}(z)$ ที่มีโอกาสแตกต่างไปจากค่าที่เป็นจริงได้มากขึ้น แต่การกำหนดกรอบเวลาที่กว้างเกินไปจะทำให้หน่วยสำคัญของค่าคอสต์เสมือนที่คำนวณได้นั้นมีค่าลดน้อยลงซึ่งส่งผลต่อความน่าเชื่อถือของเส้นทางที่ถูกเลือกใช้เช่นกัน ดังนั้นในวิทยานิพนธ์ฉบับนี้จึงเสนอให้ทำการกำหนดค่าของกรอบของเวลา t_{ref} ด้วยค่ากลางของค่า $t_{out}(l_m)$ ซึ่งได้มาจากการคำนวณค่าเฉลี่ยแบบถ่วงน้ำหนักของค่า $t_{out}(l_m)$ เมื่อ l_m เป็นลิงค์ใดๆ ในที่อยู่ในเส้นทางมัลติแคสต์ซึ่งมีค่า $t_{out}(l_m) > t_{dep}(z)$ การเฉลี่ยแบบถ่วงน้ำหนักเพื่อหาค่า t_{ref} นั้นสามารถทำได้โดยอาศัยสมการที่ 4.3

$$t_{ref} = \frac{1}{N} \sum_i [\tau_i \times n(\tau_i)] \quad (4.3)$$

เมื่อ

N เป็นจำนวนของลิงค์ l_m ทั้งหมดที่อยู่ในเส้นทางมัลติแคสต์ซึ่งมีค่า $t_{out}(l_m) > t_{dep}(z)$

- τ_i เป็นค่า t_{out} ของลิงค์ l_m ทั้งหมดที่อยู่ในเส้นทางมัลติแคสต์ที่แตกต่างกันจำนวน i ค่าและมีค่ามากกว่า $t_{dep}(z)$
- $n(\tau_i)$ เป็นค่าความถี่จำนวนลิงค์ของ τ_i

สำหรับในกรณีที่ไม่มีลิงค์ใดในเส้นทางมัลติแคสต์ที่มีค่า $t_{out}(l_m) > t_{dep}(z)$ ค่ากรอบเวลา t_{rcf} จะถูกกำหนดให้มีค่าเท่ากับ $t_{dep}(z)$ เช่นเดียวกับในอัลกอริทึม โมดิฟายด์กรีดี

ในอัลกอริทึม โมดิฟายด์กรีดีแบบใหม่จะนำค่ากรอบเวลาใหม่ที่คำนวณได้จากสมการ 4.2 และ 4.3 มาประยุกต์ใช้ในการคำนวณหาเส้นทางมัลติแคสต์ซึ่งมีแนวทางเดียวกับการคำนวณหาเส้นทางมัลติแคสต์ของอัลกอริทึม โมดิฟายด์กรีดีในแบบเดิมที่กล่าวถึงไปแล้วในหัวข้อที่ 3.3 โดยที่รายละเอียดการทำงานของอัลกอริทึม โมดิฟายด์กรีดีแบบใหม่จะได้กล่าวถึงในหัวข้อต่อไป

4.4. อัลกอริทึมโมดิฟายด์กรีดีแบบใหม่

ในหัวข้อนี้จะกล่าวถึงรายละเอียดของอัลกอริทึม โมดิฟายด์กรีดีแบบใหม่ซึ่งมีหลักการทำงานที่คล้ายคลึงกับอัลกอริทึม โมดิฟายด์กรีดีแบบเดิม แต่ได้ทำการพัฒนาเพิ่มเติมในส่วนของการคำนวณค่ากรอบเวลา t_{rcf} ให้มีประสิทธิภาพมากยิ่งขึ้นในการรองรับเหตุการณ์ที่โหนดสมาชิกมีโอกาสที่จะออกจากกลุ่มมัลติแคสต์ไม่ตรงตามเวลาที่แจ้งไว้ล่วงหน้าดังที่กล่าวถึงในหัวข้อที่ 4.3

อัลกอริทึมการสร้างเส้นทางมัลติแคสต์โมดิฟายด์กรีดีแบบใหม่ประกอบด้วยอัลกอริทึม 2 ส่วนหลักคืออัลกอริทึมการเพิ่มโหนดสมาชิกและอัลกอริทึมการลบโหนดสมาชิก รายละเอียดการทำงานของแต่ละอัลกอริทึมเป็นดังนี้

4.4.1 อัลกอริทึมการเพิ่มโหนดสมาชิก

เมื่อมีโหนด z ใดๆ ในระบบเครือข่ายต้องการเข้าร่วมเป็นสมาชิกของกลุ่มมัลติแคสต์ที่มีโหนด s เป็นโหนดต้นทาง โหนด z ดังกล่าวจะต้องส่งข้อมูลของเวลาที่ออกจากกลุ่มมัลติแคสต์ $t_{dep}(z)$ ให้โหนดต้นทาง s ทราบ หลังจากนั้นโหนดต้นทาง s จะทำการเลือกโหนด x_i ใดๆ ในเส้นทางมัลติแคสต์ที่ทำให้ค่าคอสต์เหมือนของเส้นทางจากโหนด s ไปยังโหนด z หรือ $W(s,z)$ มีค่าน้อยที่สุดซึ่งค่าคอสต์เหมือน $W(s,z)$ จะเกิดจากผลรวมของค่าคอสต์เหมือน $W(s,x_i)$ และค่าคอสต์เหมือน $W(x_i,z)$ เช่นเดียวกับในอัลกอริทึม โมดิฟายด์กรีดี และโหนดต้นทาง s จะเลือกใช้เส้นทางดังกล่าวในการเชื่อมต่อโหนด z เข้ากับเส้นทางมัลติแคสต์ที่มีอยู่ในปัจจุบัน โดยก่อนที่จะทำการคำนวณค่าคอสต์เหมือน $W(s,x_i)$ และ $W(x_i,z)$ โหนดต้นทางของกลุ่มมัลติแคสต์จะทำการกำหนด

กรอบเวลาเพื่อใช้ในการพิจารณาลิงค์ใดๆ ที่อยู่ในในเส้นทางมัลติแอสท์โดยใช้สมการที่ 4.2 และ 4.3 ซึ่งได้กล่าวถึงไปแล้วในหัวข้อที่ 4.3

ในกรณีของการคำนวณค่าคอสต์เสมือน $W(s, x_i)$ เมื่อกำหนดให้ l_m เป็นลิงค์ใดๆ บนเส้นทางจากโหนดต้นทาง s ไปยังโหนด x_i ใดๆ ซึ่งอยู่ในเส้นทางมัลติแอสท์และมีค่า $t_{out}(l_m) < t_{ref}$ ค่าคอสต์เสมือน $W(s, x_i)$ จะมีค่าตามสมการ 4.4 และลิงค์ซึ่งมีค่า $t_{out}(l_m) > t_{ref}$ จะถูกพิจารณาว่ามีค่าคอสต์เป็นศูนย์

$$W(s, x_i) = \sum_{l_m \in P(s, x_i)} \{ [t_{ref} - t_{out}(l_m)] \times C(l_m) \} \quad (4.4)$$

เมื่อ

$t_{out}(l_m)$ เป็นเวลาที่ลิงค์ l_m จะถูกยกเลิกการใช้งานในเส้นทางมัลติแอสท์

$C(l_m)$ เป็นค่าคอสต์ของลิงค์ l_m ใดๆ

และสำหรับกรณีของค่า $W(x_i, z)$ เมื่อกำหนดให้ l_n เป็นลิงค์ใดๆ ที่อยู่ในเส้นทางเส้นทางจากโหนด x_i ไปยังโหนดสมาชิกใหม่ z ค่าคอสต์เสมือน $W(x_i, z)$ จะคำนวณได้ตามสมการที่ 4.5

$$W(x_i, z) = \sum_{l_n \in P(x_i, z)} \{ [t_{ref} - t_{st}(z)] \times C(l_n) \} \quad (4.5)$$

เมื่อ

$t_{st}(z)$ เป็นเวลาที่โหนดสมาชิกใหม่ z เริ่มต้นเข้าร่วมกลุ่มมัลติแอสท์

$C(l_n)$ เป็นค่าคอสต์ของลิงค์ l_n ใดๆ

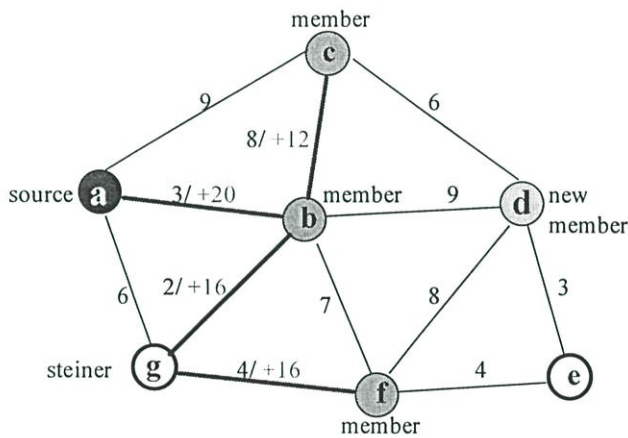
จากสมการที่ 4.4 และ 4.5 จะพบว่าอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่มีหลักการทำงานที่คล้ายคลึงกับอัลกอริทึมโมดิฟายด์กรีดีแบบเดิมแต่มีการเปลี่ยนแปลงค่ากรอบเวลาใหม่เพื่อประโยชน์ในการสงวนทรัพยากรที่มากขึ้นสำหรับแต่ละโหนดสมาชิกส่งผลให้ค่าคอสต์รวมของเส้นทางมัลติแอสท์มีการเปลี่ยนแปลงที่น้อยกว่ากรณีที่โหนดสมาชิกมีความเป็นไปได้ที่จะออกจากกลุ่มมัลติแอสท์ช้ากว่าเวลาที่แจ้งไว้

จากรูปที่ 4.4 จะแสดงถึงตัวอย่างการคำนวณหาเส้นทางมัลติแอสท์ด้วยการใช้อัลกอริทึมโมดิฟายด์กรีดีแบบใหม่ จากโหนดสมาชิกใหม่ d ไปยังเส้นทางมัลติแอสท์ที่ประกอบด้วยโหนด a, b, c และ f โดยมีโหนด a เป็นโหนดต้นทางของกลุ่มมัลติแอสท์และค่าคอสต์และค่า t_{out} ของแต่ละลิงค์จะแสดงด้วยตัวเลขที่กำกับไว้ในรูปแบบ ค่าคอสต์/ค่า t_{out} โดยลิงค์ที่อยู่ในเส้นทางมัลติแอสท์จะ

ถูกแสดงด้วยเส้นหนา ตารางเวลาของการส่งมอบพัสดุสำหรับแต่ละโหนดสมาชิกของกลุ่มมัลติแคสต์เป็นดังตารางที่ 4.1 กำหนดให้โหนดต้นทางของกลุ่มมัลติแคสต์เริ่มต้นส่งข้อมูลในเวลาเท่ากับ 0 และเวลาที่ระบุในตารางเป็นจำนวนหน่วยของเวลาที่สัมพันธ์กับเวลาที่โหนดต้นทางเริ่มต้นส่งข้อมูล

ตารางที่ 4.1 เวลาการเข้าร่วมและออกจากกลุ่มมัลติแคสต์ของโหนดสมาชิก

โหนด	เวลาที่เข้าร่วมกลุ่มมัลติแคสต์ (t_{arr})	เวลาที่ออกจากกลุ่มมัลติแคสต์ (t_{dep})
b	0	+20
c	+2	+12
f	+1	+16
d	+3	+12

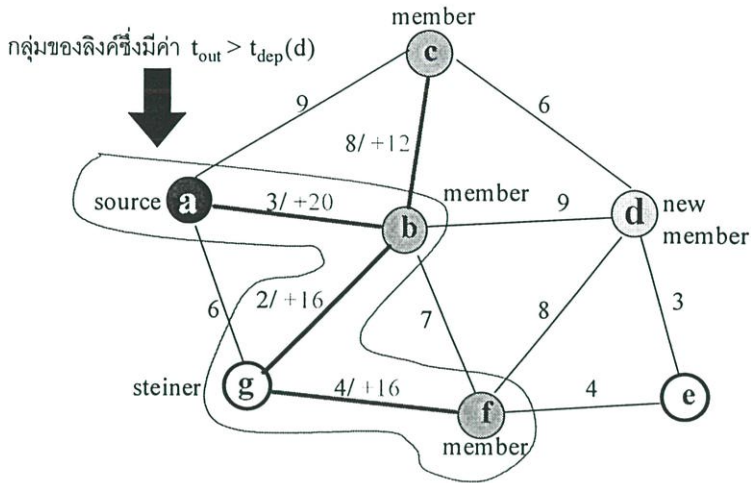


รูปที่ 4.4 ตัวอย่างการคำนวณหาเส้นทางมัลติแคสต์ด้วยอัลกอริทึมแบบใหม่

ก่อนที่จะเริ่มต้นคำนวณหาค่าคอสต์เสมือน $W(s,z)$ ของเส้นทางใดๆ ที่จะใช้ในการเชื่อมต่อจำเป็นที่จะต้องคำนวณค่ากรอบเวลา t_{cr} ก่อนเพื่อใช้เป็นเกณฑ์ในการพิจารณาถึงค่าใดๆ ที่ใช้งานอยู่ในเส้นทางมัลติแคสต์ โดยที่ค่า t_{cr} สามารถคำนวณได้โดยอาศัยสมการที่ 4.2 และ 4.3 จากเส้นทางมัลติแคสต์ในรูปที่ 4.4 สิ่งที่มีค่า t_{out} มากกว่าค่า t_{dep} ของโหนด d จะถูกแสดงในรูปที่ 4.5 ดังนั้นจากรูปที่ 4.5 ค่ากรอบเวลาหรือ t_{cr} จะมีค่าเท่ากับ $(20 \times (1/3)) + (16 \times (2/3)) = 17.33$

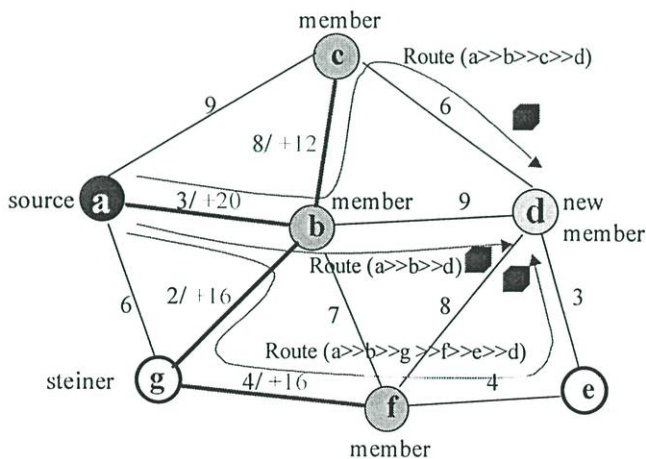
เมื่อสามารถคำนวณค่ากรอบเวลาได้แล้ว จากรูปที่ 4.6 พบว่าเส้นทางที่สามารถใช้ในการเชื่อมต่อโหนดสมาชิกใหม่ d กับเส้นทางมัลติแคสต์ประกอบด้วยเส้นทาง abcd, abd, และ abgfed ค่าคอสต์เสมือน $W(s,z)$ ของแต่ละเส้นทางสามารถคำนวณได้โดยสมการที่ 4.4 และ 4.5

จากรูปที่ 4.6 ค่าคอสต์เสมือนของเส้นทาง abcd หรือ $W(a,d)$ มีค่าเท่ากับ $W(a,c) + W(c,d)$ และเมื่อแทนค่าพารามิเตอร์ต่างๆ ของแต่ละลิงค์ในสมการที่ 4.4 และ 4.5 จะพบว่าค่าคอสต์เสมือนของเส้นทาง abcd มีค่าเท่ากับ $0 + [(17.33-12) \times 8] + [(17.33 - 3) \times 6] = 128.2$



รูปที่ 4.5 ตัวอย่างการคำนวณค่ากรอบเวลา

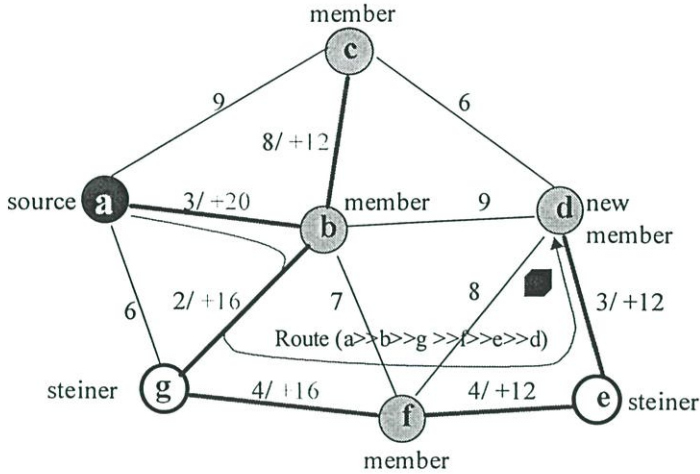
กรณีของเส้นทาง abd ค่าคอสต์เสมือน $W(a,d)$ จะมีค่าเท่ากับ $W(a,b) + W(b,d)$ และเมื่อแทนค่าพารามิเตอร์ต่างๆ ของแต่ละลิงค์ในสมการที่ 4.4 และ 4.5 จะพบว่าค่าคอสต์เสมือนของเส้นทาง abd มีค่าเท่ากับ $0 + [(17.33 - 3) \times 9] = 128.7$



รูปที่ 4.6 เส้นทาง abcd, abd, และ abgfd บนระบบเครือข่าย

ในกรณีของเส้นทาง abgfd ก็เช่นเดียวกันค่าคอสต์เสมือน $W(a,d)$ จะมีค่าเท่ากับ $W(a,b) + W(b,g) + W(g,f) + W(f,d)$ โดยเมื่อแทนค่าพารามิเตอร์ต่างๆ ของแต่ละลิงค์ในสมการที่ 4.4 และ 4.5 พบว่าค่าคอสต์เสมือนของเส้นทาง abgfd มีค่าเท่ากับ $0 + [(17.33-12) \times 2] + [(17.33 - 12) \times 4] +$

$[(17.33 - 3) \times 7] = 107.9$ และเนื่องจากเส้นทางนี้มีค่าคอสต์เสมือน $W(a,d)$ ที่น้อยที่สุด ดังนั้นจึงถูกใช้เพื่อการเชื่อมต่อโหนด d เข้ากับเส้นทางมัลติแคสต์เดิม โดยเส้นทางมัลติแคสต์ใหม่ที่รวมโหนด d เข้าเป็นโหนดสมาชิกแล้วจะเป็นดังรูป 4.7



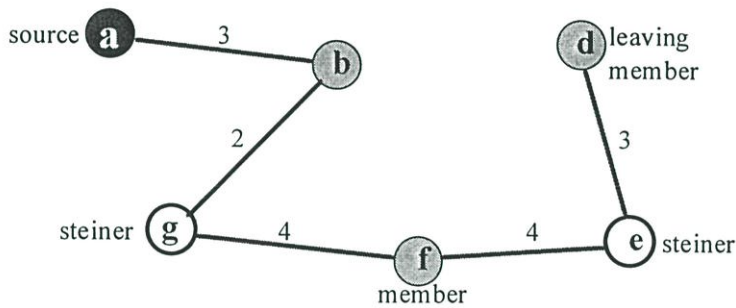
รูปที่ 4.7 เส้นทางมัลติแคสต์หลังจากเชื่อมต่อกับโหนดสมาชิกใหม่

จากตัวอย่างดังกล่าว เมื่อทำการเลือกเส้นทางโดยใช้อัลกอริทึมโมดิไฟด์กรีดีแบบเดิมจะพบว่าเส้นทางที่มีค่าคอสต์เสมือนน้อยที่สุดคือเส้นทาง $abcd$ ซึ่งการเลือกใช้เส้นทาง $abcd$ เพื่อการเชื่อมต่อจะมีค่าคอสต์รวมของเส้นทางมัลติแคสต์ที่น้อยกว่าการใช้เส้นทาง $abgfed$ เมื่อโหนดสมาชิกทุกโหนดออกจากกลุ่มมัลติแคสต์ตรงตามเวลา t_{dep} ที่แจ้งไว้แก่โหนดต้นทางในตอนแรก แต่เมื่อเกิดเหตุการณ์ที่โหนดสมาชิกบางโหนดออกจากกลุ่มมัลติแคสต์ล่าช้ากว่ากำหนด ($\Delta t_{dep}(z_i) > 0$) การเลือกใช้เส้นทาง $abcd$ อาจทำให้ค่าคอสต์รวมของเส้นทางมัลติแคสต์เพิ่มมากขึ้น ตัวอย่างเช่น ในกรณีที่โหนดสมาชิกใหม่หรือโหนด d ออกจากกลุ่มมัลติแคสต์ล่าช้ากว่ากำหนดด้วยค่า $\Delta t_{dep}(d) = +4$ (ค่า $t_{dep}(d) + \Delta t_{dep}(d)$ จะเท่ากับ $+16$) สำหรับการเลือกใช้เส้นทาง $abgfed$ จะต้องเสียค่าคอสต์ของเส้นทางมัลติแคสต์รวมทั้งหมดเท่ากับ 321 หน่วย แต่ในกรณีของเส้นทาง $abcd$ จะต้องเสียค่าคอสต์ของเส้นทางมัลติแคสต์รวมทั้งหมดเท่ากับ 340 หน่วย เป็นต้น

4.4.2 อัลกอริทึมการลบโหนดสมาชิก

ในหัวข้อนี้จะกล่าวถึงอัลกอริทึมที่ใช้ในการลบโหนดสมาชิกของกลุ่มมัลติแคสต์ที่ต้องการแยกตัวออกจากกลุ่มมัลติแคสต์ซึ่งอัลกอริทึมนี้เป็นอัลกอริทึมที่สามารถใช้ได้กับอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ทุกรูปแบบที่ได้กล่าวถึงในวิทยานิพนธ์ฉบับนี้ และเพื่อความสะดวกในการอธิบายการทำงานของอัลกอริทึมจึงให้คำจำกัดความกับโหนดและลิงค์ในเส้นทางมัลติแคสต์เพิ่มเติมดังนี้

จากรูปที่ 4.8 แสดงตัวอย่างของเส้นทางมัลติแคสต์ที่มีโหนด a, b, d และ f เป็นสมาชิกของกลุ่มมัลติแคสต์โดยที่โหนด a เป็นโหนดต้นทางของกลุ่มมัลติแคสต์ โหนด d จะถูกให้คำจำกัดความว่าเป็น “โหนดดาวน์สตรีม (Downstream)” ของโหนด e เนื่องจากโหนด e มีหน้าที่ในการส่งแพคเกจจากโหนดต้นทางให้แก่โหนด d ผ่านลิงค์ l_{ed} ซึ่งลิงค์ l_{ed} จะถูกให้คำจำกัดความว่าเป็น “ลิงค์อัพสตรีม” (Upstream) ของโหนด d และโหนด c จะถูกให้คำจำกัดความว่าเป็น “โหนดอัพสตรีม” ของโหนด d เมื่อต้องการตัดโหนดสมาชิก x ใดๆ ออกจากกลุ่มมัลติแคสต์ นอกจากจะทำการตัดโหนดดังกล่าวออกจากแล้วจะต้องทำการตัดลิงค์บางส่วนของที่ไม่ได้ใช้งานในเส้นทางมัลติแคสต์อีกด้วย อัลกอริทึมของการตัดโหนดสมาชิก x ใดๆ ออกจากกลุ่มมัลติแคสต์เป็นดังนี้



รูปที่ 4.8 ตัวอย่างการลบโหนดสมาชิกออกจากเส้นทางมัลติแคสต์

ขั้นตอนที่ 1 ตัด โหนด x ออกจากสมาชิกกลุ่มมัลติแคสต์

ขั้นตอนที่ 2 ถ้าโหนด x มีดาวน์สตรีมโหนดตั้งแต่ 1 โหนดขึ้นไป

จบการทำงาน

ถ้าโหนด x ไม่มีดาวน์สตรีมโหนดอยู่เลย

ทำการลบโหนด x ออกจากเส้นทางมัลติแคสต์

ทำการลบอัพสตรีมลิงค์ของโหนด x ออกจากเส้นทางมัลติแคสต์

ขั้นตอนที่ 3 ทำการตรวจสอบโหนดอัพสตรีมของโหนด x ที่เชื่อมต่อกับลิงค์ที่เพิ่งทำการลบออกไปจากเส้นทางมัลติแคสต์

ถ้าโหนดอัพสตรีมของโหนด x เป็นสมาชิกของกลุ่มมัลติแคสต์

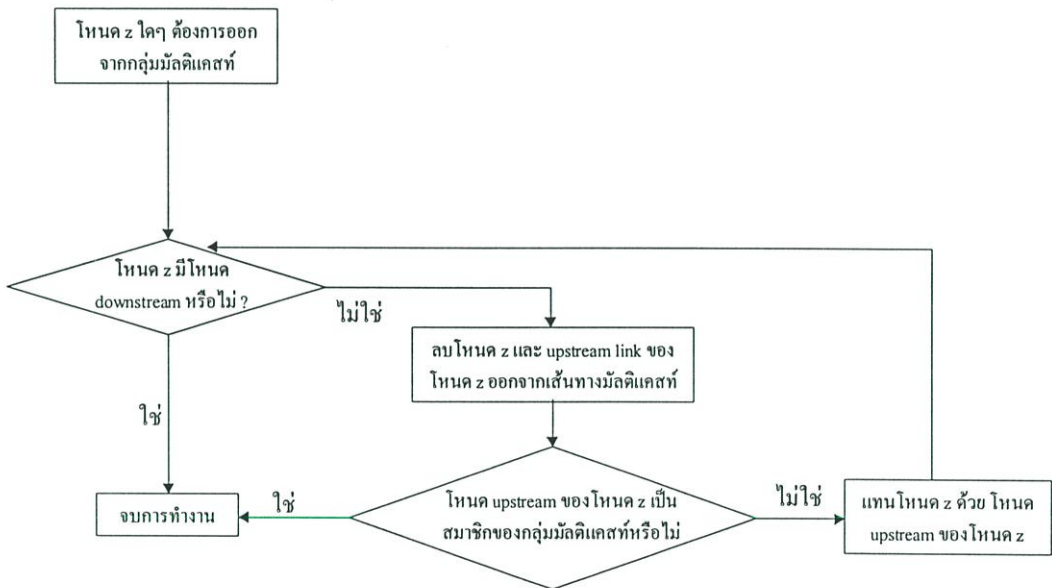
จบการทำงาน

ถ้าโหนดอัพสตรีมของโหนด x ไม่ได้เป็นสมาชิกของกลุ่มมัลติแคสต์

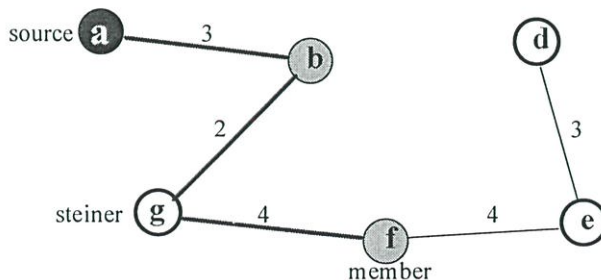
แทนโหนด x ด้วยโหนดอัพสตรีมของโหนด x

กลับไปยังขั้นตอนที่ 2

อัลกอริทึมการลบโหนดสมาชิกสามารถนำมาสร้างเป็นแผนผังการไหลได้ดังรูปที่ 4.9 และจากรูปที่ 4.8 จะแสดงตัวอย่างของการลบโหนด d ออกจากกลุ่มมัลติแคสต์โดยการใช้อัลกอริทึมที่กล่าวถึงข้างต้น เมื่อกลุ่มมัลติแคสต์ประกอบด้วยโหนดสมาชิก a, b, d และ f และมีโหนด a เป็นโหนดต้นทางของกลุ่มมัลติแคสต์ เมื่อโหนด d ต้องการออกจากกลุ่มมัลติแคสต์ ในขั้นตอนแรกให้ทำการลบโหนด d ออกจากกลุ่มมัลติแคสต์และเนื่องจากโหนด d ไม่มีโหนดคาว์นสตรีมใดๆ ดังนั้นจึงสามารถลบถึงคัพสตรีม I_{cd} ออกจากเส้นทางมัลติแคสต์ได้ ขั้นตอนต่อมาเมื่อทำการพิจารณาโหนด e ซึ่งเป็นโหนดอัพสตรีมของโหนด d พบว่าโหนด e ไม่ได้เป็นสมาชิกของกลุ่มมัลติแคสต์และไม่มีโหนดคาว์นสตรีมใดๆ จึงสามารถลบโหนด e รวมถึงถึงคัพสตรีมของโหนด e หรือถึงคัพ I_{ce} ออกจากเส้นทางมัลติแคสต์ได้ ขั้นตอนต่อไปจะเป็นพิจารณาโหนดอัพสตรีมของโหนด e คือโหนด f และเนื่องจากโหนด f เป็นสมาชิกของกลุ่มมัลติแคสต์จึงยุติการทำงานของอัลกอริทึม ดังนั้นเส้นทางมัลติแคสต์หลังจากทำการลบโหนดสมาชิก d ออกไปจะเป็นดังรูปที่ 4.10



รูปที่ 4.9 แผนผังการไหลของอัลกอริทึมการลบโหนดสมาชิก



รูปที่ 4.10 เส้นทางมัลติแคสต์หลังจากทำการลบโหนดสมาชิกออก

4.5 บทสรุป

อัลกอริทึมการสร้างเส้นทางมัลติแคสต์โม่ติฟายด์กรีดีสำหรับกลุ่มสมาชิกแบบไดนามิกภายใต้สถานะแวดล้อมแบบสแกนทรัพยากรล่วงหน้าที่น่าเสนอในงานวิจัย [14] นั้นมีสมมติฐานว่าข้อมูลด้านเวลาของการสแกนทรัพยากรล่วงหน้าที่โหนดสมาชิกของกลุ่มมัลติแคสต์แจ้งเข้ามายังโหนดต้นทางในช่วงเริ่มต้นการเข้าร่วมกลุ่มมัลติแคสต์มีความถูกต้องทั้งหมด แต่เนื่องจากในสถานการณ์จริงของระบบเครือข่ายย่อมมีความเป็นไปได้ที่โหนดสมาชิกของกลุ่มมัลติแคสต์อาจจะออกจากกลุ่มมัลติแคสต์ล่าช้าหรือเร็วกว่าเวลา t_{dep} ที่ได้ทำการแจ้งไว้ล่วงหน้าหรืออาจกล่าวได้ว่าโหนดสมาชิกกลุ่มมัลติแคสต์มีการใช้ทรัพยากรของระบบเครือข่ายในปริมาณที่ผิดจากที่ทำการสแกนทรัพยากรไว้ล่วงหน้า ซึ่งอาจส่งผลให้เส้นทางมัลติแคสต์ที่คำนวณได้มีความผิดพลาดเกิดขึ้นและส่งผลให้ค่าคอสม์รวมของเส้นทางมัลติแคสต์มีค่าสูงขึ้นได้

ดังนั้นในบทนี้จึงได้ทำการกำหนดเงื่อนไขของการสร้างเส้นทางมัลติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกภายใต้สถานะแวดล้อมแบบสแกนทรัพยากรล่วงหน้าแบบใหม่โดยคำนึงถึงความน่าจะเป็นที่โหนดสมาชิกของกลุ่มมัลติแคสต์จะออกจากกลุ่มมัลติแคสต์ไม่ตรงตามเวลาที่แจ้งไว้ในตอนแรก และได้มีการวิเคราะห์ปัญหาดังกล่าวพร้อมทั้งพัฒนาอัลกอริทึมโม่ติฟายด์กรีดีแบบใหม่ซึ่งนำเสนอแนวความคิดการขยายกรอบเวลาที่ใช้ในการพิจารณาสิ่งใด ๆ ซึ่งอยู่ในในเส้นทางมัลติแคสต์ให้มีช่วงที่กว้างขึ้นอย่างมีนัยสำคัญ เพื่อให้อัลกอริทึมโม่ติฟายด์กรีดีสามารถรองรับกับสถานการณ์ดังกล่าวได้อย่างมีประสิทธิภาพมากยิ่งขึ้น

บทที่ 5

รายงานผลการทดลอง

ในบทนี้จะเป็นรายงานผลการทดลองในการศึกษาประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์โม่ติฟายด์กรีดีแบบใหม่ที่ได้นำเสนอในวิทยานิพนธ์นี้โดยการทำซิมูเลชันระบบเครือข่ายและการสร้างเส้นทาง ซึ่งเป็นวิธีที่ใช้อย่างแพร่หลายในงานวิจัยที่เกี่ยวข้องกับอัลกอริทึมสำหรับการสร้างเส้นทางมัลติแคสต์ เนื้อหาในตอนต้นจะอธิบายถึงขั้นตอนของการทำซิมูเลชันรวมถึงการกำหนดค่าพารามิเตอร์ต่างๆ โดยอ้างอิงถึงเงื่อนไขของการสร้างเส้นทางมัลติแคสต์แบบใหม่ที่กล่าวถึงในบทที่ 4 ต่อมาจะเป็นรายงานผลการทดลอง รวมถึงการสรุปและวิเคราะห์ผลการทดลองการทดสอบประสิทธิภาพของอัลกอริทึมโม่ติฟายด์กรีดีแบบใหม่ภายใต้สภาพแวดล้อมแบบต่างๆ ของการเปลี่ยนแปลงพฤติกรรมของโหนดสมาชิกกลุ่มมัลติแคสต์

5.1 ขั้นตอนของการทำซิมูเลชัน

หัวข้อนี้จะกล่าวถึงขั้นตอนของการทำซิมูเลชันซึ่งจะอ้างอิงกับเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ที่กล่าวถึงในบทที่ 4 โดยเริ่มต้นด้วยขั้นตอนการสร้างระบบเครือข่ายแบบสุ่มหรือการกำหนดกราฟ G และการกำหนดเซต C หลังจากนั้นจะกล่าวถึงการกำหนดลำดับของการเข้าร่วมกลุ่มมัลติแคสต์หรือการกำหนดเซต R สุดท้ายจะเป็นการกล่าวถึงปริมาณแบบต่างๆ ที่ใช้ในการชี้วัดประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์ตามลำดับ

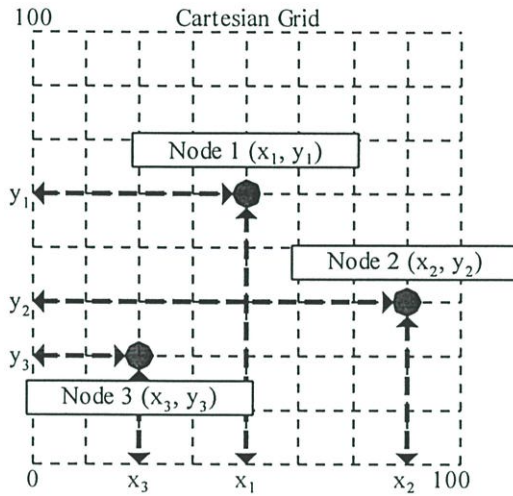
5.1.1 การสร้างระบบเครือข่ายแบบสุ่ม

ประสิทธิภาพของอัลกอริทึมการหาเส้นทางมัลติแคสต์โม่ติฟายด์กรีดีแบบใหม่ที่น่าสนใจในวิทยานิพนธ์นี้จะถูกทดสอบโดยใช้การซิมูเลชันบนโครงข่ายจำลองแบบสุ่มเช่นเดียวกับงานวิจัยอื่นๆ ที่เกี่ยวข้องกับอัลกอริทึมในการสร้างเส้นทางมัลติแคสต์เช่น งานวิจัย [6], [7], [8], [14] และ [15] เป็นต้น การสร้างเครือข่ายแบบสุ่มมีขั้นตอนดังนี้

อ้างอิงจากเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ที่กล่าวถึงในบทที่ 4 ทำการกำหนดกราฟ $G = (V, E)$ เมื่อเซต V เป็นเซตของจำนวนโหนดบนกราฟและสมมูลกับโหนดบนระบบเครือข่ายและเซต E เป็นเซตของการเชื่อมต่อระหว่างโหนดใดๆ บนกราฟซึ่งสมมูลกับลิงค์ระหว่างโหนดบนระบบเครือข่ายและลิงค์ระหว่างโหนดใดๆ ในเครือข่ายมีลักษณะสมมาตรทำให้ค่าคอสต์ของลิงค์ระหว่างใดๆ มีค่าเท่ากันใน 2 ทิศทาง หลังจากนั้นทำการสุ่มค่าพิคัดของโหนดจำนวน n โหนดซึ่งเป็นสมาชิกของเซต V ลงบนพิคัดสี่เหลี่ยมมุมฉากดังรูปที่ 5.1 และความน่าจะเป็นที่จะเกิด

การเชื่อมต่อระหว่างโหนด u และ v ใดๆ ซึ่งเป็นสมาชิกของเซต E จะถูกกำหนดโดยใช้ฟังก์ชันความน่าจะเป็นของการเชื่อมต่อดังสมการที่ 5.1

ในวิทยานิพนธ์นี้ ค่าคอสม์ของลิงค์ในระบบเครือข่ายหรือสมาชิกของเซต C จากเงื่อนไขของการสร้างเส้นทางมัลติแคสต์จะสามารถกำหนดได้โดยการใช้ค่าระยะทางระหว่างโหนดบนโครงข่ายแบบสุ่ม เช่นเดียวกับงานวิจัยอื่นๆ ที่เกี่ยวข้องกับการสร้างเส้นทางมัลติแคสต์



รูปที่ 5.1 ตัวอย่างของระบบเครือข่ายแบบสุ่ม

$$P_e(u, v) = \frac{k\varepsilon}{N} \beta \exp \frac{-d(u, v)}{L\alpha} \quad (5.1)$$

เมื่อ

- $P_e(u, v)$ เป็นความน่าจะเป็นของการเชื่อมต่อระหว่างโหนด u และ v ใดๆ ในกราฟ
- N เป็นจำนวนโหนดในระบบเครือข่าย
- $d(u, v)$ เป็นค่าระยะทางระหว่างโหนดสมาชิก u และ v ใดๆ ในเซต V และเป็นค่าคอสม์ของลิงค์ระหว่างโหนด u และ v ในระบบเครือข่าย
- L เป็นค่าระยะทางที่มากที่สุดระหว่าง 2 โหนดสมาชิกใดๆ ของเซต V
- α เป็นค่าพารามิเตอร์ที่มีค่าอยู่ระหว่าง 0.0-1.0 เมื่อค่า มีค่ามากขึ้นอัตราส่วนความหนาแน่นของการเชื่อมต่อที่มีระยะทางสั้นสัมพันธ์กับการเชื่อมต่อที่มีระยะทางยาวจะลดน้อยลง
- β เป็นค่าพารามิเตอร์ที่มีค่าอยู่ระหว่าง 0.0-1.0 เมื่อค่า มีค่ามากขึ้นจะเป็นการเพิ่มจำนวนของการเชื่อมต่อของแต่ละโหนด
- ε เป็นค่าเฉลี่ยของจำนวนการเชื่อมต่อของโหนดในระบบเครือข่าย

- k เป็นค่าตัวประกอบสเกลซึ่งมีความสัมพันธ์กับค่าเฉลี่ยของระยะทางระหว่าง 2 จุดใดๆ ในเครือข่าย มีหน้าที่ควบคุมให้ค่าเฉลี่ยของจำนวนการเชื่อมต่อของโหนดมีค่าคงที่ตามค่า e

ในวิทยานิพนธ์นี้จะทำการกำหนดค่าพารามิเตอร์ต่างๆ โดยอ้างอิงจากงานวิจัย [16] เช่นเดียวกับงานวิจัยอื่นๆ ที่เกี่ยวข้องกับการสร้างเส้นทางมัลติแคสต์ เนื่องจากงานวิจัย [16] ได้ทำการศึกษาวิจัยในการกำหนดค่าที่เหมาะสมให้กับตัวแปรเหล่านี้ให้มีความสอดคล้องกับระบบเครือข่ายสื่อสารข้อมูลซึ่งมีการใช้งานอยู่จริง โดยอ้างอิงจากแผนที่ทางภูมิศาสตร์ของโหนดหลักในระบบเครือข่ายอินเทอร์เน็ตซึ่งค่าพารามิเตอร์ต่างๆ สามารถแสดงได้ตามตารางที่ 5.1

ตารางที่ 5.1 ค่าพารามิเตอร์ที่ใช้ในการสร้างเครือข่ายแบบสุ่ม

ชนิดของพารามิเตอร์	ค่าของพารามิเตอร์
α	0.25
β	0.2
ϵ	3
N	100
K	25

5.1.2 การกำหนดลำดับการเข้าร่วมกลุ่มมัลติแคสต์ของโหนดสมาชิก

การกำหนดลำดับการเข้าร่วมกลุ่มมัลติแคสต์ของโหนดสมาชิกกลุ่มมัลติแคสต์สามารถทำได้โดยการกำหนดเซต R ซึ่งแสดงถึงลำดับของโหนดที่ต้องการเข้าร่วมกลุ่มมัลติแคสต์ จากเงื่อนไขของการสร้างเส้นทางมัลติแคสต์ที่กล่าวถึงในบทที่ 4 สมาชิกของเซต R หรือ $r(t)$ ซึ่งเป็นเหตุการณ์ที่มีโหนดเข้าร่วมสมาชิกกลุ่มมัลติแคสต์จะสามารถกำหนดได้ในลักษณะเดียวกับงานวิจัยอื่นๆ เช่นงานวิจัย [7], [8] โดยการใช้สมการที่ 5.2 ซึ่งเป็นสมการของความน่าจะเป็นที่จะมีโหนดใดๆ ในระบบเครือข่ายเข้าร่วมเป็นโหนดสมาชิกของกลุ่มมัลติแคสต์หรือเขียนแทนได้โดย $P_c(m)$

$$P_c(m) = \frac{\gamma(N - m)}{\gamma(N - m) + (1 - \gamma)m} \quad (5.2)$$

เมื่อ

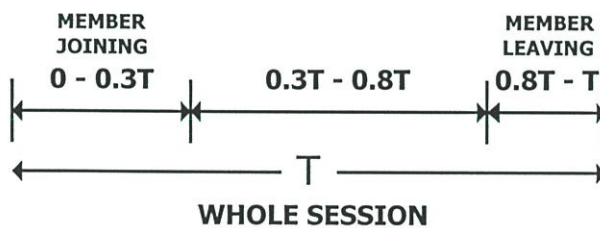
- N เป็นจำนวนโหนดทั้งหมดในระบบเครือข่าย
 m เป็นจำนวนโหนดสมาชิกของกลุ่มมัลติแคสต์ในปัจจุบัน

γ เป็นพารามิเตอร์ที่ใช้กำหนดจำนวนของโหนดสมาชิกกลุ่มมัลติแคสต์ในสถานะสมดุล

จากสมการที่ 5.2 กลุ่มมัลติแคสต์จะถูกเรียกว่าอยู่ในสถานะสมดุลเมื่อความน่าจะเป็นที่จะมีโหนดใดๆ ในระบบเครือข่ายเข้าร่วมเป็นสมาชิกกลุ่มมัลติแคสต์ γ มีค่าเท่ากับ 0.5 ซึ่งจำนวนโหนดสมาชิกของกลุ่มมัลติแคสต์ m จะมีค่าเท่ากับ γN

การกำหนดองค์ประกอบของ $r(t_i)$ ซึ่งประกอบด้วย z_i ซึ่งเป็นโหนดใดๆ ในระบบเครือข่ายที่ต้องการเข้าร่วมกลุ่มมัลติแคสต์ $t_{dep}(z_i)$ เป็นเวลาที่โหนด z_i จะออกจากกลุ่มมัลติแคสต์ และระยะเวลาที่โหนดสมาชิกจะแยกออกกลุ่มมัลติแคสต์เร็วหรือช้ากว่ากำหนด $\Delta t_{dep}(z_i)$ จะถูกกำหนดขึ้นอย่างสุ่ม โดยที่ความน่าจะเป็นที่โหนดสมาชิกจะออกจากกลุ่มมัลติแคสต์เร็วหรือช้ากว่ากำหนดจะมีค่าเท่ากันทั้งหมดสำหรับแต่ละเหตุการณ์ $r(t_i)$ แต่ระยะเวลาที่โหนดสมาชิกออกจากกลุ่มมัลติแคสต์เร็วหรือช้ากว่ากำหนดหรือ $\Delta t_{dep}(z_i)$ จะมีค่าแตกต่างกันไปในแต่ละโหนดสมาชิกของกลุ่มมัลติแคสต์

สำหรับการซิมูเลชันการสร้างเส้นทางมัลติแคสต์แบบสงวนทรัพยากรล่วงหน้าจะมีความแตกต่างไปจากการซิมูเลชันการหาเส้นทางมัลติแคสต์ในแบบปกติเช่น งานวิจัย [6], [7] และ [8] เนื่องจากโหนดสมาชิกจำเป็นจะต้องแจ้งค่าเวลาที่ออกจากกลุ่มมัลติแคสต์ $t_{dep}(z_i)$ ให้โหนดต้นทางทราบขณะที่เข้าร่วมกลุ่มมัลติแคสต์ ดังนั้นจึงต้องมีการกำหนดระยะเวลาทั้งหมดที่โหนดต้นทางทำการส่งข้อมูลมัลติแคสต์ให้แก่โหนดสมาชิกเท่ากับระยะเวลา T โดยในรูปที่ 5.2 แสดงถึงพฤติกรรมในการเข้าและออกจากกลุ่มมัลติแคสต์ของโหนดสมาชิกภายในช่วงเวลา T



รูปที่ 5.2 การเข้าและออกจากกลุ่มมัลติแคสต์ของโหนดสมาชิกภายในช่วงเวลา T

และเมื่อโหนดสมาชิกกลุ่มมัลติแคสต์มีความหนาแน่นเพิ่มมากขึ้นค่าระยะเวลา T ของการส่งข้อมูลมัลติแคสต์ก็จะมีระยะเวลาที่ยาวนานขึ้นตามตารางที่ 5.2 เพื่อให้อัตราส่วนจำนวนโหนดที่ต้องการเข้าร่วมเป็นสมาชิกของกลุ่มมัลติแคสต์ต่อช่วงเวลามีค่าอยู่ในระดับที่ใกล้เคียงกันเมื่อโหนดสมาชิกของกลุ่มมัลติแคสต์มีความหนาแน่นมากขึ้น

ตารางที่ 5.2 เวลาของการส่งข้อมูลมัดติแคสต์ T เมื่อพารามิเตอร์ γ มีค่ามากขึ้น

ความหนาแน่นของโหนดสมาชิก กลุ่มมัดติแคสต์ (γ)	เวลาการส่งข้อมูลมัดติแคสต์ (T)
0.1	100
0.3	300
0.5	600
0.7	900
0.9	1200

5.1.3 การวัดประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัดติแคสต์

การวัดประสิทธิภาพของอัลกอริทึมการสร้างเส้นทางมัดติแคสต์สำหรับกลุ่มสมาชิกแบบไดนามิกสำหรับวิทยานิพนธ์นี้จะใช้ค่าคอสม์ของเส้นทางมัดติแคสต์เป็นตัวชี้วัดประสิทธิภาพในการสร้างเส้นทางมัดติแคสต์ของอัลกอริทึมเช่นเดียวกับงานวิจัยอื่นๆ ที่เกี่ยวข้องกับการสร้างเส้นทางมัดติแคสต์แบบไดนามิก โดยที่ค่าคอสม์ของเส้นทางมัดติแคสต์สามารถแสดงได้ใน 2 รูปแบบ แบบที่ 1 คือการแสดงค่าคอสม์ของเส้นทางโดยตรง และแบบที่ 2 คือการแสดงค่าคอสม์แสดงในรูปของอัตราส่วนหรือ “ค่านอร์มาไลซ์คอสม์” เพื่อความสะดวกในการเปรียบเทียบค่าคอสม์ในการสร้างเส้นทางมัดติแคสต์ระหว่าง 2 อัลกอริทึมใดๆ โดยที่งานวิจัยส่วนใหญ่ที่เกี่ยวข้องกับการสร้างเส้นทางมัดติแคสต์เช่น งานวิจัย [6], [7], [8] จะนิยมใช้อัลกอริทึมเคเอ็มบีหรืออัลกอริทึมแบบกรีดีเป็นอัลกอริทึมอ้างอิงในการนอร์มาไลซ์ ซึ่งจะพบว่าเมื่อค่านอร์มาไลซ์คอสม์มีค่าลดลงย่อหมายถึงค่าคอสม์จะมีค่าใกล้เคียงค่าคอสม์ของอัลกอริทึมอ้างอิงมากขึ้น

5.2 การขมิูละชันเพื่อวัดประสิทธิภาพของอัลกอริทึมโมดิไฟด์กรีดีแบบใหม่

ในหัวข้อนี้จะเป็นการทดสอบประสิทธิภาพของอัลกอริทึม โมดิไฟด์กรีดีแบบใหม่ที่น่าสนใจในวิทยานิพนธ์นี้โดยการขมิูละชัน รวมทั้งทำการศึกษาวิเคราะห์ผลกระทบที่มีต่อค่าคอสม์ของเส้นทางมัดติแคสต์เมื่อพฤติกรรมการเข้าร่วมและออกจากกลุ่มมัดติแคสต์มีลักษณะแตกต่างกันไป โดยพฤติกรรมการเข้าร่วมกลุ่มมัดติแคสต์ของโหนดสมาชิกสามารถกำหนดได้โดย ความหนาแน่นของโหนดสมาชิกกลุ่มมัดติแคสต์ในสถานะสมดุลหรือค่า γ ความน่าจะเป็นที่โหนดสมาชิกจะออกจากกลุ่มมัดติแคสต์เร็วหรือช้ากว่ากำหนด (Error Probability) หรือค่า $P(\Delta_{t_{dcp}}(z))$ รวมถึงระยะเวลาที่โหนดสมาชิกจะออกจากกลุ่มมัดติแคสต์เร็วหรือช้ากว่ากำหนด $\Delta_{t_{dcp}}(z)$ ซึ่งค่า $\Delta_{t_{dcp}}(z)$ จะแสดงอยู่ในรูปของค่าช่วงเวลาที่เกิดคลาดเคลื่อน (Error Duration) เพื่อความสะดวกในการอ้างอิง โดยที่ความสัมพันธ์ระหว่างค่า $\Delta_{t_{dcp}}(z)$ และค่าช่วงเวลาที่เกิดคลาดเคลื่อนดังกล่าวสามารถแสดงได้ในสมการที่ 5.3

$$\text{Error Duration} = \frac{|\Delta t_{\text{dep}}(z)|}{t_{\text{dep}}(z) - t_{\text{st}}(z)} \times 100 \quad (5.3)$$

เมื่อ

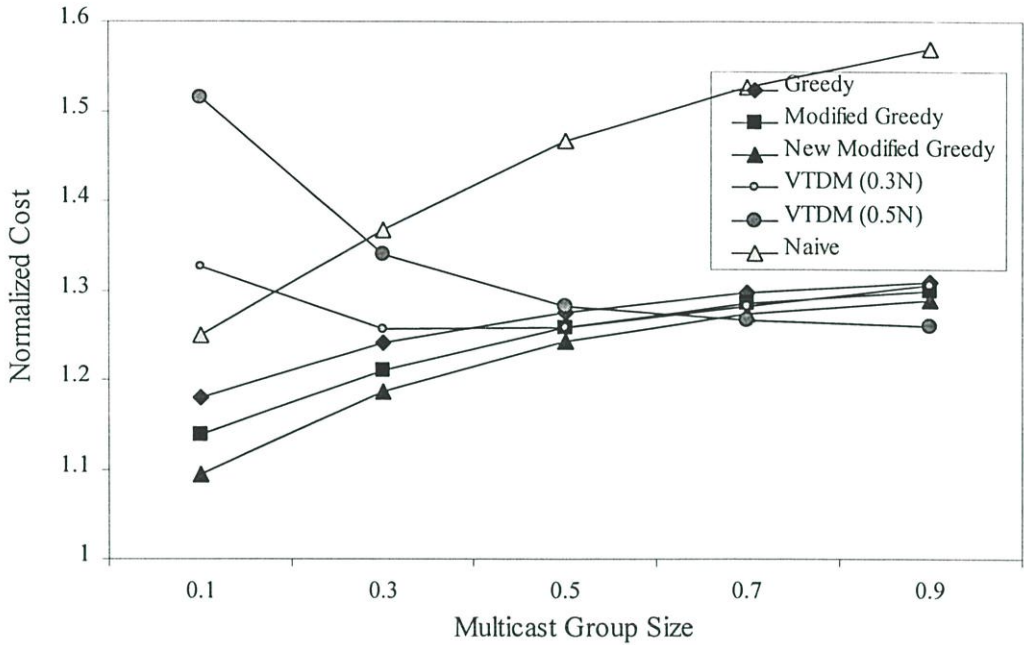
- $t_{\text{st}}(z)$ เป็นเวลาที่โหนดสมาชิก z เริ่มต้นเข้าร่วมกลุ่มมัลติแคสต์
 $t_{\text{dep}}(z)$ เป็นเวลาที่โหนดสมาชิก z จะออกจากกลุ่มมัลติแคสต์

5.2.1 ผลกระทบของพารามิเตอร์ γ ต่อค่าคอสม์ของเส้นทางมัลติแคสต์

ในหัวข้อนี้จะกล่าวถึงการทดลองเพื่อศึกษาผลกระทบของค่า γ ที่มีต่อค่าคอสม์ของเส้นทางมัลติแคสต์ เมื่อกำหนดให้ $P(\Delta t_{\text{dep}}(z)) = 0.5$ และค่าช่วงเวลาที่คลาดเคลื่อนมีค่าเท่ากับร้อยละ 30 โดยทำการทดสอบเปรียบเทียบระหว่างอัลกอริทึม โมดิฟายด์กรีดีแบบใหม่กับอัลกอริทึมแบบอื่นๆ ที่กล่าวถึงคือ อัลกอริทึมแบบทรังก์ชันเมื่อมีค่า θ เท่ากับ 0.3 และ 0.5 ตามลำดับ อัลกอริทึมแบบเนฟ อัลกอริทึมแบบกรีดี และอัลกอริทึมแบบโมดิฟายด์กรีดี ผลการทดลองจะเป็นค่าเฉลี่ยของคอสม์จากการซิมูเลชันบนโครงข่ายแบบสุ่มที่มีจำนวนโหนดเท่ากับ 100 โหนดจำนวน 5 โครงข่าย และในแต่ละโครงข่ายจะทำการทดลองด้วยตัวอย่างของโหนดต้นทางพร้อมกับเซตลำดับของการเข้าร่วมกลุ่มมัลติแคสต์ของโหนดในระบบเครือข่ายหรือเซต R ที่เกิดจากการสุ่มตัวอย่างจำนวน 1000 ตัวอย่าง (จำนวนตัวอย่างทั้งหมดเท่ากับ 5000 ตัวอย่าง) โดยผลการทดลองจะเป็นดังตารางที่ 5.3 และในรูปที่ 5.3 เป็นกราฟแสดงค่าออร์มาไลซ์คอสม์อ้างอิงกับอัลกอริทึมเคเอ็มบีภายใต้เงื่อนไขที่โหนดสมาชิกทุกโหนดออกจากกลุ่มมัลติแคสต์ตรงตามเวลาของการส่งวนทรัพยากรล่วงหน้า

ตารางที่ 5.3 ผลกระทบของค่า γ ต่อค่าคอสม์ของเส้นทางมัลติแคสต์

Algorithm	Equilibrium State (γ)				
	0.1	0.3	0.5	0.7	0.9
	$\times 10^4$	$\times 10^5$	$\times 10^5$	$\times 10^6$	$\times 10^6$
VTDM ($\theta = 0.3N$)	2.4785	1.8989	6.0489	1.1916	1.9277
VTDM ($\theta = 0.5N$)	2.8306	2.0260	6.1640	1.1771	1.8590
NAIVE	2.3345	2.0675	7.0478	1.4190	2.3147
GREEDY	2.2040	1.8750	6.1334	1.2063	1.9331
M - GREEDY	2.1280	1.8294	6.0500	1.1944	1.9167
NEW M - GREEDY	2.0423	1.7919	5.9773	1.1841	1.9019



รูปที่ 5.3 ผลกระทบของค่า γ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์

จากผลการทดลองในตารางที่ 5.3 และรูปที่ 5.3 พบว่า เมื่อโหนดสมาชิกมีโอกาสที่จะออกจากกลุ่มมัลติแคสต์ที่ไม่ตรงตามเวลาที่กำหนดด้วยเงื่อนไข $P(\Delta_{t_{dep}}(z)) = 0.5$ และค่าช่วงเวลาคลาดเคลื่อนเท่ากับร้อยละ 30 อัลกอริทึมโมดิฟายด์กรีดีแบบใหม่จะสามารถให้ค่านอร์มาไลซ์คอสต์อ้างอิงกับอัลกอริทึมเคเอ็มบีที่ต่ำกว่าอัลกอริทึมอื่นๆ ทั้งหมดในช่วงที่ค่า γ มีค่าเข้าใกล้ 0.7 เนื่องจากมีการส่งวนทรัพยากรที่มากขึ้นให้แก่โหนดสมาชิกกลุ่มมัลติแคสต์เพื่อในกรณีที่โหนดสมาชิกมีความน่าจะเป็นที่จะใช้ทรัพยากรของระบบเครือข่ายผิดจากปริมาณที่แจ้งกับโหนดต้นทางไว้ล่วงหน้า

เมื่อความหนาแน่นของโหนดสมาชิกในกลุ่มมัลติแคสต์หรือค่า γ เพิ่มสูงขึ้น (ตั้งแต่ 0.7 ถึง 0.9) ค่านอร์มาไลซ์คอสต์ของอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่จะมีค่าใกล้เคียงกับค่านอร์มาไลซ์คอสต์ของอัลกอริทึมโมดิฟายด์กรีดีและกรีดี และมีค่าสูงกว่าค่านอร์มาไลซ์คอสต์ของอัลกอริทึมแบบทังก์เสมีอนในกรณีที่ θ มีค่าเท่ากับ 0.5 เนื่องจากสำหรับอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่ อัลกอริทึมโมดิฟายด์กรีดีและอัลกอริทึมแบบกรีดีจะใช้ค่าคอสต์เสมีอนระหว่างโหนดสมาชิกใหม่กับเส้นทางมัลติแคสต์ที่มีอยู่ขณะนั้นเป็นเกณฑ์การเลือกเส้นทางในการเชื่อมต่อ โดยมีหลักการสร้างเส้นทางมัลติแคสต์โดยทำให้เกิดการเปลี่ยนแปลงกับเส้นทางที่มีอยู่ปัจจุบันให้น้อยที่สุดซึ่งทำให้ลำดับของโหนดสมาชิกที่เข้าร่วมกลุ่มมัลติแคสต์มีผลต่อการสร้างเส้นทางมัลติแคสต์ ดังนั้นแม้ในกรณีที่โหนดสมาชิกกลุ่มเดียวกันแต่มีลำดับของการเข้าร่วมกลุ่มมัลติแคสต์ที่แตกต่างกันก็จะทำให้มีเส้นทางมัลติแคสต์ที่มีลักษณะที่แตกต่างกัน ซึ่งกรณีนี้มีความแตกต่างไปจากกรณีของอัลกอริทึม

แบบทรีคี่เสมือน เนื่องจากอัลกอริทึมแบบทรีคี่เสมือนยึดหลักการสร้างเส้นทางมัลติแคสต์ในลักษณะเส้นทางต้นไม้ที่สั้นที่สุดโดยยึดเอากลุ่มของสมาชิกเป็นหลัก ดังนั้นแม้ในกรณีที่ลำดับของโหนดสมาชิกซึ่งเข้าร่วมกลุ่มมัลติแคสต์ที่มีความแตกต่างกันก็ยังคงมีเส้นทางมัลติแคสต์ที่มีลักษณะเหมือนกัน และเมื่อความหนาแน่นของโหนดสมาชิกและจำนวนทรีคี่เสมือนมีค่ามากขึ้นการสร้างเส้นทางมัลติแคสต์สำหรับโหนดสมาชิกจะมีความใกล้เคียงกับการสร้างเส้นทางต้นไม้ที่สั้นที่สุดมากขึ้นทำให้ประสิทธิภาพของอัลกอริทึมแบบทรีคี่เสมือนสูงขึ้นส่งผลให้ค่าอนุกรมมาไลซ์คอสต์อ้างอิงกับอัลกอริทึมเคเอ็มบีมีค่าลดน้อยลงด้วยเช่นกัน

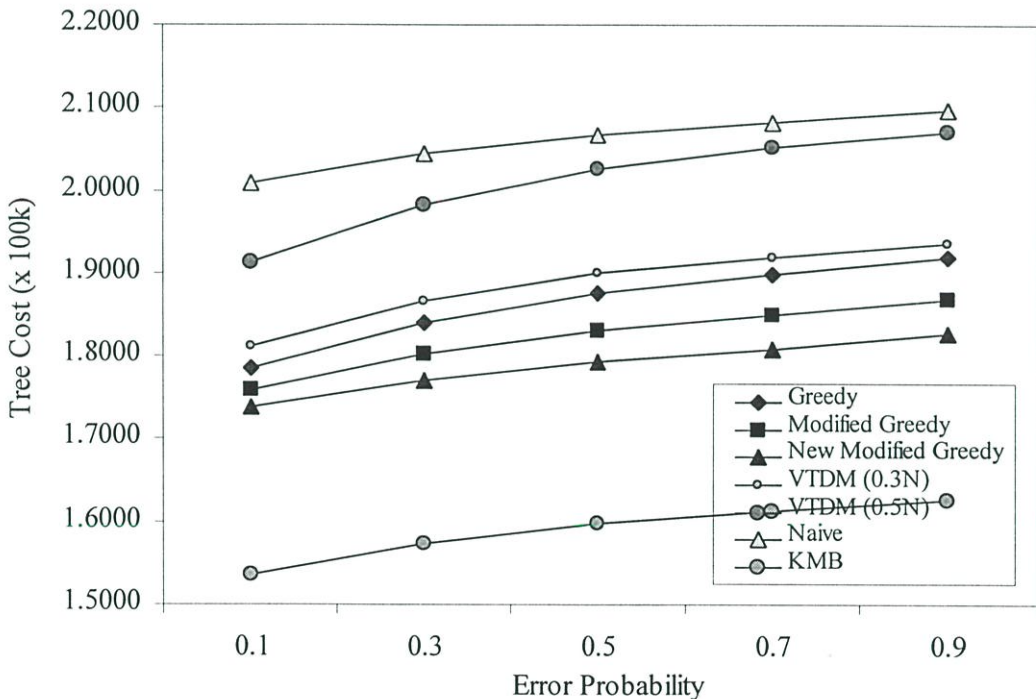
5.2.2 ผลกระทบของพารามิเตอร์ $P(\Delta t_{dep}(z))$ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์

ในหัวข้อนี้จะเป็นการทดลองเพื่อศึกษาผลของค่าพารามิเตอร์ $P(\Delta t_{dep}(z))$ ที่มีต่อค่าคอสต์ของเส้นทางมัลติแคสต์ เมื่อค่าพารามิเตอร์ γ และ $\Delta t_{dep}(z)$ ถูกกำหนดให้มีค่าคงที่คือเท่ากับ 0.3 และร้อยละ 30 ตามลำดับ โดยเป็นการเปรียบเทียบระหว่างอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่กับอัลกอริทึมแบบเคเอ็มบี แบบทรีคี่เสมือน แบบเนฟ แบบกรีดี และแบบโมดิฟายด์กรีดี ซึ่งผลการทดลองจะเป็นค่าเฉลี่ยของคอสต์ที่เกิดจากการซิมูเลชันบนโครงข่ายแบบสุ่มที่มีจำนวนโหนดเท่ากับ 100 โหนดจำนวน 5 โครงข่ายและสำหรับแต่ละโครงข่ายจะทำการทดลองด้วยตัวอย่างของโหนดต้นทางพร้อมกับเซตลำดับของการเข้าร่วมกลุ่มมัลติแคสต์ของโหนดในระบบเครือข่ายหรือเซต R ที่เกิดจากการสุ่มตัวอย่างจำนวน 1000 ตัวอย่างเช่นเดียวกับการทดลองที่ 5.2.1 ค่าคอสต์ของเส้นทางมัลติแคสต์ที่ได้จากแต่ละอัลกอริทึมแสดงดังตารางที่ 5.4 และในรูปที่ 5.4 เป็นกราฟแสดงค่าคอสต์ของเส้นทางมัลติแคสต์ของอัลกอริทึมชนิดต่างๆ

ตารางที่ 5.4 ผลกระทบของค่า $P(\Delta t_{dep}(z))$ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์

Algorithm	$P(\Delta t_{dep}(z))$				
	0.1	0.3	0.5	0.7	0.9
	$\times 10^5$				
KMB	1.5365	1.5734	1.5984	1.6121	1.6258
VTDM ($\theta = 0.3N$)	1.8114	1.8654	1.8989	1.9182	1.9359
VTDM ($\theta = 0.5N$)	1.9128	1.9825	2.0260	2.0511	2.0712
NAIVE	2.0095	2.0448	2.0675	2.0813	2.0970
GREEDY	1.7848	1.8388	1.8750	1.8977	1.9181
M - GREEDY	1.7596	1.8023	1.8294	1.8491	1.8669
NEW M - GREEDY	1.7384	1.7699	1.7919	1.8080	1.8269

จากผลการทดลองดังตารางที่ 5.4 และรูปที่ 5.4 พบว่า เมื่อความน่าจะเป็นที่โหนดสมาชิกจะออกจากกลุ่มมัลติแคสต์เร็วหรือช้ากว่ากำหนดหรือค่า $P(\Delta t_{dep}(z))$ มีค่าเพิ่มสูงขึ้นทำให้ค่าคอสต์ของเส้นทางมัลติแคสต์ที่สร้างจากอัลกอริทึมทั้งหมดมีค่าสูงขึ้นเช่นกัน โดยที่อัลกอริทึมโมดิฟายด์กรีดีแบบใหม่ยังคงให้ค่าคอสต์ของเส้นทางมัลติแคสต์ที่ใกล้เคียงกับอัลกอริทึมเคเอ็มบีมากที่สุดแม้ว่าค่าพารามิเตอร์ $P(\Delta t_{dep}(z))$ จะมีค่ามากขึ้น และจากรูปที่ 5.4 พบว่าอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่มีอัตราการเปลี่ยนแปลงของค่าคอสต์เมื่อเทียบกับค่าพารามิเตอร์ $P(\Delta t_{dep}(z))$ ที่ดีกว่าเล็กน้อย โดยอ้างอิงกับอัลกอริทึมการสร้างเส้นทางมัลติแคสต์อื่นๆ ที่มีค่าคอสต์ของเส้นทางมัลติแคสต์ที่อยู่ในระดับที่ใกล้เคียงกัน เนื่องจากอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่มีการสงวนทรัพยากรที่มากขึ้นให้แก่โหนดสมาชิกกลุ่มมัลติแคสต์ดังนั้นแม้ในกรณีที่โหนดสมาชิกมีแนวโน้มมากขึ้นที่จะออกจากกลุ่มมัลติแคสต์ไม่ตรงตามเวลาที่แจ้งไว้ ค่าคอสต์ของเส้นทางมัลติแคสต์จึงยังคงมีอัตราการเปลี่ยนแปลงที่ต่ำกว่าเมื่อเทียบกับอัลกอริทึมชนิดอื่นๆ



รูปที่ 5.4 ผลกระทบของค่า $P(\Delta t_{dep}(z))$ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์

5.2.3 ผลกระทบของพารามิเตอร์ $\Delta t_{dep}(z)$ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์

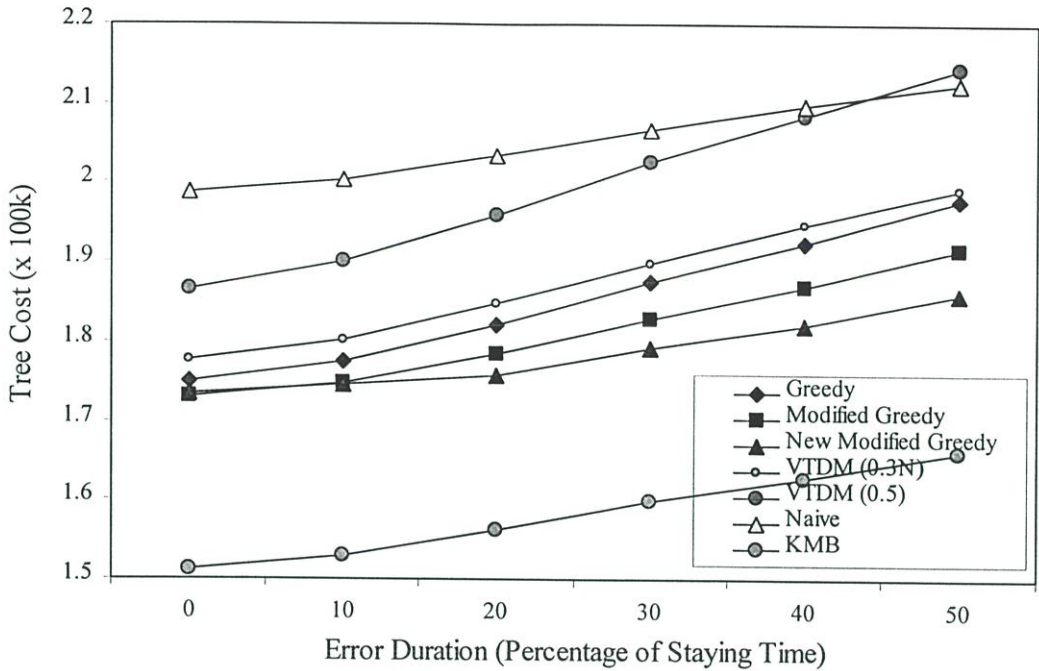
ในหัวข้อนี้จะเป็นการทดลองเพื่อศึกษาผลของค่าพารามิเตอร์ $\Delta t_{dep}(z)$ ที่มีต่อค่าคอสต์ของเส้นทางมัลติแคสต์โดยผ่านทางค่าช่วงเวลาที่คลาดเคลื่อนตามความสัมพันธ์ในสมการที่ 5.3 เมื่อค่า $\gamma = 0.3$ และ $P(\Delta t_{dep}(z)) = 0.5$ ตามลำดับ โดยทดลองเปรียบเทียบระหว่างอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่กับอัลกอริทึมแบบอื่นๆ ทั้งแบบเคเอ็มบี แบบทังก์ส์เสมือน แบบเนฟ แบบกรีดี และแบบ

โมดิฟายด์กรีดี ตามลำดับ ผลการทดลองจะเป็นค่าเฉลี่ยของค่าคอสม์ที่ได้จากการซิมูเลชันบน โครงข่ายแบบสุ่มที่มีจำนวนโหนดเท่ากับ 100 โหนดจำนวน 5 โครงข่ายและสำหรับแต่ละโครงข่ายจะทำการทดลองด้วยตัวอย่างของโหนดค้นหาพร้อมทั้งเซทลำดับของการเข้าร่วมกลุ่มมัดติเคสท์ของโหนดในระบบเครือข่ายหรือเซท R ที่เกิดจากการสุ่มตัวอย่างจำนวน 1000 ตัวอย่าง ค่าคอสม์ของเส้นทางมัดติเคสท์ที่ได้จากอัลกอริทึมแต่ละแบบเป็นดังตารางที่ 5.5 และรูปที่ 5.5 แสดงกราฟความสัมพันธ์ระหว่างค่าคอสม์ของเส้นทางมัดติเคสท์แบบต่างๆ เมื่อค่าช่วงเวลาที่คลาดเคลื่อนมีการเปลี่ยนแปลง

จากผลการทดลองดังตารางที่ 5.5 และรูปที่ 5.5 พบว่าค่าช่วงเวลาที่คลาดเคลื่อนซึ่งแปรผันโดยตรงกับ $\Delta t_{dep}(z_i)$ มีค่าเพิ่มมากขึ้น ทำให้ค่าคอสม์ของเส้นทางมัดติเคสท์จากอัลกอริทึมทั้งหมดจะมีค่าเพิ่มขึ้นอย่างรวดเร็ว โดยที่อัลกอริทึมโมดิฟายด์กรีดีแบบใหม่ยังสามารถให้ค่าคอสม์ของเส้นทางมัดติเคสท์ที่ดีที่สุดเมื่อเปรียบเทียบกับอัลกอริทึมเคเอ็มบี และอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่มีอัตราการเปลี่ยนแปลงของค่าคอสม์ของเส้นทางมัดติเคสท์เมื่อเทียบกับค่าช่วงเวลาที่คลาดเคลื่อนน้อยที่สุดเมื่อเปรียบเทียบกับอัลกอริทึมอื่นๆ ที่มีค่าคอสม์ของเส้นทางมัดติเคสท์อยู่ในระดับที่ใกล้เคียงกัน เนื่องจากอัลกอริทึมโมดิฟายด์กรีดีแบบใหม่มีการสงวนทรัพยากรที่มากขึ้นให้แก่โหนดสมาชิกกลุ่มมัดติเคสท์ ดังนั้นแม้ในกรณีที่โหนดสมาชิกกลุ่มมัดติเคสท์มีแนวโน้มที่จะออกจากกลุ่มมัดติเคสท์ที่ไม่ตรงตามเวลาด้วยค่าช่วงเวลาที่คลาดเคลื่อนที่เพิ่มสูงขึ้น อัลกอริทึมโมดิฟายด์กรีดีแบบใหม่ยังมีอัตราการเปลี่ยนแปลงของค่าคอสม์ของเส้นทางมัดติเคสท์ที่ต่ำกว่าอัลกอริทึมแบบอื่นๆ อย่างชัดเจน

ตารางที่ 5.5 ผลกระทบของค่า $\Delta t_{dep}(z)$ ต่อค่าคอสม์ของเส้นทางมัดติเคสท์

Algorithm	Error Duration (%)					
	0	10	20	30	40	50
	$\times 10^5$					
KMB	1.5111	1.5287	1.5616	1.5984	1.6285	1.6608
VTDM ($\theta = 0.3N$)	1.7765	1.8024	1.8482	1.8989	1.9460	1.9911
VTDM ($\theta = 0.5N$)	1.8659	1.8995	1.9574	2.0260	2.0846	2.1435
NAIVE	1.9867	2.0037	2.0346	2.0675	2.0980	2.1243
GREEDY	1.7502	1.7751	1.8213	1.8750	1.9228	1.9782
M - GREEDY	1.7301	1.7474	1.7846	1.8294	1.8693	1.9159
NEW M - GREEDY	1.7340	1.7453	1.7563	1.7919	1.8216	1.8603



รูปที่ 5.5 ผลกระทบของค่า $\Delta t_{dep}(z)$ ต่อค่าคอสต์ของเส้นทางมัลติแคสต์ขึ้น

5.3 บทสรุป

ในบทนี้ได้กล่าวถึงการทดสอบประสิทธิภาพของอัลกอริทึม โมดิไฟด์กรีดีแบบใหม่ที่ได้นำเสนอในวิทยานิพนธ์นี้ด้วยการซิมูเลชันบนระบบเครือข่ายแบบสุ่มภายใต้พฤติกรรมการเข้าร่วมกลุ่มมัลติแคสต์แบบต่างๆ ซึ่งกำหนดโดยความหนาแน่นของโหนดสมาชิกกลุ่มมัลติแคสต์ในสถานะสมดุลหรือค่า γ ความน่าจะเป็นที่โหนดสมาชิกจะออกจากกลุ่มมัลติแคสต์เร็วหรือช้ากว่ากำหนดหรือค่า $P(\Delta t_{dep}(z))$ รวมถึงระยะเวลาที่โหนดสมาชิกจะออกจากกลุ่มมัลติแคสต์เร็วหรือช้ากว่ากำหนด $\Delta t_{dep}(z)$ และพบว่า อัลกอริทึม โมดิไฟด์กรีดีแบบใหม่สามารถให้ค่าคอสต์ของเส้นทางมัลติแคสต์ที่ดีกว่าเมื่อเปรียบเทียบกับอัลกอริทึมการสร้างเส้นทางมัลติแคสต์แบบอื่นๆ ที่ได้กล่าวถึงในวิทยานิพนธ์นี้

บทที่ 6

รายงานสรุปและข้อเสนอแนะ

6.1 รายงานสรุป

ในวิทยานิพนธ์นี้ได้นำเสนออัลกอริทึม โมดิฟายด์กรีดีแบบใหม่ซึ่งเป็นอัลกอริทึมการสร้างเส้นทางมัลติแอสต์สำหรับกลุ่มสมาชิกแบบไดนามิกด้วยการสงวนทรัพยากรล่วงหน้าและได้ทำการศึกษาผลกระทบจากพฤติกรรมการเข้าร่วมกลุ่มมัลติแอสต์ที่มีต่อค่าคอสม์ของเส้นทางมัลติแอสต์เปรียบเทียบกับอัลกอริทึมการสร้างเส้นทางมัลติแอสต์แบบอื่นๆ จากผลการทดลองพบว่า เมื่อค่าพารามิเตอร์ที่เกี่ยวข้องกับพฤติกรรมการเข้าร่วมกลุ่มมัลติแอสต์ของโหนดสมาชิกทั้ง 3 ค่าคือ ความหนาแน่นของโหนดสมาชิกกลุ่มมัลติแอสต์ในสถานะสมดุลหรือค่า γ ความน่าจะเป็นที่โหนดสมาชิกจะออกจากกลุ่มมัลติแอสต์เร็วหรือช้ากว่ากำหนดหรือค่า $P(\Delta t_{dep}(z))$ และระยะเวลาที่โหนดสมาชิกจะออกจากกลุ่มมัลติแอสต์เร็วหรือช้ากว่ากำหนดหรือค่า $\Delta t_{dep}(z)$ มีค่าเพิ่มมากขึ้นจะส่งผลให้ค่าคอสม์ของเส้นทางจากมัลติแอสต์ของแต่ละอัลกอริทึมมีค่าเพิ่มสูงขึ้นด้วยเช่นกัน

สำหรับกรณีที่ค่าความหนาแน่นของโหนดสมาชิกกลุ่มมัลติแอสต์ในสถานะสมดุลหรือค่าพารามิเตอร์ γ มีค่าเพิ่มมากขึ้น อัลกอริทึม โมดิฟายด์กรีดีแบบใหม่สามารถให้ค่าคอสม์ของเส้นทางมัลติแอสต์ที่ใกล้เคียงกับค่าคอสม์ของเส้นทางมัลติแอสต์จากอัลกอริทึมเคเอ็มบีซึ่งเป็นอัลกอริทึมอ้างอิงมากที่สุด เมื่อค่าพารามิเตอร์ $\gamma \leq 0.7$ โดยประมาณและเมื่อค่าพารามิเตอร์ γ มีค่าเพิ่มมากขึ้นจนถึง 0.9 อัลกอริทึมแบบทังก์เสมือนสามารถให้ค่าคอสม์ของเส้นทางมัลติแอสต์ที่ใกล้เคียงกับค่าคอสม์ของเส้นทางมัลติแอสต์จากอัลกอริทึมเคเอ็มบีมากที่สุด โดยอาศัยหลักการการหาเส้นทางต้นไม้ที่เล็กที่สุดซึ่งยึดกลุ่มสมาชิกเป็นหลักในการสร้างเส้นทางมัลติแอสต์ แต่ในทางปฏิบัติเมื่อโหนดสมาชิกของกลุ่มมัลติแอสต์เพิ่มจำนวนมากขึ้น โปรโตคอลการหาเส้นทางมัลติแอสต์มักจะนิยมใช้วิธีการฟลัดดิ้ง (Flooding) ในการส่งข้อมูลให้โหนดสมาชิกมากกว่าการสร้างเส้นทางต้นไม้สำหรับโหนดสมาชิก เนื่องจากมีความต้องการลดภาระที่เพิ่มขึ้นในการประมวลผลสถานะกลุ่มมัลติแอสต์ของโหนดต้นทางเมื่อมีจำนวนโหนดสมาชิกเพิ่มมากขึ้นดังรายละเอียดใน [2], [3] รวมถึงยังสามารถใช้ประโยชน์จากรางการเลือกเส้นทางสำหรับการส่งข้อมูลยูนิแอสต์ได้ด้วย

สำหรับกรณีที่ความน่าจะเป็นที่โหนดสมาชิกจะออกจากกลุ่มมัลติแอสต์เร็วหรือช้ากว่ากำหนดหรือค่า $P(\Delta t_{dep}(z))$ และระยะเวลาที่โหนดสมาชิกจะออกจากกลุ่มมัลติแอสต์เร็วหรือช้ากว่ากำหนดหรือค่า $(\Delta t_{dep}(z))$ มีค่าเพิ่มสูงขึ้น เมื่อค่าพารามิเตอร์ γ มีค่าน้อยกว่า 0.7 จะพบว่าอัลกอริทึม โมดิฟายด์กรีดีแบบใหม่ยังคงให้ค่าคอสม์ของเส้นทางมัลติแอสต์ที่มีค่าใกล้เคียงกับค่าคอสม์ของ

เส้นทางมัลติแอสต์จากอัลกอริทึมเคเอ็มบีมากที่สุดและมีอัตราการเปลี่ยนแปลงของค่าคอสม์ของเส้นทางมัลติแอสต์ที่น้อยกว่าอย่างชัดเจนเมื่อเทียบกับอัลกอริทึมการสร้างเส้นทางมัลติแอสต์แบบอื่นๆ ที่ได้กล่าวถึงในวิทยานิพนธ์ฉบับนี้

6.2 ข้อเสนอแนะเพื่องานวิจัยในอนาคต

แม้ว่าจากงานวิจัยนี้รวมถึงงานวิจัย [14] ได้แสดงถึงประโยชน์ในการประยุกต์ใช้แนวคิดการสงวนทรัพยากรล่วงหน้าเข้ากับอัลกอริทึมของการสร้างเส้นทางมัลติแอสต์ แต่อัลกอริทึมในการสร้างเส้นทางมัลติแอสต์ที่ถูกนำเสนอในวิทยานิพนธ์นี้รวมถึงอัลกอริทึมในการสร้างเส้นทางมัลติแอสต์อื่นๆ ที่ถูกกล่าวถึงยังมีลักษณะแบบรวมศูนย์ซึ่งอาจจะเกิดปัญหาด้านความสามารถของการขยายขนาดได้ เมื่อนำอัลกอริทึมในลักษณะนี้ไปใช้งานจริงในระบบเครือข่าย หากเกิดเหตุการณ์ที่มีจำนวนโหนดสมาชิกของกลุ่มมัลติแอสต์เพิ่มมากขึ้น โหนดต้นทางจะต้องรับภาระหนักในการประมวลผลข้อมูลทั้งหมดของโหนดสมาชิกเพื่อสร้างเส้นทางมัลติแอสต์รวมถึงยังต้องเสียทรัพยากรในระบบเครือข่ายอีกจำนวนหนึ่งเพื่อใช้ในการส่งข้อมูลต่างๆ เกี่ยวกับสถานะของสมาชิกกลุ่มมัลติแอสต์อีกด้วย

ดังนั้นแนวคิดในการพัฒนางานวิจัยนี้ต่อไปในอนาคตจึงมีจุดมุ่งหมายในการออกแบบและพัฒนาอัลกอริทึมการหาเส้นทางมัลติแอสต์ด้วยการสงวนทรัพยากรล่วงหน้าแบบใหม่ที่มีลักษณะการทำงานเป็นแบบกระจายมากขึ้นกว่าเดิม เพื่อการลดปัญหาด้านความสามารถในการขยายตัวของกลุ่มมัลติแอสต์รวมถึงพยายามรักษาประสิทธิภาพในการสร้างเส้นทางให้ใกล้เคียงกับอัลกอริทึมที่มีลักษณะการทำงานแบบรวมศูนย์ให้มากที่สุด ซึ่งนำไปสู่การพัฒนาโปรโตคอลของการสร้างเส้นทางมัลติแอสต์ด้วยการสงวนทรัพยากรล่วงหน้าเพื่อการใช้งานในระบบเครือข่ายต่อไปในอนาคต

เอกสารอ้างอิง

- [1] T. Cormen, “Introduction to Algorithms”, The MIT Press, Cambridge, Massachusetts, 1990.
- [2] R. Perlman, “Interconnections: Bridges, Routers, Switches, and Internetworking Protocols (2nd edition)”, Addison-Wesley Pub Co, 1999.
- [3] M. Ramalho, “Intra- and Inter- Domain Multicast Routing Protocols: A Survey and Taxonomy”, IEEE Communications Surveys & Tutorials, vol. 3, no. 1, First Quarter, 2000
- [4] L. Kou, G. Markowsky, and L. Berman, “A Fast Algorithm for Steiner Tree”, Acta Informatica, vol. 15, pp. 141-145, 1981.
- [5] B.M. Waxman, “Routing of Multiple Connections,” IEEE Journal on Selected Areas in Communications, Vol.6, No.9, pp.1617-1622 December, 1988.
- [6] M. Doar and I. Leslie, “How Bad is Naïve Multicast Routing,” Proceedings of IEEE INFOCOMM, San Francisco, CA., pp.82-89, April 1993.
- [7] H. Lin and S. Lai., “VTDM—A Dynamic Multicast Routing Algorithm,” IEEE INFOCOMM’ 98, pp.1426-1432, March-April 1998.
- [8] T. Asaka, T. Miyoshi, and Y. Tanaka, “Dynamic Multicast Routing Algorithm Using Predetermined Path Search,” IEICE Transactions on Communications, Vol.E83-B, No.5, pp. 1128-1135, May 2000.
- [9] R. Guerin and A. Orda, “Networks with Advance Reservations: The Routing Perspective,” IEEE INFOCOMM, Tel-Aviv, Israel, March 26-30, 2000.
- [10] G. Apostolopoulos, “Cost and Performance trade offs of Quality of Service Routing”, PhD. Dissertation, University of Maryland, College Park, 1999.
- [11] R. Widyono, “The Design and Evaluation of Routing Algorithms for Real-time Channels,” Technical Report TR-94-024, University of California at Berkeley, June, 1994.
- [12] A. Gunabhibal, C. Pornavalai, and R. Varakulsiripunth, “Dynamic Multicast Routing under Inaccurate Links State Information”, Proceeding of Student Conference on Research and Development (SCOReD 2001), Kuala Lumpur, Malaysia, February 20-21, 2001, paper no.37 on CD-ROM.

- [13] Z. Wang, and J. Crowcroft, "Quality of Service Routing for Supporting Multimedia Applications", IEEE Journal on Selected Area in Communication, 14(7):1228-1234, 1996.
- [14] C. Pornavalai, D. Chakraborty, G. Chakraborty, N. Shiratori, "Dynamic Multicast Routing in Advance Resource Reservation Environment" Proc. in ICPAD, Iwate Prefectural Universtiy, Japan, 4-6 July, 2000.
- [15] A. Gunabhibal, C. Pornavalai, and R. Varakulsiripunth, "Dynamic Multicast Routing with Advance Reservations", ISCIT, Chiang-Mai, Thailand, 14-16 November, 2001.
- [16] J. M. S. Doar, "Multicasting in The Asynchronous Transfer Mode Environment", Computer Laboratory Technical Report, no. 298, University Cambridge, 1993. (Ph.D. dissertation)
- [17] W. Stallings, "Data & Computer Communications (6th Edition)", Prentice Hall, 1999

ภาคผนวก ก

รหัสเทียมของอัลกอริทึมการสร้างเส้นทางมัลติแคสต์โอมิฟาइट์กรีดีแบบใหม่

อินพุท: $G = (V, E)$ เมื่อ V เป็นเซตของโหนดในระบบเครือข่ายและ เป็นเซตของลิงค์ที่เชื่อม
 ระหว่าง 2 โหนดใดๆ ในระบบเครือข่าย
 C เป็นเซตของค่าคอสม์ของลิงค์ในระบบเครือข่าย
 T เป็นเวลาของทั้งหมดของการส่งข้อมูลมัลติแคสต์
 s เป็นโหนดต้นทางของกลุ่มมัลติแคสต์
 γ เป็นค่าความหนาแน่นของโหนดสมาชิกกลุ่มมัลติแคสต์

เอาต์พุท: ค่าคอสม์ของเส้นทางมัลติแคสต์ในแต่ละช่วงเวลาจาก 0 ถึง T ($TreeCost(t)$)

NEW_MODIFIED_GREEDY(G, T, C, s, γ)

```

1>    $D \leftarrow \{s\}$ 
2>    $N \leftarrow V - \{s\}, XcastRouteTbl \leftarrow \emptyset$ 
3>   for every edge  $e \in E$ 
4>        $LinkTagTbl(e) \leftarrow 0$ 
5>
6>   for  $t \leftarrow 0$  to  $T$ 
7>       for  $i \leftarrow 0$  to  $0.3T$ 
8>           if  $p_c(m) > RAND \mid (0 - 1)$ 
9>                $\{z, t_{dep}(z), \Delta t_{dep}(z)\} \leftarrow RAND \mid (z \in N) \text{ and } (z \neq s)$ 
10>
11>                $t_{ref} \leftarrow FND\_REFERENCE\_TIME(G, LinkTagTbl, t_{dep}(z))$ 
12>                $H \leftarrow FND\_HOOK\_NODE(G, C, D, XcastRouteTbl, LinkTagTbl, t_{ref})$ 
13>                $P_{z \rightarrow H} \leftarrow DIJKSTRA(G, C, z, H)$ 
14>                $P_{H \rightarrow s} \leftarrow XcastRouteTbl(P_{H \rightarrow s})$ 
15>                $P_{z \rightarrow s} \leftarrow P_{z \rightarrow H} \oplus P_{H \rightarrow s}$ 
16>                $D \leftarrow D \cup \{z\}$ 
17>                $N \leftarrow N - \{z\}$ 
18>                $XcastRouteTbl \leftarrow XcastRouteTbl \cup P_{z \rightarrow s}$ 
19>                $UPDATE\_TAG(LinkTagTbl, P_{z \rightarrow s}, t_{dep}(z), \Delta t_{dep}(z))$ 
20>
21>    $TIME\_TRIGGER(G, LinkTagTbl)$ 

```

```

22>      TreeCost(t) ← COST_CALCULATION(G, C, LinkTagTbl)
23>
24> FND_REFERENCE_TIME(G, LinkTagTbl, tdep(z))
25>   TempLinkTagTbl(e) ← 0
26>   for every edge e ∈ E
27>       if LinkTagTbl(e) < tdep(z)
28>           LinkTagTbl(e) ← 0
29>
30>   TempLinkTagTbl ← LinkTagTbl(l)
31>   FreqLinkTag ← HISTOGRAM(TempLinkTagTbl)
32>   tref ← WEIGHTED_AVG(FreqLinkTag)
33>   return tref
34>
35> FND_HOOK_NODE(G, C, D, XcastRouteTbl, LinkTagTbl, tref)
36>   for every vertex di ∈ D
37>       VCost(di) ← 0
38>       Pz→di ← DIJKSTRA(G, C, z, di)
39>       Pdi→s ← XcastRouteTbl(Pdi→s)
40>       Pz→s ← Pz→di ⊕ Pdi→s
41>       VCost(di) ← VCOST_CALCULATION(G, C, LinkTagTbl, Pz→s, di, tref)
42>
43>   H = di | min(VCost(di))
44>   return H
45>
46> VCOST_CALCULATION(G, C, LinkTagTbl, Pz→s, di, tref)
47>   VCost(di) = 0
48>   for every edge li ∈ Pz→s
49>       If LinkTagTbl(li) ≠ 0 and LinkTagTbl(li) < tref
50>           VCost(di) ← VCost(di) + {[tref - LinkTagTbl(li)] x C(li)}
51>       elseif LinkTagTbl(li) = 0
52>           VCost(di) ← VCost(di) + {[tref - t] x C(li)}

```

```

53>         return VCost(d)
54>
55> UPDATE_TAG(LinkTagTbl,  $P_{z \rightarrow s}$ ,  $t_{dep}(z)$ ,  $\Delta t_{dep}(z)$ )
56>   for every edge  $l_i \in P_{z \rightarrow s}$ 
57>     if LinkTagTbl( $l_i$ ) <  $t_{dep}(z) + \Delta t_{dep}(z)$ 
58>       LinkTagTbl( $l_i$ )  $\leftarrow t_{dep}(z) + \Delta t_{dep}(z)$ 
59>
60> TIME_TRIGGER(G, LinkTagTbl)
61>   for every edge  $e \in E$ 
62>     if LinkTagTbl( $e$ )  $\neq 0$ 
63>       LinkTagTbl( $e$ )  $\leftarrow$  LinkTagTbl( $e$ ) - 1
64>
65> COST_CALCULATION(G, C, LinkTagTbl, t)
66>   TreeCost(t)  $\leftarrow 0$ 
67>   for every edge  $e \in E$ 
68>     if LinkTagTbl( $e$ )  $\neq 0$ 
69>       TreeCost(t)  $\leftarrow$  TreeCost(t) + C( $e$ )
70>   return TreeCost(t)

```

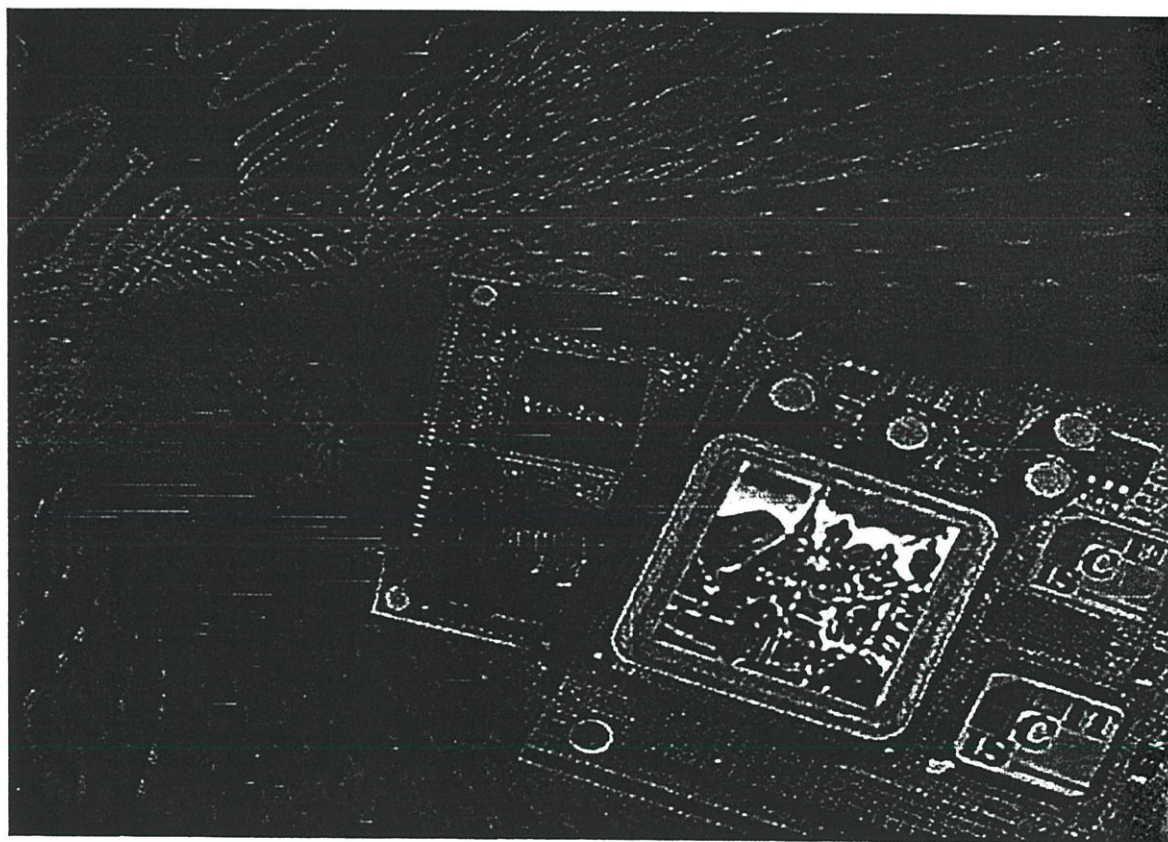
ภาคผนวก ข

ผลงานวิจัยที่ได้รับการตีพิมพ์

- A. Gunabhibal, C. Pornavalai, and R. Varakulsiripunth, “Dynamic Multicast Routing with Advance Reservations”, ISCIT, Chiang-Mai, Thailand, 14-16 November, 2001.

PROCEEDINGS

2001 International Symposium on Communications and Information Technology ISCIT 2001



*The Merging Decade of
Communication Systems and Information Technology*

November 14 - 16, 2001

Chiang Mai Orchid Hotel, Chiang Mai, Thailand



DYNAMIC MULTICAST ROUTING WITH ADVANCE RESERVATIONS

Apipol Gunabhibal[†] Chotipat Pornavalai^{††} and Ruttikorn Varakulsiripunth^{†††}

^{†,†††}Faculty of Engineering and Research Center
for Communication and Information Technology
(ReCCIT)

King Mongkut's Institute of Technology
Ladkrabang, Bangkok 10520, Thailand.

E-mail: [†]s2061039@kmitl.ac.th

^{†††}kvruttik@kmitl.ac.th

^{††}Faculty of Information Technology
King Mongkut's Institute of Technology
Ladkrabang, Bangkok 10520, Thailand.

E-mail: ^{††}chotipat@it.kmitl.ac.th

ABSTRACT

Reserving the resources for the requested applications is one of the most effective schemes that were proposed to offer the time-critical multimedia applications recently. Because the resource is limited, an ability to provide resource reservations in advance, is essential in multi-party applications with dynamic accessing and leaving of user such as modern integrated (voice, video and data) collaboration system. Modified Greedy (MG) algorithm [4] was proposed for multicast routing in advance reservations environment. However, the performance of MG algorithm was quite sensitive with precision of the information given by user. In this work, we improve MG algorithm in order to give a good performance, even the inaccuracy of information given by user was happened, as shown in our simulation results.

Key words: multicast, routing, dynamic, algorithm advance reservations

1. INTRODUCTION

As communication networks capabilities increase, their usage is also expanding. Therefore, the wide range of requirements of the many applications using them calls for mechanisms to offer the quality of service guarantee. For example, STII and RSVP provided the resource reservations. However, since the resource is limited, it is not possible to admit arbitrary number of connections, and at the same time, to ensure that the performance of the applications that already reserved the resources will not be degraded. Then, resource may be requested in advance whenever they are needed. This can be a useful service. There are so many works that related to providing support for advance reservations in communication networks.

Advance reservations are also required for multicast communications. In multicast communications, messages are concurrently sent to multiple destinations of all members of the same multicast group. Mechanisms to support such a form of communication are becoming an increasingly important component of the design

and implementations of future communication networks. Depending on the application, the multicast group can be either static or dynamic. For static group, the members of the multicast group remain unchanged during the lifetime of the multicast session. For dynamic group, members can join or leave the group during the lifetime of the multicast session. Dynamic multicast routing is important for actual multicast applications. In terms related to the efficient use of network resources involved in a multicast session, typical approaches to multicast routing require the transmission of packets along the branches of a tree spanning between sources and destinations nodes. For ensuring network efficiency, the smallest possible amount of network resources should be used to set up the multicast connection. The smallest tree is known as the steiner tree and finding such a tree is a well known NP-hard problem. In this work, we focus on dynamic routing with advance reservations problems.

As we know, there are few works that having done on routing with advance reservations [4], [6]. Modified Greedy (MG) algorithm is dynamic multicast routing algorithm that works well in advance reservations environment. In the advance reservations environment, the requests made by the application specify, not only the parameters that define the traffic flow specifications and its quality of service requirements, but also the following two quantities: the starting time, and the duration time. But we found that MG algorithm has less flexibility when it facing with inaccuracy of information of joining and duration time. Here, we proposed the new version of MG algorithm that can coping with inaccuracy of information of joining and duration time (such as real situation).

In section 2, we define the dynamic multicast routing problem, then describing it with examples. In section 3, we present our algorithm, as a new version of MG algorithm. And we evaluate simulation environment and show results, when new version of MG algorithm is facing with inaccuracy of joining and leaving time in section 4. Finally, in section 5, we present conclusion from the results of simulation.

2. DYNAMIC MULTICAST ROUTING PROBLEM SPECIFICATION

Our problem specification are same as [4]. By given graph $G = (V, E)$, where V is the set of vertices and E is the set of edges. Each edge $l \in E$ has a cost function: $Cost(l) \rightarrow R^+$ (where R^+ stands for set of positive real numbers). And $Z (Z \subset V, s \notin Z)$ is a set of destination nodes that can join in the multicast session. $R_t = \{r_t(1), r_t(2), r_t(3), \dots, r_t(m)\}$ is the set of requests asked by participants at time t . $r_t(i)$ is the request number i at time t , and each request is composed of z_i and $t_{dep}(z_i)$ (advance reservations concept [6]), where $z_i \in Z$ is the destination node that want to join at time t , and $t_{dep}(z_i)$ is the departure time of node z_i . T is the total session duration. But for this work, actual leaving time can either accuracy or inaccuracy with some of sooner or later by randomization.

Given a source node $s \in V$ and a sequence of request R_t , starting from the beginning of the session. At each time instant t , a multicast tree as defined as a subtree of the graph $G(V, E)$, rooted from s , that contains the destination nodes which join before (or at) time t , and have not departed from the session yet. We aim to minimize tree cost over the session time T (without re-routing [4]), as shown in equation (1). Because total tree cost are used to measure bandwidth utilizing efficiency.

$$\min \left\{ \int_0^T Cost_{tree}(t) dt / T \right\} \quad (1)$$

3. ALGORITHM DESCRIPTION

In this work, we have considered only one type of arriving patterns, where every individual node coming separately. The session will begin only with the source node s . The connection requested from destination node z comes to source node in the form R_t . With every link l , we associate a pair $[Cost(l), t_{dep}(l)]$. This $t_{dep}(l)$ represents time that link l will have to be connected to the multicast tree. To add new node z to an existing connection, we choose a node x , in the connection, which minimizes the function

$$\begin{aligned} W(z, s) &= W(z, x) + W(x, s) \\ W(z, x) &= (1 - \eta) \times \\ &\left\{ \sum_{l_i \in P(z, x)} [(t_{ref} - t_{current}) \times Cost(l_i)] \right\} \end{aligned} \quad (2)$$

For every link l in path $P(x, s)$, if departure time of new node z is more than $t_{dep}(l_i)$ then

$$\begin{aligned} W(x, s) &= (1 - \eta) \times \\ &\left\{ \sum_{l_i \in P(x, s)} [(t_{ref} - t_{dep}(l_i)) \times Cost(l_i)] \right\} \end{aligned} \quad (3)$$

And if departure time of new node z is less than $t_{dep}(l_i)$ then

$$\begin{aligned} W(x, s) &= (\eta) \times \\ &\left\{ \sum_{l_i \in P(x, s)} [(t_{ref} - t_{dep}(l_i)) \times Cost(l_i)] \right\} \end{aligned} \quad (4)$$

Where

t_{ref} is maximum value between $t_{dep}(z)$ and maximum value of $t_{dep}(l)$ of link l that still have connected to the multicast tree.

$t_{current}$ is current time

$W(z, s)$ defines virtual cost from new node z to source node s

$P(z, s)$ means path from new node z to source node s

$W(z, x)$ defines virtual cost from new node z to leaf node x

$P(z, x)$ means path from new node z to leaf node x

$W(x, s)$ defines virtual cost from leaf node x to source node s

$P(x, s)$ means path from leaf node x to source node s

η means weight of virtual cost of each path [3]. In this work, we set it to be 0.4

The equation (2), (3) and (4) are adapted from original version of MG algorithm. The equation (4) is not presented in original version because the information that given by users is assumed to be 100 percents accurate. We use all information in the existing tree by weighted technique [3] that has ever used for dynamic multicast routing. By this technique, the original version is added flexibility to perform good performance even it is in the situation of inaccurate information.

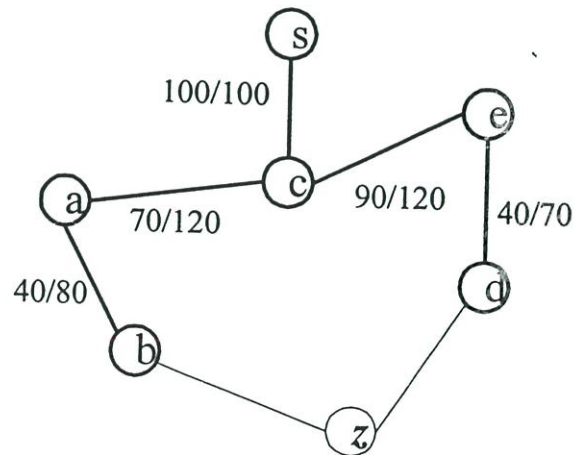


Fig. 1 New node requests to join with multicast tree

From the equation (2), it shows that new version of MG algorithm will choose the shortest path from the new node to an existing tree. And further more, it also prefer a path that have the longest live in the connection, by referring to maximum value of $t_{dep}(l)$. Example of choosing a path from the new node z to the source node s is shown in figure 1.

In figure 1, bolded link and node mean link and node that have already been existed in multicast tree. New node z requests for joining with multicast tree. Source node is s and leaf node is a , b , c and d . The x and y in the notation x/y on each link means leaving time and link cost respectively. For example of selecting path in according to our algorithm, we assign cost of link $z-b$ and $z-d$ to be 70 and 80 respectively. And the current time is 10 and leaving time of new node z is 60.

From the figure 1, t_{ref} is equal to 100 and $t_{current}$ is equal to 10. We substitute all parameters into the equation set (2), (3) and (4) to calculate virtual cost from new node z to existing multicast tree. For path $s-c-a-b-z$, the virtual cost path is 8100 and path $s-c-e-d-z$, the virtual cost is 7320. So we choose path $s-c-e-d-z$ for establish connection from new node z to multicast tree.

4. SIMULATION ENVIRONMENT AND RESULTS

At first, we use the network model that used in several works about multicast routing algorithm [3], [4], [7] and [8]. The networks are constructed by distributing n nodes across a Cartesian coordinate grid. Edges were added to the graph by considering all possible pairs (u, v) of nodes and using the equation (5) to create an edge.

$$P_e(u, v) = \beta \exp\left(\frac{-d(u, v)}{\alpha L}\right) \quad (5)$$

Here, $d(u, v)$ is the Euclidean distance between node's location, L is the maximum possible distance between two nodes, α and β are parameters in the range $0 \leq \alpha \leq 1$ and $0 \leq \beta \leq 1$. A large value of α increases the number of connections to nodes further away from it, while a value of β increases the number of edges from each node. Each graph was tested to ensure that every node is connected.

We used a probability model to generate the sequence of requests. The probability for adding a node to multicast tree, was determined by equation (6)

$$P_c(q) = \frac{\gamma(N-q)}{\gamma(N-q) + (1-\gamma)q} \quad (6)$$

q means the current number of the nodes in the multicast tree. And γ ($0 \leq \gamma \leq 1$) is parameter that determines the size of multicast group in equilibrium.

Each node had the different rate to join the multicast tree. We defined A as a set of non-joining nodes and the joining rate of the i th node was given by equation (7) and (8);

$$P_{join}(i) = (1-\mu)\mu^{i-1} / F_0 \quad \text{for } i \in A \quad (7)$$

and

$$F_0 = \sum_{i \in A} (1-\mu)\mu^{i-1} \quad (8)$$

A parameter μ ($0 \leq \mu \leq 1$) determined the bias of joining rate. In simulations, if a joining event is generated by (6), a node that will join is determined with (7) and (8). And removal of node is randomly determined within session duration T .

In our simulations, we created one hundred nodes network by using equation (5) and set $\alpha = 0.4$ and $\beta = 0.4$. We defined weight (cost) of each link by randomization. We used equation (6), (7) and (8) for setting request from any node in the network ($\gamma = 0.2$). The requests of node for joining to multicast tree must be inform their leaving time. We set the limit of inaccuracy between actual leaving time and leaving time that is informed, not over 30% of its duration time. The result of simulation is shown in figure 2 and 3.

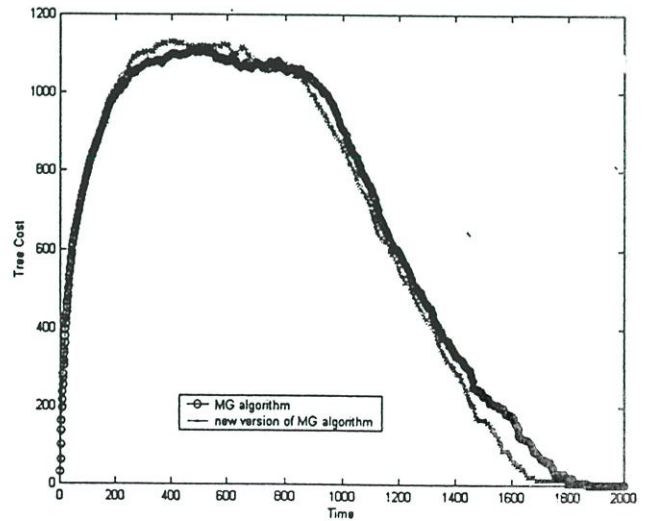


Fig.2 Tree cost versus session duration

Figure 2 shows the plot of tree cost versus time during multicast session. New version of MG algorithm spends more cost for constructing tree at first. But our algorithm gives a lower cost when the member nodes start to leave the session until

the end of session. Figure 3 shows the plot of cumulative tree cost versus time during multicast session. Cumulative tree cost of our algorithm is lower than original version of MG algorithm, when member nodes start to leave multicast session. And our algorithm gives lower overall tree cost at the end of the session.

From the simulation results in figure 2 and 3, new version of MG algorithm has more flexibility. It gives a lower overall multicast tree cost, when nodes in tree are not leaving the tree as time to be informed.

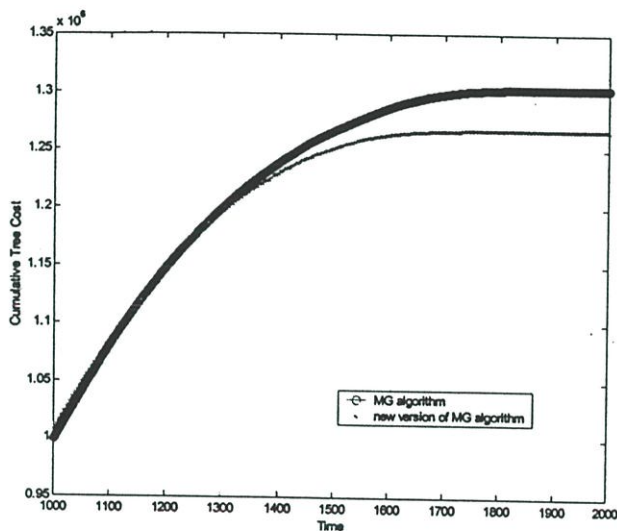


Fig.3 Cumulative tree cost versus session duration

5. CONCLUSION

In this work, we improve performance of MG multicast routing algorithm by proposing the mechanism to solve multicast routing problem with advance reservations environment. Since, MG algorithm is sensitive with precision of leaving time of nodes in multicast tree. Therefore, we proposed new version of MG algorithm that gives a better performance, when inaccuracies of leaving time of nodes in multicast tree are happened. Our algorithm try to find a shortest path from new node to source node and select a path that composed of longer live link for support changing of leaving time of both new node and leaf node.

However, our algorithm is centralized algorithm. It requires many query/reply messages between nodes when implemented in a distributed environment. The performance will be degraded when network grows up to be large-scale and it makes central server overloaded easily. In the future, we try to develop dynamic multicast routing with advance reservations that can give an optimal performance in distributed environment.

6. ACKNOWLEDGEMENT

We would like to gratitude Japan International Cooperation Agency (JICA) for research supports.

7. REFERENCES

- [1] T. Cormen, "Introduction to Algorithms", The MIT Press, Cambridge, Massachusetts, 1990.
- [2] M. Doar and I. Leslie, "How Bad is Naïve Multicast Routing," Proceedings of IEEE INFOCOMM, San Francisco, CA., pp.82-89, April 1993.
- [3] B.M. Waxman, "Routing of Multiple Connections," IEEE Journal on Selected Areas in Communications, Vol.6, No.9, pp.1617-1622 December, 1988.
- [4] D. Chakraborty, C. Pornavalai, G. Chakraborty and N. Shiratori, "An Efficient Routing to Minimize The Cost for Dynamic Multicast," IEEE APCCAS, pp.463-466, 1998.
- [5] R. Widyono, "The Design and Evaluation of Routing Algorithms for Real-time Channels," Technical Report TR-94-024, University of California at Berkeley, June, 1994.
- [6] R. Guerin and A. Orda, "Networks with Advance Reservations: The Routing Perspective," IEEE INFOCOMM, Tel-Aviv, Israel, March 26-30, 2000.
- [7] T. Asaka, T. Miyoshi, and Y. Tanaka, "Dynamic Multicast Routing Algorithm Using Predetermined Path Search," IEICE Transactions on Communications, Vol.E83-B, No.5, pp. 1128-1135, May 2000
- [8] H. Lin and S. Lai., "VTDM—A Dynamic Multicast Routing Algorithm," IEEE INFOCOMM' 98, pp.1426-1432, March-April 1998.
- [9] A. Gunabhibal, C. Pornavalai, and R. Varakulsiripunth, "Dynamic Multicast Routing under Inaccurate Links State Information", Proceeding of Student Conference on Research and Development (SCoReD 2001), Kuala Lumpur, Malaysia, February 20-21, 2001, paper no.37 on CD-ROM.

ประวัติผู้เขียน

นายอภิพล คุณาภิบาล เกิดเมื่อวันที่ 7 ตุลาคม พ.ศ.2519 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษา วิศวกรรมศาสตรบัณฑิต (อิเล็กทรอนิกส์) จากสถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง ปีการศึกษา 2539

ปี พ.ศ.2540 ได้รับการบรรจุเป็นพนักงานรัฐวิสาหกิจ องค์กรโทรศัพท์แห่งประเทศไทย ในตำแหน่งวิศวกร 4 สังกัดฝ่ายวิศวกรรมระบบโทรคม และปัจจุบันทำงานอยู่ในตำแหน่ง วิศวกร 5 สังกัดสำนักงานมาตรฐานเชื่อมต่อโครงข่าย บริษัท ทศท. คอร์ปอเรชั่น จำกัด(มหาชน)