การประสานเวลาที่ทนทานต่อความบกพร่อง
สำหรับโครงข่ายเซนเซอร์ไร้สาย

FAULT-TOLERANT TIME SYNCHRONIZATION
FOR WIRELESS SENSOR NETWORKS

ศักดิ์พงษ์ เสรีเศวตรัตน์

SAKPONG SEAREESAVETRAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2552

KMITL-2009-IT-M-001-001

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การประสานเวลาที่มีความทนทานต่อความบกพร่อง สำหรับโครงข่ายเซนเซอร์ไร้สาย

FAULT-TOLERANT TIME SYNCHRONIZATION FOR WIRELESS SENSOR NETWORKS

T095049

ศักดิ์พงษ์ เสรีเศวตรัตน์

SAKPONG SEAREESAVETRAT

เลขหมู่..............................
เลขทะเบียน 95049
วัน,เดือน,ปี 20 พ.ค. 2552

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

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2552

KMITL-2009-IT-M-001-001

FAULT-TOLERANT TIME SYNCHRONIZATION FOR WIRELESS SENSOR NETWORKS

SAKPONG SEAREESAVETRAT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE IN INFORMATION TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2009

KMITL-2009-IT-M-001-001

COPYRIGHT 2009
FACULTY OF INFORMATION TECHNOLOGY
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อวิทยานิพนธ์	การประสานเวลาที่มีความทนทานต่อความบกพร่องสำหรับโครงข่ายเซนเซอร์ไร้สาย
นักศึกษา	นายศักดิ์พงษ์ เสรีเศวตรัตน์
รหัสนักศึกษา	47066150
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
พ.ศ.	2552
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร.โชติพัชร์ ภรณวลัย
อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม	รศ.ดร.รัตติกร วรากูลศิริพันธุ์

บทคัดย่อ

ระบบโครงข่ายเซนเซอร์ไร้สายกำลังได้รับความสนใจเป็นอย่างมากทั้งในด้านการวิจัย และการใช้งานในชีวิตประจำวัน ซึ่งระบบนี้ต้องการการประสานเวลาอย่างสม่ำเสมอเพื่อจะทำงานได้อย่างถูกต้องและมีประสิทธิภาพ โพรโทคอลทีพีเอสเอ็น (TPSN) ใช้เทคนิคการแลกเปลี่ยนข้อมูลเวลาเพื่อประสานเวลาระหว่างคู่โหนด แล้วทำการประสานเวลาต่อกันไปตามโครงสร้างลำดับชั้นซึ่งเหมาะสมกับระบบโครงข่ายเซนเซอร์ไร้สายที่มีทรัพยากรอันจำกัด โดยมีค่าผิดพลาดหลังการประสานเวลาระหว่างคู่โหนดต่ำกว่า 20 ไมโครวินาที แต่เนื่องจากแต่ละโหนดประสานเวลาต่อตามโครงสร้างลำดับชั้น จึงมีการสะสมของค่าเวลาผิดพลาดตามจำนวนฮ็อพที่เพิ่มขึ้นและระยะเวลาที่นานขึ้น และเมื่อในระบบมีบางโหนดที่มีนาฬิกาไม่เที่ยงตรงสูงกว่าปกติ (เราเรียกว่า Fault Clock Node หรือ ฟอลคล็อกโหนด) ซึ่งมีผลให้ค่าเวลาคลาดเคลื่อนสูงนี้ถูกสะสมต่อไปตามการประสานเวลา และส่งผลให้ค่าเวลาเฉลี่ยผิดพลาดจากการประสานเวลาของโครงข่ายโดยรวมสูงขึ้น

งานวิทยานิพนธ์นี้ทำการศึกษาปัจจัยของค่าผิดพลาดจากการประสานเวลา และเสนออัลกอลิทึมที่อาศัยหลักการทางสถิติเพื่อระบุฟอลคล็อกโหนด จากนั้นปรับปรุงโพรโทคอลทีพีเอสเอ็นให้โหนดพยายามหลีกเลี่ยงการประสานเวลากับฟอลคล็อกโหนด ซึ่งจะทำให้การประสานเวลาของโครงข่ายโดยรวมมีความทนทานต่อสถานการณ์ที่มีฟอลคล็อกโหนดปะปนอยู่มากขึ้น และจากผลการทดลองกับแบบจำลองระบบ โพรโทคอลที่นำเสนอมีค่าเฉลี่ยผิดพลาดจากการประสานเวลาดีกว่าโพรโทคอลทีพีเอสเอ็นแบบเน้นโครงสร้างลำดับชั้นที่สั้น 1-2%, 4-6% และ 4-6% เมื่อระบบมีฟอลคล็อกโหนดที่มีอัตราค่าความคลาดเคลื่อนสัมพัทธ์ 3, 6, 10 เท่าตามลำดับและมีฟอลคล็อกโหนดปะปนอยู่จำนวนร้อยละ 5-30 ของโหนดทั้งหมดในระบบ และดีกว่า 18-19%, 20-23% และ 21-23% ตามลำดับเมื่อเปรียบเทียบกับโพรโทคอลทีพีเอสเอ็นดั้งเดิมที่เลือกโหนดแม่แบบสุ่ม

Thesis	Fault-Tolerant Time Synchronization for Wireless Sensor Networks
Student	Mr. Sakpong Seareesavetrat
Student ID.	47066150
Degree	Master of Science
Programme	Information Technology
Year	2009
Thesis Advisor	Assoc. Prof. Dr. Chotipat Pornavalai
Thesis Co-advisor	Assoc. Prof. Dr. Ruttikorn Varakulsiripunth

ABSTRACT

Wireless sensor networks (WSN) have received a lot of attention recently due to their enormous potential in different applications. Many applications in WSN require time synchronization. Timing-sync Protocol for Sensor Networks (TPSN) can synchronize a pair of neighboring motes to an average synchronization error less than 20 μs. Also providing network-wide time synchronization in a sensor network by a pair wise synchronization performed along the edges of a hierarchical structure. However, the drawback of TPSN is cumulative synchronization error that will be highly influenced by delays during synchronization along multi-hops. When the error of each pair synchronized nodes is strongly affected by clock drift from inside node. In practical, node can have a bad clock (we called Fault Clock Node or FCN). For example when node has low power energy the node's clock may work abnormally, which could cause high synchronization error and create mores time error along multi-hops tree.

In this paper, we studied factors of synchronization error and present fault-tolerant time-synchronization protocol for wireless sensor network (FTTS-WSN) which works in a way similar to NTP. The protocol is able to detect itself node in hierarchical tree by a statistic method, then modified existing TPSN to avoid of FCN. Our simulation results show that the average synchronization error using the proposed protocol is 1-2%, 4-6% and 4-6% less than TPSN shortest tree fashion (parent selection algorithm by shortest level), in case of FCN has clock drift rate x3, x6 and x10 times of normal drift rate respectively and amount of FCN is 5-30% from total nodes. Also the average synchronization error less than 18-19%, 20-23% and 21-23% compared with TPSN traditional (parent selection algorithm by randomize).

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก รศ.ดร. โชติพัชร์ ภรณวลัย อาจารย์ผู้ควบคุมวิทยานิพนธ์, รศ. ดร. รัตติกร วรากูลศิริพันธุ์ อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม ตลอดจนคณาจารย์จากห้องวิจัย Communication Network Lab สำนักวิจัยการสื่อสารและเทคโนโลยีสารสนเทศ (ReCCIT) ข้าพเจ้ารู้สึกทราบซึ้งในความอนุเคราะห์จากอาจารย์ทุกท่าน และขอขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณคณาจารย์คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณเงินสนับสนุนเพื่อเป็นทุนวิจัยจากคณะเทคโนโลยีสารสนเทศ โดยข้าพเจ้าได้รับในตำแหน่งผู้ช่วยวิจัยของโครงงานวิจัย รศ.ดร. โชติพัชร์ ภรณวลัย

ขอขอบคุณห้องวิจัยและปฏิบัติการ Internet Intelligent and Informatics Research Lab (I-Cube) คณะเทคโนโลยีสารสนเทศ และห้องวิจัย Communication Network Lab (ReCCIT) ที่ได้สนับสนุนเครื่องคอมพิวเตอร์ ตลอดจนข้อมูล และหนังสือต่างๆ ที่ใช้ในการทำวิจัย

ขอขอบคุณ พี่ๆ เพื่อนๆ น้องๆ ในสาขาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ และห้องวิจัย Communication Network Lab (ReCCIT) ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

ขอขอบคุณบัณฑิตศึกษาและบัณฑิตวิทยาลัย คณะเทคโนโลยีสารสนเทศที่ให้ความช่วยเหลือในการช่วยดำเนินการและประสานงาน สำหรับการสอบและจัดทำรูปเล่มวิทยานิพนธ์จนแล้วเสร็จ

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่อง ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

ศักดิ์พงษ์ เสรีเศวตรัตน์

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	2
1.3 สมมติฐานของการศึกษา	2
1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย	4
1.5 การเปรียบเทียบวิธีการที่นำเสนอกับวิธีการแบบพื้นฐาน	6
1.6 ขอบเขตการวิจัย	7
1.7 ขั้นตอนการศึกษา	8
1.8 ข้อตกลงเบื้องต้น	8
1.9 คำจำกัดความที่ใช้ในการศึกษา	8
บทที่ 2 ทฤษฎีพื้นฐานที่ใช้ในการวิจัย	11
2.1 โพรโทคอลประสานเวลาสำหรับระบบเครือข่ายเซนเซอร์ไร้สาย	11
2.1.1 Timing-sync Protocol for Sensor Networks	11
2.1.2 Cluster-based Hierarchical Time Synchronization for Multi-hop Wireless Sensor Networks	14
2.1.3 Reliable Time Synchronization Protocol for Wireless Sensor Network	15
2.2 คุณลักษณะและการวิเคราะห์ปัจจัยที่ส่งผลต่อค่าผิดพลาดจากวิธีการประสานเวลา	15
2.2.1 ผลกระทบจากค่าตัวแปรเวลาระหว่างการสื่อสาร	15
2.2.2 ผลกระทบจากนาฬิกาที่เที่ยงตรงทั้งปกติและไม่ปกติ	17

สารบัญ (ต่อ)

หน้า

2.3 การประเมินสมรรถนะของโพรโทคอลการประสานเวลา.............................. 18

2.3.1 ค่าผิดพลาดจากการประสานเวลา 18

2.3.2 ค่าเฉลี่ยผิดพลาดจากการประสานเวลาของทั้งระบบ.............................. 19

2.3.3 ปริมาณข้อมูลของโพรโทคอลการประสานเวลา 20

2.4 การวิเคราะห์ผลกระทบจากการมีโหนดซึ่งมีนาฬิกาไม่เที่ยงตรงมากกว่าปกติ...... 20

บทที่ 3 วิธีการระบุและหลีกเลี่ยงฟอลคล็อกโหนด 24

3.1 วิธีการพิจารณาฟอลคล็อกโหนด 24

3.2 อัลกอลิทึมการเลือกโหนดแม่ที่ดีที่สุดโดยหลีกเลี่ยงฟอลคล็อกโหนด 31

3.3 การปรับปรุงโพรโทคอลทีพีเอสเอ็นให้ทนทานต่อฟอลคล็อกโหนด 32

3.3.1 การสร้างโครงสร้างลำดับชั้น 32

3.3.2 การประสานเวลาให้สอดคล้องกันทั้งโครงข่าย.................................... 35

3.3.3 การจัดการกับกรณีพิเศษในการประสานเวลา 37

บทที่ 4 การประเมินสมรรถนะของระบบ 39

4.1 แบบจำลองที่ใช้ในการจำลองระบบ 39

4.2 ค่าชี้วัดสมรรถนะ 39

4.3 พารามิเตอร์ที่ใช้ในการจำลองระบบ... 39

4.4 ปัจจัยที่มีผลต่อสมรรถนะของระบบ.. 40

4.5 สมรรถนะของระบบ.. 47

4.5.1 สมรรถนะเอเอสอีและปัจจัยที่สนับสนุน 47

4.5.2 สมรรถนะค่าผิดพลาดจากการประสานเวลาสูงสุดและปัจจัยสนับสนุน 50

4.5.3 สมรรถนะในมิติความหนาแน่นของเครือข่าย 51

4.5.4 ความถูกต้องของการระบุฟอลคล็อกโหนดมีผลต่อสมรรถนะ................. 54

4.5.5 วิเคราะห์ผลกระทบที่ส่งผลต่อสมรรถนะการระบุฟอลคล็อกโหนด......... 58

4.5.6 วิเคราะห์สมรรถนะในมิติของโครงสร้างลำดับชั้นและเวลา 60

4.5.7 สมรรถนะปริมาณข้อมูลการประสานเวลา 65

สารบัญ (ต่อ)

หน้า

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ.. 69

5.1 สรุปผลการวิจัย.. 69

5.2 ข้อเสนอแนะ และแนวทางที่ควรพัฒนาต่อ.. 70

5.2.1 ตัวอย่างแอพพลิเคชันที่มีความต้องการความแม่นยำของเวลาสูง 70

5.2.2 วิธีการระบุฟอลคล็อกโหนดวิธีอื่น.. 71

5.2.2.1 Grubbs Test 71

5.2.2.2 Voting Algorithm.. 71

5.2.3 วิธีการอาศัยการสื่อสารที่มีอยู่เพื่อแลกเปลี่ยนข้อมูลเวลา...................... 73

5.2.4 ตำแหน่งและลักษณะการกระจายตัวของฟอลคล็อกโหนด.................... 74

บรรณานุกรม 76

ภาคผนวก 78

ภาคผนวก ก. ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่ 79

ประวัติผู้เขียน.. 85

สารบัญตาราง

ตารางที่ หน้า

4.1 พารามิเตอร์ที่สำคัญในการจำลองระบบ... 40

4.2 ผลกระทบจากฟอลคล็อกโหนดต่อค่าเอเอสอี 44

4.3 เปรียบเทียบสมรรถนะค่าเอเอสอีของโพรโทคอลประสานเวลาที่ค่าความคลาดเคลื่อนของฟอลคล็อกโหนดระดับต่างๆ (หน่วยไมโครวินาที) 46

4.4 สมรรถนะค่าเอเอสอีของโพรโทคอลประสานเวลาที่ความหนาแน่นเครือข่ายระดับต่างๆ... 53

สารบัญรูป

รูปที่		หน้า
1.1	การวิเคราะห์ที่ผิดพลาดเมื่อโหนดเซนเซอร์ปราศจากการประสานเวลา	1
1.2	เวลานาฬิกาของแต่ละโหนดจะแตกต่างกันไปเรื่อยๆ เมื่อเทียบกับเวลาอุดมคติ	3
1.3	การประสานเวลาแบบทีพีเอสเอ็นที่ต่อเนื่องเป็นทอดๆ ตามโครงสร้างลำดับชั้น	4
1.4	การสะสมเวลาที่ผิดพลาดจากการประสานเวลาแบบมีโครงสร้างลำดับชั้น	5
1.5	ตัวอย่างวิธีการเลือกโหนดแม่	7
2.1	ตัวอย่างโครงสร้างลำดับชั้นและโพรโทคอลการประสานเวลาทีพีเอสเอ็น	12
2.2	วิธีทีดับบิวเอ็มอี การแลกเปลี่ยนข้อมูลเวลาระหว่างสองโหนด	13
2.3	โพรโทคอลการประสานเวลาอาร์บีเอส	14
2.4	โครงข่ายแบบกลุ่มของโพรโทคอลซีเอชทีเอส	14
2.5	การเลือกโหนดแม่ของโหนด C ตามโพรโทคอลอาร์ทีเอสพี	15
2.6	โครงสร้างตัวแปรเวลาระหว่างการสื่อสาร	17
2.7	ผลกระทบจากความเที่ยงตรงของนาฬิกาทั้งปกติและไม่ปกติ	18
2.8	ค่าผิดพลาดจากการประสานเวลาเชิงสะสม หรือค่าเอสอี	19
2.9	ผลกระทบของบีพีเอ็นและไอบีพีเอ็นต่อค่าเอเอสอี	22
2.10	สมรรถนะที่ดีขึ้นเพราะค่าเอเอสอีลดลง เมื่อพยายามหลีกเลี่ยงบีพีเอ็นและไอบีพีเอ็น	23
3.1	ผลกระทบของฟอลคล็อกโหนดที่มีผลต่อค่าเอเอสอี	25
3.2	สมรรถนะโพรโทคอลประสานเวลาเมื่อในระบบมีฟอลคล็อกโหนดปะปนอยู่	26
3.3	กระบวนการการระบุฟอลคล็อกโหนด	28
3.4	ตัวอย่างกระบวนการการระบุฟอลคล็อกโหนดที่โหนด C	29
3.5	ตัวอย่างกระบวนการการระบุฟอลคล็อกโหนดที่โหนด B	29
3.6	ตัวอย่างค่า *LV* และ *BIL* ในโครงสร้างลำดับชั้น	30
3.7	วิธีการ การเลือกโหนดแม่ที่ดีที่สุด	32
3.8	โครงสร้างข้อมูลโครงสร้างลำดับชั้น	33
3.9	วิธีการ การแก้ไขค่า *LV* และ *BIL* ให้ทันสมัย	34
3.10	ตัวอย่างโครงสร้างลำดับชั้นของระบบโครงข่ายเซนเซอร์ไร้สาย	35
3.11	การแลกเปลี่ยนข้อมูลเวลา (ทีดับบิวเอ็มอี) ระหว่างโหนด A และโหนด R	36
3.12	โครงสร้างข้อมูลการประสานเวลา	36

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.1 จำนวนโหนดที่มากขึ้นทำให้ค่าเอเอสอีสูงขึ้น 41

4.2 จำนวนโหนดที่มากขึ้นทำให้ความยาวของโครงสร้างลำดับชั้นยาวขึ้น 41

4.3 จำนวนโหนดที่มากขึ้นทำให้ระยะเวลาการประสานเวลาทั้งระบบนานขึ้น 42

4.4 ค่าเอเอสอีลดลงจากโพรโทคอล TPSN (Shortest tree) 42

4.5 ความยาวโครงสร้างลดลงจากโพรโทคอล TPSN (Shortest tree) 43

4.6 เวลาในการประสานเวลาทั้งระบบสั้นลงจากโพรโทคอล TPSN (Shortest tree) 43

4.7 ผลกระทบจากฟอลคล็อกโหนดต่อค่าเอเอสอี 44

4.8 ค่าเอเอสอีเมื่อฟอลคล็อกโหนดมีค่าความคลาดเคลื่อนมากกว่าปกติ 3 เท่า 45

4.9 โพรโทคอลการประสานเวลาเมื่อมีฟอลคล็อกโหนดในเครือข่าย 10% 48

4.10 โพรโทคอลการประสานเวลาเมื่อมีฟอลคล็อกโหนดในเครือข่าย 5% 48

4.11 ระยะเวลาที่ใช้ในการประสานเวลา 49

4.12 จำนวนชั้นหรือความยาวของโครงสร้างลำดับชั้น 49

4.13 ค่าผิดพลาดจากการประสานเวลาสูงสุด 51

4.14 ชั้นของโหนดที่มีค่าผิดพลาดจากการประสานเวลาสูงสุด 51

4.15 ค่าเอเอสอีของการประสานเวลาที่ความหนาแน่นเครือข่ายระดับต่างๆ 52

4.16 ความยาวของโครงสร้างลำดับชั้นที่ความหนาแน่นเครือข่ายระดับต่างๆ 53

4.17 ระยะเวลาที่ใช้ประสานเวลาของทั้งระบบที่ความหนาแน่นเครือข่ายระดับต่างๆ 54

4.18 ค่าผิดพลาดจากการประสานเวลาสูงสุดที่ความหนาแน่นเครือข่ายระดับต่างๆ 54

4.19 จำนวนฟอลคล็อกโหนดที่ความหนาแน่นเครือข่ายระดับต่างๆ 55

4.20 สมรรถนะค่าเอเอสอีจากวิธีการระบุฟอลคล็อกโหนดของโพรโทคอลที่นำเสนอ 56

4.21 สมรรถนะค่าเอเอสอีที่ความหนาแน่นเครือข่ายระดับต่างๆ 57

4.22 จำนวนฟอลคล็อกโหนดและโหนดที่ประสานเวลาต่อมาจากฟอลคล็อกโหนด 58

4.23 จำนวน BPN หรือ IBPN ที่มีโหนดอื่นประสานเวลาตาม 58

4.24 ความเหมาะสมในการคำนวณค่า Faulty_Drift ที่มีผลต่อค่าเอเอสอี 59

4.25 ความเหมาะสมในการคำนวณค่า Faulty_Drift ที่มีผลต่อค่าผิดพลาดสูงสุด 59

4.26 ระยะเวลา Synced_Time ส่งผลต่อสมรรถนะการระบุฟอลคล็อกโหนด 60

4.27 จำนวนโหนดลูกเฉลี่ยต่อโหนดแม่ในแต่ละโครงสร้างลำดับชั้น 61

สารบัญรูป (ต่อ)

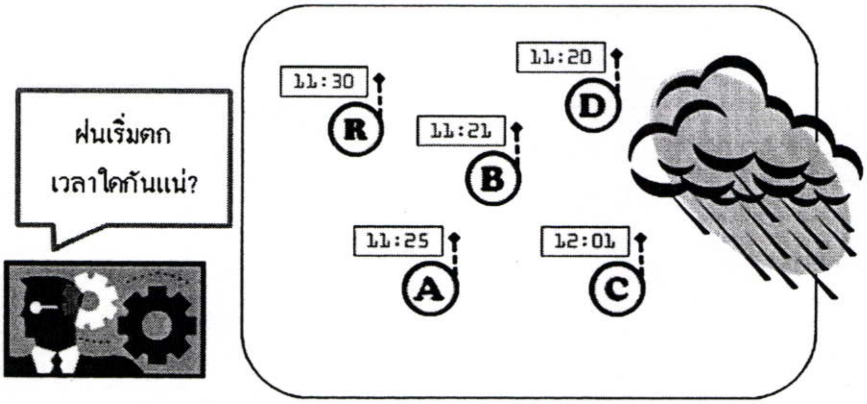
รูปที่		หน้า
4.28	จำนวนโหนดที่เป็นตัวเลือกในแต่ละโครงสร้างลำดับชั้น	62
4.29	อัตราการกระจายโหนดแม่ในโหนดที่เป็นตัวเลือกในแต่ละโครงสร้างลำดับชั้น	62
4.30	ค่าเฉลี่ยเอสอีของโหนดในแต่ละโครงสร้างลำดับชั้น	63
4.31	ค่าชั้นโดยเฉลี่ยของโหนดที่ได้รับการประสานเวลา ณ เวลาต่างๆ	64
4.32	ค่าจำนวนโหนดโดยเฉลี่ยที่ได้รับการประสานเวลา ณ เวลาต่างๆ	64
4.33	ค่าเฉลี่ยเอสอีของโหนด ณ เวลาที่ผ่านไป	65
4.34	ปริมาณข้อมูลที่ถูกส่ง หากไม่นับข้อมูลที่ใช้ระบุฟอลคล็อกโหนด	66
4.35	ปริมาณข้อมูลที่ถูกส่ง เมื่อนับรวมข้อมูลที่ใช้ระบุฟอลคล็อกโหนด	67
4.36	ปริมาณข้อมูลที่ชนกัน กรณีรวมข้อมูลที่ใช้ระบุฟอลคล็อกโหนด	68
5.1	ตำแหน่งของโหนดเซนเซอร์และมือปืน	70
5.2	ค่าเวลาที่โหนด C แตกต่างจากโหนดข้างเคียง	72
5.3	ค่าเวลาที่โหนด B แตกต่างจากโหนดข้างเคียง	73
5.4	ตำแหน่งของโหนดเซนเซอร์ที่สุ่มให้มีการกระจายตัวอิสระ	74
5.5	ตำแหน่งของฟอลคล็อกโหนดที่สุ่มให้กระจายตัวใกล้โหนดรูต	75
5.6	ตำแหน่งของฟอลคล็อกโหนดที่สุ่มให้กระจายตัวไกลจากโหนดรูต	75

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การประสานเวลา (Time Synchronization) มีความจำเป็นต่อระบบงานแบบกระจายทุกรูปแบบ โดยเฉพาะอย่างยิ่งต่อระบบโครงข่ายเซนเซอร์ไร้สาย (Wireless Sensor Networks : WSNs) ซึ่งการวิเคราะห์ข้อมูลโดยส่วนใหญ่นั้นอยู่บนพื้นฐานเวลาที่แม่นยำ ดังตัวอย่างเช่น การคำนวณความเร็วการเคลื่อนที่ของวัตถุ, การคำนวณเวลาการเคลื่อนที่ของเสียง, การวิเคราะห์รูปแบบการกระจายตัว, การจำแนกข้อมูลที่ซ้ำซ้อนซึ่งได้รับจากเซนเซอร์หลายๆ โหนด, ข้อตกลงด้านความปลอดภัยของข้อมูล (Secure Cryptographic Scheme) ที่อาศัยเวลาที่แม่นยำเพื่อยืนยันความถูกต้องหรือที่เรียกว่า Message Authentication Code : MAC, การประสานความร่วมมือกันทำงาน และตลอดจนการค้นหาสาเหตุของปัญหาโดยเรียงเหตุการณ์ตามเวลา เป็นต้น

รูปที่ 1 แสดงตัวอย่างการตรวจจับตำแหน่งและเวลาที่เกิดฝนตก เมื่อโหนดเซนเซอร์มีเวลาที่ไม่สอดคล้องกัน ทำให้การวิเคราะห์ผิดพลาดอาจจะเข้าใจว่าฝนตกเกิดในพื้นที่เซนเซอร์ D แล้วจึงเคลื่อนไปยังพื้นที่เซนเซอร์ B และ C ทั้งที่โดยความจริงแล้วฝนตกในพื้นที่เซนเซอร์ D และ C เกือบจะพร้อมกัน



รูปที่ 1.1 การวิเคราะห์ที่ผิดพลาดเมื่อโหนดเซนเซอร์ปราศจากการประสานเวลา

เนื่องด้วยโหนดเซนเซอร์ไร้สาย หรือขอเรียกว่าโหนด นั้นมีข้อจำกัดอันควรทราบ [1], [2] ได้แก่ หน่วยประมวลผลมีขนาดเล็กไม่เหมาะสมจะประมวลผลงานที่ซับซ้อน, การสื่อสารแบบไร้สายที่มีความเร็วต่ำ, โพรโทคอลดาต้าลิงค์เป็นแบบหลีกเลี่ยงการชนกันและอาจไม่มีบัฟเฟอร์พักข้อมูลรอส่ง (CSMA/CA), ขนาดพื้นที่หน่วยความจำมีขนาดเล็ก และสำคัญที่สุดคือพลังงานที่ได้

จากแบตเตอรี่ที่มีปริมาณจำกัด หากหมดแล้วก็จะยากต่อการเปลี่ยนใหม่ นั่นก็คือสถานะโหนดจะตายถาวร ดังนั้นงานวิจัยจำนวนมากจึงเน้นพัฒนาประสิทธิภาพพร้อมทั้งคำนึงถึงข้อจำกัดที่สำคัญเหล่านี้ด้วย

โพรโทคอลประสานเวลาสำหรับระบบโครงข่ายคอมพิวเตอร์ในปัจจุบัน ดังเช่น เน็ตเวิร์กทามโพรโทคอล (Network Time Protocol : NTP : เอ็นทีพี) [3] และระบบระบุตำแหน่งทางภูมิศาสตร์ (Global Positioning System : GPS : จีพีเอส) [4] เป็นต้น ล้วนไม่เหมาะสมที่จะนำมาใช้ระบบโครงข่ายเซนเซอร์ไร้สายโดยทันที เนื่องจากข้อจำกัดที่กล่าวมาข้างต้น [2] ดังเช่น ข้อจำกัดระยะทางการสื่อสาร, ข้อจำกัดด้านพลังงาน, ความไม่เที่ยงตรงของนาฬิกา (Clock skew) อันมีสาเหตุจากคุณภาพของควอทซ คริสตัลและผลกระทบจากสิ่งแวดล้อม, ระบบจีพีเอสก็มีราคาสูงมากเมื่อเทียบกับราคาโหนดเซนเซอร์ เป็นต้น จึงจำเป็นต้องปรับแต่งวิธีการที่มีอยู่ หรือคิดค้นพัฒนาและวิจัยวิธีการใหม่ สำหรับวิธีการประสานเวลาให้เหมาะสมกับระบบโครงข่ายเซนเซอร์ไร้สาย

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

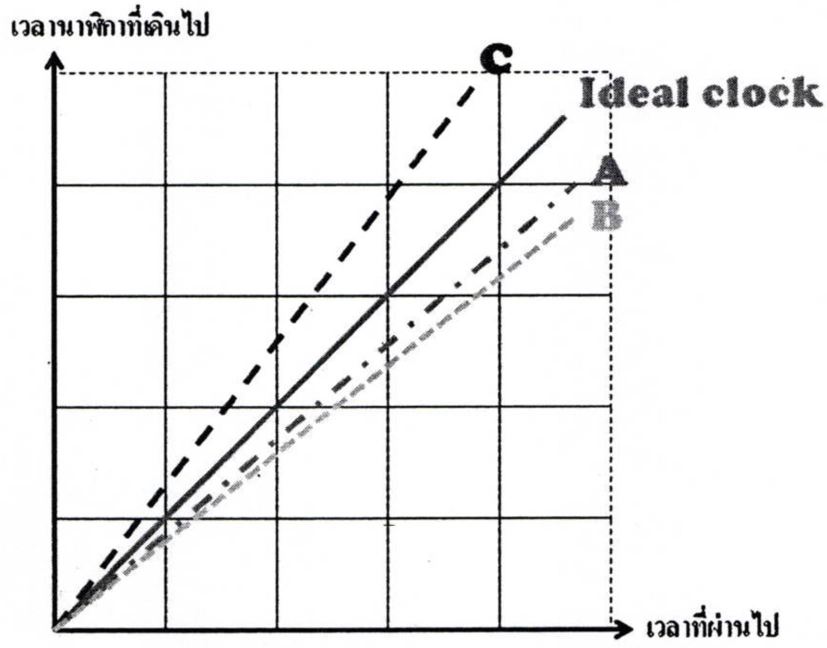
1. เพื่อศึกษาตัวชี้วัดสมรรถนะของโพรโทคอลการประสานเวลา ของระบบโครงข่ายเซนเซอร์ไร้สาย โดยตัวชี้วัดสมรรถนะที่เรากล่าวถึงนี้คือ ค่าเฉลี่ยผิดพลาดจากการประสานเวลา (Average Synchronization Error : ASE : เอเอสอี)
2. เพื่อศึกษาค่าชี้วัดสมรรถนะ “เอเอสอี” (ตามข้อที่ 1) ในสถานการณ์ที่โหนดมีนาฬิกาเที่ยงตรงระดับปกติ และในสถานการณ์ที่มีโหนดซึ่งมีนาฬิกาไม่เที่ยงตรงปะปนอยู่ในระบบ
3. เพื่อศึกษาและหาปัจจัยที่ส่งผลต่อค่าชี้วัดสมรรถนะ “เอเอสอี” ตามสถานการณ์ทั้งสอง (ตามข้อที่ 2)
4. เพื่อวิจัยและพัฒนาวิธีการที่มีประสิทธิภาพ โดยการลดค่า “เอเอสอี” ลงเมื่อเทียบกับวิธีการที่มีอยู่ในปัจจุบันโดยเพิ่มภาระหรือเกิดผลเสียต่อระบบน้อยที่สุด
5. เพื่อพัฒนาปรับปรุงโพรโทคอลการประสานเวลาสำหรับระบบโครงข่ายเซ็นเซอร์ไร้สายให้มีความทนทานต่อโหนดซึ่งมีนาฬิกาไม่เที่ยงตรงปะปนอยู่ในระบบ

1.3 สมมติฐานของการศึกษา

ปกติแล้วความเที่ยงตรงของนาฬิกานั้นขึ้นอยู่กับคุณภาพของควอทซ คริสตัล [5], [6] ซึ่งมีค่าความคลาดเคลื่อนประมาณ 20 มิลลิวินาทีต่อชั่วโมง (หรือ 5.5 ไมโครวินาทีต่อวินาที) ดังนั้นเมื่อเวลาผ่านไประยะหนึ่ง เพื่อให้ทุกโหนดมีเวลาที่สอดคล้องกันมากที่สุด หรือสอดคล้องกันเพียงพอ

ต่อการใช้งานตามความต้องการของแอพพลิเคชั่น นาฬิกาเหล่านี้จึงควรทำการประสานเวลาเพื่อให้เวลาที่แตกต่างกันนั้นลดลงหรือหมดไป

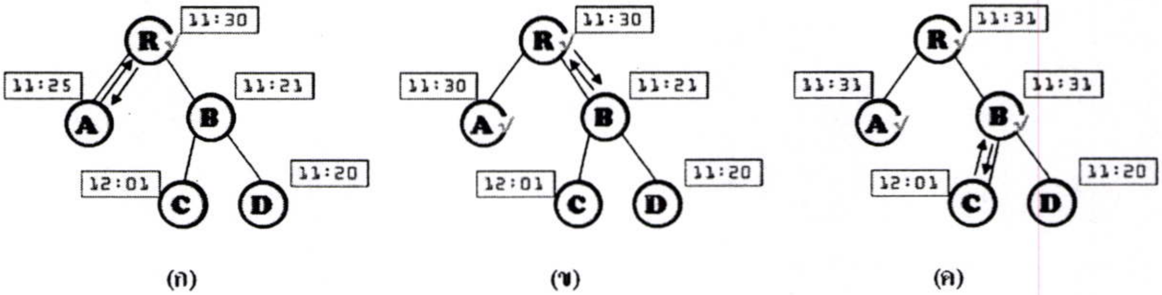
รูปที่ 1.2 แสดงลักษณะการเบี่ยงเบนเวลาของนาฬิกาโหนด A, B และ C ที่ไม่เที่ยงตรง ทำให้เมื่อเวลาผ่านไป เวลาของนาฬิกาเหล่านี้จึงต่างกันมากขึ้นเรื่อยๆ



รูปที่ 1.2 เวลานาฬิกาของแต่ละโหนดจะแตกต่างกันไปเรื่อยๆ เมื่อเทียบกับเวลาอุดมคติ

นอกจากความไม่เที่ยงตรงของนาฬิกาที่ทำให้เวลาของแต่ละโหนดแตกต่างกันมากขึ้นตามระยะเวลาที่ผ่านไป วิธีการ Timing-Sync Protocol for Sensor Networks (TPSN : ทีพีเอสเอ็น) ทำการทดลองจริงระหว่างโหนดเซนเซอร์ไร้สายรุ่น MICA 2 โหนด พบว่าหลังการประสานเวลา (Synchronization Error) จะมีค่าผิดพลาด (drift error) ประมาณ 17 ไมโครวินาที [7] รูปที่ 1.3 แสดงตัวอย่างโครงสร้างลำดับชั้นตามโพรโทคอลทีพีเอสเอ็น และทุกโหนดประสานเวลากับโหนด R ตามโครงสร้างลำดับชั้นเป็นทอดๆ รูปที่ 1.3 (ก) โหนด A ทำการประสานเวลากับโหนด R ที่เป็นโหนดเวลาอ้างอิงเป็นโหนดคู่แรก รูปที่ 1.3 (ข) โหนด B หาจังหวะที่จะไม่ส่งข้อมูลชนกับโหนด A เพื่อทำการประสานกับโหนด R และรูปที่ 1.3 (ค) เมื่อโหนด B ประสานเวลาเสร็จ โหนด C และ D ที่เป็นโหนดลูกของโหนด B จะทำการประสานเวลากับโหนด B ตามลำดับต่อไป

ซึ่งเมื่อประสานเวลาต่อกันในแบบทำต่อเนื่องกันเป็นทอดๆ ผลที่ตามมาคือจะเกิดการสะสมเวลาที่แตกต่างมากยิ่งขึ้น ดังนั้นเมื่อเปรียบเทียบเวลาของทุกโหนดกับรูตโหนด (Root Node หรือ Sink Node) ค่าเอเอสอี (Average Synchronization Error : ASE) จึงมีค่าสูงขึ้นตามจำนวนโหนดและจำนวนชั้นในโครงสร้างลำดับชั้น



รูปที่ 1.3 การประสานเวลาแบบทีพีเอสเอ็นที่ต่อเนื่องเป็นทอดๆ ตามโครงสร้างลำดับชั้น

นาฬิกาโหนดใดๆ สามารถเกิดสภาวะนาฬิกาไม่เที่ยงตรงมากกว่าปกติทั้งชั่วคราวและถาวรได้ โดยมีผลกระทบจากสิ่งแวดล้อมหรือความผิดปกติภายในนาฬิกาและโหนดเอง ซึ่งเราขอเรียกโหนดแบบนี้ว่า Fault Clock Node หรือ ฟอลคล็อกโหนด เมื่อมีฟอลคล็อกโหนดปะปนระหว่างการประสานเวลาแบบทีพีเอสเอ็น จะทำให้ค่าผิดพลาดหลังการประสานเวลา ตลอดจนค่า เอเอสอีสูงขึ้นด้วย

เพื่อลดผลกระทบของฟอลคล็อกโหนด เราได้เสนอวิธีการที่ไม่ซับซ้อนเพื่อระบุและหลีกเลี่ยงฟอลคล็อกโหนด โดยแต่ละโหนดคำนวณเวลาแตกต่างเฉลี่ยเทียบกับโหนดข้างเคียง แล้วโหนดใดที่มีค่าเฉลี่ยนี้สูงกว่าค่าที่เราได้จากการศึกษาว่าสมควรต่อการเฝ้าระวัง โหนดนั้นจะระบุตัวเองเป็นฟอลคล็อกโหนด และด้วยพื้นฐานการไว้ใจกัน สถานะฟอลคล็อกโหนดจะถูกประกาศเพื่อให้โหนดข้างเคียงสามารถหลีกเลี่ยงฟอลคล็อกโหนดดังกล่าว

การสร้างโครงสร้างลำดับชั้นที่สามารถหลีกเลี่ยงฟอลคล็อกโหนด (หรือการพยายามให้ฟอลคล็อกโหนดเป็นใบของโครงสร้างต้นไม้) จะสามารถลดการสะสมค่าผิดพลาดจากวิธีการประสานเวลาจากการเชื่อมต่อกันเป็นทอดๆ ลงได้ ซึ่งค่าเอเอสอีที่เคยสูงจากวิธีการทั่วๆ ไปก็จะลดลง อีกทั้งในงานวิจัยต่างๆ ในปัจจุบันพยายามลดโครงสร้างลำดับชั้นให้สั้น เพื่อลดค่าผิดพลาดสะสมในทางอ้อม แต่การที่พยายามหลีกเลี่ยงฟอลคล็อกโหนดจนอาจทำให้โครงสร้างลำดับชั้นยาวกว่า ก็สามารถลดค่าผิดพลาดสะสมได้ดีกว่าไม่แพ้กัน จุดนี้เองที่เป็นความแตกต่างที่สำคัญของงานชิ้นนี้เมื่อเปรียบเทียบกับงานที่มีอยู่ในปัจจุบัน

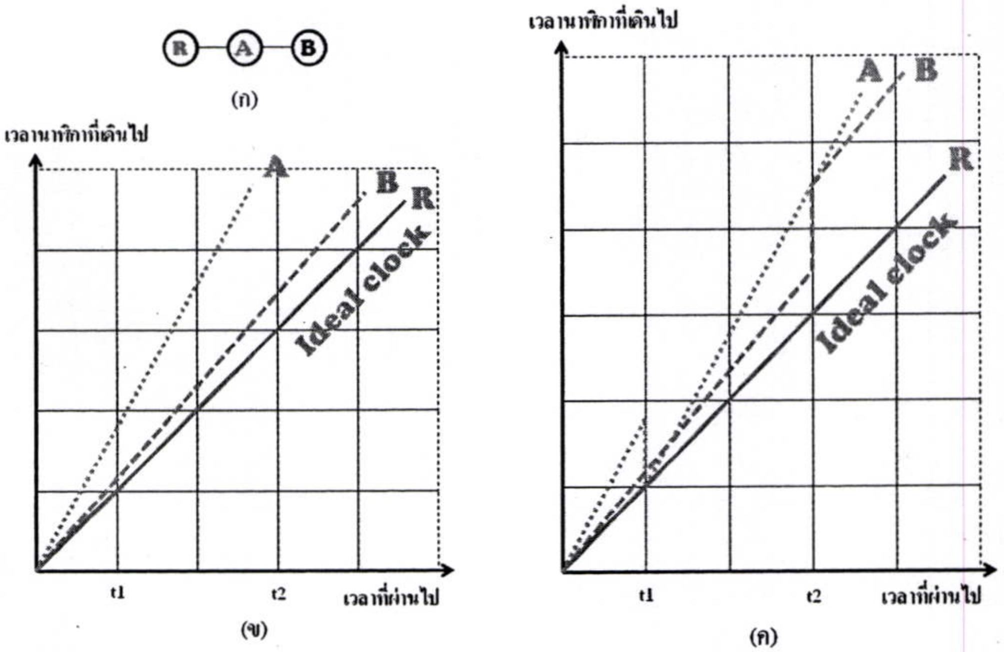
1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

1. ในการประสานเวลา ต้องมีการบอกเวลาของโหนดอ้างอิงเพื่อให้โหนดอื่นๆ ได้ปรับเวลาตาม แต่เมื่อข้อมูลถูกสื่อสารผ่านสื่อที่ใช้ร่วมกัน จึงเกิดตัวแปรขึ้นระหว่างการสื่อสารที่ยากจะหลีกเลี่ยงและกำจัดให้หมดไปได้ เพื่อลดค่าผิดพลาดจากวิธีการประสานเวลา จึงมีวิธีการที่

หลากหลายพยายามกำจัดตัวแปรเวลาขณะสื่อสารข้อมูล (Unreliable network transmission) รวมไปถึงตัวแปรเวลาขณะจัดการกับข้อมูล (Unbounded message latencies) [5] ,[7] และ [8]

2. เพื่อให้ทุกโหนดสามารถทำการประสานเวลาได้ แม้จะไม่สามารถสื่อสารโดยตรงกับโหนดอ้างอิง จึงจำเป็นที่จะอาศัยการสื่อสารแบบมัลติฮ๊อพ แต่ด้วยพลังงานของโหนดที่มีจำกัด หากจะให้ทุกโหนดพยายามสื่อสารมายังโหนดอ้างอิง โหนดตัวกลางระหว่างมัลติฮ๊อพก็จะต้องรับภาระนำส่งต่อข้อมูลจำนวนมาก และสุดท้ายโหนดจะตายพลังงานหมด เมื่อเป็นเช่นนี้จึงมีหลากหลายวิธีที่พยายามประสานเวลาแบบทีละคู่ แล้วให้โหนดที่ยังไม่ได้ประสานเวลาสามารถประสานเวลาต่อกับโหนดที่ผ่านการประสานเวลาเสร็จแล้วจนทั่วทุกโหนด [7], [9] และ [10]

3. ในการประสานเวลาแบบประสานกันทีละคู่เป็นทอดๆ ไปเรื่อยๆ นี้จะเกิดการสะสมค่าเวลาที่ผิดพลาดตามจำนวนฮ๊อพที่สูงขึ้นและเวลาที่ผ่านไป ท้ายที่สุดเมื่อทุกโหนดทำการประสานเวลากันเสร็จ ค่าผิดพลาดเฉลี่ย (ค่าเอเอสอี) จึงมีค่าสูงตามความยาวโครงสร้างลำดับชั้นและเวลาที่ใช้ในการประสานเวลา ดังรูปที่ 1.4 (ก) โหนด A ประสานเวลาตามโหนด R (ณ เวลา t_1) แล้วโหนด B ก็ประสานเวลาตามโหนด A (ณ เวลา t_2) รูปที่ 1.4 (ข) แสดงความเที่ยงตรงของนาฬิกาทั้งสามโหนด โดยโหนด R มีความเที่ยงตรงอย่างอุดมคติ ขณะที่นาฬิกาโหนด A เดินเร็วกว่าโหนด B และโหนด R ตามลำดับ รูปที่ 1.4 (ค) เมื่อโหนด A ประสานเวลาตามโหนด R ณ เวลา t_1 จะสังเกตเห็นว่าโหนดทั้งสองมีเวลาที่สอดคล้องกัน แต่เวลาที่ผ่านไปจนกระทั่งเวลา t_2 ที่โหนด B ประสานเวลาตามโหนด A นาฬิกาโหนด A ก็เดินเร็วไปมากแล้ว และเมื่อโหนด B ประสานเวลากับโหนด A เวลาโหนด B ยิ่งมีเวลาแตกต่างจากเวลาในอุดมคติมากเสียกว่าก่อนประสานเวลาเสียอีก



รูปที่ 1.4 การสะสมเวลาที่ผิดพลาดจากการประสานเวลาแบบมีโครงสร้างลำดับชั้น

เพื่อลดค่าเอเอสอี จึงพยายามลดโครงสร้างลำดับชั้นโดยสร้างโครงสร้างแบบกลุ่ม [9] หรือแต่ละโหนดพยายามเลือกโหนดแม่ที่อยู่ชั้นที่ต่ำที่สุดในโครงสร้างลำดับชั้น [10]

4. ตามธรรมชาตินาฬิกาโหนดใดๆ สามารถเกิดสภาวะนาฬิกาไม่เที่ยงตรงมากกว่าปกติ หรือที่เราเรียกว่า ฟอลคล็อกโหนด การไม่คำนึงถึงโหนดดังกล่าวในโครงสร้างที่อาศัยการประสานเวลาต่อเนื่องกันไปเรื่อยๆ จะทำให้ค่าเอเอสอีมีค่าสูงขึ้น เพื่อลดค่าเอเอสอีในสถานการณ์มีฟอลคล็อกโหนดปะปนอยู่ในระบบโครงข่ายเซนเซอร์ไร้สาย เราจึงเสนอวิธีการที่จะสามารถระบุและหลีกเลี่ยงฟอลคล็อกโหนด ซึ่งเราคาดว่าจะช่วยลดค่าเอเอสอีสำหรับวิธีการประสานเวลาแบบมีโครงสร้างลำดับชั้นได้

1.5 การเปรียบเทียบวิธีการที่นำเสนอกับวิธีการแบบพื้นฐาน

เริ่มต้นเราจะเปรียบเทียบค่าผิดพลาดเฉลี่ย (ค่าเอเอสอี) จากกรณีที่ทุกโหนดในระบบมีนาฬิกาที่เที่ยงตรงตามมาตรฐาน กับกรณีเมื่อมีโหนดที่มีนาฬิกาไม่เที่ยงตรง (ฟอลคล็อกโหนด) ปะปนอยู่ในระบบ เพื่อจะชี้ให้เห็นถึงผลกระทบของฟอลคล็อกโหนดที่มีต่อค่าเอเอสอี จากนั้นเราจะทำการประเมินสมรรถนะของโพรโทคอลประสานเวลาวิธีต่างๆ ที่จะสามารถทนทานต่อฟอลคล็อกโหนดที่แสดงออกมาทางค่าเอเอสอี

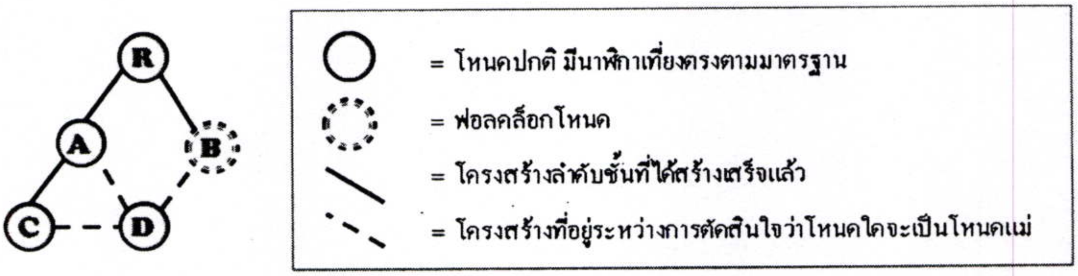
วิธีการประเมินสมรรถนะนั้นทำการเปรียบเทียบค่าเอเอสอีของทั้ง 3 โพรโทคอลการประสานเวลา ได้แก่

1. โพรโทคอลทีพีเอสเอ็นประสานเวลาตามโครงสร้างลำดับชั้น [7] โดยมีวิธีการเลือกโหนดแม่ (Parent Node) ด้วยการสุ่ม (Randomize) ดังกรณีตัวอย่างรูป 1.5 โหนด D สามารถเลือกโหนด A, B และ C เป็นโหนดแม่ที่ความน่าจะเป็นเท่ากัน

2. โพรโทคอลทีพีเอสเอ็นที่พยายามสร้างโครงสร้างลำดับชั้นให้สั้นที่สุด วิธีนี้จะเลือกโหนดแม่ที่มีหมายเลขระบุชั้นค่าต่ำที่สุด ดังกรณีตัวอย่างรูป 1.5 โหนด D สามารถเลือกโหนด A หรือ B เป็นโหนดแม่ที่ความน่าจะเป็นเท่ากัน โดยไม่สนใจโหนด C เพราะโหนด C มีหมายเลขระบุชั้นสูงกว่าโหนด A และ B

3. วิธีที่เรานำเสนอ “FTTS-WSN” คือ โพรโทคอลทีพีเอสเอ็นที่โครงสร้างลำดับชั้นพยายามหลีกเลี่ยงโหนดแม่ที่นาฬิกาไม่เที่ยงตรงมากกว่าปกติ (ฟอลคล็อกโหนด) ทั้งในทางตรงและทางอ้อมโดยพยายามหลีกเลี่ยงโหนดที่สืบทอดต่อมาจากฟอลคล็อกโหนด เมื่อให้ความสำคัญกับเงื่อนไขข้างต้นแล้วมีค่าสัมประสิทธิ์ที่เท่ากัน จึงให้ความสำคัญในการเลือกโหนดแม่ที่มีหมายเลขระบุชั้นที่ต่ำสุดเป็นความสำคัญระดับถัดมา ดังกรณีตัวอย่างรูป 1.5 โหนด D สามารถเลือกโหนด A หรือ C เป็นโหนดแม่ได้ แต่หลังจากพิจารณาหลีกเลี่ยงฟอลคล็อกโหนด (โหนด B) แล้ว

จึงดูว่าตัวเลือกที่เหลือมีโหนดใดที่หมายเลขระบุชั้นต่ำที่สุด กรณีนี้โหนด A (ที่อยู่ชั้นต่ำกว่าโหนด C) จะถูกเลือกเป็นโหนดแม่



รูปที่ 1.5 ตัวอย่างวิธีการเลือกโหนดแม่

ด้วยวิธีเปรียบเทียบประเมินประสิทธิภาพ หากวิธีการใดมีค่าเอเสอีต่ำที่สุด ย่อมแสดงว่าโพรโทคอลประสานเวลาดังกล่าวมีประสิทธิภาพดีที่สุด หรือเป็นวิธีการที่ทนทานต่อสถานการณ์ที่มีฟอลคล็อกโหนดปะปนในระบบได้ดีที่สุด

นอกจากนั้นยังประเมินผลด้านอื่นๆ ดังเช่น ค่าเวลาผิดพลาดสูงสุด, ค่าเวลาผิดพลาดเฉลี่ยในแต่ละชั้น, ปริมาณข้อมูลที่เกิดในการส่งและรับในระบบ ซึ่งจะสื่อถึงพลังงานที่ต้องใช้ ตลอดจนศึกษาภาระงานที่โหนดแม่จะได้รับในวิธีการต่างๆ เป็นต้น

1.6 ขอบเขตการวิจัย

ในวิทยานิพนธ์ฉบับนี้ได้นำเสนอเหตุและผล เพื่อให้เกิดการตระหนักและหลีกเลี่ยงโหนดที่มีนาฬิกาไม่เที่ยงตรงมากกว่าปกติ (ฟอลคล็อกโหนด) ซึ่งจะส่งผลต่อโครงสร้างลำดับชั้นที่จะมีเส้นทางการเชื่อมต่อ (route path) ระหว่างโหนดนาฬิกาเที่ยงตรงปกติและไม่ปกติ เมื่อแต่ละโหนดสามารถประเมินสถานะนาฬิกาที่ผิดปกติของตนเองได้แล้ว จะมีการประกาศข้อมูลที่สอดคล้องกับสถานะดังกล่าวให้โหนดข้างเคียงได้ทราบ ค่าข้อมูลนี้เราตั้งชื่อว่า “บิล” ซึ่งมาจาก *BIL* (*Bad_Inherited_Level* จะได้อธิบายโดยละเอียดในบทที่ 3 ต่อไป) ด้วยค่าของ “บิล” จะช่วยให้การตัดสินใจเลือกโหนดแม่เพื่อจะสร้างโครงสร้างลำดับชั้นต่อไป สามารถตระหนักและหลีกเลี่ยงฟอลคล็อกโหนดได้ง่าย

เราทำการประเมินสมรรถนะด้วยการเปรียบเทียบวิธีการที่เรานำเสนอกับวิธีการที่สำคัญในปัจจุบันผ่านแบบจำลองระบบ NESLsim ซึ่งทำงานร่วมกับ PARSEC (โปรแกรมสร้างสิ่งแวดล้อมการทำงานแบบขนาน) โดยรายละเอียดจะอยู่ในบทที่ 4 ต่อไป

1.7 ขั้นตอนของการศึกษา

วิทยานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ การเปรียบเทียบวิธีการที่นำเสนอกับวิธีการแบบพื้นฐาน ขอบเขตของการวิจัย ขั้นตอนการศึกษา ข้อตกลงเบื้องต้น และคำจำกัดความที่ใช้ในการศึกษา

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานของการประสานเวลาสำหรับระบบโครงข่ายเซนเซอร์ไร้สาย ปัจจัยที่มีผลต่อค่าผิดพลาดจากวิธีการประสานเวลาตลอดจนค่าเอเอสอีและผลกระทบจากการมีฟอลคล็อกโหนดปะปนอยู่

บทที่ 3 กล่าวถึงวิธีการระบุและหลีกเลี่ยงฟอลคล็อกโหนด ตลอดจนโหนดที่เชื่อมต่อกันมาเป็นทอดๆ จากฟอลคล็อกโหนด วิธีการผนวกงานของเราเข้ากับวิธีการที่ได้รับการยอมรับอยู่แล้ว (ทีพีเอสเอ็น [7]) ให้สามารถระบุและหลีกเลี่ยงฟอลคล็อกโหนด

บทที่ 4 กล่าวถึงการประเมินสมรรถนะของวิธีการของเราเปรียบเทียบกับงานที่มีอยู่ในปัจจุบัน ซึ่งจะเกี่ยวข้องกับค่าต่างๆ ดังต่อไปนี้ ค่าชี้วัดสมรรถนะของระบบ พารามิเตอร์ต่างๆ และผลการเปรียบเทียบตามการทดลองผ่านแบบจำลองระบบ

บทที่ 5 เป็นบทสรุปผลการวิจัยและข้อเสนอแนะ

1.8 ข้อตกลงเบื้องต้น

1. ทุกโหนดในระบบให้ความร่วมมือในการตรวจสอบความเที่ยงตรงของนาฬิกา และซื่อสัตย์ที่จะบอกกับโหนดอื่นๆ หากพบว่าโหนดตนเองเป็นฟอลคล็อกโหนด

2. ทุกโหนดที่ไม่ใช่โหนดรูตถูกสมมุติให้นาฬิกามีค่าคลาดเคลื่อนสัมพัทธ์แบบคงที่ ซึ่งในความเป็นจริงนาฬิกาสามารถเปลี่ยนแปลงอัตรานี้ทีละน้อยหรือมากได้ตลอดเวลา

1.9 คำจำกัดความที่ใช้ในการศึกษา

1. นาฬิกา (Clock) คืออุปกรณ์ที่ทำหน้าที่ดูแลและบอกเวลา ([11] แต่เนื่องจากอุปกรณ์นี้ไม่เที่ยงตรงตามอุดมคติ บางครั้งจึงถูกเรียกว่า Timer)

2. ควอทซ คริสตัล (Quartz Crystal) คือ แร่ที่เมื่อถูกอัดภายใต้สภาวะแรงดันสูงจะเกิดการสั่นสร้างจังหวะที่มีความถี่เกือบจะคงที่ แต่ด้วยขนาดและคุณภาพที่ต่างกันจะทำให้แร่คริสตัลของนาฬิกาแต่ละเรือนสร้างความถี่ที่แตกต่างกันไป

3. สภาวะเสถียรภาพ (The stability of a clock) คือ สภาวะที่นาฬิกาสามารถสร้างและดูแลความถี่ให้คงที่ตามที่ได้รับการออกแบบมา

4. เวลาที่เที่ยงตรง (Accuracy) คือ การที่นาฬิกาสามารถดูแลรักษาเวลาให้ตรงกับเวลาในอุดมคติ หรือตรงกับนาฬิกาอ้างอิงที่ระบบยอมรับร่วมกัน

5. ค่าเวลาที่แตกต่างจริง (Offset) คือ ค่าเวลาจริงที่แตกต่างกันระหว่างนาฬิกา

6. ค่าความถี่ที่แตกต่าง (Skew) คือ ค่าความถี่จากควอทซ คริสตัลที่แตกต่างกันระหว่างนาฬิกา 2 เรือน ทำให้เมื่อเวลาผ่านไประยะหนึ่ง จากเวลาที่ตรงกัน เวลาของนาฬิกาทั้งสองเรือนจะเดินแตกต่างกัน

7. ค่าเวลาที่คาดว่าต่างกัน (Drift) คือ ค่าเวลาที่เบี่ยงเบนหรือเวลาที่แตกต่างกันระหว่างนาฬิกาสองเรือน โดยสามารถทราบได้จากการคำนวณ

8. การปรับความถี่ประสานจังหวะ (Frequency Synchronization) คือ การปรับความถี่ของตัวให้จังหวะ เช่น ควอทซ คริสตัล เป็นต้น ให้มีความถี่เดียวกัน

9. การประสานเวลา (Time Synchronization) คือ การประสานเวลาระหว่างนาฬิกาให้ตรงกัน ณ เวลาขณะนั้น

10. การทำนาฬิกาประสานเวลา (Clock Synchronization) คือ การทั้งปรับเวลาและความถี่ของนาฬิกาให้ประสานกัน

11. ค่าผิดพลาดจากวิธีการประสานเวลา (Synchronization Error) คือ ค่าเวลาที่แตกต่างกันจริง ณ หลังจากการประสานเวลาแล้ว โดยค่าในอุดมคติคือ ศูนย์ หรือน้อยกว่า 1 ไมโครวินาที แต่ในทางปฏิบัติทำได้ยาก เพราะระหว่างโหนดจะไม่ทราบเวลาที่แตกต่างกันจริง (offset) แต่จะทราบเพียงเวลาที่คาดว่าต่างกัน (drift) จึงเป็นสาเหตุหนึ่งทำให้นาฬิกาไม่สามารถประสานเวลากันได้อย่างสมบูรณ์

12. นาฬิกาเที่ยงตรงตามมาตรฐาน [6] คือ ค่ามาตรฐานซึ่งถูกกำหนด หรือแสดงความแม่นยำของนาฬิกาที่ได้ผลิตจากโรงงานอุตสาหกรรม โดยเบื้องต้นแล้วนาฬิกาที่ใช้ควอทซ คริสตัลเป็นส่วนประกอบ จะมีความผิดพร่องหรือความแม่นยำที่เศษหนึ่งในหนึ่งหมื่นไปจนถึงหนึ่งในหนึ่งแสน เช่น นาฬิกาเรือนหนึ่งแจ้งว่ามีความเที่ยงตรงระดับ ± 40 p.p.m กล่าวได้ว่าทุกๆ 1 ชั่วโมงนาฬิกาชนิดนี้จะเดินช้าหรือเร็วขึ้นประมาณ 20 มิลลิวินาที หรือทุกๆ 1 วินาทีจะเดินช้าหรือเร็วขึ้นประมาณ 5.5 ไมโครวินาที

13. นาฬิกาไม่เที่ยงตรงผิดปกติ (Faulty Clock) คือ นาฬิกามีความผิดพร่องเกินระดับปกติหรือนาฬิกาซึ่งมีสภาวะไม่เสถียรภาพ โดยมีสาเหตุอันได้แก่ ความถี่เปลี่ยนแปลงมาก, ปัจจัยจากสิ่งแวดล้อมอย่างเช่นอุณหภูมิ การถูกกระทบกระเทือน ตลอดจนความผิดพลาดอันมาจากซอฟแวร์ที่ติดต่อกับนาฬิกาเป็นต้น โดยความผิดพร่องนี้อาจเป็นช่วงระยะเวลาสั้นๆ หรือคงอยู่เป็นเวลานานได้ ซึ่งนาฬิกาผิดพร่องยิ่งมีค่ามากจะยิ่งส่งผลให้ค่าผิดพลาดจากการประสานเวลามีค่าสูง

14. โหนดที่มีนาฬิกาไม่เที่ยงตรงมากกว่าปกติ (Fault Clock Node : FCN) หรือฟอลคล็อกโหนด คือโหนดที่มีนาฬิกาไม่เที่ยงตรงผิดปกติ

15. โหนดแม่ในโครงสร้างลำดับชั้นที่เป็นฟอลคล็อกโหนด (Bad Parent Node : BPN) หรือ บีพีเอ็น คือ ฟอลคล็อกโหนดที่ถูกเลือกให้เป็นโหนดแม่ให้กับโหนดลูกใดๆ ในโครงสร้างลำดับชั้น

16. โหนดลูกที่ประสานเวลาตามเส้นทางภายในโครงสร้างลำดับชั้นที่มีฟอลคล็อกโหนดอยู่ในสายทั้งทางตรงและทางอ้อม (Inherited Bad Parent Node : IBPN) หรือ ไอบีพีเอ็น คือ โหนดที่เลือกโหนดแม่ที่เป็นฟอลคล็อกโหนดทางตรง หรือโหนดแม่ที่เป็นไอบีพีเอ็นซึ่งเป็นลักษณะการสืบทอดทางอ้อม

17. โหนดรูต (Root Node) หรือซิงค์โหนด (Sink Node) คือ โหนดที่เชื่อมกับเครือข่ายคอมพิวเตอร์ภายนอก ทำหน้าที่รวบรวมข้อมูลจากโหนดในระบบโครงข่ายเซนเซอร์ไร้สาย ซึ่งบางกรณีอาจจะไม่ใช่โหนดเซนเซอร์แต่เป็นโมดูลทางการสื่อสาร (Access Point) ที่เชื่อมต่อระหว่างโหนดเซนเซอร์ และเครื่องคอมพิวเตอร์

18. โหนดแม่ที่เป็นตัวเลือก (Parent's Candidature) คือ โหนดข้างเคียงที่ได้อยู่ในโครงสร้างลำดับชั้นแล้ว และพร้อมจะเป็นโหนดแม่ให้กับโหนดอื่นๆ ที่ต้องการร่วมโครงสร้างลำดับชั้น

บทที่ 2
ทฤษฎีพื้นฐานที่ใช้ในการวิจัย

บทนี้จะกล่าวถึงทฤษฎีพื้นฐานที่เกี่ยวข้องกับงานวิจัย ซึ่งประกอบไปด้วยโพรโทคอลการประสานเวลาสำหรับระบบโครงข่ายเซนเซอร์ไร้สาย, คุณลักษณะและการวิเคราะห์ปัจจัยที่ส่งผลต่อค่าผิดพลาดจากวิธีการประสานเวลา, การประเมินสมรรถนะของโพรโทคอลการประสานเวลา และการวิเคราะห์ผลกระทบจากการมีโหนดซึ่งมีนาฬิกาไม่เที่ยงตรงมากกว่าปกติ (Fault Clock Node หรือ ฟอลคล็อกโหนด) ปะปนอยู่ในระบบ

2.1 โพรโทคอลประสานเวลาสำหรับระบบเครือข่ายเซนเซอร์ไร้สาย

วิธีการประสานเวลาสำหรับระบบเครือข่ายเซนเซอร์ไร้สายในปัจจุบัน มีพัฒนาการมาจากของ Network Time Protocol [3] ซึ่งได้รับการยอมรับและใช้กันอย่างแพร่หลายในโครงข่ายอินเทอร์เน็ต และยังคงพัฒนาอย่างต่อเนื่องจวบจนปัจจุบัน แต่ด้วยเหตุผลจากข้อจำกัดของโครงข่ายเซนเซอร์ไร้สายที่มีหลายข้อ ดังเช่น ข้อจำกัดทางการสื่อสารในด้านระยะทางและความเร็ว, พลังงานที่มีอยู่จำกัด, ความไม่เที่ยงตรงของนาฬิกา (Clock skew) อันมีสาเหตุจากคุณภาพของควอทซ คริสตัล (Quartz Crystal Oscillator) หรือผลกระทบจากสิ่งแวดล้อม, หรือจะอาศัยระบบจีพีเอสก็มีราคาสูงมากหากเปรียบเทียบกับราคาโหนดเซนเซอร์ เป็นต้น จึงจำเป็นต้องปรับแต่งวิธีการที่มีอยู่หรือพัฒนาและวิจัยวิธีการใหม่ ให้เหมาะสมต่อวิธีการประสานเวลาสำหรับระบบโครงข่ายเซนเซอร์ไร้สาย

เนื่องจากแต่ละโหนดไม่ทราบเวลาปัจจุบันที่แท้จริงของโหนดใดๆ ผ่านการสื่อสารข้อมูลเพราะระหว่างการบอกเวลาให้อีกโหนดทราบ เวลาก็เดินผ่านไปด้วย ข้อมูลที่ได้รับที่โหนดปลายทางจึงไม่ทันสมัยแล้ว นอกจากโหนดปลายทางจะทราบว่าเวลาที่ใช้ในการสื่อสารเป็นเท่าใดเพื่อจะใช้ปรับเวลาที่ได้รับมาให้ตรงกัน อีกทั้งเมื่อแต่ละโหนดไม่สามารถสื่อสารกับโหนดอ้างอิงได้โดยตรง ยิ่งเป็นไปไม่ได้เลยที่จะทราบเวลาของโหนดอ้างอิง ด้วยเหตุนี้โพรโทคอลประสานเวลาต่างๆ จึงพัฒนาเพื่อคาดเดาเวลาที่แตกต่างกันระหว่างโหนดภายใต้ข้อผิดพลาดต่ำที่สุด

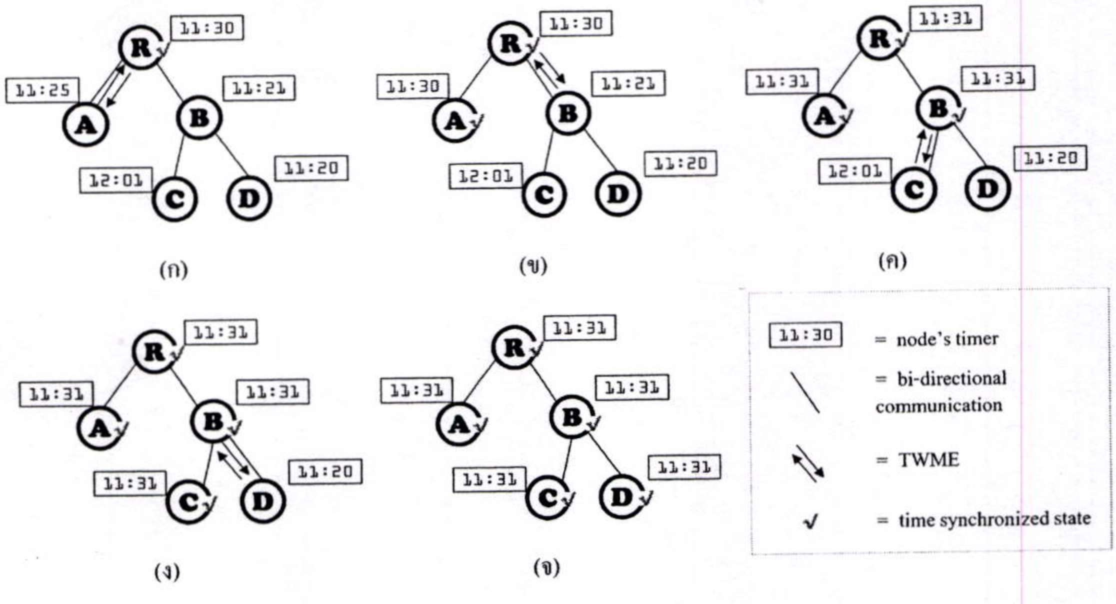
งานที่เสนอวิธีการประสานเวลาสำหรับโครงข่ายเซนเซอร์ไร้สาย และมีความเกี่ยวข้องกับวิทยานิพนธ์นี้ประกอบไปด้วย

2.1.1 Timing-sync Protocol for Sensor Networks (TPSN – ทีพีเอสเอ็น) [7]

โพรโทคอลทีพีเอสเอ็นมีความคล้ายคลึงกับ NTP [3] เพราะมีลักษณะเป็นโครงสร้างลำดับชั้นเช่นเดียวกัน โดยโหนดจะเชื่อมต่อกับโหนดข้างเคียงภายใต้รัศมีคลื่นวิทยุของทั้งสอง

โหนด และสร้างโครงสร้างลำดับชั้นจากโหนดที่ให้เวลาอ้างอิง จากนั้นโหนดจะประสานเวลากับโหนดที่อยู่ชั้นเหนือกว่าจากชั้นบนสุดสู่ล่างสุดจนครบทั้งโครงสร้างลำดับชั้น เพื่อ ให้ทุกโหนดในโครงสร้างมีเวลาที่ประสานกันทั้งหมด ดังแสดงในรูปที่ 2.1 ภายหลังเมื่อโหนด R สั่งให้ทุกโหนดสร้างโครงสร้างลำดับชั้นเสร็จแล้ว ต่อมาโหนด R ซึ่งอยู่ตำแหน่งบนสุดของโครงสร้างลำดับชั้นจะส่งสัญญาณให้โหนดอื่นๆ ที่อยู่ลำดับชั้นต่ำลงไปทำการประสานเวลากับโหนดตน ในรูป 2.1 (ก) โหนด A แลกเปลี่ยนเวลากับโหนด R เพื่อคำนวณหาเวลาที่โหนด A คาดว่าต่างกับโหนด R (เราเรียกค่าเวลาที่คาดว่าต่างกันนี้ว่า Clock Drift และแสดงวิธีการคำนวณดังสมการ (2.1)) เมื่อทราบเวลาที่แตกต่างกันแล้ว โหนด A จึงประสานเวลาให้ตรงกับโหนด R เมื่อถึงตรงนี้ โหนด A ก็พร้อมที่จะให้โหนดที่อยู่ในลำดับชั้นต่ำลงไป (หากมี) ประสานเวลากับตนต่อไป

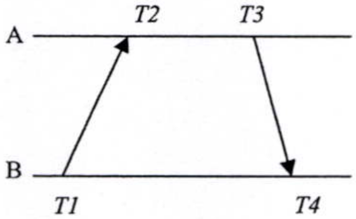
จากนั้นโหนดอื่นๆ (B, C และ D) จะประสานเวลากับโหนดที่อยู่เหนือขึ้นไปตามโครงสร้างลำดับชั้นด้วยวิธีการเดียวกันกับโหนด A ดังรูป 2.1 (ข), 2.1 (ค), 2.1 (ง) และ 2.1 (จ) ตามลำดับ แน่นอนว่าการจะประสานเวลากับทุกโหนดแบบประสานเวลาต่อเป็นทอดๆ นี้ย่อมต้องใช้เวลา (เริ่มประสานเวลาตอนโหนด R เวลา 11:30 แล้วเสร็จสิ้นตอนโหนด R เวลา 11:31) และหลายครั้งที่เมื่อทุกโหนด ประสานเวลาเสร็จ เวลาแต่ละโหนดอาจจะไม่ตรงกันก็ได้



รูปที่ 2.1 ตัวอย่างโครงสร้างลำดับชั้นและโพรโทคอลการประสานเวลาทีพีเอสเอ็น

รูปที่ 2.2 โพรโทคอลทีพีเอสเอ็นสามารถคำนวณค่าเวลาที่คาดว่าแตกต่างกันระหว่างโหนด (Clock Drift - Δ) จากวิธีการแลกเปลี่ยนข้อมูลเวลาระหว่างโหนด ดังสมการ (2.1) ถูกเรียกกว่า Two Way Message Exchange algorithm (TWME – ทีดับบิวเอ็มอี) หรือเรียกวิธีนี้เป็นการประสานเวลาระหว่างโหนดส่งกับโหนดรับ (Sender-Receiver Synchronization [3]) และด้วย อัลกอลิทึมการแลกเปลี่ยนเวลาระหว่างโหนด สามารถคำนวณเวลาการส่งข้อมูลจากต้นทางไปยังปลายทาง

(Communication delay หรือ Network transmission delay - *d*) ได้ดังสมการ (2.2) โดยค่านี้สื่อถึงความน่าเชื่อถือของผลลัพธ์ค่าเวลาที่คาดว่าต่างกันระหว่างสองโหนด เพราะหากเวลาในระหว่างส่งข้อมูลไม่คงที่หรือมีค่ามาก มีความเป็นไปได้สูงที่ค่าเวลาที่คาดว่าต่างกันระหว่างสองโหนดมีค่าผิดพลาดมากขึ้นด้วย



- เวลา *T2* และ *T3* คือ เวลาที่ได้รับและส่งของโหนด A ตามลำดับ
- เวลา *T1* และ *T4* คือ เวลาที่ส่งและได้รับของโหนด B ตามลำดับ

รูปที่ 2.2 วิธีทีดับบิวเอ็มอี การแลกเปลี่ยนข้อมูลเวลาระหว่างสองโหนด

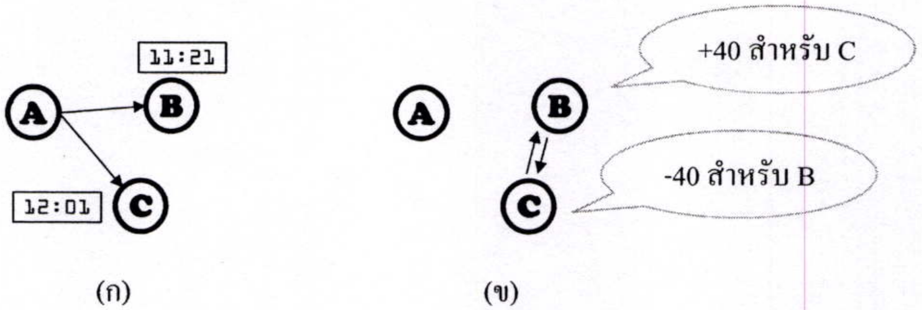
$$\Delta = \frac{(T2-T1)-(T4-T3)}{2} \quad (2.1)$$

$$d = \frac{(T2-T1)+(T4-T3)}{2} \quad (2.2)$$

สมรรถนะความถูกต้องจากโพรโทคอลทีพีเอสเอ็นที่ได้ทำการทดลองจริง ซึ่งพยายามเน้นที่ความถูกต้องระหว่างเซนเซอร์โหนด พบว่ามีค่าผิดพลาดจากวิธีการประสานเวลาระหว่างโหนดใดๆ (Synchronization Error : *SE* : เอสอี) ต่ำกว่า 2 เท่าเมื่อเทียบกับ Reference-Broadcast Synchronization (RBS – อาร์บีเอส) [8] โดยโพรโทคอลทีพีเอสเอ็น และโพรโทคอล อาร์บีเอสมีค่าเอสอีประมาณ 17-20 ไมโครวินาทีและ 34-40 ไมโครวินาที ตามลำดับ แม้ที่ค่าเอสดีดังกล่าวทีพีเอสเอ็นมีเงื่อนไขว่า ต้องกำจัดเวลาที่เกิดจากขั้นตอนการเตรียมข้อมูลและขั้นตอนเวลารอคอยก่อนจะส่ง (รายละเอียดจะกล่าวในหัวข้อ 2.2.1 ต่อไป) โดยเวลาที่บันทึกตอนส่งและรับจะบันทึกขณะข้อความพร้อมส่งออกไปจากโหนดและได้รับจริงๆ (ที่ชั้นดาต้าลิงค์ตามโพรโทคอลสื่อสาร) เพราะจะสามารถลดตัวแปรที่ไม่แน่นอนอันเป็นสาเหตุหนึ่งของการเกิดค่าเอสอี เทคนิคดังกล่าวมีหลักการคล้ายกับโพรโทคอลอาร์บีเอส [8] ที่เสนอการกำจัดตัวแปรที่ไม่แน่นอนในขณะส่งข้อมูล โพรโทคอลทั้งสองให้ความสำคัญอย่างมากต่อความถูกต้องของเวลาระหว่างโหนดซึ่งจะใช้เพื่อคำนวณเวลาที่คาดว่าต่างกัน (drift)

วิธีการประสานเวลาตามโพรโทคอลอาร์บีเอส [8] นี้เรียกว่า การประสานเวลาระหว่างโหนดรับกับโหนดรับ (Receiver-Receiver Synchronization) ดังแสดงในรูป 2.3 (ก) ด้วยวิธีการสุ่มให้โหนดหนึ่ง (เช่น โหนด A) ส่งข้อความหนึ่งออกไปให้โหนดข้างเคียง ซึ่งโหนด B บันทึกเวลา 11:21 ตามนาฬิกาตนเองและโหนด C บันทึกเวลา 12:01 ตามนาฬิกาตนเองเช่นกัน ต่อมาตามรูป 2.3 (ข) โหนด B และ C แลกเปลี่ยนเวลาที่ได้รับระหว่างกันเพื่อเปรียบเทียบคำนวณหาเวลาที่แตกต่างระหว่างโหนด ณ กรณีนี้ โหนด B จะทราบว่านาฬิกาตนนั้นเดินช้ากว่านาฬิกาโหนด C อยู่ 40 นาที

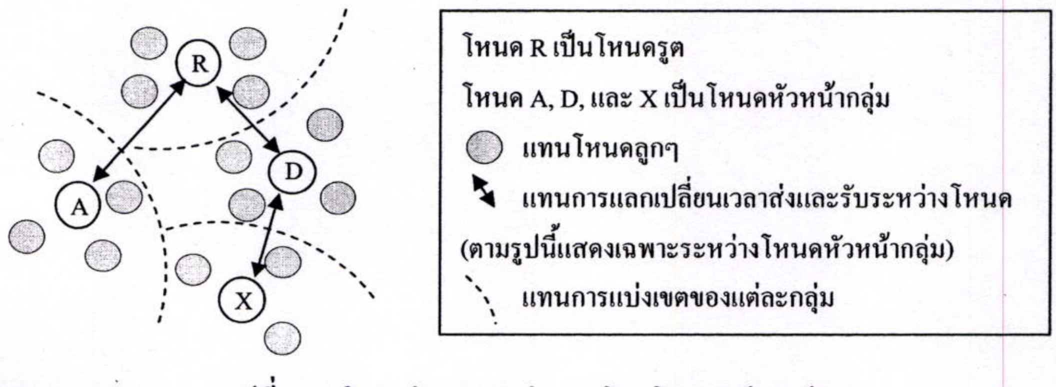
และในทางกลับกันโหนด C ก็ทราบว่านาฬิกาตนเดินเร็วกว่า 40 นาทีเช่นกัน โพรโทคอลอาร์บีเอสจะไม่ทำการประสานเวลาให้ตรงกันเหมือนโพรโทคอลทีพีเอสเอ็น แต่ทุกโหนดจะคำนวณหาความสัมพันธ์ของเวลาที่แตกต่างนี้กับโหนดอื่นตลอดทั้งโครงข่าย



รูปที่ 2.3 โพรโทคอลการประสานเวลาอาร์บีเอส

2.1.2 Cluster-based Hierarchical Time Synchronization for Multi-hop Wireless Sensor Networks (CHTS – ซีเอชทีเอส) [9]

โพรโทคอลซีเอชทีเอสจะสร้างกลุ่มของเซนเซอร์ในลักษณะสุ่ม ด้วยโครงสร้างแบบกลุ่มจะช่วยแบ่งเบางานเพราะมีการกระจายงานได้ดีและลดจำนวนฮ๊อพ (Hop count) ในโครงข่ายได้ แสดงดังในรูป 2.4 หลังจากได้โครงสร้างแบบกลุ่มและระหว่างกลุ่ม การแลกเปลี่ยนข้อมูลเวลาจะอาศัยรูปแบบการประสานเวลาแบบผู้ส่งและผู้รับ (ทีดับบิวเอ็มอี) เริ่มระหว่างโหนดหัวหน้าแต่ละกลุ่มกับโหนดอ้างอิง (โหนดรูต) ก่อน ซึ่งถ้าโหนดหัวหน้ากลุ่มใดไม่สามารถสื่อสารถึงโหนดรูตได้โดยตรง ก็จะประสานเวลากับเวลาหัวหน้ากลุ่มอื่นที่ได้ประสานเวลากับโหนดรูตเรียบร้อยแล้วแทน หลังจากนั้นจะทำการประสานเวลาด้วยวิธีทีดับบิวเอ็มอีกันภายในกลุ่มระหว่างโหนดหัวหน้ากลุ่มกับโหนดลูกๆ ภายในกลุ่ม วิธีการนี้จะทำให้ทุกโหนดในโครงข่ายได้รับการประสานเวลากับโหนดรูตไม่ทางตรงก็ทางอ้อม



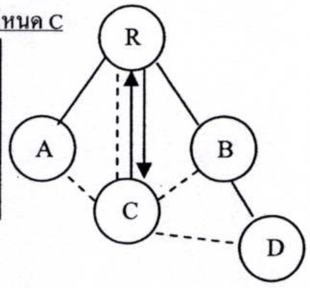
รูปที่ 2.4 โครงข่ายแบบกลุ่มของโพรโทคอลซีเอชทีเอส

2.1.3 Reliable Time Synchronization Protocol for Wireless Sensor Network (RTSP – อาร์ทีเอสพี) [10]

โพรโทคอลอาร์ทีเอสพีมีพื้นฐานมาจากโพรโทคอลทีพีเอสเอ็น โดยมีโครงสร้างลำดับชั้น และวิธีการประสานเวลาแบบแลกเปลี่ยนข้อมูลเวลา-ทีดับบิวเอ็มอี ประเด็นสำคัญที่ได้เสนอเพิ่มเติมสำหรับงานนี้คือ การพยายามสร้างโครงสร้างลำดับชั้นให้สั้นที่สุด (Lowest Depth) เนื่องจากการมีโครงสร้างที่สั้น สามารถลดค่าเฉลี่ยผิดพลาดจากการประสานเวลาของทั้งระบบ (เอเอสอี) ได้ดีกว่าโครงสร้างลำดับชั้นที่ยาวกว่า และระหว่างการเรียนรู้โหนดแม่ (parent node) ในขั้นตอนการสร้างโครงสร้างลำดับชั้น โหนดลูก (child node) จะเก็บข้อมูลของทุกโหนดเพื่อนบ้านที่สามารถเป็นโหนดแม่ (candidate parent node) เพื่อสามารถเปลี่ยนโหนดแม่ได้ทันทีหากเกิดกรณีที่ไม่สามารถติดต่อโหนดแม่ปัจจุบันได้ ทำให้ระบบทนทานต่อการเปลี่ยนแปลงโครงสร้าง ตัวอย่างการเลือกโหนดแม่ของโหนด C แสดงดังในรูป 2.5 เมื่อโหนด C ได้เรียนรู้โครงสร้างลำดับชั้นจากโหนดข้างเคียงทั้ง 4 โหนด อันได้แก่ R, A, B และ D พบว่าโหนด R เหมาะสมจะเป็นโหนดแม่ที่ดีที่สุดในกรณีนี้ เนื่องจากโหนด C จะมีอยู่ลำดับชั้นที่ 1 เมื่อเลือกโหนด R เป็นโหนดแม่ ขณะที่ถ้าเลือกโหนด A, B หรือ D แล้ว โหนด C จะมีลำดับชั้นในโครงสร้างลำดับชั้นเป็น 2, 2 และ 3 ตามลำดับ

ตารางการเรียนรู้โหนดแม่ ของโหนด C

Parent_ID	*Parent's level*
R	0
A	1
B	1
D	2



โหนด "R" เป็นโหนดรูต
โหนด "A, B, C และ D" เป็นโหนดลูก
แทนระยะที่สามารถสื่อสารระหว่างโหนด
การแลกเปลี่ยนข้อมูลเวลา หรือทีดับบิวเอ็มอี ระหว่างโหนด C และโหนด R (ซึ่งเป็นโหนดแม่ของ C)

รูปที่ 2.5 การเลือกโหนดแม่ของโหนด C ตามโพรโทคอลอาร์ทีเอสพี

2.2 คุณลักษณะและการวิเคราะห์ปัจจัยที่ส่งผลต่อค่าผิดพลาดจากวิธีการประสานเวลา

2.2.1 ผลกระทบจากค่าตัวแปรเวลาระหว่างการสื่อสาร

การประสานเวลาด้วยวิธีการแลกเปลี่ยนข้อมูลเวลา-ทีดับบิวเอ็มอี (ตามรูปที่ 2.2) อาศัยการรับส่งข้อมูลผ่านโพรโทคอลการสื่อสาร และระหว่างการส่งหรือแลกเปลี่ยนข้อมูลจะมีตัวแปรเวลาที่ไม่แน่นอนปรากฏอยู่ในระหว่างชั้นต่างๆ ของโพรโทคอลการสื่อสาร ซึ่งตัวแปรเวลาเหล่านี้เป็นปัจจัยสำคัญส่วนหนึ่งของค่าผิดพลาดจากวิธีการประสานเวลา (ค่าเอสอี) [7] เราสามารถแจกแจงตัวแปรเวลาที่ไม่แน่นอนได้ดังนี้

- ค่าเวลาเตรียมข้อมูล (Send time) เป็นเวลาที่ใช้ในการสร้างแพ็กเก็ตที่ชั้นแอพพลิเคชัน (Application Layer) ตามโพรโทคอลการสื่อสาร ตามรูปที่ 2.2 เมื่อบันทึกเวลา *T1* (หรือ *T3*) แล้ว ข้อมูลนี้จะต้องถูกจัดเตรียมก่อนจะถูกส่งไปยังชั้นต่ำลงตามโพรโทคอลการสื่อสาร ค่าของตัวแปรนี้มีค่าที่ไม่แน่นอน เนื่องจากเกี่ยวข้องกับซอฟแวร์และระบบปฏิบัติการ

- ค่าเวลาที่ต้องรอก่อนใช้สื่อ (Access time) หมายถึงเวลาที่เฟรมข้อมูลพร้อมส่งแล้ว แต่ต้องรอจนกว่าสื่อกลางจะว่าง เพราะโหนดได้ยินว่าโหนดอื่นกำลังใช้สื่อเพื่อส่งข้อมูลอยู่ ซึ่งเวลาการจัดการในชั้นดาต้าลิงค์ (Data Link Layer) หรือบางทีเรียกว่า MAC Layer ของโพรโทคอลการสื่อสาร เป็นค่าตัวแปรที่มีความสำคัญมากที่สุดเพราะอาจจะมีค่าต่ำหรือสูงมากขึ้นอยู่กับความหนาแน่นของโหนด และปริมาณความต้องการที่จะส่งข้อมูล อีกทั้งค่าตัวแปรเวลานี้ได้รับความสนใจอย่างมากในโพรโทคอลอาร์บีเอส และโพรโทคอลทีพีเอสเอ็นซึ่งพยายามเสนอวิธีกำจัดค่าตัวแปรนี้

- ค่าเวลาที่ใช้ส่งข้อมูล (Transmission time) ตัวแปรนี้สามารถคำนวณหรือคาดการณ์ได้โดยอาศัยความสัมพันธ์ระหว่างขนาดข้อมูลในหน่วยบิตและความเร็วของอุปกรณ์ที่สามารถส่งข้อมูลเข้าไปในสื่อกลางในหน่วยบิตต่อวินาที ดังสมการที่ (2.3) หากสมมติว่าต้องการส่งข้อมูลขนาด 128 บิต [7] และความเร็วที่อุปกรณ์เครือข่ายสามารถส่งข้อมูลมีอัตราความเร็ว 19.2 กิโลบิตต่อวินาที ดังนั้นการส่งข้อมูลชิ้นนี้ใช้เวลาประมาณ 6.6 มิลลิวินาที

$$Transmission\ (Reception) time = \frac{\text{ขนาดข้อมูล (บิต)}}{\text{ความเร็วในการส่งข้อมูลผ่านสื่อ (บิตต่อวินาที)}} \tag{2.3}$$

- ค่าเวลาที่ข้อมูลเดินทางจากอุปกรณ์สื่อสารฝั่งส่งไปจนถึงอุปกรณ์ฝั่งรับ (Propagation time) หรือเวลาที่ข้อมูลเดินทางบนอากาศกรณีการสื่อสารด้วยคลื่นวิทยุ ซึ่งค่าตัวแปรนี้สามารถคำนวณหรือคาดเดาได้เช่นกันจากสมการ (2.4) ความเร็วของคลื่นวิทยุในอากาศมีความเร็วใกล้เคียงความเร็วแสง เมื่อระยะทางระหว่างโหนดเซนเซอร์มักจำกัดอยู่เพียงประมาณ 10 ถึง 150 เมตร [1] ทำให้เวลาของตัวแปรนี้มีค่าที่น้อยกว่า 1 ไมโครวินาที ดังนั้นตัวแปรนี้สามารถละทิ้งได้

$$Propagation\ time = \frac{\text{ระยะทางระหว่างโหนดส่งและรับ (เมตร)}}{\text{ความเร็วคลื่นวิทยุ (เมตรต่อวินาที)}} \tag{2.4}$$

- ค่าเวลาที่อุปกรณ์สื่อสารปลายทางใช้รับข้อมูล (Reception time) โดยมีหน่วยความเร็วเป็นบิตต่อวินาที โดยทั่วไปจะมีค่าเท่ากับเวลาที่อุปกรณ์สื่อสารต้นทางใช้ส่งข้อมูล จึงสามารถคำนวณค่าเวลาได้ดังสมการข้อ (2.3)

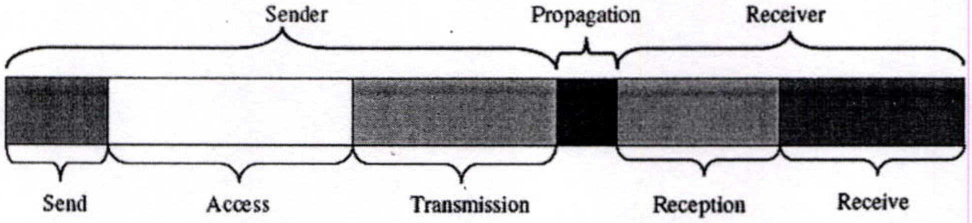
- ค่าเวลาเพื่อใช้ถอดข้อมูล (Receive time) คือตั้งแต่เวลาที่ข้อความได้รับครบถ้วน ณ ปลายทางแล้ว ข้อมูลจะถูกถอดออกจากข้อความที่มีส่วนประกอบอ้างอิง (Header – เฮดเดอร์) ตามข้อกำหนดของแต่ละชั้นโพรโทคอลการสื่อสาร สุดท้ายจะส่งขึ้นไปจนถึงโปรแกรมที่รอรับข้อมูลนี้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ที่อยู่ชั้นแอพพลิเคชัน (ฝั่งปลายทาง) ค่าเวลาส่วนนี้อาจมีค่าไม่แน่นอนเนื่องจากการขัดจังหวะของระบบปฏิบัติการที่จัดการงานหลายงานพร้อมกัน

รูปที่ 2.6 ตัวแปรต่างๆ ที่กล่าวมาข้างต้น ตั้งแต่ส่งข้อมูลจากต้นทางไปยังปลายทางสามารถสรุปได้เป็น 3 ส่วน [7] อันได้แก่ (สัดส่วนในภาพอาจจะไม่สมจริงตามแต่ละค่าตัวแปร)

- S^{UC} – ค่าตัวแปรเวลาที่ไม่แน่นอนของฝั่งโหนดส่ง
- P^{UC} – ค่าตัวแปรเวลาขณะข้อมูลเดินทางบนตัวกลาง หรือบนสื่อ
- R^{UC} – ค่าตัวแปรเวลาที่ไม่แน่นอนของฝั่งโหนดปลายทาง



รูปที่ 2.6 โครงสร้างตัวแปรเวลาระหว่างการสื่อสาร

2.2.2 ผลกระทบจากนาฬิกาที่เที่ยงตรงทั้งปกติและไม่ปกติ

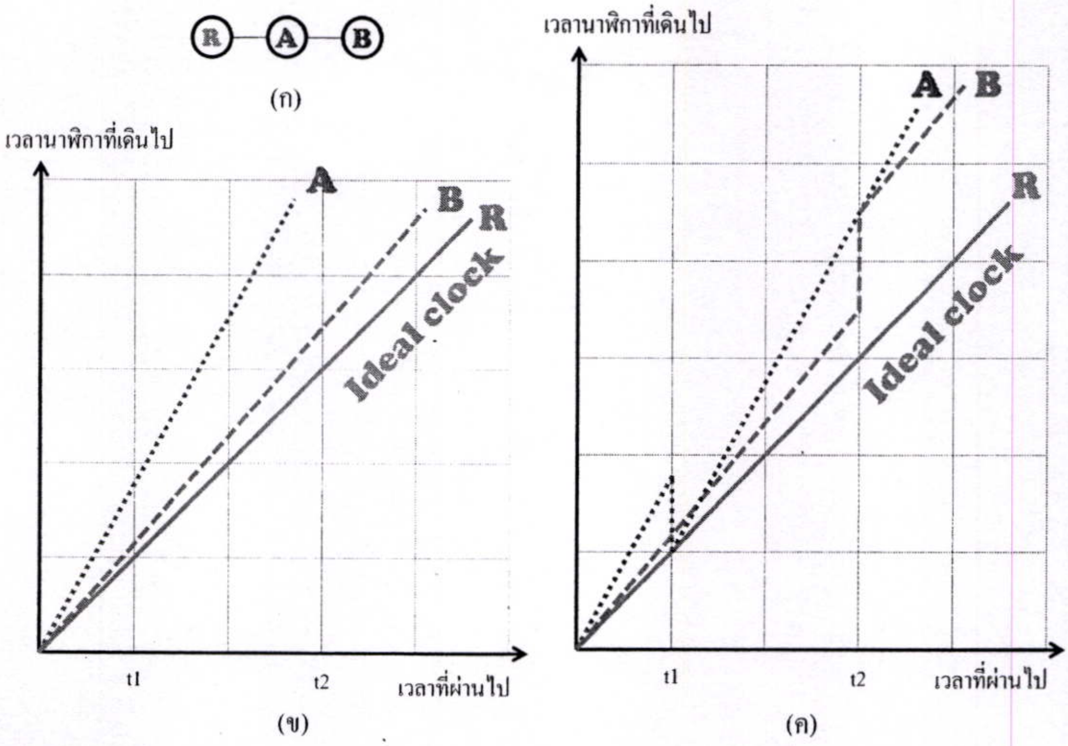
ความเที่ยงตรงของนาฬิกานั้นขึ้นอยู่กับคุณภาพของควอทซ คริสตัล [6] ซึ่งตามมาตรฐานทั่วไป นาฬิกาที่มีความเที่ยงตรงปกติจะมีค่าความคลาดเคลื่อนประมาณ 20 มิลลิวินาทีต่อชั่วโมง (หรือ 5.5 ไมโครวินาทีต่อวินาที) สำหรับนาฬิกาโหนดใดๆ ที่เกิดสภาวะนาฬิกาไม่เที่ยงตรงมากกว่าปกติทั้งชั่วคราวและถาวร อันมีผลกระทบจากสิ่งแวดล้อมและความผิดปกติภายในนาฬิกาและโหนดเอง ซึ่งอาจทำให้นาฬิกามีค่าความคลาดเคลื่อนมากกว่าปกติหลายเท่าตัว ผลของความคลาดเคลื่อนไม่ว่าจะปกติหรือไม่ปกตินี้จะส่งผลต่อโหนดอื่นๆ ที่ทำการประสานเวลาด้วยไม่มากก็น้อย ดังตัวอย่างในรูปที่ 2.7 (ก-ค) ต่อไปนี้

รูปที่ 2.7 (ก) โหนด A และ B ทำการประสานเวลากับโหนด R (โหนดเวลาอ้างอิง) โดยโหนด A จะประสานเวลาตามโหนด R (ณ เวลา *t1*) แล้วโหนด B ก็ประสานเวลาตามโหนด A (ณ เวลา *t2*) อีกต่อหนึ่ง

รูปที่ 2.7 (ข) สมมติความเที่ยงตรงของนาฬิกาทั้งสามโหนด โดยโหนด R มีความเที่ยงตรงในอุดมคติ ขณะที่นาฬิกาโหนด A เที่ยงตรงไม่ปกติ และโหนด B มีนาฬิกาที่เที่ยงตรงปกติ

รูปที่ 2.7 (ค) เมื่อโหนด A ประสานเวลาตามโหนด R เสร็จ ณ เวลา *t1* จะเห็นว่าในขณะนั้นนาฬิกาโหนด A เหมือนจะตรงกับนาฬิกาโหนด R แต่ด้วยความเที่ยงตรงที่ไม่ปกติทำให้ระหว่างเวลาที่ผ่านไปจนกระทั่งเวลา *t2* นาฬิกาโหนด A จึงเบี่ยงเบนไปมาก เมื่อโหนด B ประสานเวลาได้ตรงตามโหนด A ณ เวลา *t2* แต่นาฬิกาของโหนด B กลับยิ่งแตกต่างจากนาฬิกาโหนด R ก่อนทำประสานเวลา

95049



รูปที่ 2.7 ผลกระทบจากความเที่ยงตรงของนาฬิกาทั้งปกติและไม่ปกติ

ในการประสานเวลาแบบประสานกันทีละคู่ไปเรื่อยๆ นี้จะเกิดการสะสมค่าเวลาที่ผิดพลาดตามจำนวนฮ๊อพที่สูงขึ้นและเวลาที่ผ่านไป เราขอเรียกค่าผิดพลาดลักษณะนี้ว่า "ค่าผิดพลาดเมื่อเวลาผ่านไป" ท้ายที่สุดเมื่อทุกโหนดทำการประสานเวลากันเสร็จ นาฬิกาของแต่ละโหนดจึงไม่สอดคล้องกันตามวัตถุประสงค์

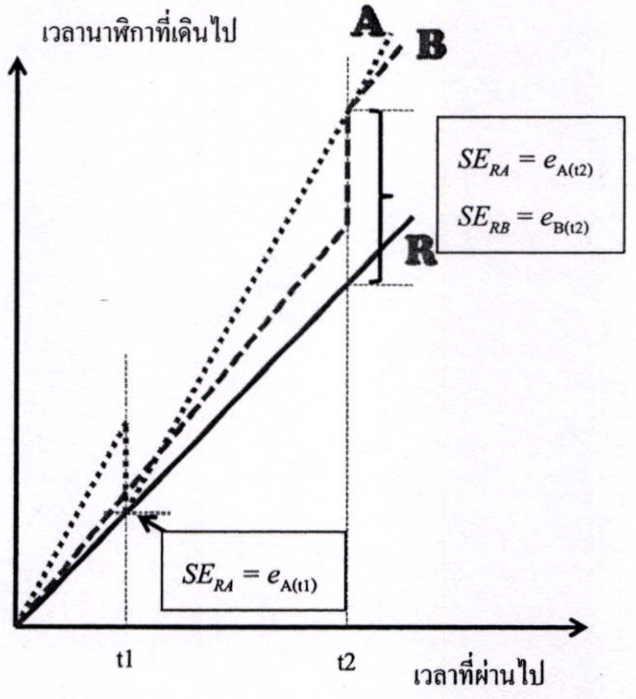
2.3 การประเมินสมรรถนะของโพรโทคอลการประสานเวลา

2.3.1 ค่าผิดพลาดจากการประสานเวลา (Synchronization Error : *SE* : เอสอี)

งานวิจัยจำนวนมากพยายามพัฒนาวิธีการที่ทำให้ค่าเอสอีมีค่าใกล้เคียงกับค่าในอุดมคติ ที่ต้องการให้ปราศจากค่าเอสอีหรืออย่างน้อยมีค่าเอสอีน้อยกว่า 1 ไมโครวินาที แม้ว่าโหนดเซนเซอร์ เช่น MICAx, IRIS เป็นต้น ที่ระบบปฏิบัติการเป็น TinyOS จะมีหน่วยเวลาที่เล็กที่สุด คือ 1,000 ไมโครวินาที หรือ 1 มิลลิวินาที [1] ซึ่งค่อนข้างหยาบเกินกว่าจะเปรียบเทียบกับค่าเอสอีในอุดมคติในทางปฏิบัติก็ตาม แต่งานวิจัยจำนวนมากยังคงมีเป้าหมายให้การประสานเวลาระหว่างโหนดสามารถทำงานได้ดีที่สุด พร้อมสำหรับการใช้งานทั่วไปจนไปถึงงานที่ต้องการความละเอียดสูง ดังนั้นในการประเมินสมรรถนะของโพรโทคอลการประสานเวลา เรายังคงพิจารณาตัวชี้วัดค่าผิดพลาดจากวิธีการประสานเวลาด้วยเช่นกัน และเราขอใช้ชื่อย่อว่า "เอสอี" แต่เอสอีที่เรากำลังพูดถึงนี้แตกต่างจากนิยามเดิมเล็กน้อย โดยแต่เดิมนั้นพิจารณาค่าผิดพลาดที่เกิดระหว่างโหนดที่ทำการ

ประสานเวลากัน (offset of synchronization pair nodes) ที่มักมุ่งความสนใจไปที่นาฬิกาโหนด 2 โหนด ขณะในงานวิจัยชิ้นนี้จะหมายถึงค่าผิดพลาดเชิงสะสมระหว่างโหนดที่ประสานเวลาเสร็จนั้นกับโหนดอ้างอิง (offset of synchronized node and reference node) ซึ่งมองถึงโหนดจำนวนมากในระบบ ว่ามีค่าความแตกต่างกันเพียงใด ณ เวลาหนึ่ง ดังรูป 2.8 (ซึ่งสืบเนื่องมาจากรูปที่ 2.7) ค่า SE_{RA} ณ เวลา *t1* สมมุติมีค่า $e_{A(t1)}$ หน่วยเวลา ซึ่งมีค่าเล็กมากๆ แต่ในเวลาต่อมา (*t2*) ทุกโหนดในระบบซึ่งในกรณีนี้ คือโหนด A และ B ทำการประสานเวลาเสร็จ ค่า SE_{RA} และ SE_{RB} สมมุติมีค่า $e_{A(t2)}$ และ $e_{B(t2)}$ หน่วยเวลา ตามลำดับ ซึ่งมีค่าค้อนข้างมากเมื่อเทียบกับ $e_{A(t1)}$ สาเหตุนั้นดังคำอธิบายก่อนหน้านี้ในหัวข้อ 2.2.2

เหตุผลที่เราพิจารณาค่าเอสอีของทุกโหนด ณ โหนดสุดท้ายเสร็จสิ้นการทำการประสานเวลา นั้นเพราะตามวัตถุประสงค์ที่ต้องการให้การประสานเวลาทุกโหนดได้สอดคล้องกับโหนดอ้างอิง แต่เมื่อโหนดใดๆ ประสานเวลาเสร็จแล้ว ค่าเวลาที่แตกต่าง (offset) เมื่อเทียบกับโหนดอ้างอิงไม่ได้อยู่นิ่งที่ค่านั้นตลอดไป อีกทั้งโหนดอื่นๆ อาจยังไม่ได้รับการประสานเวลา ดังนั้นเราควรมาพิจารณา ณ เวลาที่ทุกโหนดประสานเวลากันเสร็จ ซึ่งหลังจากนี้ทุกโหนดก็พร้อมที่จะทำงานตามวัตถุประสงค์ของเซนเซอร์หรืองานอื่นๆ ต่อไป



รูปที่ 2.8 ค่าผิดพลาดจากการประสานเวลาเชิงสะสม หรือค่าเอสอี

2.3.2 ค่าเฉลี่ยผิดพลาดจากการประสานเวลาของทั้งระบบ (Average Synchronization Error – เอเอสอี)

ระบบโครงข่ายเซนเซอร์ไร้สายมักประกอบด้วยเซนเซอร์โหนดจำนวนมาก และทุกโหนดต้องการมีเวลานาฬิกาที่ประสานสอดคล้องกัน เมื่อโครงข่ายการเชื่อมต่อโดยทั่วไปเป็นแบบทั่วถึงเพียงบางส่วน (partial mesh) หรือเมื่อทุกโหนดไม่สามารถสื่อสารกับโหนดรูตได้โดยตรง วิธีการประสานเวลาทั้งระบบจึงต้องผ่านรูปแบบโครงข่ายมัลติฮ๊อพ (Multi-hop Topology) การประสานเวลาจะทำกันทีละคู่ไปเรื่อยๆ หรือทีละกลุ่ม อย่างไรก็ตามไม่ว่าจะประสานเวลาด้วยวิธีการใดจะสะสมค่าผิดพลาด 2 ส่วน คือ จากวิธีการประสานเวลาทีดับบิวเอ็มอี ตามข้อ 2.2.1 และจากค่าผิดพลาดเมื่อเวลาผ่านไประหว่างการประสานเวลา ตามข้อ 2.2.2 เพื่อจะแสดงสมรรถนะของโพรโทคอลประสานเวลาในภาพรวมทั้งระบบ จึงควรให้ความสนใจ ณ เวลาที่ทุกโหนดประสานเวลาเสร็จ ว่าแต่ละโหนดนั้นจะมีเวลาต่างจากนาฬิกาโหนดอ้างอิง (เช่น รูตโหนด) เพียงใด ตัวชี้วัดสมรรถนะสำหรับโพรโทคอลการประสานเวลานี้คือ ค่าเฉลี่ยผิดพลาดจากการประสานเวลาของทั้งระบบ ซึ่งใช้ชื่อย่อว่า Average Synchronization Error – "เอเอสอี" และสามารถคำนวณได้ตามสูตร (2.5)

$$ASE = \frac{\sum_{i=1}^{n} SE_{Ri}}{n} \quad ; \qquad (2.5)$$

เมื่อ n คือจำนวนโหนดที่ผ่านการประสานเวลา

R คือรูตโหนด หรือโหนดที่เป็นเวลาอ้างอิง

i คือหมายเลขโหนดหรือลำดับโหนดที่ประสานเวลาเสร็จ

ดังรูปที่ 2.8 ค่าเอเอสอี จากโหนดที่ผ่านการประสานเวลาที่มีเพียงสองโหนดจะเท่ากับ $(SE_{RA} + SE_{RB})/2 = (e_{A(t2)} + e_{B(t2)})/2$ เป็นต้น

2.3.3 ปริมาณข้อมูลของโพรโทคอลการประสานเวลา (Communication Messages)

นอกจากความแม่นยำถูกต้องของโพรโทคอลการประสานเวลา ปริมาณข้อมูลที่ใช้สำหรับแลกเปลี่ยนกันตามโพรโทคอลก็เป็นปัจจัยสำคัญต่อสมรรถนะ เพราะจากที่ทราบกันแล้วว่าระบบโครงข่ายเซนเซอร์ไร้สายมีข้อจำกัดด้านพลังงาน ซึ่งงานสื่อสารโดยเฉพาะงานส่งข้อมูลนั้นต้องใช้พลังงานสูงกว่างานประมวลผล ดังนั้นจำนวนการสื่อสารที่น้อยที่สุดแต่ให้โพรโทคอลทำงานได้ถูกต้อง จึงเป็นปัจจัยที่ควรคำนึงถึงและท้าทายเป็นอย่างยิ่ง

2.4 การวิเคราะห์ผลกระทบจากการมีโหนดซึ่งมีนาฬิกาไม่เที่ยงตรงมากกว่าปกติ

ในสิ่งแวดล้อมปกติ หรือที่อุณหภูมิ 20 – 40 องศาเซลเซียส นาฬิกาแต่ละเรือนจะมีความน่าเชื่อถือหรือความเที่ยงตรงระดับหนึ่ง ซึ่งทางผู้ผลิตนาฬิกากำหนดค่าความเสถียรภาพไว้ประมาณ ±40 p.p.m (เมื่อตัวสร้างจังหวะการสั่นๆ ที่ความถี่ประมาณ 10,000 ครั้งต่อวินาที) ค่าความไม่แน่นอนนี้จะทำให้นาฬิกาเดินเร็วหรือช้าไปประมาณ 5.5 ไมโครวินาทีต่อวินาที หรือประมาณ 20

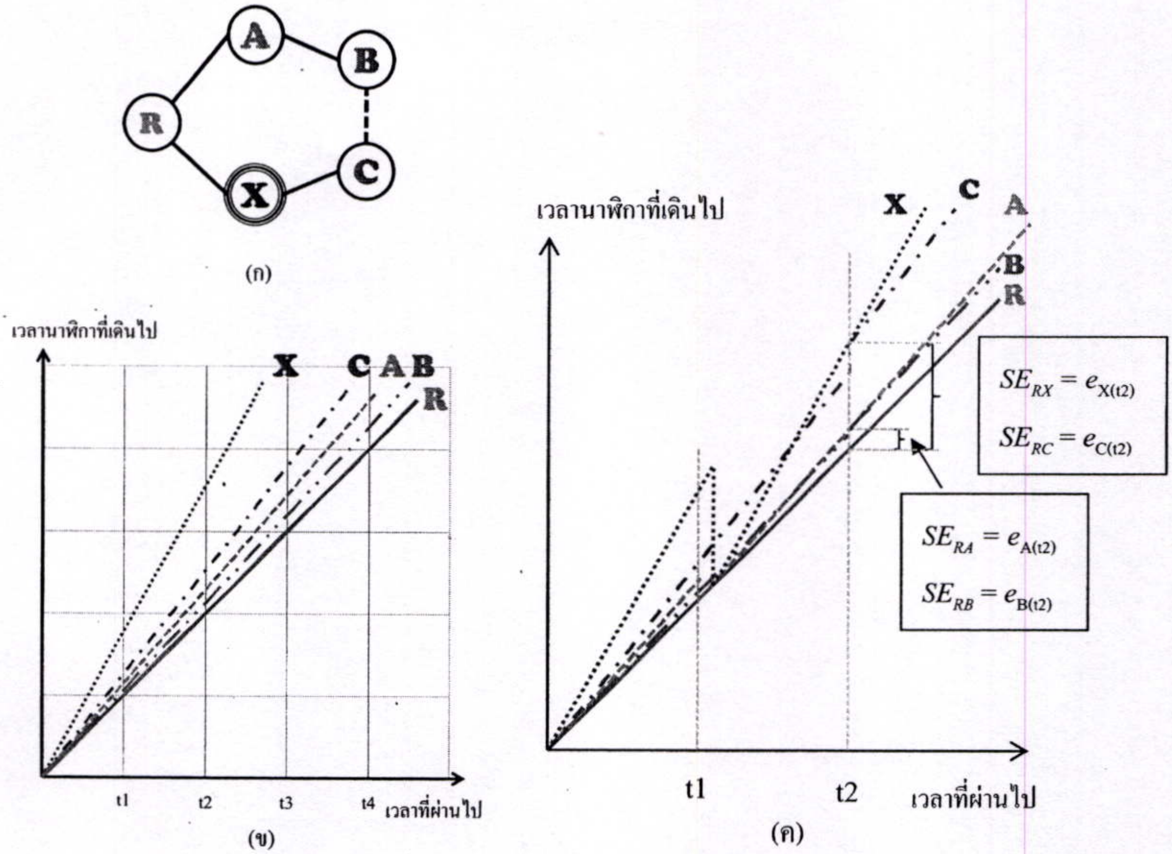
มิลลิวินาทีต่อชั่วโมง [6] และด้วยผลกระทบอื่นๆ ดังเช่น อุณหภูมิที่เปลี่ยนไป, กำลังไฟที่จ่ายเลี้ยงวงจร, การกระทบกระเทือน และความผิดปกติในตัวสร้างจังหวะแบบสั่น (Oscillator) เป็นต้น ปัจจัยเหล่านี้ล้วนส่งผลต่อความถี่ให้ไม่เสถียร และผลที่ตามมาคือนาฬิกาเดินไม่เที่ยงตรงมากกว่าปกติ

เมื่อนาฬิกาของบางโหนดเดินไม่เที่ยงตรงมากกว่าปกติ เวลาที่ผ่านไปก่อนและหลังการประสานเวลาจะทำให้เวลาของโหนดดังกล่าวแตกต่างไปจากโหนดอื่นๆ โดยเฉพาะเมื่อเทียบกับเวลาของโหนดอ้างอิง (รูตโหนด) เป็นอย่างมาก เราจึงเรียกโหนดที่มีนาฬิกาไม่เที่ยงตรงมากกว่าปกติ (หรือความถี่ที่ไม่เสถียรภาพอย่างมาก) นี้ว่า Fault Clock Node – ฟอลคล็อกโหนด (ซึ่งในงานวิจัยด้านความทนทานต่อความผิดพลาด มักเรียกโหนดที่ไม่ปกติว่า Faulty Node)

อีกทั้งเราได้นิยาม Bad Parent Node (บีพีเอ็น) คือ ฟอลคล็อกโหนดที่มีโหนดใดๆ มาเชื่อมต่อในฐานะโหนดลูก จึงทำให้ฟอลคล็อกโหนดนี้เป็นโหนดแม่ที่ไม่ดี ขณะที่ Inherited Bad Parent Node (ไอบีพีเอ็น) คือโหนดลูกที่ต่อเชื่อมกับบีพีเอ็นหรือไอบีพีเอ็น ซึ่งในวิทยานิพนธ์นี้ เราต้องการเสนอความสำคัญของการตระหนักและหลีกเลี่ยงโหนดแม่ทั้งสองประเภท (บีพีเอ็นและไอบีพีเอ็น)

รูปที่ 2.9 (ก, ข) แสดงตัวอย่างสมมุติกรณีมีโหนด X เป็นฟอลคล็อกโหนด และเป็นบีพีเอ็นให้กับโหนด C ประสานเวลาด้วย ทำให้โหนด C เป็นไอบีพีเอ็น ขณะที่โหนด A และ B เป็นโหนดที่นาฬิกาเที่ยงตรงปกติ เมื่อโครงสร้างลำดับชั้นนั้นสร้างตามแนวคิดโพรโทคอลซีเอชทีเอส [9] และอาร์ทีเอสพี [10] ที่พยายามให้โครงสร้างมีลักษณะสั้น ดังแสดงการเชื่อมต่อของโครงสร้างลำดับชั้นตามเส้นตรงทึบ ขณะที่เส้นตรงประหมายถึงเส้นทางสื่อสารสำรองที่โหนดสามารถเลือกใช้สื่อสารได้เช่นกัน

ผลของการประสานเวลาแสดงดังรูป 2.9 (ค) นั้นจะเริ่มที่เวลา *t1* โหนด A และโหนด X ต่างพยายามประสานเวลากับโหนด R แต่ด้วยการหลีกเลี่ยงการชนกันของโพรโทคอลชั้นดาต้าลิงค์ทำให้จะเห็นการเหลื่อมเวลากันเล็กน้อยของการประสานเวลาของโหนดทั้งสอง ต่อมาสมมุติให้ ณ เวลา *t2* ที่โหนด B และโหนด C ต่างประสานเวลาเสร็จพร้อมกันพอดี (ซึ่งโดยปกติจะไม่พร้อมกันเช่นนี้) ในการคำนวณหาค่าเอเอสอี สมมุติว่าพบค่า SE_{RA} , SE_{RB} , SE_{RC} และ SE_{RX} มีค่าผิดพลาดเทียบกับโหนดอ้างอิงเป็น $e_{A(t2)}$, $e_{B(t2)}$, $e_{C(t2)}$ และ $e_{X(t2)}$ หน่วยเวลา ตามลำดับ ดังนั้นค่าเอเอสอี จากโหนดที่ผ่านการประสานเวลาทั้งสี่โหนดจะเท่ากับสมการ (2.6)



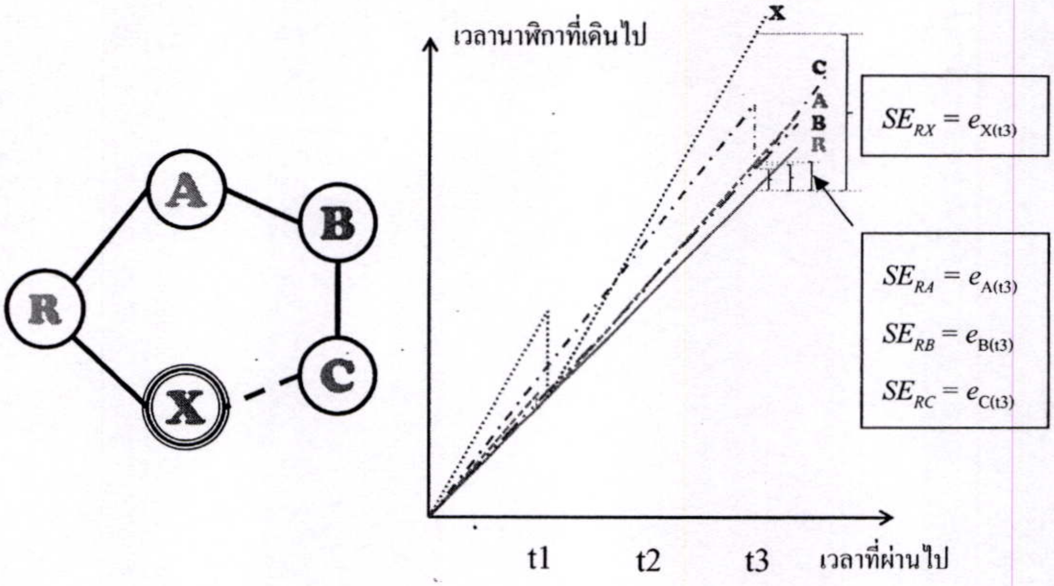
รูปที่ 2.9 ผลกระทบของบีพีเอ็นและไอบีพีเอ็นต่อค่าเอเอสอี

$$(SE_{RA} + SE_{RB} + SE_{RC} + SE_{RX})/4 \;=\; (e_{A(t2)} + e_{B(t2)} + e_{C(t2)} + e_{X(t2)})/4 \qquad (2.6)$$

เมื่อสังเกตความสัมพันธ์ของค่าผิดพลาดเมื่อเวลาผ่านไปของทั้งสี่โหนดจะพบว่าค่าเอสอีของโหนด C และ X มีค่าที่สูงกว่าโหนด A และ B มาก หรืออาจแสดงได้ดังสมการ (2.7)

$$(e_{A(t2)} \approx e_{B(t2)}) \quad << \quad (e_{C(t2)} \approx e_{X(t2)}) \qquad (2.7)$$

รูปที่ 2.10 เมื่อเราให้ความสำคัญต่อการเลือกโหนดแม่ที่พยายามหลีกเลี่ยงฟอลคล็อกโหนด (บีพีเอ็น) หรือไอบีพีเอ็นมากกว่าการมุ่งเน้นโครงสร้างลำดับชั้นที่สั้นเพียงอย่างเดียวนั้นสามารถทำให้ค่าเอสอีของโหนดส่วนใหญ่มีค่าที่เล็กลง และนั่นจะทำให้ค่าเอเอสอีลดลงด้วยแม้ในระบบจะมีฟอลคล็อกโหนด ซึ่งตรงตามความมุ่งหมายและวัตถุประสงค์ของวิทยานิพนธ์ และเกิดความแตกต่างจากงานที่ผ่านๆ มา ที่แม้โครงสร้างลำดับชั้นจะไม่สั้นที่สุด แต่สามารถทำให้ผลลัพธ์ดีกว่า



(ก) (ข)

รูปที่ 2.10 สมรรถนะที่ดีขึ้นเพราะค่าเอเอสอีลดลง เมื่อพยายามหลีกเลี่ยงบีพีเอ็นและไอบีพีเอ็น

เมื่อสังเกตความสัมพันธ์ของค่าเอสอีของทั้งสี่โหนด หลังจากแก้ไขโครงสร้างลำดับชั้นตามรูป 2.10 ที่แม้โครงสร้างลำดับชั้นและระยะเวลาที่ใช้ในการประสานเวลาของทุกโหนดจะสูงขึ้น แต่พบว่าค่าเอสอีของโหนด X ที่เป็นฟอลคล็อกโหนดเพียงโหนดเดียวที่มีค่าที่สูงกว่าโหนด A, B และ C มาก หรืออาจแสดงได้ดังสมการ (2.8)

$$(e_{A(t3)} \approx e_{B(t3)} \approx e_{C(t3)}) \quad << \quad e_{X(t3)} \tag{2.8}$$

บทที่ 3
วิธีการระบุและหลีกเลี่ยงฟอลคล็อกโหนด

โพรโทคอลประสานเวลาสำหรับโครงข่ายเซนเซอร์ไร้สายในปัจจุบัน พยายามให้โหนดมีเวลาที่สอดประสานหรือมีเวลาที่สอดคล้องกันมากที่สุด แม้นาฬิกาบางโหนดจะมีความไม่เที่ยงตรงมากกว่าปกติ หรือที่เรานิยามว่าเป็นฟอลคล็อกโหนด ซึ่งในบทนี้เราจะอธิบายรายละเอียดวิธีการประเมิน, ระบุ และหลีกเลี่ยงฟอลคล็อกโหนดเพื่อให้การประสานเวลาเกิดประสิทธิภาพสูงสุดภายใต้สถานการณ์ดังกล่าว

ในบทที่ 3 นี้ประกอบด้วย 3 หัวข้อ อันได้แก่ 1) วิธีการพิจารณาฟอลคล็อกโหนด 2) อัลกอลิทึมการเลือกโหนดแม่ที่ดีที่สุดโดยหลีกเลี่ยงฟอลคล็อกโหนด และ 3) การปรับปรุงโพรโทคอลทีพีเอสเอ็นให้มีคุณสมบัติตามทั้ง 2 ข้อ

3.1 วิธีการพิจารณาฟอลคล็อกโหนด (FCN's identification)

งานวิจัยทางด้านการทนทานต่อโหนดที่บกพร่องได้นิยามลักษณะโหนดที่บกพร่อง [12] [13] คือ โหนดที่มีพฤติกรรมร้าย, ก่อกวน, โจมตี และ ไม่ให้ความร่วมมือ เป็นต้น โดยมีส่วนทำให้สมรรถนะระบบแย่ลงตลอดจนเสี่ยงต่อความเสียหาย ซึ่งโหนดเหล่านี้มักถูกเรียกว่า ฟอลตี้โหนด (Faulty Node) แต่สำหรับฟอลตี้โหนดตามวัตถุประสงค์ของวิทยานิพนธ์เล่มนี้ คือโหนดที่มีนาฬิกาบกพร่องโดยไม่อาจหลีกเลี่ยงได้ หรือที่เราเรียกว่า ฟอลคล็อกโหนด โดยความผิดปกตินี้มีผลมาจากสิ่งแวดล้อมและปัจจัยทางคุณภาพของนาฬิกาโหนดเอง แต่กระนั้นโหนดก็ยังให้ความร่วมมือในการทำการประสานเวลาอย่างเต็มที่

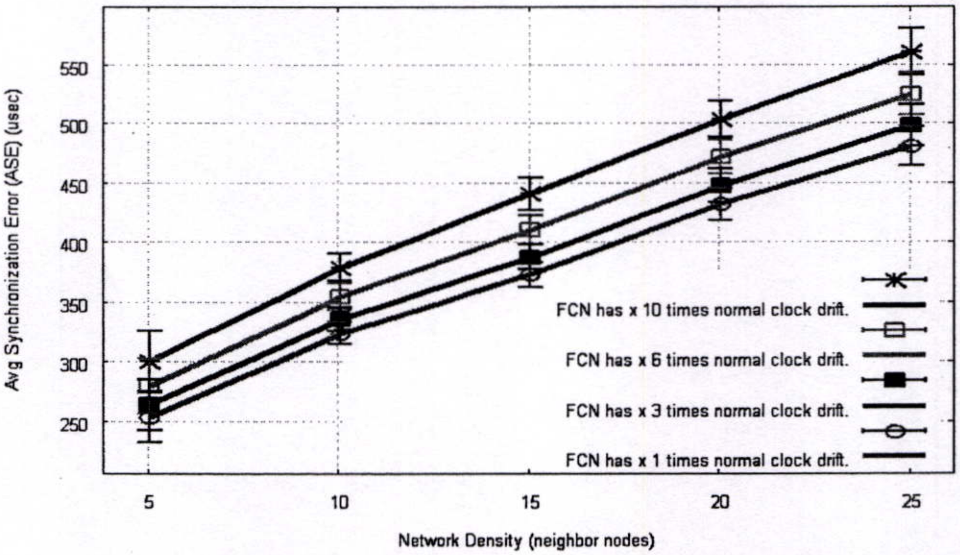
จากข้อสมมติเบื้องต้นที่ทุกโหนดให้ความร่วมมืออย่างเต็มที่ ดังนั้นข้อมูลเวลาของทุกโหนดจึงมีความน่าเชื่อถือระดับหนึ่ง (แม้จะไม่ถูกต้อง 100% แต่ทุกโพรโทคอลก็พยายามขจัดความไม่แน่นอนอันเกิดจากตัวแปรระหว่างการสื่อสาร ดังที่กล่าวไว้ในหัวข้อ 2.1.1 และ 2.2.1 ของบทที่ 2) และด้วยข้อมูลเวลาของโหนดข้างเคียง (หรือโหนดเพื่อนบ้าน) เราสามารถคำนวณหาเวลาที่คาดว่าโหนดใดๆ จะแตกต่างจากโหนดข้างเคียงได้ แล้วเปรียบเทียบกับค่าคลาดเคลื่อนสัมพัทธ์ของนาฬิกาในกรณีปกติ

ค่าคลาดเคลื่อนสัมพัทธ์นั้นสอดคล้องกับความเที่ยงตรงของนาฬิกา ผู้ผลิตนาฬิกาได้รับประกันค่าคลาดเคลื่อนสัมพัทธ์ไว้ที่ประมาณ 20 มิลลิวินาทีต่อชั่วโมง [6] หรือหมายความว่า ทุกๆ หนึ่งชั่วโมง ค่าคลาดเคลื่อนสัมพัทธ์ของนาฬิกานี้จะทำให้นาฬิกาเดินไม่ช้าก็เร็วประมาณ 20

มิลลิวินาที (หรือเท่ากับ 20,000 ไมโครวินาทีต่อชั่วโมง หรือเท่ากับ 333 ไมโครวินาทีต่อนาที หรือเท่ากับ 5.5 ไมโครวินาทีต่อวินาที)

สำหรับค่าคลาดเคลื่อนสัมพัทธ์ในงานวิจัยนี้ สมมุติให้ทุกโหนดที่ไม่ใช่โหนดรูตมีค่าคลาดเคลื่อนสัมพัทธ์แบบคงที่ ซึ่งกรณีเช่นนี้เกิดขึ้นได้ยากในความเป็นจริง เนื่องจากนาฬิกาสามารถเปลี่ยนแปลงค่าคลาดเคลื่อนสัมพัทธ์นี้ทีละน้อยหรือมากได้ตลอดเวลา ดังนั้นจึงเป็นการยากที่จะจำลองการทดลองให้ค่าคลาดเคลื่อนสัมพัทธ์เปลี่ยนแปลงได้เหมือนดังความเป็นจริง เพื่อควบคุมตัวแปรที่ไม่แน่นอนนี้สำหรับงานวิจัย จึงกำหนดให้แต่ละโหนดมีค่าคลาดเคลื่อนสัมพัทธ์ที่แตกต่างกันแต่มีค่าคงที่ในแต่ละการทดลอง

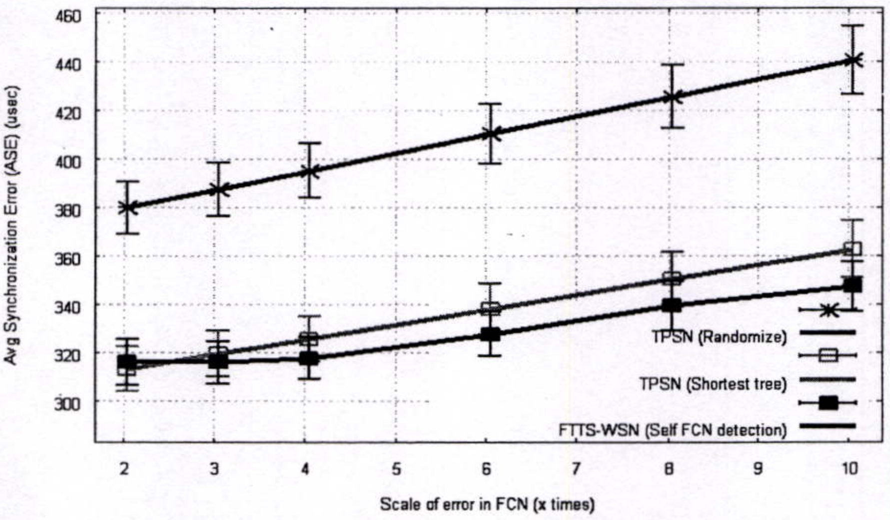
จากการทดลองพบว่าเมื่อค่าเฉลี่ยผิดพลาดจากการประสานเวลาของทั้งระบบ (หรือ "ค่าเอเอสอี") ของโพรโทคอลทีพีเอสเอ็น ในกรณีมีฟอลคล็อกโหนดปะปนเข้ามาจำนวนร้อยละ 3 ของ 200 โหนด โดยแสดงดังรูปที่ 3.1 แกนนอนคือ จำนวนโหนดข้างเคียงโดยเฉลี่ย หรือค่าที่สะท้อนความหนาแน่นของโหนด ส่วนแกนตั้งคือค่าเอเอสอี และเส้นกราฟทั้ง 5 เส้น แสดงอัตราค่าความคลาดเคลื่อนสัมพัทธ์ เส้นล่างสุดคือไม่มีฟอลคล็อกโหนดหรือทุกโหนดนาฬิกาเที่ยงตรงปกติ ส่วนเส้นที่อยู่เหนือขึ้นไปกำหนดให้ฟอลคล็อกโหนดทั้ง 6 โหนดใน 200 โหนดที่มีปริมาณความคลาดเคลื่อนสัมพัทธ์ 3, 6 และ 10 เท่าตามลำดับ จะเห็นว่าค่าความคลาดเคลื่อนสัมพัทธ์ของนาฬิกาฟอลคล็อกโหนดที่เพิ่มขึ้น จะทำให้ค่าเอเอสอีสูงขึ้น



รูปที่ 3.1 ผลกระทบของฟอลคล็อกโหนดที่มีผลต่อค่าเอเอสอี

สถานการณ์ดังเช่นรูปที่ 3.1 เราพบว่าโพรโทคอลตามวิทยานิพนธ์นี้ FTTS-WSN [14] สามารถช่วยลดค่าเอเอสอี และมีผลที่ดีกว่าโพรโทคอลประสานเวลาสำหรับโครงข่ายเซนเซอร์ไร้สายในปัจจุบัน เมื่อแต่ละโหนดสามารถระบุและหลีกเลี่ยงการประสานเวลากับฟอลคล็อกโหนดได้ ตามรูปที่ 3.2 แสดงผลการทดลองเปรียบเทียบสมรรถนะค่าเอเอสอีระหว่างโพรโทคอลที่เรา

นำเสนอกับโพรโทคอลทีพีเอสเอ็น (2.1.1) ซึ่งเลือกโหนดแม่แบบสุ่ม และโพรโทคอลที่มีแนวคิดเรื่องเน้นโครงสร้างลำดับชั้นให้สั้น ดังเช่น โพรโทคอลซีเอชทีเอส (2.1.2) หรือ โพรโทคอลอาร์ทีเอสพี (2.1.3) เป็นต้น สำหรับกรณีที่ยกมานี้เป็นเพียงกรณีหนึ่งจากหลายๆ กรณี โดยมีฟอลคล็อกโหนดร้อยละ 3 ของ 200 โหนด ความหนาแน่นเฉลี่ยจะมีโหนดข้างเคียง 15 โหนด แกนนอนคือจำนวนเท่าของค่าความคลาดเคลื่อนของนาฬิกาฟอลคล็อกโหนด และแกนตั้งคือสมรรถนะของโพรโทคอลการประสานเวลา (ค่าเอเอสอี) ซึ่งตามผลในรูปที่ 3.1 เราทราบว่าจำนวนและปริมาณค่าคลาดเคลื่อนของฟอลคล็อกโหนดทำให้ค่าเอเอสอีสูงขึ้นสำหรับโพรโทคอลทีพีเอสเอ็น ในรูปที่ 3.2 นี้แสดงด้วยเส้นกราฟเส้นบนสุด โพรโทคอลของเราได้แสดงให้เห็นว่าที่ปริมาณค่าคลาดเคลื่อน 2-3 เท่า ค่าเอเอสอีไม่แตกต่างจากโพรโทคอลที่พยายามสร้างโครงสร้างให้สั้น (แม้ค่าเฉลี่ยมีแนวโน้มต่ำกว่า แต่ที่ค่าความเชื่อมั่น 99% นั้นหมายถึงว่า ทั้งสองวิธีมีผลลัพธ์ไม่แตกต่างกัน) แต่เมื่อค่าคลาดเคลื่อนเพิ่มมากกว่า 3 เท่า วิธีการของเราแสดงสมรรถนะที่เหนือกว่าอย่างชัดเจน (ซึ่งผลในสถานการณ์อื่นๆ ก็ให้ผลในลักษณะที่คล้ายกันนี้ และจะแสดงอยู่ในบทที่ 4 ต่อไป)



รูปที่ 3.2 สมรรถนะโพรโทคอลประสานเวลาเมื่อในระบบมีฟอลคล็อกโหนดปะปนอยู่

ผลจากรูปที่ 3.2 นอกจากจะแสดงนัยทางสมรรถนะของโพรโทคอลแล้ว ยังแฝงความหมายของความจำเป็นในการระบุฟอลคล็อกโหนดอีกด้วย เพราะที่ค่าความคลาดเคลื่อน 2-3 เท่า อาจหมายถึงการระบุและหลีกเลี่ยงฟอลคล็อกโหนดที่ระดับนั้นไม่ได้ช่วยให้ผลดีขึ้นได้ เป็นต้น ซึ่งข้อมูลนี้ช่วยเรากำหนดค่าจุดเริ่มต้นของการประเมินฟอลคล็อกโหนด (*Faulty_Threshold*) เช่น เราควรพิจารณาฟอลคล็อกโหนดที่มีค่าไม่เที่ยงตรงสูงกว่าปกติ 2-3 เท่าเป็นต้นไป

การระบุฟอลคล็อกโหนดโดยการเปรียบเทียบค่าเฉลี่ยคลาดเคลื่อนสัมพัทธ์ของนาฬิกา กับค่าคลาดเคลื่อนสัมพัทธ์ของนาฬิกาฟอลคล็อกโหนด ประกอบด้วยกระบวนการโดยรวมแสดงดังรูปที่ 3.3 เริ่มจากการทำประสานเวลาเพื่อให้นาฬิกาทุกโหนดสอดประสานกันมากที่สุด หลังจากนั้นจะ

มีการรอด้วยเวลา *Synced_Time* เพื่อให้ฟอลคล็อกโหนดแสดงตัว เพราะด้วยเวลาที่ผ่านไป นาฬิกาของฟอลคล็อกโหนดจะแตกต่างจากโหนดข้างเคียงมากยิ่งขึ้น (ดังที่กล่าวไว้ใน 2.2.2) เวลา *Synced_Time* ที่เพียงพอจนสามารถแยกฟอลคล็อกโหนดออกจากโหนดปกติทั่วไปได้ ในขั้นตอนที่ 4 (รูปที่ 3.3) ทุกโหนด i จะทำการแลกเปลี่ยนเวลาระหว่างโหนดข้างเคียง j ที่มีจำนวน m_i โหนด ซึ่งจะได้ค่าเวลาที่คาดว่าแตกต่าง $drift_{ij}$ และคำนวณค่าเฉลี่ยเวลาที่คาดว่าแตกต่าง ($Average_Drift_i$) ตามสมการ (3.1)

$$Average_Drift_i = ABS\left(\frac{\sum_{j=1}^{m_i} drift_{ij}}{m_i}\right) \quad (3.1)$$

ค่าคลาดเคลื่อนสัมพัทธ์ของนาฬิกาฟอลคล็อกโหนด (*Faulty_Drift*: FD) ซึ่งเป็นเกณฑ์ที่เราใช้พิจารณานาฬิกาของฟอลคล็อกโหนด ถูกคำนวณในขั้นตอนที่ 5 (รูปที่ 3.3) มีค่าเท่ากับ จำนวนเท่าของค่าความคลาดเคลื่อนของนาฬิกา (*Faulty_Threshold*) คูณกับค่าความคลาดเคลื่อนของนาฬิกาปกติ (*Drift_Rate*) ที่สอดคล้องกับระยะเวลาหลังจากผ่านการทำการประสานเวลา (*Synced_Time*) ดังสมการที่ (3.2)

$$Faulty_Drift\ (FD) = Faulty_Threshold\ x\ Drift_Rate\ x\ Synced_Time \quad)$$

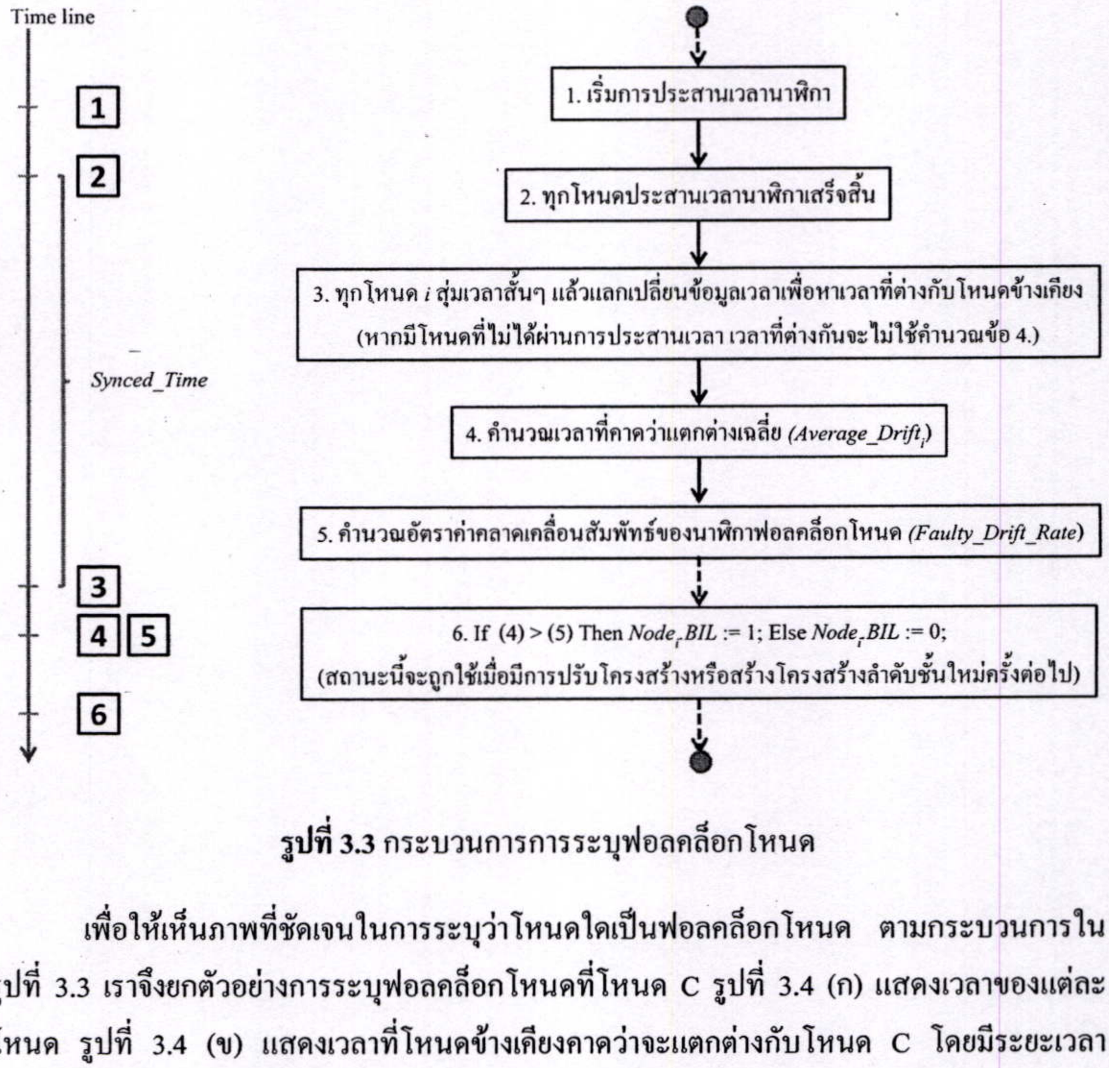
แต่มีคำถามสำหรับค่าที่เหมาะสมที่ใช้คำนวณค่า *FD* อยู่ 2 ข้อ คือ ค่าจำนวนเท่าของค่าคลาดเคลื่อนของนาฬิกาสำหรับประเมินฟอลคล็อกโหนด (*Faulty_Threshold*) ควรมีค่าเป็นเท่าใด? ตามรูปที่ 3.2 ค่าที่ 2.0 ถึง 3.0 สามารถแยกแยะฟอลคล็อกโหนดได้อย่างถูกต้องกว่า 96% เมื่อตัวแปร *Synced_Time* มีค่า 180 วินาที แต่เนื่องด้วย *Synced_Time* สามารถเปลี่ยนแปลงได้ ค่า *Faulty_Threshold* จึงควรปรับตามได้ เนื่องจากระยะเวลาภายหลังการประสานเวลา จะมีผลต่อความโดดเด่นของฟอลคล็อกโหนดและความแม่นยำในการระบุฟอลคล็อกโหนด เพื่อจะระบุฟอลคล็อกโหนดได้แม่นยำและไม่เข้าใจผิดเมื่อปล่อยให้ระยะเวลา *Synced_Time* มีค่ามากกว่านี้ ดังนั้นค่า *Faulty_Threshold* สามารถคำนวณได้ตามสมการ (3.3)

$$Faulty_Threshold\ (FT) = 2.0 + \frac{Synced_Time\ (second)}{360\ (second)} \quad (3.3)$$

คำถามอีกข้อคือ ระยะเวลา *Synced_Time* นานเท่าใด? จึงจะควรแลกเปลี่ยนข้อมูลเวลากับโหนดข้างเคียงเพื่อหาค่า $drift_{ij}$ ซึ่งจากการทดลองที่ระยะเวลา 20, 180 และ 360 วินาที ให้ความแม่นยำในการระบุฟอลคล็อกโหนดเป็น 87%, 96% และ 98% ตามลำดับ และที่ 180 วินาทีมีค่าความผิดพลาดจากการระบุฟอลคล็อกโหนด (Fault Positive หรือ Fall-Out) อยู่ในระดับที่ดีกว่า 360 วินาที ดังนั้นเราจึงเลือกค่า *Synced_Time* เป็น 180 วินาที

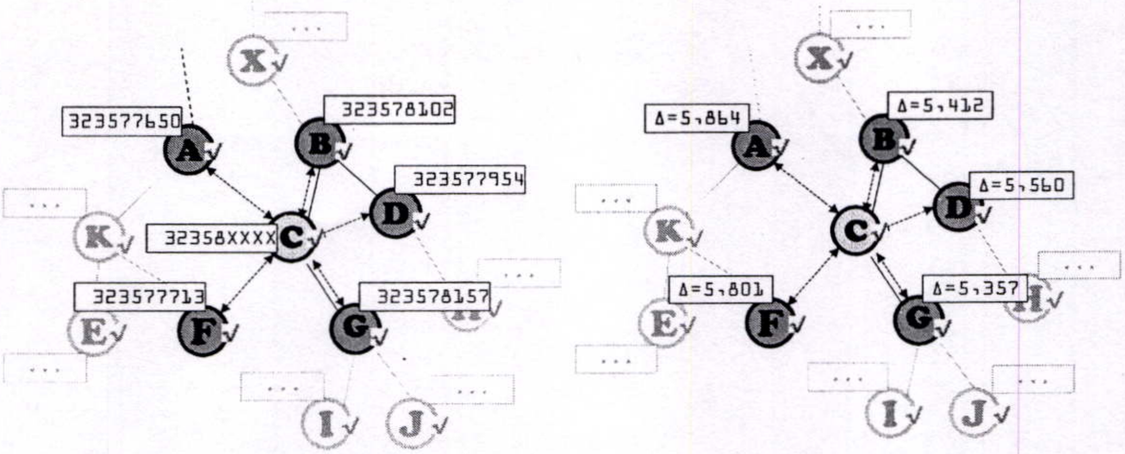
ค่าความคลาดเคลื่อนของนาฬิกาปกติ (*Drift_Rate*) ที่หน่วยต่อวินาทีมีค่าประมาณ 5.5 ไมโครวินาที [6] ถึงตรงนี้เราก็จะมีตัวแปรครบทุกตัวสำหรับคำนวณค่าคลาดเคลื่อนสัมพัทธ์ของนาฬิกาฟอลคล็อกโหนด (*Faulty_Drift* : FD) ได้ตามสมการ (3.2)

และเมื่อคำนวณผลลัพธ์ของสมการ (3.1) และ (3.2) ตามขั้นตอนที่ 4 และ 5 ดังในรูปที่ 3.3 แล้ว แต่ละโหนดจะมีข้อมูลสำหรับประเมินการเป็นฟอลคล็อกโหนดตามขั้นตอนที่ 6 ของรูปที่ 3.3 ซึ่งขั้นตอนนี้จะเกิดขึ้นเมื่อโพรโทคอลทำการปรับปรุงโครงสร้างลำดับชั้น หรือทำการสร้างโครงสร้างลำดับชั้นใหม่ (โดยจะกล่าวโดยละเอียดในหัวข้อ 3.3 ต่อไป)



รูปที่ 3.3 กระบวนการการระบุฟอลคล็อกโหนด

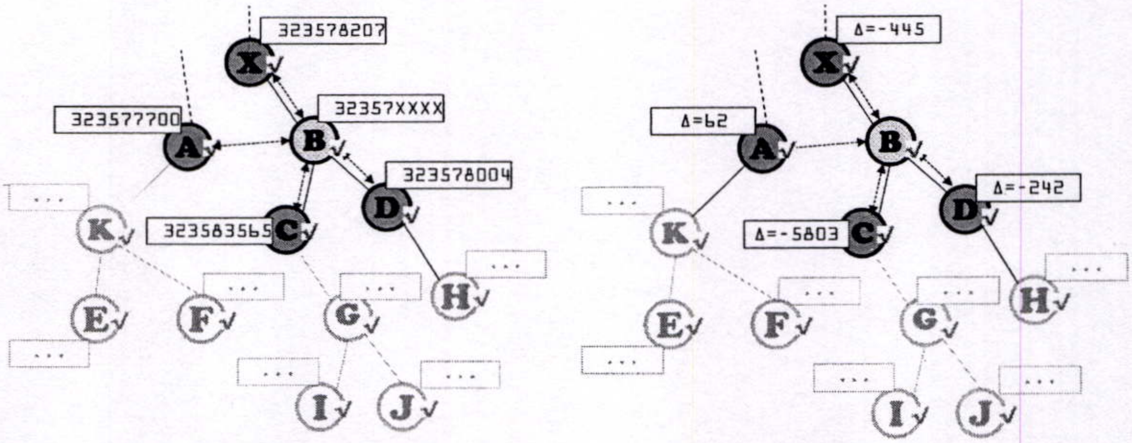
เพื่อให้เห็นภาพที่ชัดเจนในการระบุว่าโหนดใดเป็นฟอลคล็อกโหนด ตามกระบวนการในรูปที่ 3.3 เราจึงยกตัวอย่างการระบุฟอลคล็อกโหนดที่โหนด C รูปที่ 3.4 (ก) แสดงเวลาของแต่ละโหนด รูปที่ 3.4 (ข) แสดงเวลาที่โหนดข้างเคียงคาดว่าจะแตกต่างกับโหนด C โดยมีระยะเวลา *Synced_Time* = 180 วินาที, $Average_Drift_C$ = ABS(5864 + 5560 + 5357 + 5412 + 5801)/5 = 5,599 และ *FD* = 2.50 x 180 x 5.5 = 2,475 เมื่อวิเคราะห์ตามขั้นตอนที่ 6 พบว่า $Average_Drift_C$ มีค่ามากกว่า *FD* ดังนั้นโหนด C จึงเป็นฟอลคล็อกโหนด



(ก) (ข)

รูปที่ 3.4 ตัวอย่างกระบวนการการระบุฟอลคล็อกโหนดที่โหนด C

และตัวอย่างการระบุฟอลคล็อกโหนดที่โหนด B รูปที่ 3.5 (ก) แสดงเวลาของแต่ละโหนด รูปที่ 3.5 (ข) แสดงเวลาที่โหนดข้างเคียงคาดว่าจะแตกต่างกับโหนด B โดยมีระยะเวลา *Synced_Time* = 180 วินาที, $Average_Drift_B$ = ABS(-445 - 242 + 62 - 5803)/4 = 1,607 และ *FD* = 2.50 x 180 x 5.5 = 2,475 เมื่อวิเคราะห์ตามขั้นตอนที่ 6 พบว่า $Average_Drift_B$ มีค่าน้อยกว่า *FD* ดังนั้นโหนด B จึงเป็นโหนดที่มีนาฬิกาคลาดเคลื่อนปกติ หรือไม่เป็นฟอลคล็อกโหนด



(ก) (ข)

รูปที่ 3.5 ตัวอย่างกระบวนการการระบุฟอลคล็อกโหนดที่โหนด B

จากตัวอย่างที่ยกมาทั้งสองกรณี จะสังเกตุเห็นว่าสำหรับฟอลคล็อกโหนด เมื่อปล่อยเวลาผ่านไประยะหนึ่ง ค่าความคลาดเคลื่อนของนาฬิกาซึ่งสูงผิดปกติจะทำให้นาฬิกาเดินแตกต่างจากโหนดข้างเคียงอย่างชัดเจน ขณะที่โหนดปกติจะมีเวลาที่แตกต่างจากโหนดข้างเคียงน้อยแม้จะมีฟอลคล็อกโหนดอยู่ข้างเคียงอยู่จำนวนหนึ่ง แต่วิธีการนี้มีจุดอ่อนที่หากมีฟอลคล็อกโหนดเป็น

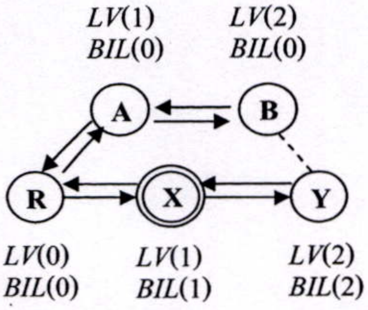
โหนดเพื่อนบ้านจำนวนมากอาจทำให้ค่า $Average_Drift_i$ สูงกว่า *FD* จนเข้าใจผิดจากโหนดดีกลายเป็นฟอลคล็อกโหนดได้

แนวคิดการระบุและหลีกเลี่ยงฟอลคล็อกโหนด โพรโทคอลกำหนดให้ทุกโหนดจะมีตัวแปรที่บ่งบอกสถานะการเป็นฟอลคล็อกโหนด หรือสถานะการสืบทอด ที่เรียกว่า *Bad_Inherited_Level* (*BIL* หรือเราเรียกว่า "บิล") สำหรับติดตามสถานะฟอลคล็อกโหนดที่เป็นโหนดแม่ในโครงสร้างลำดับชั้น (บีพีเอ็น – Bad Parent Node) และโหนดใดๆ ที่ประสานเวลากับบีพีเอ็นหรือไอบีพีเอ็น (ไอบีพีเอ็น – Inherited Bad Parent Node) โดยทำหน้าที่คล้ายตัวแปรที่ใช้บอกลำดับชั้นของตน (*own_level* หรือเราเรียกโดยย่อว่า *LV*)

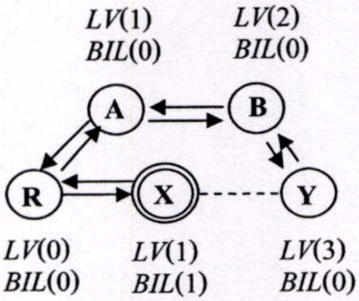
ณ เวลาเริ่มต้นหรือเมื่อโหนดเข้ามาสู่ระบบใหม่ ทุกโหนดจะกำหนดค่าเริ่มต้นให้ *LV* เป็น -1 และ *BIL* เป็น 0 ตามลำดับ ค่าดังกล่าวแสดงว่าโหนดนั้นยังไม่อยู่ในโครงสร้างลำดับชั้น หากได้พิจารณาโหนดตัวเองแล้วว่ามีนาฬิกาไม่เที่ยงตรงสูงกว่าปกติ (เป็นจริงตามขั้นตอนที่ 6 ของรูปที่ 3.3) โหนดดังกล่าวจะเปลี่ยนค่า *BIL* จากเดิมที่เป็น 0 เป็น 1 และเมื่อมีโหนดใดๆ มาขอเป็นโหนดลูกเชื่อมต่อกับฟอลคล็อกโหนดดังกล่าว (กำลังจะเป็นบีพีเอ็น) โหนดลูกนี้จะกลายเป็นไอบีพีเอ็น และจะนำค่า *BIL* ของโหนดแม่มาเพิ่มอีกหนึ่งแล้วใส่ให้กับตัวแปร *BIL* ของตน แต่ถ้าโหนดลูกมีนาฬิกาเที่ยงตรงปกติแล้วเลือกโหนดแม่ที่มีค่า *BIL* เป็นศูนย์ โหนดลูกจะสืบทอดค่า *BIL* ที่เป็นศูนย์จากโหนดแม่โดยทันที

รูป 3.6 (ก) โหนด Y เลือกโหนด X (บีพีเอ็นที่มี *BIL* เป็น 1) เป็นโหนดแม่ ส่วนค่า *LV* นั้นจะเริ่มโหนดรูตที่มีค่าเป็นศูนย์และจะเพิ่มขึ้นทีละหนึ่งทุกๆ ชั้นที่เชื่อมต่อกันไป

รูป 3.6 (ข) โหนด Y ตระหนักถึงโหนด X ซึ่งมีค่า BIL มากกว่าศูนย์ แสดงว่าโหนด X เป็นฟอลคล็อกโหนดจึงไม่ใช่ผลดีตามที่ได้วิเคราะห์กันในข้อ 2.4 เพราะจะทำให้ค่าเอเอสอีสูงขึ้น ดังนั้นจึงควรเลือกโหนดแม่ที่ดีที่สุดโดยพยายามหลีกเลี่ยงโหนดแม่ที่เป็นบีพีเอ็นและไอบีพีเอ็นนี้ ในกรณีนี้ โหนด Y เลือกโหนด B แม้จะมีเส้นทางที่ยาวกว่า แต่มีค่าผิดพลาดสะสมที่ต่ำกว่าเมื่อเทียบกับเส้นทางที่มีโหนด X เป็นโหนดแม่



(ก)



(ข)

รูปที่ 3.6 ตัวอย่างค่า *LV* และ *BIL* ในโครงสร้างลำดับชั้น (ก) โหนด Y มีค่า *BIL* เป็น 2 เพราะเพิ่มขึ้นหนึ่งจากค่า BIL ของโหนดแม่ (บีพีเอ็นหรือไอบีพีเอ็น) (ข) โหนด Y เลือกโหนด B เพื่อหลีกเลี่ยงบีพีเอ็นจึงมีค่า *BIL* เป็น 0 ตามค่า *BIL* ของโหนด B ขณะที่ค่า *LV* สูงขึ้นเป็น 3

อาจมีข้อสงสัย ในเมื่อสามารถระบุบีพีเอ็นและไอบีพีเอ็นแล้ว เหตุใดจึงยังประกาศสร้างโครงข่ายต่อจากโหนดบีพีเอ็นและไอบีพีเอ็นนี้ (ซึ่งถ้าหยุดก็จะไม่มีโหนดลูกมาเลือกโหนดเหล่านี้เป็นโหนดแม่ต่อโดยอัตโนมัติ) เหตุผลที่บีพีเอ็นและไอบีพีเอ็นยังคงจำเป็น เพราะบางโหนดที่ไม่มีโหนดเพื่อนบ้านอื่นให้เชื่อมต่อยกเว้นบีพีเอ็นและไอบีพีเอ็น โหนดนี้จะไม่เคยได้รับการประสานเวลาเลย จึงอาจจะมีค่าผิดพลาดที่สูงกว่าได้รับการประสานเวลา ซึ่งผลเซนเซอร์ที่อาศัยเวลาจากโหนดที่ไม่เคยประสานเวลาจะส่งผลเสียต่อการวิเคราะห์ข้อมูลเป็นอย่างมาก

3.2 อัลกอลิทึมการเลือกโหนดแม่ที่ดีที่สุดโดยหลีกเลี่ยงฟอลคล็อกโหนด (Best parent selection)

ขั้นตอนการเลือกโหนดแม่นี้พบในสามกรณี คือ กรณีที่หนึ่ง เมื่อโหนดรูตสั่งให้สร้างโครงสร้างลำดับชั้น (Level Discovery) กรณีที่สอง กรณีโหนดใดๆ เข้ามาใหม่ในขณะที่โหนดอื่นๆ ในระบบสร้างโครงสร้างลำดับชั้นเสร็จแล้ว หรือไม่สามารถติดต่อกับโหนดแม่เดิมได้ กรณีนี้จึงต้องค้นหาโหนดแม่ใหม่ (Panic Level Discovery) และกรณีที่สาม มีความคล้ายกับกรณีที่สองแต่แตกต่างที่โหนดแม่เปลี่ยนแปลงข้อมูลที่ส่งผลกระทบต่อการตัดสินใจโครงสร้างลำดับชั้น อันได้แก่ LV หรือ BIL เป็นต้น ทำให้โหนดลูกต้องปรับโครงสร้างตามหรือเปลี่ยนไป

ในสองกรณีแรกก่อนการตัดสินใจเลือกโหนดแม่ จะมีช่วงเวลาขณะหนึ่งสำหรับรอรับข้อมูลโหนดที่เป็นตัวเลือก (Parent's candidature) จากการตอบของโหนดข้างเคียงที่ได้อยู่ในโครงสร้างลำดับชั้นแล้ว ซึ่งคำตอบจากโหนดข้างเคียงเหล่านี้จะถูกจัดเก็บไว้เป็นฐานข้อมูลของโหนด เพื่อนำไปใช้กับอัลกอลิทึมเลือกโหนดแม่ต่อไป

รูปที่ 3.7 แสดงชุดคำสั่งในการเลือกโหนดแม่ที่ดีที่สุดด้วยการตรวจสอบตัวเลือกโหนดแม่ที่เก็บไว้ในฐานข้อมูลทุกตัวเลือก *โหนดที่เป็นตัวเลือกลำดับที่หนึ่ง* ที่ดีที่สุดคือ มีค่า *BIL* เป็นศูนย์ (*candidate[i].state* ไม่เป็น BPN/IBPN คือ *BIL* = 0 แต่ *candidate[i].state* ที่เป็น BPN/IBPN คือ *BIL* > 0) และมีค่า *LV* ต่ำสุดในบรรดาโหนดที่เป็นตัวเลือกอื่นๆ, แต่ถ้าหาก *BIL* เป็นศูนย์แต่ค่า *LV* เท่าเทียมกันหมด ถือว่าเป็น *โหนดที่เป็นตัวเลือกลำดับที่สอง* (ที่สามารถเลี่ยงบีพีเอ็นและไอบีพีเอ็นได้ แต่เส้นทางไม่สั้นที่สุด), แต่หากไม่ตรงตามเงื่อนไขทั้งสองข้อแรกนี้แล้ว *โหนดที่เป็นตัวเลือกลำดับที่สาม*นี้เป็นบีพีเอ็นหรือไอบีพีเอ็นซึ่งก็ต้องพยายามเลือกโหนดแม่ที่มี *BIL* และ *LV* ต่ำที่สุด

เพราะยิ่งลำดับชั้นมากยิ่งสะสมค่าผิดพลาดมากขึ้น, สำหรับ*โหนดที่เป็นตัวเลือกลำดับที่สี่* คือเป็นบีพีเอ็นหรือไอบีพีเอ็น ที่มี *BIL* ต่ำสุดและค่า *LV* เท่าเทียมกันหมด

เมื่อเปรียบเทียบโหนดที่เป็นตัวเลือกทุกตัวแล้ว จะทำการตัดสินใจเลือกประสานเวลากับโหนดแม่ที่ดีที่สุดตามลำดับความสำคัญที่พบ ซึ่งเรียงจาก *โหนดที่เป็นตัวเลือกลำดับที่หนึ่ง*, *โหนดที่เป็นตัวเลือกลำดับที่สอง*, *โหนดที่เป็นตัวเลือกลำดับที่สาม* และ *โหนดที่เป็นตัวเลือกลำดับที่สี่* ตามลำดับ

```
For i=1 To numbers of candidate parent received (m_i) Do
  If NOT ((candidate[i].state = = BPN) OR (candidate[i].state = = IBPN))
    If (candidate[i].level is lowest)
      first_choice ← candidate[i];
      /* This is a best candidate parent, also do shortest path to root node. */
    Else
      second_choice ← candidate[i];
      /* This parent may do not shortest path to root node. */
    Endif
  Else If (candidate[i].state is lowest BIL)
    third_choice ← candidate[i];
    /* This parent is a best of BPN or IBPN. */
  Else
    last_choice ← candidate[i]; is last choice to be the parent;
    /* This parent is very bad parent, because its do longest path to root and a huge of
cumulative error. */
  Endif
Endfor
/* Then set Parent_ID follow as priority choice were found from previous conditions */
  Parent_ID ← first_choice or second_choice or third_choice or last_choice respectively;
```

รูปที่ 3.7 วิธีการ การเลือกโหนดแม่ที่ดีที่สุด

3.3 การปรับปรุงโพรโทคอลทีพีเอสเอ็นให้ทนทานต่อฟอลคล็อกโหนด

โพรโทคอลทีพีเอสเอ็นนั้นมีขั้นตอนพื้นฐานในการประสานเวลาให้สอดคล้องกันอยู่ 2 ส่วน คือ ส่วนการสร้างโครงสร้างลำดับชั้น และส่วนการประสานเวลาให้สอดคล้องกัน ซึ่งมีรายละเอียดดังต่อไปนี้

3.3.1 การสร้างโครงสร้างลำดับชั้น (Level discovery)

ระบบโครงข่ายเซนเซอร์ไร้สายยังที่ไม่เคยสร้างโครงสร้างลำดับชั้น (Hierarchical tree) หรือมีความต้องการจะสร้างภาพโครงสร้างลำดับชั้นขึ้นใหม่ การร้องขอเพื่อสร้างโครงสร้างลำดับชั้นจะได้รับการปฏิบัติตามคำขอของโหนดรูต ซึ่งส่วนการทำงานนี้เป็นส่วนที่วิทยานิพนธ์นี้ได้ตระหนักและเพิ่มเติมการคัดเลือกโหนดแม่เพิ่มเติมเข้าไป มีรายละเอียดขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 โหนดรูตจะสร้างข้อมูลชนิด *LEVEL_DISCOVERY* และกำหนดให้ชั้นตัวเองเป็นจุดเริ่มต้นในโครงสร้างลำดับชั้น (มีค่า *LV* เป็น 0) แล้วประกาศข้อมูลนี้ออกไปให้โหนดเพื่อนบ้านผ่านโมดูลการสื่อสารของตน

สำหรับโหนดรูตมักมีนาฬิกาที่มีความเสถียรภาพมาก หรือผ่านการประสานเวลากับระบบนาฬิกาสากล เช่น NTP เป็นต้น จึงมีค่า *BIL* เป็น 0 เสมอ เพราะหากค่า *BIL* ของโหนดรูตมีค่ามากกว่าศูนย์ จะทำให้โหนดอื่นๆ ในระบบเข้าใจผิดและเกิดโครงสร้างลำดับชั้นที่ผิดปกติ

รูปที่ 3.8 เป็นโครงสร้างข้อมูลที่ถูกประกาศใช้เพื่อสร้างโครงสร้างลำดับชั้นจากโหนดรูต และโหนดใดๆ ที่ได้อยู่ในโครงสร้างลำดับชั้นแล้ว เพื่อให้โหนดอื่นๆ ได้มาเชื่อมต่อต่อไป

ขั้นตอนที่ 2 เมื่อโหนดใกล้เคียงใดๆ ได้รับข้อความที่มี DataType เป็น *LEVEL_DISCOVERY* ในครั้งแรก จะตั้งเวลานับถอยหลังแบบสุ่มระยะหนึ่ง แล้วบันทึกข้อมูลจากโหนดแม่ที่ได้รับครั้งแรกและครั้งต่อๆ ไปลงฐานข้อมูลไว้เป็นโหนดที่เป็นตัวเลือก

	0	1	2	3
0	DataType	BIL	LV	
4	Source address			
8	Destination address			
12	Optional			

รูปที่ 3.8 โครงสร้างข้อมูลโครงสร้างลำดับชั้น

<u>คำอธิบายฟิลด์ต่างๆ ของโครงสร้างขนาด 128 บิต (16 ไบต์)</u>

ฟิลด์ DataType ขนาด 1 ไบต์

0000 0110 (6) = *LEVEL_DISCOVERY*

0000 1000 (8) = *PANIC_LEVEL_REQUEST*

ฟิลด์ BIL ขนาด 1 ไบต์ มีค่าตั้งแต่ 0..255 ระบุสถานะของบีพีเอ็นหรือไอบีพีเอ็น

ฟิลด์ LV ขนาด 2 ไบต์ มีค่าตั้งแต่ -256..255 ระบุชั้นของโหนด

ฟิลด์ Source address ขนาด 4 ไบต์ ระบุหมายเลขโหนดแม่ที่พร้อมให้โหนดใดๆ มาเป็นโหนดลูกตน

ฟิลด์ Destination address ขนาด 4 ไบต์ ระบุหลายเลขโหนดปลายทาง แต่สำหรับการประกาศแบบไม่ระบุโหนดจะแทนค่าเป็น 4,294,967,295 หรือ FFFF FFFF ในระบบเลขฐาน 16

ขั้นตอนที่ 3 เมื่อรอจนครบเวลาสุ่มที่ตั้งไว้ โหนดนี้จะนำข้อมูลในฐานข้อมูลไปใช้คัดเลือกโหนดแม่ที่ดีที่สุดตามรูปที่ 3.7 ซึ่งจะพิจารณาโหนดแม่จากค่า *BIL* ที่น้อยที่สุดก่อน แล้วจึงพิจารณาค่า *LV* ต่อไป

ขั้นตอนที่ 4 หลังจากได้โหนดแม่ที่ดีที่สุดแล้ว ในรูปที่ 3.9 จะเป็นวิธีการแก้ไขค่า *LV* และค่า *BIL* ของโหนดให้ทันสมัย

โดยค่า *LV* นั้นเพียงแค่นำค่า *LV* ของโหนดแม่ (สมมติเป็น *i*) มาทำการบวกเพิ่มขึ้นจากเดิมหนึ่งชั้น (เป็น *i*+1) แล้วแทนค่าให้กับ *LV* ของตนเอง ยกตัวอย่างเช่นโหนดใดๆ ที่อยู่ในชั้นถัดจากโหนดรูต จะมีค่า *LV* เป็น 0 + 1 = 1 เป็นต้น

ส่วนค่า *BIL* นั้นก่อนจะแทนค่าให้กับตัวแปร จำเป็นต้องตรวจสอบ 2 เรื่องด้วยกัน (1) ค่า *BIL* ของโหนดแม่มีค่ามากกว่า 0 หรือไม่ (สมมติค่า *BIL* ของโหนดแม่ *j* คือ *BILj* > 0) หากจริงจะบวกเพิ่มขึ้นจากเดิมหนึ่งค่า (เป็น *j*+1) แต่ถ้าเงื่อนไขแรก (1) ไม่จริง จึงจะตรวจสอบเงื่อนไขที่ (2) ว่าโหนดดังกล่าวเป็นฟอลคล็อกโหนดหรือไม่ (ตามขั้นตอนที่ 6 รูปที่ 3.3 ข้อ 3.1) หากเป็นจริง จะแทนค่า *BIL* เป็น 1 แต่ถ้าไม่ใช่ จะแทนค่า *BIL* เป็น 0

```
lv ← pkt.parent.lv +1;             /* Own level is increased one from parent's lv. */
  /* (1) The selected parent is BPN or IBPN. */
  If (pkt.parent.bil ≥ 1)
    bil ← pkt.parent.bil+1;        /* bil in increased one from parent's bil. */
  Else
    /* (2) Own node is a BCN or a BPN. */
    If (isBPN(BIL, Average_Drift_node_id, Faulty_Drift_Rate))
      bil ← 1;                     /* bil is assigned to 1. */
    Else                           /* Own node is not a BCN or a BPN. */
      bil ← 0;                     /* bil is assigned to 0. */
    Endif
```

รูปที่ 3.9 วิธีการ การแก้ไขค่า *LV* และ *BIL* ให้ทันสมัย

ขั้นตอนที่ 5 หลังจากสร้างข้อมูลชนิด *LEVEL_DISCOVERY* ที่ใส่ข้อมูลที่แก้ไขให้ทันสมัยของโหนดตนแล้ว จะประกาศเพื่อส่งต่อให้โหนดข้างเคียงที่ยังไม่ได้อยู่ในโครงสร้างลำดับชั้นได้ทราบและจัดการตามขั้นตอนที่ 2-5 ต่อไป

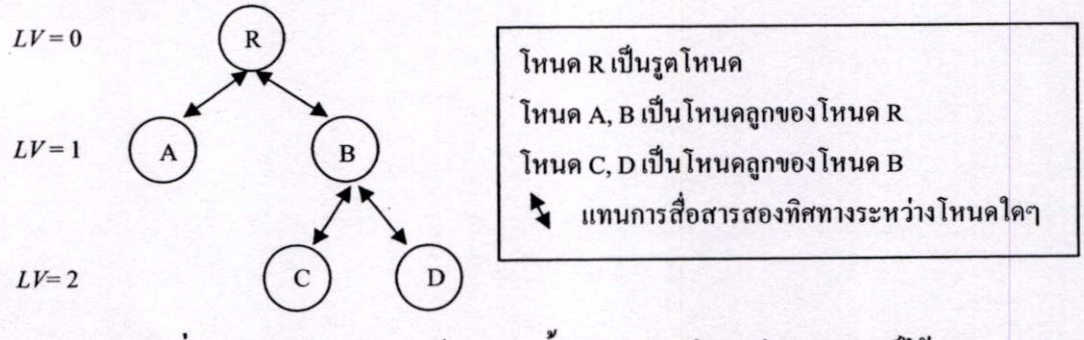
ทุกโหนดที่ได้ยินข้อมูลชนิด *LEVEL_DISCOVERY* นี้ จะทำตามขั้นตอนที่ 2-5 ไปเรื่อยๆ จนครบทุกโหนด เป็นอันเสร็จสิ้นขั้นตอนการสร้างโครงสร้างลำดับชั้นที่ได้รับการปรับปรุงเพื่อหลีกเลี่ยงฟอลคล็อกโหนด ขั้นตอนต่อไปจะเป็นการประสานเวลา หรืองานอื่นๆ ต่อไป

ส่วนโหนดใดที่ไม่ได้ยินข้อมูลชนิด *LEVEL_DISCOVERY* หรือเพราะเหตุใดก็ตามแต่ ทำให้ยังไม่ได้อยู่ในโครงสร้างลำดับชั้น เราจะกล่าวถึงวิธีการรองรับกรณีนี้ ในส่วนของข้อ 3.3.3 ต่อไป

3.3.2 การประสานเวลาให้สอดคล้องกันทั้งโครงข่าย (Time Synchronization)

ในขั้นตอนการประสานเวลา ภายหลังจากกำหนดความสัมพันธ์ระหว่างโหนดแม่กับโหนดลูกตามโครงสร้างลำดับชั้นแบบหลีกเลี่ยงฟอลคล็อกโหนดไปแล้วนั้น จะใช้การสื่อสารรูปแบบการแลกเปลี่ยนข้อมูลเวลา (TWME) เพื่อคำนวณหาเวลาที่คาดว่าจะต่างกันระหว่างโหนดลูกกับโหนดแม่ ซึ่งโหนดลูกจะประสานเวลาเข้าหาโหนดแม่ จากชั้นติดกับโหนดรูตและชั้นถัดต่อไปเรื่อยๆ จนครบทุกชั้นของโครงสร้างลำดับชั้นหรือทั่วทั้งโครงข่าย จากรูปที่ 3.10 ตัวอย่างโครงสร้างลำดับชั้น เพื่อประกอบขั้นตอนการประสานเวลาตามโพรโทคอลทีพีเอสเอ็น โดยมีขั้นตอนดังต่อไปนี้

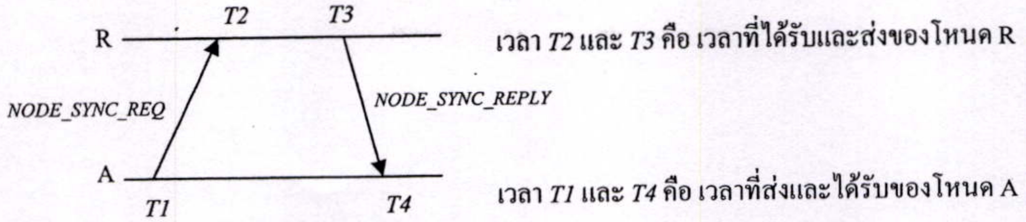
ขั้นตอนที่ 1 โหนด R (โหนดรูตหรือโหนดแม่ของทั้งโครงข่าย) เป็นโหนดที่ให้เวลาอ้างอิงสำหรับโหนดลูกใดๆ ตามโครงสร้างลำดับชั้น ส่งสัญญาณโดยประกาศข้อมูลชนิด *NODE_SYNC_MESSAGE* ให้โหนดข้างเคียงได้ทราบ



รูปที่ 3.10 ตัวอย่างโครงสร้างลำดับชั้นของระบบโครงข่ายเซนเซอร์ไร้สาย

ขั้นตอนที่ 2 โหนดลูกที่ได้ยินข้อมูล *NODE_SYNC_MESSAGE* จากโหนดแม่ของตน ซึ่งในที่นี้คือ โหนด A และ โหนด B จะสุ่มเวลาขึ้นมาค่าหนึ่ง สมมุติโหนด A ครบเวลาที่สุ่มก่อน ก็จะสร้างข้อมูลชนิด *NODE_SYNC_REQ* ที่จะบันทึกเวลาของตนเอง (*T1*) แล้วเพื่อส่งข้อมูลนี้ออกไปแลกเปลี่ยนกับโหนดแม่ (โหนด R) ดังในรูปที่ 3.11

ขั้นตอนที่ 3 โหนดแม่เมื่อได้ยินข้อมูลชนิด *NODE_SYNC_REQ* จากโหนดลูกของตน ซึ่งในที่นี้คือจากโหนด A ซึ่งจะบันทึกเวลาที่ได้ยินด้วยค่า *T2* แล้วสุ่มเวลาสั้นๆ เพื่อส่งข้อมูลนี้กลับคืน เมื่อครบเวลาที่สุ่มก็สร้างข้อมูลชนิด *NODE_SYNC_REPLY* ที่บันทึก *T1* และ *T2* จาก *NODE_SYNC_REQ* ที่ได้รับ พร้อมบันทึกเวลา *T3* ที่จะส่งข้อมูลออกไป แล้วจึงส่งข้อมูลนี้คืนให้กับโหนดลูก (โหนด A) ดังในรูปที่ 3.11



รูปที่ 3.11 การแลกเปลี่ยนข้อมูลเวลา (ทีดับบิวเอ็มอี) ระหว่างโหนด A และโหนด R

ขั้นตอนที่ 4 โหนดลูก (โหนด A) ได้รับข้อมูลชนิด *NODE_SYNC_REPLY* จากโหนดแม่ (โหนด R) กลับมา และบันทึกเวลาที่รับไว้ ณ เวลา *T4* โดยมีเวลา *T1* สอดคล้องกับข้อมูล *T1* ที่ได้บันทึกไว้ตอนส่ง *NODE_SYNC_REQ* ด้วยข้อมูลเวลา *T1*, *T2*, *T3* และ *T4* จะสามารถประสานเวลาของโหนดตนตามผลต่างของเวลาที่คาดว่าแตกต่าง (Clock Drift - Δ) กับโหนดแม่ ได้ดังตามสมการ (3.4)

$$\Delta = \frac{(T2-T1)-(T4-T3)}{2} \qquad (3.4)$$

ขั้นตอนที่ 5 ขณะที่โหนดอื่นกำลังประสานเวลากันอยู่ โหนดลูกใดๆ ที่ได้ยินข้อมูลชนิด *NODE_SYNC_REQ* จากโหนดแม่ของตน เช่น โหนด C และโหนด D ได้ยินข้อมูลดังกล่าวจากโหนด B ทำให้โหนดทั้งสองทราบว่าอีกสักครู่โหนดแม่ของตนจะประสานเวลากับโหนดที่อยู่ชั้นเหนือโหนดแม่เสร็จ ดังนั้นจึงสุ่มเวลาที่คาดว่านานพอ ที่โหนดแม่ของตน (โหนด B) จะทำการประสานเวลากับโหนดแม่ของโหนด B เสร็จ ซึ่งเมื่อครบตามเวลาที่สุ่มขึ้นก็จะทำขั้นตอน 2 – 4 เพื่อประสานเวลาให้สอดคล้องกับโหนดแม่ตามโครงสร้างลำดับชั้นต่อไป

รูปที่ 3.12 แสดงถึงโครงสร้างข้อมูลที่ใช้แลกเปลี่ยนข้อมูลเวลาระหว่างโหนด เพื่อใช้กับสมการ (3.4) คำนวณหาเวลาที่คาดว่าแตกต่างกันระหว่างโหนดใดๆ

	0	1	2	3
0	DataType	Optional	LV	
4	Source address			
8	Destination address			
12	T1			
16	T2			
20	T3			

รูปที่ 3.12 โครงสร้างข้อมูลการประสานเวลา

คำอธิบายฟิลด์ต่างๆ ของโครงสร้างขนาด 192 บิต (24 ไบต์)

ฟิลด์ DataType ขนาด 1 ไบต์

0000 0010 (2) = *NODE_SYNC_REQ*

0000 0011 (3) = *NODE_SYNC_REPLY*

0000 0100 (4) = *NODE_SYNC_MESSAGE*

ฟิลด์ LV ขนาด 2 ไบต์ มีค่าตั้งแต่ -256..255 ระบุชั้นของโหนด

ฟิลด์ Source address ขนาด 4 ไบต์ ระบุหมายเลขโหนดที่ส่งข้อมูล

ฟิลด์ Destination address ขนาด 4 ไบต์ ระบุหลายเลขโหนดปลายทาง

ฟิลด์ T1, T2 และ T3 ขนาดฟิลด์ละ 4 ไบต์ บันทึกเวลาของโหนดลูกตอนส่ง *NODE_SYNC_REQ*, บันทึกเวลาโหนดแม่เมื่อได้รับ *NODE_SYNC_REQ* และบันทึกเวลาโหนดแม่เมื่อส่ง *NODE_SYNC_REPLY* กลับ ตามลำดับ

สำหรับขั้นตอนที่ 2 และ 4 หากโหนดลูกใดๆ สร้าง *NODE_SYNC_REQ* เพื่อส่งไปยังโหนดแม่ของตนแล้วไม่ได้รับคำตอบกลับมาในช่วงเวลาหนึ่ง จะมีความพยายามส่ง *NODE_SYNC_REQ* ซ้ำ (บันทึก *T1* ใหม่) แต่หากพยายามครบ 4 ครั้งแล้วยังไม่ได้รับ *NODE_SYNC_REPLY* ตอบกลับมาจากโหนดแม่ อาจหมายความว่าโหนดแม่ตามโครงสร้างลำดับชั้นได้หายไป จะต้องค้นหาโหนดแม่ใหม่ ซึ่งจะมีรายละเอียดตามส่วน 3.3.3 ถัดไป

3.3.3 การจัดการกับกรณีพิเศษในการประสานเวลา

กรณีพิเศษที่มีโหนดถูกติดตั้งเข้ามาใหม่ หรือโหนดพึ่งตื่นภายหลังจากการสร้างโครงสร้างลำดับชั้นเสร็จไปแล้ว สถานการณ์เช่นนี้โหนดจะรอข้อมูลชนิด *LEVEL_DISCOVERY* ระยะเวลาหนึ่ง เมื่อไม่ได้ยินข้อมูลชนิดดังกล่าว หรือระหว่างรอนั้นได้ยินข้อมูลชนิด *NODE_SYNC_REQ* โหนดจะประกาศข้อมูลชนิด *PANIC_LEVEL_REQUEST* ไปยังโหนดข้างเคียงที่อยู่ในโครงสร้างลำดับชั้น คำตอบที่ได้รับกลับมาจะถูกบันทึกไว้และรอรับข้อมูลจากโหนดอื่นๆ อีกระยะเวลาหนึ่ง ก่อนจะตัดสินใจเลือกโหนดแม่ที่ดีที่สุดตามวิธีการข้อ 3.2

นอกจากนั้นในกรณีที่โหนดแม่ในโครงสร้างชั้นลำดับได้หายไป ทำให้โหนดลูกที่อยู่ในชั้นล่างลงไปหนึ่งชั้นไม่ได้ยิน *NODE_SYNC_REQ* จากโหนดแม่ของตนได้ หรือการส่ง *NODE_SYNC_REQ* ซ้ำครบ 4 ครั้ง โดยปราศจาก *NODE_SYNC_REPLY* ใดๆ ตอบกลับมาจากโหนดแม่ของตน โหนดลูกนี้จะพยายามแก้ไขโครงสร้างลำดับชั้นของตน โดยประกาศข้อมูลชนิด *PANIC_LEVEL_REQUEST* ไปยังโหนดข้างเคียงที่อยู่ในโครงสร้างลำดับชั้น และใช้คำตอบที่ได้รับกลับมาตัดสินใจหาโหนดแม่ที่ดีที่สุด ตามวิธีการข้อ 3.2 แล้วจึงประสานเวลาให้สอดคล้องตามวิธีการ 3.3.2 ต่อไป

ทีพีเอสเอ็น [7] กำหนดจำนวนความพยายามในการส่ง *NODE_SYNC_REQ* ซ้ำไว้ 4 ครั้ง จากซอร์สโค้ดโพรโทคอลทีพีเอสเอ็นในตัวแบบจำลอง ซึ่งเราคาดเดาเหตุผลที่กำหนดจำนวนความพยายามดังกล่าวเพราะในความพยายามที่มากเกินไป จะต้องใช้เวลานานและไม่สำเร็จ แต่หากพยายามน้อยเกินไป อาจต้องเสียเวลาและเพิ่มปริมาณข้อมูลในการปรับปรุงโครงสร้างลำดับชั้นใหม่ ซึ่งจำนวนครั้งความพยายาม (4 ครั้ง) ดังกล่าวน่าจะเป็นค่าที่เหมาะสมระหว่างความพยายามรอกับความสำเร็จที่จะได้รับมาสูงที่สุด

ในกรณีที่โหนดรูตหายไป แม้จะมีความเป็นไปได้ต่ำกว่ากรณีโหนดแม่หายไปจากโครงสร้างลำดับชั้น เนื่องมาจากโหนดรูตมีความสำคัญมากกว่าการเป็นโหนดอ้างอิงทางเวลา เพราะงานของระบบเซนเซอร์ยกให้โหนดรูตมีความสำคัญเทียบเท่าหัวใจของระบบ ซึ่งการหาโหนดรูตมาทดแทนโหนดเดิม จะกระทบกับงานทุกประเภท สำหรับการประสานเวลาเมื่อมีโหนดรูตใหม่จะต้องสร้างโครงสร้างลำดับชั้นใหม่หมด และทำการประสานเวลากันอย่างน้อยหนึ่งครั้ง

บทที่ 4

การประเมินสมรรถนะของระบบ

ในบทนี้กล่าวถึงแบบจำลองระบบ พารามิเตอร์ ค่าชี้วัดสมรรถนะ ผลการทดลองที่สนับสนุนสมมุติฐาน และผลการเปรียบเทียบสมรรถนะระหว่างโพรโทคอลที่นำเสนอกับโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize) เมื่อสมมุติสถานการณ์ให้ระบบโครงข่ายเซนเซอร์ไร้สายมีฟอลคล็อกโหนดปะปนอยู่จำนวนหนึ่ง

4.1 แบบจำลองที่ใช้ในการจำลองระบบ

แบบจำลองระบบเพื่อประเมินค่าสมรรถนะของระบบ คือ NESLsim [15] ซึ่งอาศัย PARSEC [16] เป็นแพลตฟอร์ม

PARSEC (PARallel Simulation Environment for Complex systems) คือ ภาษาตัวแบบจำลองเหตุการณ์แบบไม่ต่อเนื่องที่เป็นรูปแบบโปรแกรมภาษาซี ซึ่งสามารถจำลองเหตุการณ์ที่เกิดขึ้นพร้อมๆ กันแบบขนาน และให้ NESLsim ที่เป็นโมเดลของโหนดเซนเซอร์ที่มีโมดูลเป็นชั้นคล้ายโพรโทคอลเครือข่าย ประกอบด้วยโมดูลของโหนดเทียบได้กับชั้นแอพพลิเคชั่นผสมกับชั้นทรานสปอร์ต โมดูลของเรดิโอเทียบได้กับชั้นเครือข่าย และโมดูลของแชนแนลเทียบได้กับชั้นดาต้าลิงก์ และสามารถสั่งให้แต่ละโหนดทำงานได้อย่างอิสระจากกัน

4.2 ค่าชี้วัดสมรรถนะ

สำหรับการประเมินสมรรถนะสำหรับงานนี้ จะใช้ค่าชี้วัดสมรรถนะอันได้แก่

- ค่าเฉลี่ยผิดพลาดจากการประสานเวลา (Average Synchronization Error – *ASE* หรือค่า *เอเอสอี*)
- ค่าผิดพลาดจากการประสานเวลาสูงสุด (Maximum Synchronization Error – SE_{MAX})
- ปริมาณข้อมูลการประสานเวลา (Communication Messages)

4.3 พารามิเตอร์ที่ใช้ในการจำลองระบบ

พารามิเตอร์ส่วนใหญ่ที่ใช้ในการจำลองระบบโครงข่ายเซนเซอร์ไร้สายนี้แสดงดังในตารางที่ 4.1 โดยกำหนดพารามิเตอร์ทางกายภาพ (Physical Layer) ให้คล้ายกับอุปกรณ์เซนเซอร์ไร้สาย (UC Berkeley MICA Motes) และคล้ายกับงานวิจัยที่มีในปัจจุบันมากที่สุด ซึ่งในแต่ละการทดลองใช้การสุ่มลักษณะการ

กระจายอย่างมีรูปแบบ (Uniformly Random Distribution) แล้วคำนวณหาค่าเฉลี่ยจากการทดลอง 300 รอบ พร้อมค่าความเชื่อมั่น 99% ทุกการทดลองโหนดจะถูกสุ่มตำแหน่ง เพื่อให้ได้ความหนาแน่นของโหนดตามที่พารามิเตอร์ที่กำหนด โหนดจะมีสถานะตื่นตลอดเวลาการทดลอง และโหนดไม่เคลื่อนที่

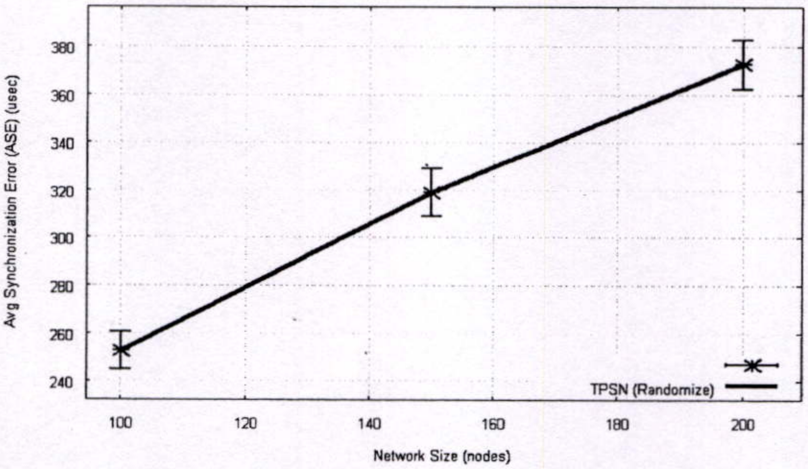
ตารางที่ 4.1 พารามิเตอร์ที่สำคัญในการจำลองระบบ

Parameter	**Value**
MAC Layer	CSMA MAC (no RTS/CTS/ACK และ no radio buffer)
Bit Error Ratio (BER)	ไม่มี หรือ การส่งรับจะต้องสำเร็จหรือไม่ก็ล้มเหลว เนื่องจากการชนกันของคลื่นวิทยุอย่างใดอย่างหนึ่ง
Random Model	Uniformly Distribution
Number of Sensor Node (N)	100, 150, 200
Node Density	5, 10, 15, 20 และ 25
Radio Speed	19.2 กิโลบิตต่อวินาที
Radio Transmission Range	20 เมตร
Message Size	128 บิต (16 ไบต์) และ 192 บิต (24 ไบต์)
Minimum Accuracy Attainable	1 ไมโครวินาที
Clock model	ความเที่ยงตรง ± 40 p.p.m [6] โหนดปกติสุ่มค่าความคลาดเคลื่อนสัมพัทธ์ไม่เกิน 5.5 ไมโครวินาทีต่อวินาที
Faulty nodes	3%, 5%, 10%, 20%, 30%, 40%, 50% และ 60%
Fault Clock Rate (times)	x2, x3, x4, x6, x8 และ x10
Detection Faulty Delay	20, 180 และ 360 วินาที
ทำการสร้างโครงสร้างลำดับชั้นทุกครั้งก่อนการประสานเวลา	ใช่
Time Stamp Layer	ทำการบันทึกข้อมูลเวลา ณ ชั้นแอพพลิเคชัน

4.4 ปัจจัยที่มีผลต่อสมรรถนะของระบบ

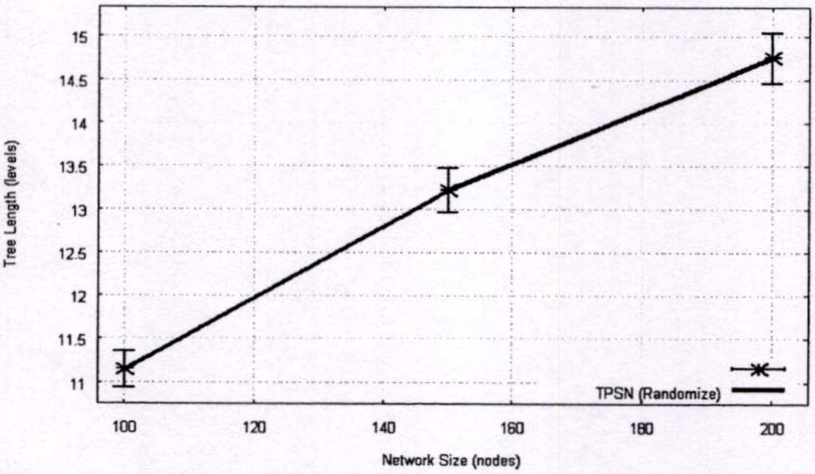
วิทยานิพนธ์นี้ตั้งสมมติฐานว่า (1) การประสานเวลาเป็นทอดๆ จะเกิดการสะสมเวลาที่แตกต่างมากขึ้นเนื่องจากค่าผิดพลาดจากการประสานเวลา และค่าผิดพลาดเมื่อเวลาผ่านไป (2) ฟอลคล็อกโหนดทำให้ค่าเอสอี และค่าเอเอสอีสูงขึ้น ดังการวิเคราะห์ในข้อ 2.3.1 และ ข้อ 2.3.2 (3) การหลีกเลี่ยงฟอลคล็อกโหนดดังที่ได้นำเสนอจะสามารถลดค่าเอเอสอีได้

เพื่อแสดงผลที่สอดคล้องกับข้อสมมุติฐานข้อแรก จึงทำการทดลองให้เครือข่ายมีขนาดใหญ่ขึ้น โดยคงความหนาแน่นของโหนดไว้คงที่ ทำให้เกิดการประสานเวลาแบบเป็นทอดๆ ที่ยาวขึ้น และใช้เวลานานขึ้น รูปที่ 4.1 แสดงค่าเอเอสอีที่เพิ่มขึ้นเนื่องจากโครงสร้างลำดับชั้นที่ยาวขึ้นและจำนวนโหนดที่เพิ่มขึ้น

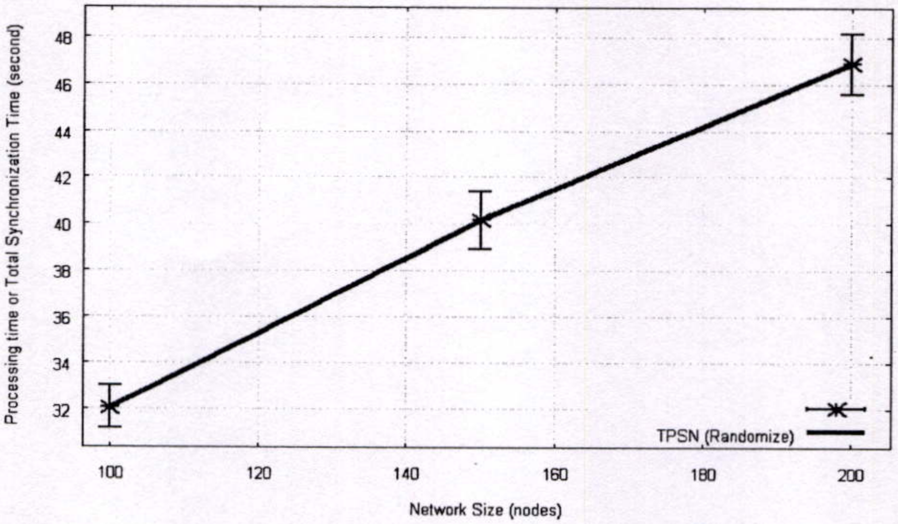


รูปที่ 4.1 จำนวนโหนดที่มากขึ้นทำให้ค่าเอเอสอีสูงขึ้น

รูปที่ 4.2 เมื่อเครือข่ายใหญ่ขึ้นจะทำให้ความยาวของโครงสร้างลำดับชั้นยาวขึ้น และในขณะเดียวกัน รูปที่ 4.3 เมื่อเครือข่ายใหญ่ขึ้นจะทำให้ระยะเวลาในการทำการประสานเวลานานขึ้น ซึ่งทั้งสองเป็นสาเหตุให้ค่าเอเอสอีสูงขึ้น ดังการวิเคราะห์ในข้อ 2.2

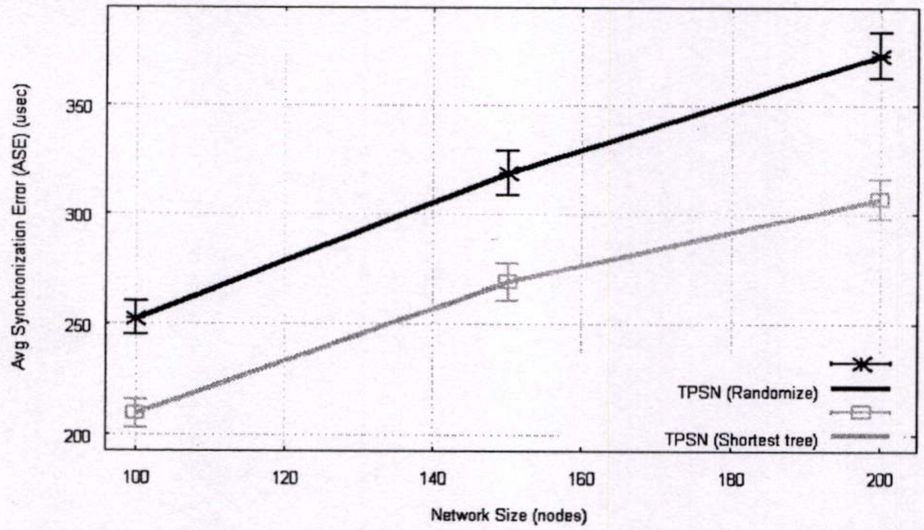


รูปที่ 4.2 จำนวนโหนดที่มากขึ้นทำให้ความยาวของโครงสร้างลำดับชั้นยาวขึ้น

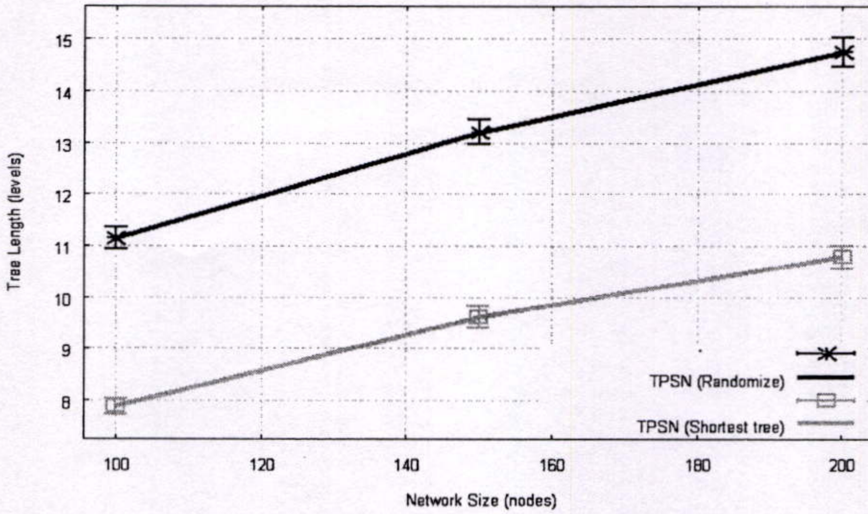


รูปที่ 4.3 จำนวนโหนดที่มากขึ้นทำให้ระยะเวลาการประสานเวลาทั้งระบบนานขึ้น

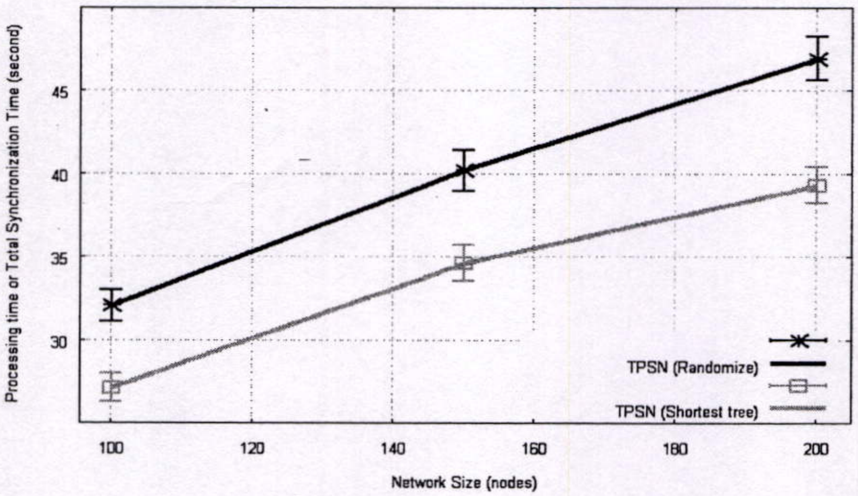
รูปที่ 4.4 แสดงโพรโทคอลที่มีอยู่ปัจจุบันที่มีแนวคิดให้โครงสร้างลำดับชั้นสั้นลง (TPSN - Shortest tree) สามารถช่วยลดค่าเอเอสอีได้ 15-17% โดยมีเหตุผลสอดคล้องตามรูปที่ 4.5 แสดงจำนวนโครงสร้างลำดับชั้นลดลง 27-29% และตามรูปที่ 4.6 ใช้ระยะเวลาในการประสานเวลาทั้งระบบสั้นลง 16-19% อีกด้วย



รูปที่ 4.4 ค่าเอเอสอีลดลงจากโพรโทคอล TPSN (Shortest tree)



รูปที่ 4.5 ความยาวโครงสร้างลดลงจากโพรโทคอล TPSN (Shortest tree)

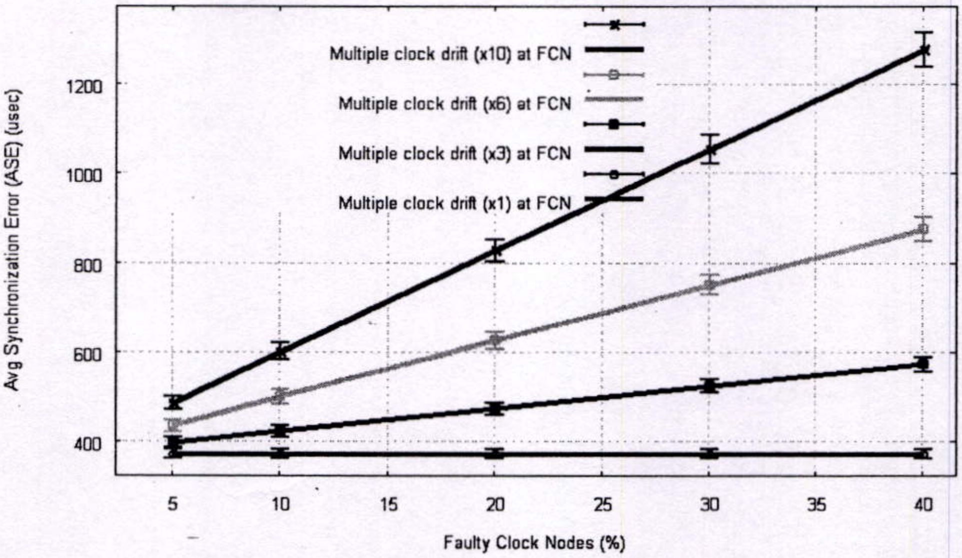


รูปที่ 4.6 เวลาในการประสานเวลาทั้งระบบสั้นลงจากโพรโทคอล TPSN (Shortest tree)

ด้วยผลการทดลองก่อนหน้านี้ แสดงให้เห็นว่าการประสานเวลาที่มีโครงสร้างลำดับชั้นเป็นปัจจัย ที่สมรรถนะจะสูงขึ้นได้โดยการพยายามบีบโครงสร้างให้สั้นลง

ข้อสมมุติฐานต่อมาคือฟอลคล็อกโหนดส่งผลกระทบต่อค่าเอเอสอีของโพรโตคอลทีพีเอสเอ็น โดยการทดลองต่อไปนี้ ทำการควบคุมค่าพารามิเตอร์ให้มีโหนดเซนเซอร์ 200 โหนด และความหนาแน่นให้แต่ละโหนดมีโหนดข้างเคียงเฉลี่ย 15 โหนด จากนั้นกำหนดให้มีฟอลคล็อกโหนดปริมาณต่างๆ (Fault Clock Node %) และแต่ละฟอลคล็อกโหนดจะมีอัตราค่าความคลาดเคลื่อนสัมพัทธ์ (Multiply clock drift) ในอัตราต่างๆ แสดงดังรูปที่ 4.7 ผลกระทบจากฟอลคล็อกโหนดต่อค่าเอเอสอี เมื่อทุกโหนดยังมีนาฬิกาเที่ยงตรง หรือมีอัตราค่าความคลาดเคลื่อนสัมพัทธ์ที่ 1 เท่าของค่าปกติ กราฟค่าเอเอสอีมีค่าเท่ากับ 372 ไมโครวินาที หลังจากทำการเพิ่มปริมาณฟอลคล็อกโหนดและอัตราค่าความคลาดเคลื่อนสัมพัทธ์ ค่าเอสอีนั้นสูงขึ้นดังตารางที่ 4.2

จากผลลัพธ์เบื้องต้นนี้สรุปได้ว่าทั้งปริมาณและอัตราค่าความคลาดเคลื่อนสัมพัทธ์ในฟอลคล็อกโหนด ต่างส่งผลให้ค่าเอเอสอีจากโพรโทคอลทีพีเอสเอ็นสูงขึ้น โดยยิ่งมีปริมาณและอัตราค่าความคลาดเคลื่อนสัมพัทธ์สูงขึ้น ยิ่งทำให้การประสานเวลาเกิดความผิดพลาด (ค่าเอเอสอี) สูงยิ่งขึ้นด้วย

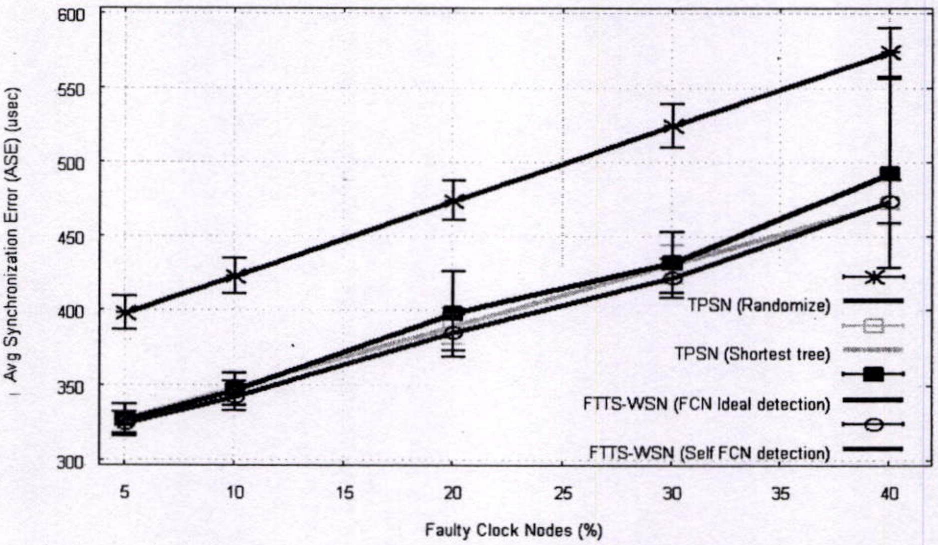


รูปที่ 4.7 ผลกระทบจากฟอลคล็อกโหนดต่อค่าเอเอสอี

ตารางที่ 4.2 ผลกระทบจากฟอลคล็อกโหนดต่อค่าเอเอสอี (หน่วยไมโครวินาที)

		False Clock Node (%)				
		5	10	20	30	40
Multiply clock drift at FCN (x เท่าของอัตราปกติ)	1	372.4 (0%)	372.4 (0%)	372.4 (0%)	372.4 (0%)	372.4 (0%)
	3	397.7 (+7%)	423.1 (+14%)	473.6 (+27%)	523.8 (+41%)	573.5 (+54%)
	6	435.4 (+17%)	499.0 (+34%)	625.1 (+68%)	750.6 (+102%)	874.9 (+135%)
	10	485.8 (+30%)	600.3 (+61%)	827.3 (+122%)	1,053.2 (+183%)	1,276.9 (+243%)

รูปที่ 4.8 เปรียบเทียบสมรรถนะค่าเอเอสอีของโพรโทคอลประสานเวลา เมื่อฟอลคล็อกโหนดมีค่าความคลาดเคลื่อนมากกว่าปกติ 3 เท่า (จากรูปที่ 4.7) ซึ่งโพรโทคอลที่นำเสนอลดค่าผิดพลาดได้ 18-19% เมื่อเทียบกับโพรโทคอลทีพีเอสเอ็นที่เลือกโหนดแม่แบบสุ่ม (TPSN - Randomize) อีกทั้งทำได้ดีกว่าประมาณ 1% เมื่อเทียบกับโพรโทคอลทีพีเอสเอ็นที่พยายามสร้างโครงสร้างให้สั้น (TPSN - Shortest tree) ขณะที่ผลการเปรียบเทียบสมรรถนะโพรโทคอลสำหรับกรณีฟอลคล็อกโหนดระดับต่างๆ แสดงรายละเอียดดังตารางที่ 4.3



รูปที่ 4.8 ค่าเอเอสอีเมื่อฟอลคล็อกโหนดมีค่าความคลาดเคลื่อนมากกว่าปกติ 3 เท่า

จากผลในตารางที่ 4.3 โพรโทคอลที่นำเสนอช่วยเพิ่มสมรรถนะให้โพรโคคอลประสานเวลาโดยลดผลกระทบจากฟอลคล็อกโหนดได้ดีกว่าโพรโทคอลที่ถูกเปรียบเทียบด้วย และเพิ่มความแตกต่างมากยิ่งขึ้นเมื่อจำนวนและอัตราค่าคลาดเคลื่อนของฟอลคล็อกโหนดสูงขึ้น

สมมุติฐานข้อที่สาม เพื่อให้การประสานเวลาทำงานได้ดีแม้มีฟอลคล็อกโหนด วิธีการที่ได้เสนอ คือ วิธีการระบุและหลีกเลี่ยงฟอลคล็อกโหนดดังที่ได้กล่าวถึงในบทที่ 3 สามารถช่วยลดค่าเอเอสอีได้ดีกว่าโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize) ดังรูปที่ 4.8 และตารางที่ 4.3 ซึ่งเหตุผลสนับสนุนสมมุติฐานข้อนี้จะแสดงในหัวข้อต่อไปนี้

ตารางที่ 4.3 เปรียบเทียบสมรรถนะค่าเอเอสอีของโพรโทคอลประสานเวลาที่ค่าความคลาดเคลื่อนของฟอลคล็อกโหนดระดับต่างๆ (หน่วยไมโครวินาที)

ค่าความคลาดเคลื่อนของฟอลคล็อกโหนด			จำนวนฟอลคล็อกโหนด (ร้อยละของโหนดทั้งหมด 200 โหนด)															
			5				10				20				30			
			(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)
	1		372.4				372.4				372.4				372.4			
	3		397.7	327.3	326.94	323.73	423.1	348.15	346.78	342.41	473.6	389.4	397.5	384.75	523.8	431.27	432.25	421.55
		(*)		(-17.7%)	(-17.8%)	(-18.6%)		(-17.7%)	(-18.0%)	(-19.1%)		(-17.8%)	(-16.1%)	(-18.8%)		(-17.7%)	(-17.5%)	(-19.5%)
		(**)			(-0.1%)	(-1.1%)			(-0.4%)	(-1.6%)			(2.1%)	(-1.2%)			(0.2%)	(-2.3%)
	6		435.4	357.52	347.93	342.18	499.0	409.62	390.97	382.42	625.1	512.69	498.16	496.61	750.6	617.39	585.36	590.22
		(*)		(-17.9%)	(-20.1%)	(-21.4%)		(-17.9%)	(-21.7%)	(-23.4%)		(-18.0%)	(-20.3%)	(-20.6%)		(-17.7%)	(-22.0%)	(-21.4%)
		(**)			(-2.7%)	(-4.3%)			(-4.6%)	(-6.6%)			(-2.8%)	(-3.1%)			(-5.2%)	(-4.4%)
	10		485.8	397.82	375.92	381.51	600.3	491.59	449.87	462.4	827.3	677.17	632.34	653.23	1,053.2	865.59	789.52	829.36
		(*)		(-18.1%)	(-22.6%)	(-21.5%)		(-18.1%)	(-25.1%)	(-23.0%)		(-18.1%)	(-23.6%)	(-21.0%)		(-17.8%)	(-25.0%)	(-21.3%)
		(**)			(-5.5%)	(-4.1%)			(-8.5%)	(-5.9%)			(-6.6%)	(-3.5%)			(-8.8%)	(-4.2%)

(1) : TPSN (Randomize)　　(2) : TPSN (Shortest tree)

(3) : FTTS-WSN (FCN Ideal detection)　　(4) : FTTS-WSN (Self FCN detection)

(*) : เปรียบเทียบกับ TPSN (Randomize)　　(**) : เปรียบเทียบกับ TPSN (Shortest tree)

4.5 สมรรถนะของระบบ

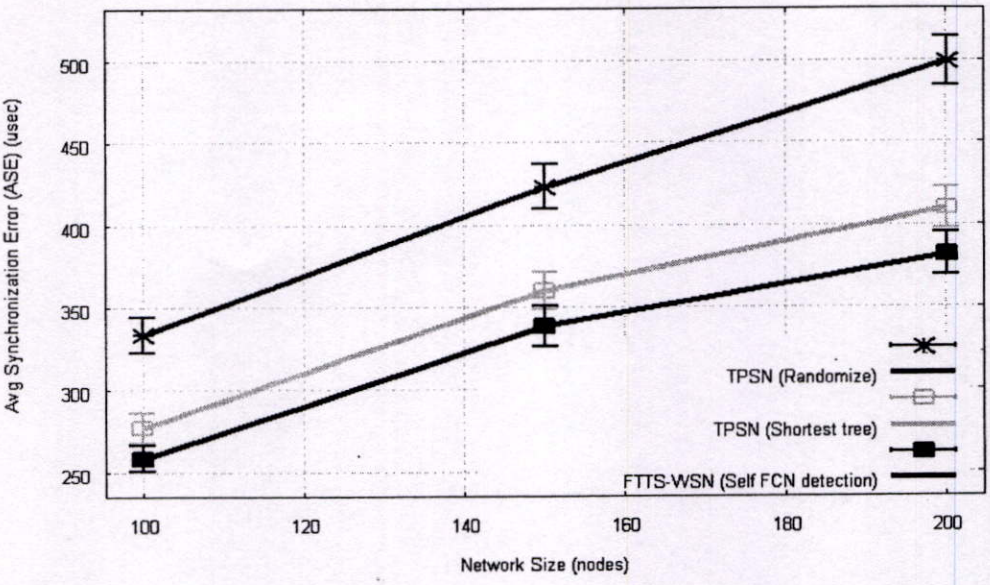
เนื่องจากมีตัวแปรพารามิเตอร์โครงข่ายเซนเซอร์ไร้สายจำนวนมาก เช่น จำนวนโหนด, ความหนาแน่นของเครือข่าย, ปริมาณฟอลคล็อกโหนดในเครือข่าย, อัตราค่าความคลาดเคลื่อนสัมพัทธ์ในฟอลคล็อกโหนด, โพรโทคอลประสานเวลาสำหรับโครงข่ายเซนเซอร์ไร้สาย เป็นต้น การจะนำเสนอผลการทดลองทุกกรณีในวิทยานิพนธ์นี้จึงเยอะเกินความเหมาะสม ซึ่งจากการสังเกตุผลลัพธ์ของกรณีตัวแปรดังที่กล่าวมาในเบื้องต้นมักแสดงออกในรูปแบบ หรือแนวโน้มเดียวกัน ในหัวข้อนี้ จึงเลือกนำเสนอกรณีที่เห็นว่าฟอลคล็อกโหนดได้ส่งผลกระทบอย่างมีนัยสำคัญต่อสมรรถนะของโพรโทคอลประสานเวลา แทนการนำเสนอจากทุกกรณีที่ได้ทำการทดลองหรือจากทุกความเป็นไปได้

4.5.1 สมรรถนะเอเอสอีและปัจจัยที่สนับสนุน

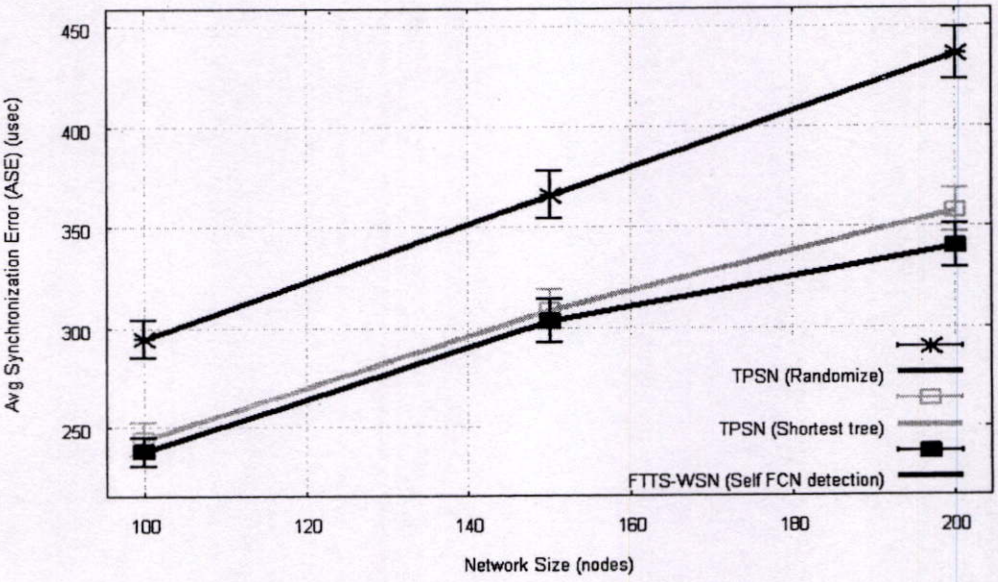
เพื่อจะแสดงผลลัพธ์ที่สนับสนุนข้อสมมุติฐานข้อสุดท้ายที่ว่า "โพรโทคอลที่สามารถระบุและหลีกเลี่ยงฟอลคล็อกจะประสานเวลาได้ดีกว่าโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize)" กำหนดให้มีโหนด 100, 150 และ 200 โหนด ความหนาแน่นให้แต่ละโหนดมีโหนดข้างเคียงเฉลี่ย 15 โหนด สุ่มให้มีฟอลคล็อกโหนด 10% ของโหนดทั้งหมด และฟอลคล็อกโหนดมีค่าความคลาดเคลื่อนสัมพัทธ์ 6 เท่าของอัตราปกติ

รูปที่ 4.9 เปรียบเทียบโพรโทคอลการประสานเวลาเมื่อมีฟอลคล็อกโหนดในเครือข่าย โดยโพรโทคอลที่นำเสนอ (โครงสร้างลำดับชั้นมีการระบุและหลีกเลี่ยงฟอลคล็อกโหนด, FTTS-WSN) ลดค่าเอเอสอีลงได้ 7% และ 25-30% เมื่อเปรียบเทียบกับวิธีการอันได้แก่ โพรโทคอลที่มีแนวคิดเรื่องเน้นโครงสร้างลำดับชั้นให้สั้น (TPSN-Shortest tree) ดังเช่น โพรโทคอลซีเอชทีเอส (2.1.2) หรือ โพรโทคอลอาร์ทีเอสพี (2.1.3) และโพรโทคอลทีพีเอสเอ็น (2.1.1) ซึ่งเลือกโหนดแม่แบบสุ่ม (TPSN-Randomize) ตามลำดับ

รูปที่ 4.10 คล้ายกับรูปที่ 4.9 จะแตกต่างที่กำหนดให้จำนวนฟอลคล็อกโหนดลดลงจาก 10% เหลือเพียง 5% ของโหนดทั้งหมด ซึ่งผลการเปรียบเทียบโพรโทคอลที่นำเสนอ (FTTS-WSN) ยังคงลดค่าเอเอสอีลงได้ 2-5% และ 20-28% เมื่อเทียบกับโพรโทคอล TPSN (Shortest tree) และโพรโทคอล TPSN (Randomize) ตามลำดับ และพบว่าเมื่อฟอลคล็อกโหนดมีจำนวนน้อยลงความแตกต่างของสมรรถนะโพรโทคอลที่นำเสนอกับโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize) จะด้อยลง หรือที่ความเชื่อมั่นระดับ 99% สามารถเรียกได้ว่าไม่แตกต่างอย่างชัดเจน ยกเว้นที่กรณี 200 โหนดซึ่งผลแตกต่างชัดเจน แต่อย่างไรก็ตามค่าเอเอสอีของโพรโทคอลที่นำเสนอก็มีแนวโน้มที่ต่ำกว่าโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize)



รูปที่ 4.9 โพรโทคอลการประสานเวลาเมื่อมีฟอลคล็อกโหนดในเครือข่าย 10%

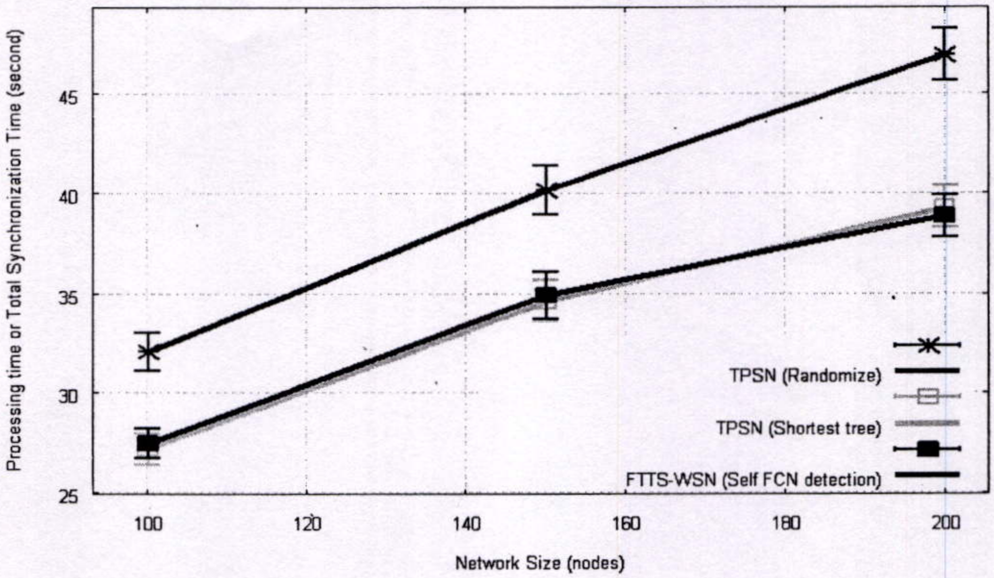


รูปที่ 4.10 โพรโทคอลการประสานเวลาเมื่อมีฟอลคล็อกโหนดในเครือข่าย 5%

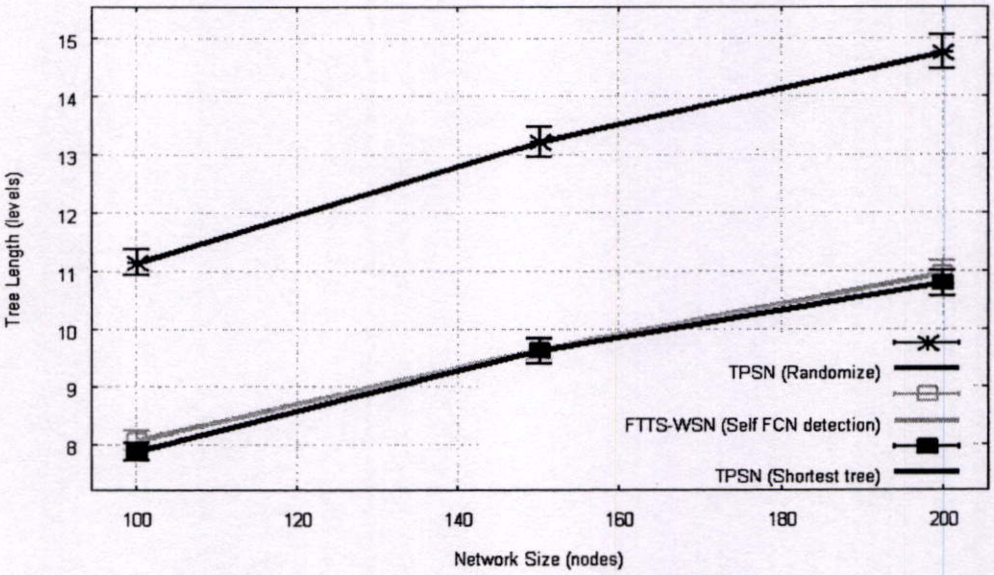
ก่อนจะสรุปว่าโพรโทคอลที่นำเสนอนี้ สามารถเพิ่มสมรรถนะให้การประสานเวลากรณีมีฟอลคล็อกโหนดได้จริง ควรดูผลลัพธ์ในมิติของระยะเวลาที่ใช้ในการประสานเวลา และจำนวนชั้นหรือความยาวโครงสร้างลำดับชั้นของแต่ละโพรโทคอล ซึ่งประกอบด้วยรูปที่ 4.11 และรูปที่ 4.12 ตามลำดับ (จากกรณีเดียวกับรูปที่ 4.9 คือ ฟอลคล็อกโหนด 10% ของจำนวนโหนดทั้งหมด)

ดังจะเห็นโพรโทคอลที่นำเสนอ (FTTS-WSN) มีระยะเวลาในการประสานเวลาไม่แตกกับโพรโทคอล TPSN (Shortest tree) ซึ่งทั้งสองโพรโทคอลนี้ใช้เวลาต่ำกว่าโพรโทคอล TPSN

(Randomize) ถึง 15-20% เมื่อดูความยาวโครงข่ายลำดับชั้นหรือจำนวนชั้นพบว่าโพรโทคอลที่นำเสนอ มีโครงสร้างลำดับชั้นที่ยาวกว่าโพรโทคอล TPSN (Shortest tree) เล็กน้อย แต่ทั้งสองโพรโทคอลนี้โครงสร้างลำดับชั้นสั้นกว่าโพรโทคอล TPSN (Randomize) 35-41%



รูปที่ 4.11 ระยะเวลาที่ใช้ในการประสานเวลา



รูปที่ 4.12 จำนวนชั้นหรือความยาวของโครงสร้างลำดับชั้น

เมื่อพิจารณาระหว่างโพรโทคอลที่นำเสนอกับโพรโทคอล TPSN-Shortest tree ซึ่งโพรโทคอลที่นำเสนอนั้นใช้ระยะเวลาประสานเวลาไม่ได้เร็วกว่า และไม่ได้มีจำนวนชั้นหรือโครงสร้างลำดับชั้นที่สั้นกว่า แต่กลับแสดงผลสมรรถนะหรือค่าเอเอสอีได้ต่ำกว่าโพรโทคอล

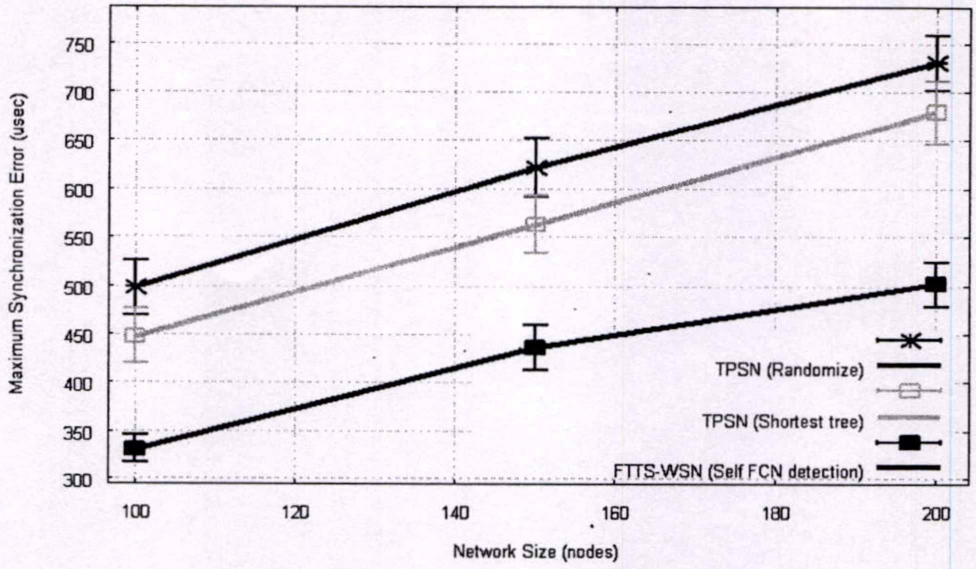
TPSN-Shortest tree (ดังรูปที่ 4.9) ซึ่งนั่นแสดงว่า การหลีกเลี่ยงฟอลคล็อกโหนด (บีพีเอ็น และไอบีพีเอ็น) พร้อมพยายามสร้างโครงสร้างลำดับชั้นให้สั้นเท่าที่จะทำได้ เป็นปัจจัยที่ส่งผลดีต่อสมรรถนะค่าเอเอสอีในโพรโทคอลประสานเวลาจริง โดยเฉพาะกรณีที่เครือข่ายมีฟอลคล็อกโหนดปะปนอยู่ และตรงดังสมมติฐานข้อ 1.3

4.5.2 สมรรถนะค่าผิดพลาดจากการประสานเวลาสูงสุดและปัจจัยสนับสนุน

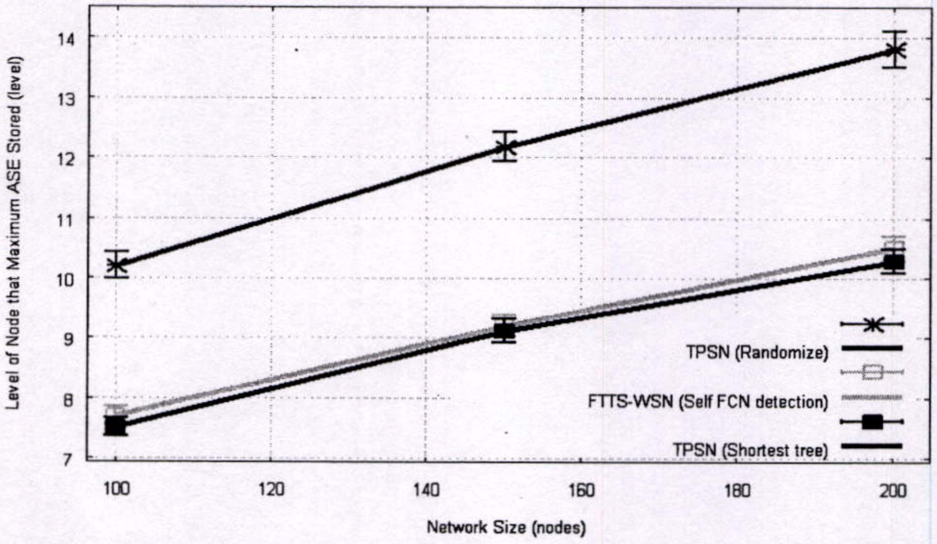
นอกจากค่าชี้วัดสมรรถนะค่าเอเอสอี (สมรรถนะในภาพรวม) ค่าผิดพลาดจากการประสานเวลาสูงสุด หรือ SE_{MAX} ก็เป็นอีกค่าที่มีความน่าสนใจ อย่างที่ทราบกันว่าค่าเอสอีนี้จะสะสมตามระยะเวลาและจำนวนความยาวของลำดับชั้นที่มากขึ้น แต่ด้วยสมมุติฐานที่บอกว่าโหนดสุดท้ายที่ประสานเวลากับฟอลคล็อกโหนด หรือประสานเวลาในเส้นทางที่มีฟอลคล็อกโหนดจะมีเวลาแตกต่างจากโหนดอ้างอิงมาก (ดังรูป 2.8) ดังนั้นเมื่อหลีกเลี่ยงการประสานเวลากับฟอลคล็อกโหนดค่า SE_{MAX} ดังกล่าวควรต่ำลงด้วย แต่การพิจารณาค่า SE_{MAX} นี้จะไม่นับค่าเอสอี ซึ่งมาจากฟอลคล็อกโหนด เนื่องจากฟอลคล็อกโหนดจะมีค่าเอสอีสูงเร็วมาก เพราะมีค่าความคลาดเคลื่อนสัมพัทธ์ที่สูงกว่าโหนดปกติทั่วไป ดังนั้นผลลัพธ์ต่อไปนี้จึงพิจารณาเฉพาะโหนดนาฬิกาปกติที่มีค่าเวลาที่แตกต่างกับโหนดอ้างอิงสูงที่สุด (โดยใช้สถานการณ์การทดลองเช่นเดียวกับรูปที่ 4.9)

จากรูปที่ 4.13 แสดงสมรรถนะโพรโทคอลที่นำเสนอ (FTTS-WSN) มีค่า SE_{MAX} ต่ำกว่า 30-35 และ 43-50% เมื่อเทียบกับโพรโทคอล TPSN (Shortest tree) และโพรโทคอล TPSN (Randomize) ตามลำดับ ถึงแม้ว่าโหนดที่มีเวลาแตกต่างจากโหนดอ้างอิงสูงนี้จะมีจำนวนไม่มาก แต่ด้วยผลที่แตกต่างกันมากนี้เอง ย่อมส่งผลต่อความถูกต้องและการทำงานของโหนดเซนเซอร์ไม่มากก็น้อย

รูปที่ 4.14 แสดงชั้นของโหนดที่มีค่า SE_{MAX} ซึ่งสอดคล้องถึงความยาวของโครงสร้างลำดับชั้นในแต่ละโพรโทคอล (รูปที่ 4.12) เพราะโหนดที่มีค่าเอสอีสูงที่สุดหรือเป็น SE_{MAX} มักอยู่ปลายของโครงสร้างลำดับชั้น ซึ่งผลการทดลองนี้โพรโทคอลที่นำเสนอมีโหนด SE_{MAX} อยู่ชั้นที่สูงกว่าโพรโทคอล TPSN (Shortest tree) เล็กน้อย และทั้งสองโพรโทคอลนี้มีโหนด SE_{MAX} อยู่ในชั้นที่ต่ำกว่าโพรโทคอล TPSN (Randomize) ถึง 33-35%



รูปที่ 4.13 ค่าผิดพลาดจากการประสานเวลาสูงสุด



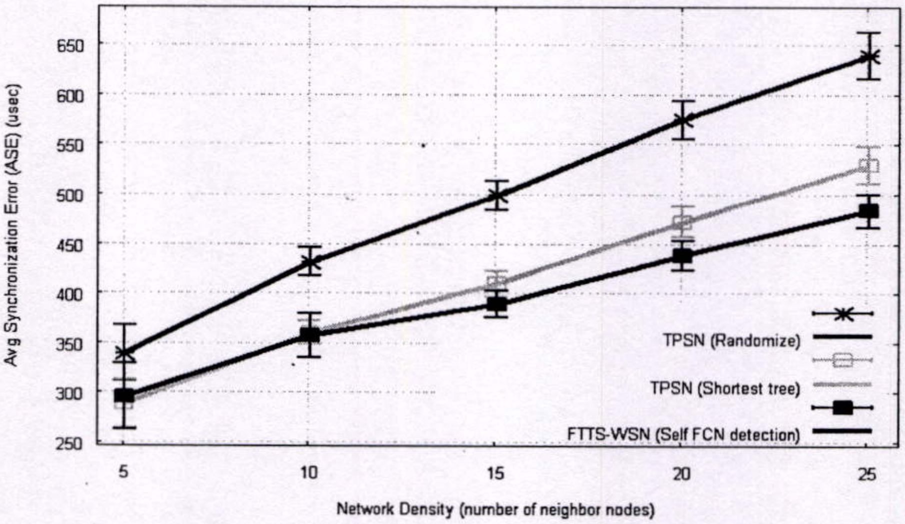
รูปที่ 4.14 ชั้นของโหนดที่มีค่าผิดพลาดจากการประสานเวลาสูงสุด

4.5.3 สมรรถนะในมิติความหนาแน่นของเครือข่าย

ระดับความหนาแน่นของเครือข่าย อันได้แก่ 5, 10, 15, 20 และ 25 เป็นต้น ซึ่งตัวเลขเหล่านี้บอกถึงจำนวนโหนดข้างเคียงโดยเฉลี่ยของโหนดใดๆ ในเครือข่าย (number of neighbor nodes) เมื่อค่าความหนาแน่นมีค่าน้อยหมายถึงเครือข่ายเบาบาง และค่ายิ่งสูงแสดงถึงเครือข่ายที่หนาแน่นมากขึ้นตามลำดับ

ในการทดลองที่เกี่ยวกับความหนาแน่นระดับต่างๆ นี้ เราได้ยกตัวอย่างกรณีโหนด 200 โหนด และปริมาณฟอลคล็อกโหนดที่มีค่าความคลาดเคลื่อน 6 เท่าจำนวน 10% ของจำนวนโหนดทั้งหมด แสดงดังรูปที่ 4.15 ค่าเอเอสอีที่ความหนาแน่นของเครือข่ายระดับต่างๆ

โดยปกติหากเครือข่ายหนาแน่นขึ้น ความยาวของโครงสร้างจะสั้นลงเสมอ และจากสมมุติฐานที่ว่า ค่าผิดพลาดจากการประสานเวลานั้นขึ้นอยู่กับความยาวของโครงสร้างที่ต้องประสานเวลากันเป็นทอดๆ และระยะเวลาทั้งหมดที่ใช้ในการประสานเวลา แต่ผลการทดลองนี้แสดงความหมายว่าเมื่อความหนาแน่นสูงขึ้น ค่าเอเอสอีก็สูงขึ้นด้วย ซึ่งสร้างความน่าแปลกใจเป็นอย่างยิ่ง เพราะยิ่งความหนาแน่นสูงขึ้น (โครงสร้างลำดับชั้นสั้นลง) ค่าเอเอสอีน่าจะลดลง สำหรับสาเหตุที่เป็นเช่นนั้นจะอธิบายในลำดับต่อไป



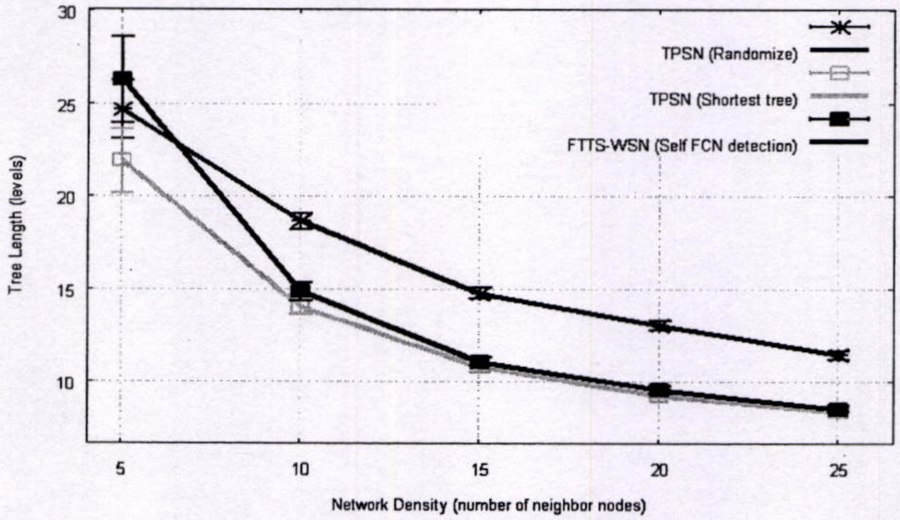
รูปที่ 4.15 ค่าเอเอสอีของการประสานเวลาที่ความหนาแน่นเครือข่ายระดับต่างๆ

ตารางที่ 4.4 (แสดงค่าสอดคล้องตามรูปที่ 4.15) ในสถานการณ์ค่าความหนาแน่นของเครือข่ายส่วนใหญ่นั้น โพรโทคอลที่นำเสนอเพิ่มสมรรถนะโดยการลดค่าเอเอสอีได้มากกว่าโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize) แต่มีเพียงช่วงเครือข่ายเบาบาง (จำนวนโหนดข้างเคียงโดยเฉลี่ย 5 โหนด) ที่โพรโทคอลที่นำเสนอมีผลที่แย่กว่าโพรโทคอล TPSN (Shortest tree) สาเหตุนั้นมาจากสองสาเหตุ ได้แก่ ข้อมูลเวลาระหว่างโหนดข้างเคียงมีจำนวนน้อยทำให้ความถูกต้องในการระบุฟอลคล็อกโหนดผิดพลาดไป และเมื่อการระบุฟอลคล็อกโหนดผิดพลาด โครงสร้างลำดับชั้นที่ถูกสร้างขึ้นจึงยาวกว่าปกติ (ดังรูปที่ 4.16)

รูปที่ 4.16 แสดงค่าเฉลี่ยความยาวของโครงสร้างลำดับชั้นของโพรโทคอลประสานเวลาในสถานการณ์ความหนาแน่นต่างๆ ดังจะเห็นว่า กรณีความหนาแน่นน้อยโพรโทคอลที่นำเสนอ ระบุจำนวนฟอลคล็อกโหนดมากผิดปกติ และมีความพยายามหลีกเลี่ยงฟอลคล็อกโหนดดังกล่าวทำให้โครงสร้างลำดับชั้นมีความยาวกว่าปกติ แต่เมื่อความหนาแน่นของเครือข่ายมากขึ้นการสร้างโครงสร้างที่สามารถหลีกเลี่ยงฟอลคล็อกโหนดและมีลักษณะสั้น ก็ให้ผลที่ใกล้เคียงกับโพรโทคอล TPSN (Shortest tree)

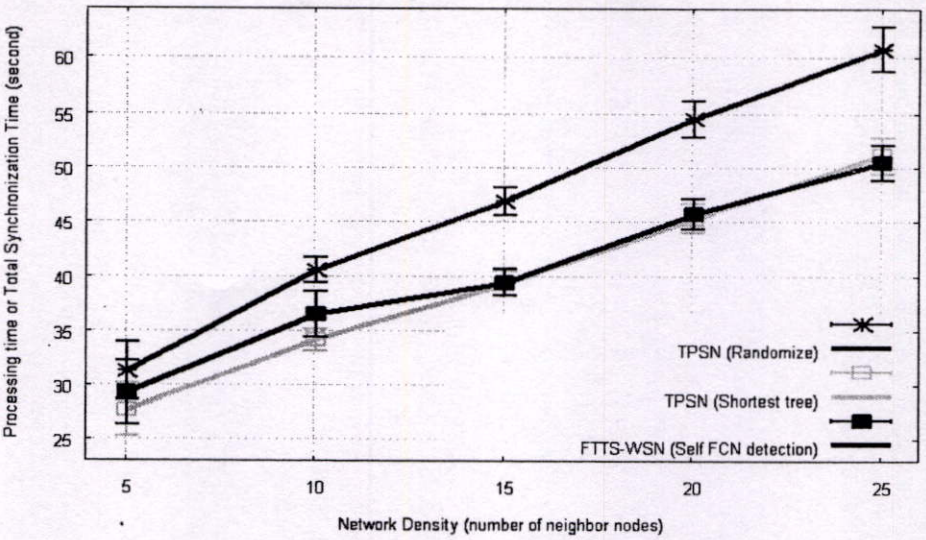
ตารางที่ 4.4 สมรรถนะค่าเอเอสอีของโพรโทคอลประสานเวลาที่ความหนาแน่นเครือข่ายระดับต่างๆ (หน่วยไมโครวินาที)

		Network Density (number of neighbor nodes)				
		5	10	15	20	25
Time Synchronization Protocols for WSNs.	1. TPSN (Randomize)	339.3	431.1	499.0	574.8	638.8
	2. TPSN (Shortest tree)	288.4	360.2	409.6	471.7	529.5
	3. FTTS-WSN	299.4	357.2	381.8	438.5	483.4
	เปรียบเทียบ (1)	(-11.8%)	(-17.1%)	(-23.5%)	(-23.7%)	(-24.3%)
	เปรียบเทียบ (2)	(+3.8%)	(-0.8%)	(-6.8%)	(-7.0%)	(-8.7%)



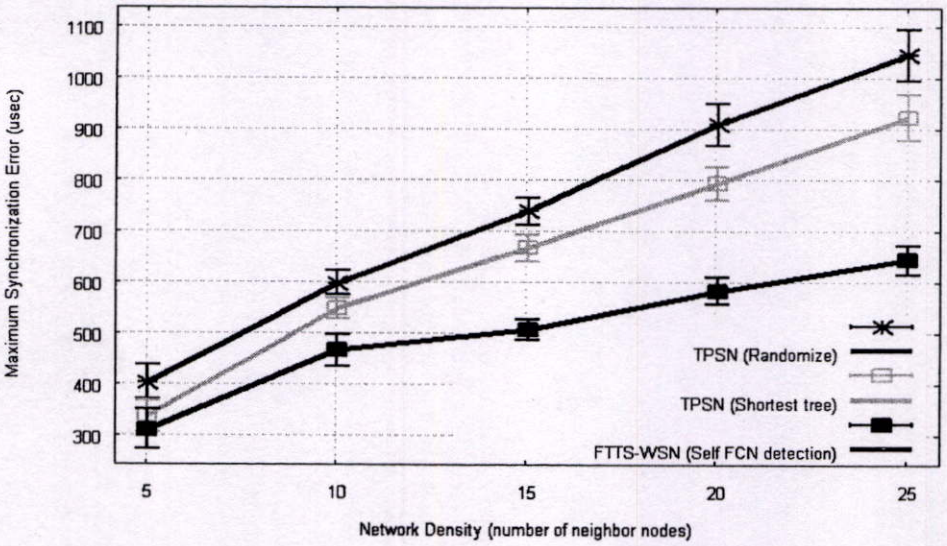
รูปที่ 4.16 ความยาวของโครงสร้างลำดับชั้นที่ความหนาแน่นเครือข่ายระดับต่างๆ

รูปที่ 4.17 แสดงระยะเวลาที่ใช้ประสานเวลาของทั้งระบบที่เพิ่มขึ้นในทุกโพรโทคอลประสานเวลา ดังจะเห็นว่ายิ่งเครือข่ายหนาแน่นมากขึ้น ระยะเวลาที่ต้องใช้เพื่อประสานเวลาทั้งระบบก็เพิ่มขึ้น ซึ่งปัจจัยเรื่องระยะเวลานี้สำคัญไม่แพ้ปัจจัยอื่น (ดังที่เคยวิเคราะห์ในหัวข้อ 2.3) ด้วยเหตุนี้จึงสามารถอธิบายว่าที่ความหนาแน่นมากขึ้นค่าสมรรถนะเอเอสอีของทุกโพรโทคอลทำไมจึงสูงขึ้น (ดังรูปที่ 4.15)



รูปที่ 4.17 ระยะเวลาที่ใช้ประสานเวลาของทั้งระบบที่ความหนาแน่นเครือข่ายระดับต่างๆ

รูปที่ 4.18 ค่าเวลาที่แตกต่างสูงสุดของโพรโทคอลที่นำเสนอ (FTTS-WSN) มีค่า SE_{MAX} ต่ำกว่า 7-30 และ 22-38% เมื่อเทียบกับโพรโทคอล TPSN (Shortest tree) และโพรโทคอล TPSN (Randomize) ตามลำดับ

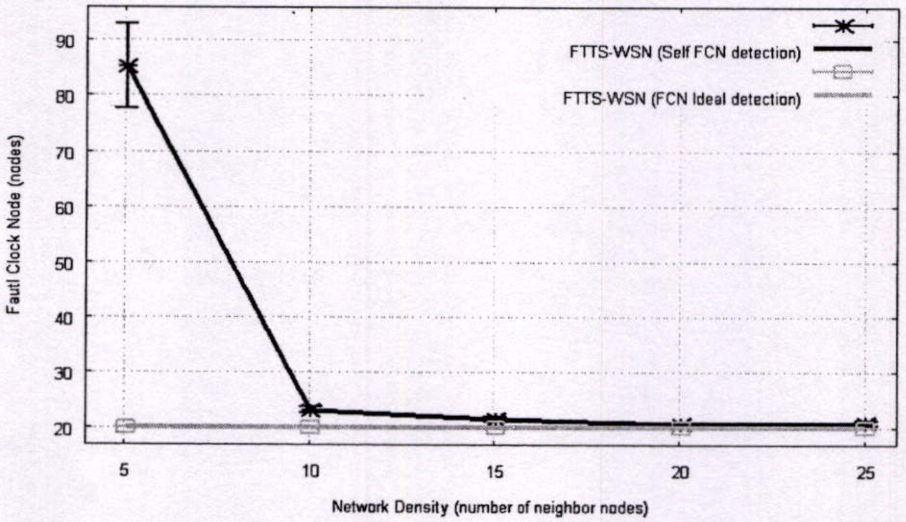


รูปที่ 4.18 ค่าผิดพลาดจากการประสานเวลาสูงสุดที่ความหนาแน่นเครือข่ายระดับต่างๆ

4.5.4 ความถูกต้องของการระบุฟอลคล็อกโหนดมีผลต่อสมรรถนะ

เมื่อสมรรถนะของโพรโทคอลที่นำเสนอ มีปัจจัยความแม่นยำการระบุฟอลคล็อกโหนดและหลีกเลี่ยงมาเกี่ยวข้อง ผลการทดลองต่อไปนี้จะแสดงความน่าเชื่อถือของโพรโทคอลจากวิธีการระบุและหลีกเลี่ยงฟอลคล็อกโหนด ซึ่งกรณีที่ยกมานี้สอดคล้องกับผลการทดลองที่ผ่านมา คือ โหนดเซนเซอร์ 200 โหนด แสดงในมิติความหนาแน่นเครือข่ายระดับต่างๆ

รูปที่ 4.19 แสดงจำนวนฟอลคล็อกโหนดที่สามารถระบุได้จากโพรโทคอลที่นำเสนอ หรือ FTTS-WSN (Self FCN Detection) เปรียบเทียบกับจำนวนฟอลคล็อกโหนดที่มีจริง หรือ FTTS-WSN (FCN Ideal detection) พบว่าโพรโทคอลที่นำเสนอมีปัญหาที่ความหนาแน่นเครือข่ายเบาบาง ดังกรณีความหนาแน่นที่มีโหนดข้างเคียงเฉลี่ย 5 โหนด การเข้าใจผิดว่ามีฟอลคล็อกโหนดมากกว่าที่กำหนดไว้ถึง 4 เท่า เนื่องจากวิธีการระบุฟอลคล็อกโหนดของโพรโทคอลที่นำเสนอ ใช้ค่าเฉลี่ยเวลาที่แตกต่างจากโหนดข้างเคียง โอกาสที่โหนดจะเข้าใจผิดว่าตนเองเป็นฟอลคล็อกโหนดเมื่อมีโหนดข้างเคียงน้อย เนื่องจากค่าเวลาที่แตกต่างสูงจากฟอลคล็อกโหนดเพียงโหนดเดียวสามารถโน้มน้าวค่าเฉลี่ยได้ แต่เมื่อความหนาแน่นของโหนดมากขึ้น ตั้งแต่โหนดข้างเคียงเฉลี่ย 10 โหนดขึ้นไป ค่าเฉลี่ยซึ่งได้จากโหนดจำนวนมากขึ้นทำให้ค่าจากฟอลคล็อกโหนดส่งผลกระทบจนเข้าใจผิดน้อยลง และเมื่อความหนาแน่นของเครือข่ายสูงขึ้น ความถูกต้องของโพรโทคอลที่นำเสนอก็สามารถระบุฟอลคล็อกโหนดได้ถูกต้องใกล้เคียงกับจำนวนจริงที่สมมุติสถานการณ์ไว้ คือ กรณีนี้สุ่มให้มีฟอลคล็อกโหนดร้อยละ 10 จาก 200 โหนด หรือ 20 โหนดนั่นเอง



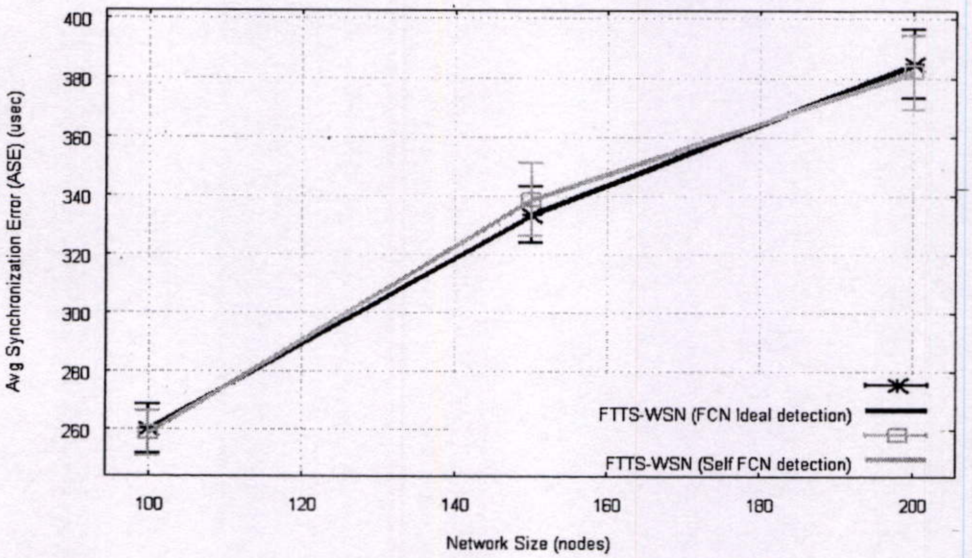
รูปที่ 4.19 จำนวนฟอลคล็อกโหนดที่ความหนาแน่นเครือข่ายระดับต่างๆ

เพื่อแสดงความน่าเชื่อถือของโพรโทคอลที่นำเสนอ จึงได้ทำการเปรียบเทียบกับกรณีที่สามารถระบุฟอลคล็อกโหนดได้อย่างแม่นยำ หรือ FTTS-WSN (FCN Ideal detection) คือ เมื่อทำการสุ่มโหนดที่เป็นฟอลคล็อกโหนดแล้ว โหนดเหล่านั้นจะประกาศ *BIL* ที่มีค่า 1 ให้โหนดข้างเคียงโดยอัตโนมัติ ทำให้โครงข่ายสามารถระบุฟอลคล็อกโหนดได้อย่างสมบูรณ์ แล้วเปรียบเทียบกับโพรโทคอลที่นำเสนอ FTTS-WSN (Self FCN Detection) แต่ละโหนดสามารถตรวจสอบฟอลคล็อกโหนดได้ด้วยตัวเอง (ซึ่งรายละเอียดได้อธิบายไว้ในหัวข้อ 3.1)

รูปที่ 4.20 แสดงสมรรถนะการระบุฟอลคล็อกโหนดด้วยตนเองของโพรโทคอลที่ได้นำเสนอนี้ สามารถทำงานได้ดีไม่แพ้โพรโทคอลการตรวจสอบแบบอุดมคติ โดยสถานการณ์การ

ทดลองที่กำหนดให้มีฟอลคล็อกโหนดร้อยละ 10 ของจำนวนโหนดต่างๆ และความหนาแน่นของเครือข่ายมีจำนวนโหนดข้างเคียงเฉลี่ย 15 โหนด ผลการทดลองนี้แสดงสมรรถนะค่าเอเอสอีที่ไม่แตกต่างกัน ณ ระดับนัยสำคัญที่ 99%

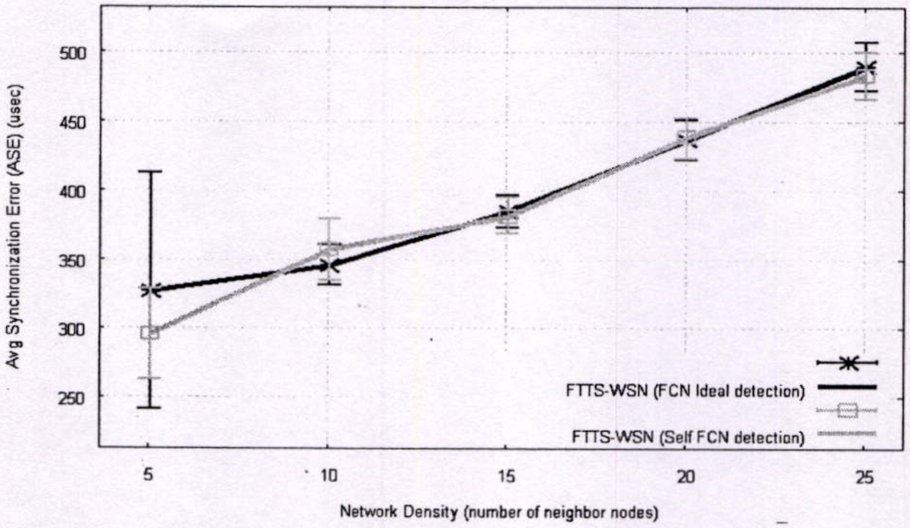
แม้ในบางครั้งเราคาดหวังว่า FTTS-WSN (Self FCN Detection) จะทำสมรรถนะได้ดีกว่าเล็กน้อย สาเหตุก็เนื่องมาจาก การตัดสินใจที่อยู่บนพื้นฐานข้อมูลจริง ซึ่งแตกต่างจาก FTTS-WSN (FCN Ideal detection) ที่ถูกกำหนดอย่างตายตัว เมื่อโหนดบางโหนดที่ถูกกำหนดให้เป็นโหนดนาฬิกาเที่ยงตรงปกติ แต่นาฬิกากลับแตกต่างค่อนข้างมากจนเกินเกณฑ์ที่กำหนดและถึงแม้ว่าจะไม่สูงเท่าฟอลคล็อกโหนด โหนดดังกล่าวสามารถประกาศตนเป็นฟอลคล็อกโหนดเพื่อให้โหนดข้างเคียงได้หลีกเลี่ยง ซึ่งจะเป็นผลดีต่อการประสานเวลา และน่าจะทำให้ผลสมรรถนะดีกว่าเล็กน้อย



รูปที่ 4.20 สมรรถนะค่าเอเอสอีจากวิธีการระบุฟอลคล็อกโหนดของโพรโทคอลที่นำเสนอ

รูปที่ 4.21 แสดงการเปรียบเทียบสมรรถนะค่าเอเอสอีที่ส่วนใหญ่ไม่แตกต่างระหว่างของโพรโทคอลที่นำเสนอ FTTS-WSN (Self FCN Detection) กับ FTTS-WSN (FCN Ideal detection) ที่ค่าความหนาแน่นเครือข่ายระดับต่างๆ โดยกำหนดสถานการณ์เซนเซอร์โหนด 200 โหนด จากผลก่อนหน้านี้ (รูปที่ 4.19) ทำให้ทราบแล้วว่าโพรโทคอลที่นำเสนอนี้สามารถระบุฟอลคล็อกโหนดได้ดีที่ความหนาแน่นเครือข่ายตั้งแต่โหนดข้างเคียงเฉลี่ย 10 โหนดขึ้นไป ซึ่งก็สอดคล้องกับรูปที่ 4.21 ที่ค่าสมรรถนะไม่แตกต่างกันตั้งแต่ช่วงความหนาแน่นเครือข่ายดังกล่าว ขณะที่ในเครือข่ายเบาบาง โพรโทคอลที่นำเสนอพยายามหลีกเลี่ยงโหนดส่วนใหญ่ที่คาดว่าจะเป็นฟอลคล็อกโหนด ตามปริมาณการระบุฟอลคล็อกโหนดที่เยอะกว่าปกติ (รูปที่ 4.19) ซึ่งการหลีกเลี่ยงฟอลคล็อกโหนดที่เข้าใจถูกและผิดนี้ยังคงทำให้ค่าเอเอสอีต่ำกว่าเล็กน้อย การทดลองทั้ง

สองโพรโทคอลมีตำแหน่งโหนดที่เหมือนกันแต่โครงสร้างลำดับชั้นจะแตกต่างกัน เนื่องจากจำนวนฟอลคล็อกโหนดที่ต่างกัน ซึ่งความแตกต่างนี้ต้องส่งผลต่อค่าสมรรถนะไม่มากก็น้อย และจะสังเกตุว่าผลการทดลองช่วงเครือข่ายเบาบางดังกล่าวมีค่าการกระจายตัวที่สูงมาก และหากพิจารณาตามค่าความเชื่อมั่น 99% อาจกล่าวได้ว่าสมรรถนะของทั้งสองวิธีไม่แตกต่างกันก็ได้

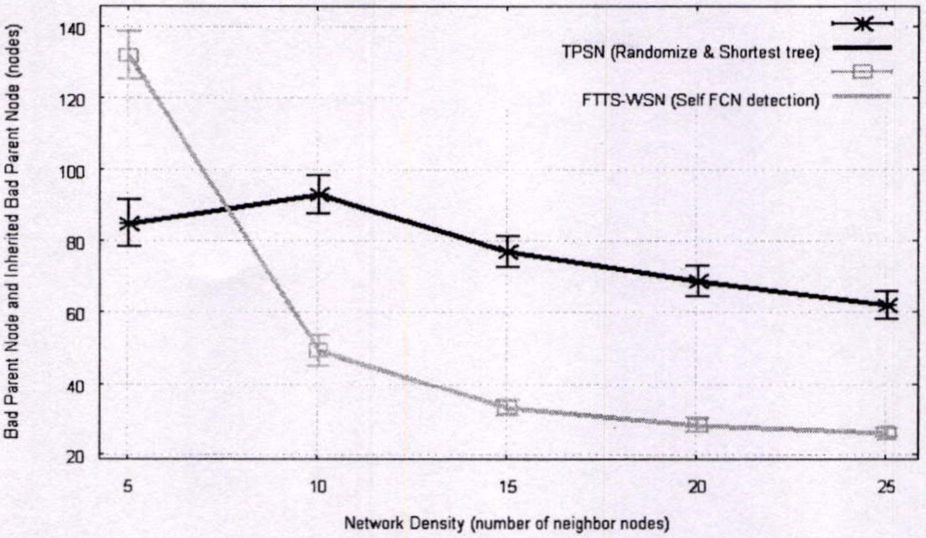


รูปที่ 4.21 สมรรถนะค่าเอเอสอีที่ความหนาแน่นเครือข่ายระดับต่างๆ

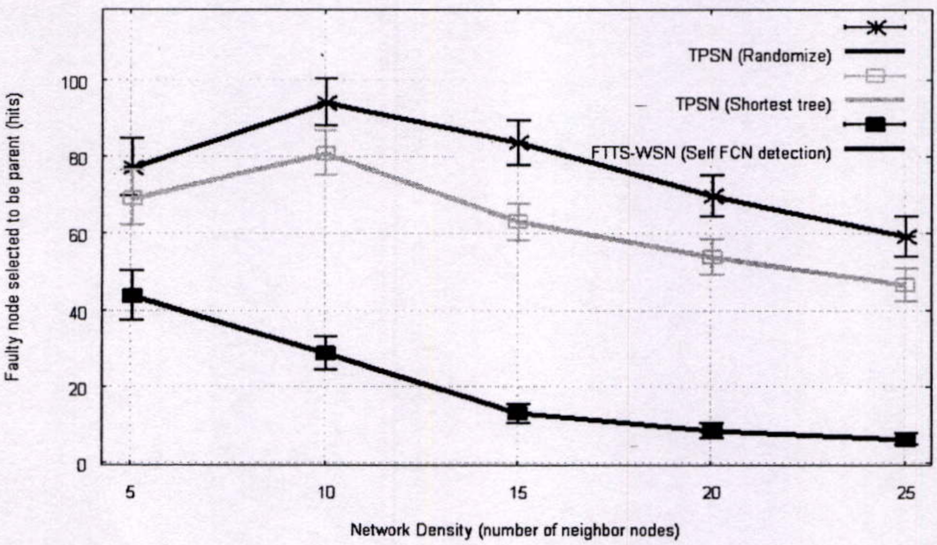
เมื่อมีฟอลคล็อกโหนดปะปนในเครือข่าย โหนดใดก็ตามที่ประสานเวลาต่อจากฟอลคล็อกโหนดแล้วเกิดการประสานเวลาต่อให้โหนดอื่นอีก หรือที่เรียกว่า Inherited Bad Parent Node (IBPN) โหนดเหล่านี้จะมีเวลาที่แตกต่างจากโหนดอ้างอิงมากด้วย ซึ่งไม่ดีต่อวัตถุประสงค์ของการประสานเวลา

รูปที่ 4.22 แสดงจำนวนฟอลคล็อกโหนดและโหนดที่ประสานเวลาต่อมาจากฟอลคล็อกโหนด โดยเปรียบเทียบระหว่างโพรโทคอลที่นำเสนอกับโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize) ซึ่งไม่รู้จักการระบุและหลีกเลี่ยงฟอลคล็อกโหนด มีโอกาสสูงที่จะเลือกโหนดดังกล่าวเป็นโหนดแม่ในโครงสร้างลำดับชั้น แล้วประสานเวลาด้วยจึงมีกราฟค่อนข้างสูง ขณะที่โพรโทคอลที่นำเสนอจะสามารถหลีกเลี่ยงและเกิดโหนดลักษณะที่เป็น IBPN ต่ำกว่า โดยยิ่งความหนาแน่นเครือข่ายสูง มีความแม่นยำในการระบุและหลีกเลี่ยงได้มากขึ้นปริมาณ IBPN จึงยิ่งน้อยลงด้วย แต่ที่ความหนาแน่นของเครือข่ายต่ำเกิดการเข้าใจผิดทำให้มีจำนวนฟอลคล็อกโหนดที่มาก อีกทั้งไม่สามารถหลีกเลี่ยงได้ง่ายเพราะจำกัดด้วยโหนดที่เป็นตัวเลือกน้อย

รูปที่ 4.23 แสดงจำนวนโหนดแม่ที่เป็น BPN หรือ IBPN แล้วถูกประสานเวลาด้วย ซึ่งโพรโทคอลที่นำเสนอมีการใช้ข้อมูล *BIL* ที่ประกาศให้โหนดข้างเคียงทราบถึงความเป็น BPN และ IBPN จึงสามารถหลีกเลี่ยงโหนดแม่ดังกล่าวได้ดีกว่า 37-86% และ 43-89% เมื่อเทียบกับโพรโทคอล TPSN (Shortest tree) และโพรโทคอล TPSN (Randomize) ตามลำดับ



รูปที่ 4.22 จำนวนฟอลคล็อกโหนดและโหนดที่ประสานเวลาต่อมาจากฟอลคล็อกโหนด

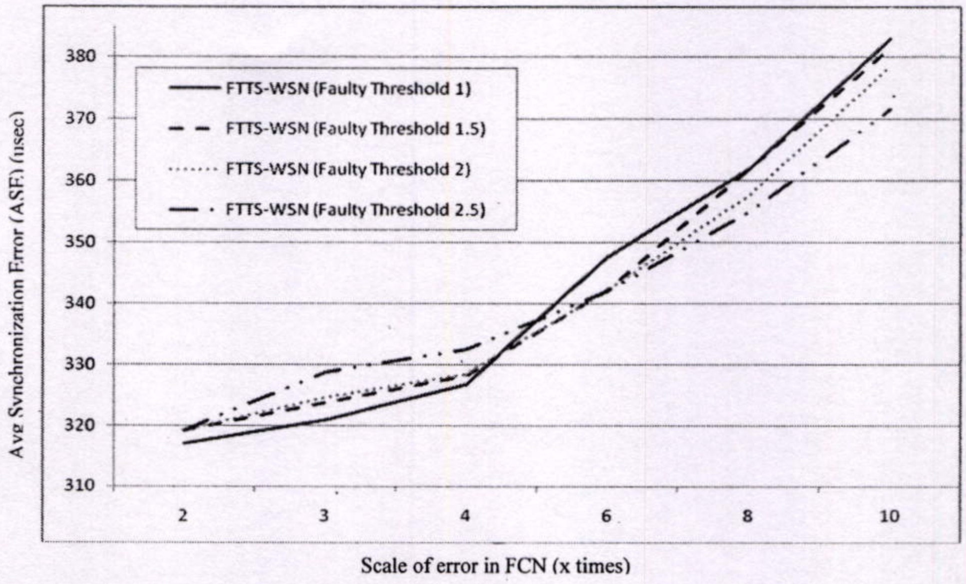


รูปที่ 4.23 จำนวน BPN หรือ IBPN ที่มีโหนดอื่นประสานเวลาตาม

4.5.5 วิเคราะห์ผลกระทบที่ส่งผลต่อสมรรถนะการระบุฟอลคล็อกโหนด

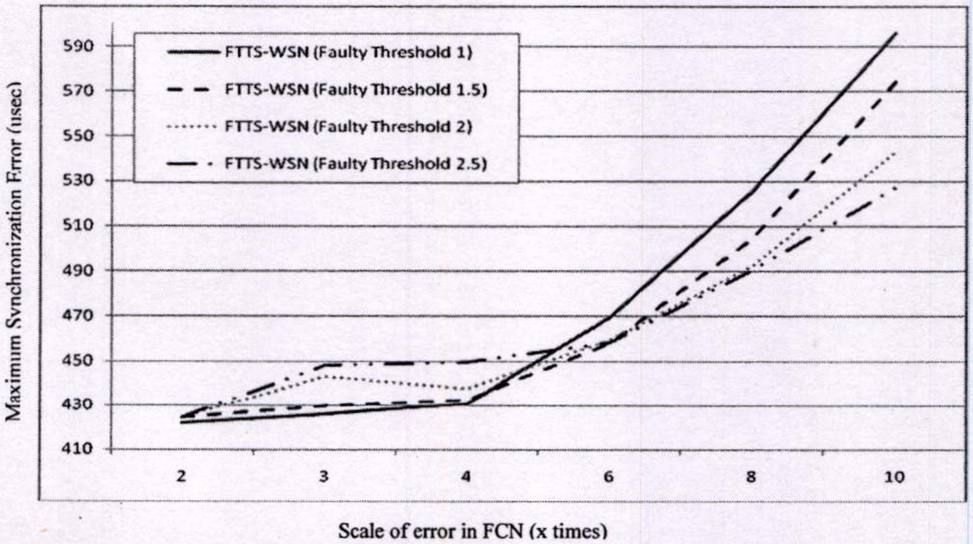
รูปที่ 4.24 แสดงผลกระทบเล็กน้อยต่อค่าเอเอสอีจากค่าตัวแปร *Faulty_Threshold* (จากสมการ 3.2) ซึ่งตัวแปรนี้มีผลต่อการกำหนดขอบเขตเพื่อระบุฟอลคล็อกโหนด สำหรับตัวอย่างนี้ยกมาจากค่าความหนาแน่น 15 และมีฟอลคล็อกโหนดร้อยละ 10 ของโหนดทั้งหมด 200 โหนด โดยทำให้เห็นว่าค่าตัวแปรที่ต่ำ สามารถมีสมรรถนะโพรโทคอลในการระบุฟอลคล็อกโหนดได้ดี ถ้าค่าความคลาดเคลื่อนของฟอลคล็อกโหนดอยู่ระหว่าง 2-4 เท่า แต่เมื่อฟอลคล็อกโหนดมีค่าความคลาดเคลื่อนสูงกว่าช่วงดังกล่าว สมรรถนะโพรโทคอลจะแย่ยิ่งขึ้น เนื่องจากการเข้าใจผิดของโหนดปกติที่ถูกเข้าใจว่าเป็นฟอลคล็อกโหนด เพราะค่าเฉลี่ยเวลาที่แตกต่างมีโอกาสสูงกว่าเกณฑ์ที่ตั้งไว้

ค่อนข้างต่ำได้ง่าย และผลก็แสดงในทางตรงกันข้ามว่าหากค่าตัวแปร *Faulty_Threshold* มีค่าสูง ดังนั้นจึงเป็นอีกปัญหาหนึ่งของโพรโทคอลที่นำเสนอ เนื่องจากเราไม่ทราบว่าฟอลคล็อกโหนดที่มีปะปนอยู่นั้นมีค่าความคลาดเคลื่อนสัมพัทธ์เป็นเช่นใด การจะกำหนดค่าตัวแปร *Faulty_Threshold* ให้สอดคล้องเพื่อประสิทธิภาพของโพรโทคอลสูงสุดจึงเป็นเรื่องยาก



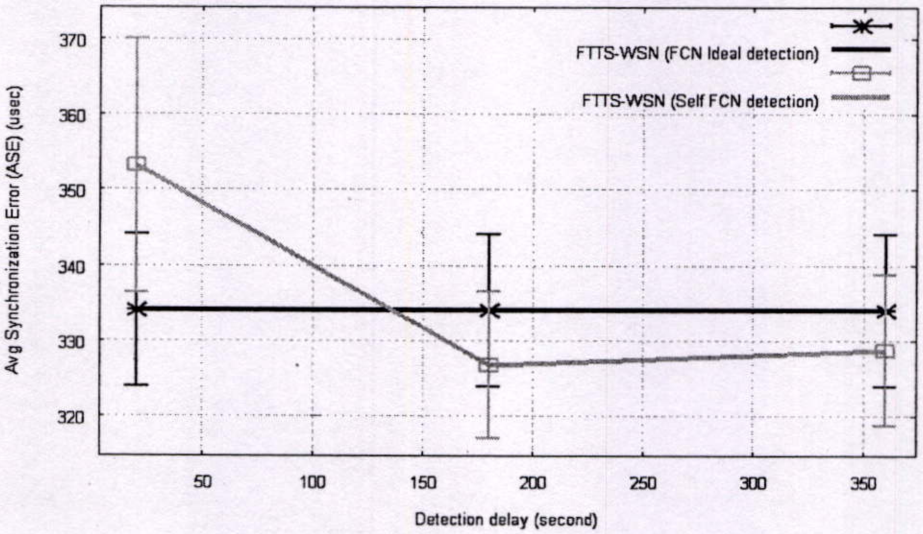
รูปที่ 4.24 ความเหมาะสมในการคำนวณค่า *Faulty_Drift* ที่มีผลต่อค่าเอเอสอี

รูปที่ 4.25 มีความหมายคล้ายกับผลก่อนหน้านี้ (รูปที่ 4.24) โดยแสดงความสัมพันธ์ของค่าตัวแปร *Faulty_Threshold* ที่มีผลต่อสมรรถนะการระบุฟอลคล็อกโหนดซึ่งมีความแตกต่างชัดเจนขึ้นหากพิจารณาในค่าสมรรถนะของค่าผิดพลาดสูงสุด



รูปที่ 4.25 ความเหมาะสมในการคำนวณค่า *Faulty_Drift* ที่มีผลต่อค่าผิดพลาดสูงสุด

รูปที่ 4.26 แสดงถึงระยะเวลา *Synced_Time* ที่รอคอยก่อนจะทำการแลกเปลี่ยนข้อมูลเวลากับโหนดข้างเคียงเพื่อคำนวณหาฟอลคล็อกโหนด (ตามข้อ 3.1) ซึ่งมีผลต่อการระบุฟอลคล็อกโหนดและมีผลต่อสมรรถนะการประสานเวลาตามโพรโทคอลที่นำเสนอต่อไป จากกราฟแสดงให้เห็นว่าระยะเวลา *Synced_Time* ที่นานระยะหนึ่งมีผลที่ดีกว่าระยะเวลา *Synced_Time* ที่สั้น สำหรับกราฟนี้ยกมาจากค่าความหนาแน่น 15 และมีฟอลคล็อกโหนดมีค่าความคลาดเคลื่อน 4 เท่าจำนวนเพียงร้อยละ 4 ของโหนดทั้งหมด 200 โหนด และมีค่า *Faulty_Threshold* เท่ากับ 1.0



รูปที่ 4.26 ระยะเวลา *Synced_Time* ส่งผลต่อสมรรถนะการระบุฟอลคล็อกโหนด

4.5.6 วิเคราะห์สมรรถนะในมิติของโครงสร้างลำดับชั้นและเวลา

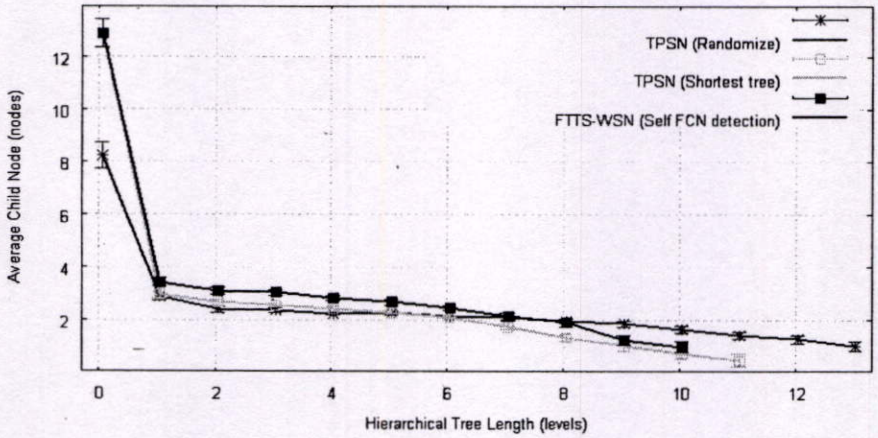
จากผลการทดลองที่ยกมานำเสนอ ส่วนใหญ่เป็นมุมมองของผลลัพธ์ที่ต้องการสื่อถึงสมรรถนะที่ดีกว่าและมีความแตกต่างที่ระดับนัยสำคัญ หรือต้องให้ได้เห็นภาพโดยรวม ซึ่งหากต้องการวิเคราะห์สมรรถนะว่ามีที่มาของแนวโน้มอย่างไร? ทำไมผลลัพธ์จึงเป็นดังเช่นที่เห็น? ผลการทดลองในส่วนนี้จะช่วยตอบคำถามเหล่านี้ได้

เพื่อจะพิจารณาแนวโน้มของสมรรถนะโดยละเอียด จึงเลือกผลการทดลองซึ่งประกอบด้วยโหนด 200 โหนด ความหนาแน่นเครือข่ายที่มีโหนดข้างเคียงเฉลี่ย 15 โหนด กำหนดให้มีฟอลคล็อกโหนดที่มีค่าความคลาดเคลื่อนสัมพัทธ์ 6 เท่าจากปกติในจำนวนร้อยละ 10 ของจำนวนโหนดทั้งหมด ระยะเวลาหลังจากประสานเวลา 180 วินาทีแล้วเก็บข้อมูลเวลาที่แตกต่างกับโหนดข้างเคียงเพื่อใช้ในการพิจารณาระบุฟอลคล็อกโหนด

รูปที่ 4.27 แสดงความหนาแน่นของโหนดลูกที่รอประสานเวลาด้วยในแต่ละโพรโทคอล ซึ่งจะพบว่าโหนดแม่ในชั้นแรก (ชั้น 0) หรือโหนดรูตมีโหนดลูกขอประสานเวลาด้วยประมาณ 13 โหนด ในโพรโทคอลที่นำเสนอ และโพรโทคอล TPSN-Shortest tree ขณะที่โพรโท

คอล TPSN-Randomize มีเพียงประมาณ 8 โหนดเท่านั้น นั่นแสดงว่าสองโพรโทคอลแรกจะสร้างโครงข่ายที่หนาแน่นในช่วงต้น เพื่อให้โหนดในชั้นถัดจากโหนดรูตมีค่าเวลาผิดพลาดน้อยที่สุด และช่วยลดความยาวโครงสร้างลำดับชั้นได้อีกด้วย

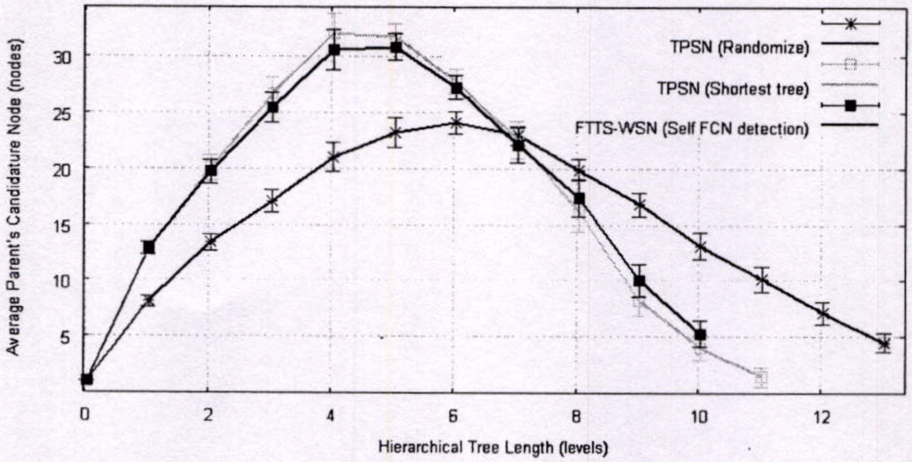
โพรโทคอลที่นำเสนอมีแนวโน้มที่จะมีความหนาแน่นของโหนดลูกที่สูงกว่าทุกโพรโทคอล เนื่องจากมีข้อจำกัดของเงื่อนไขในการเลือกโหนดแม่ที่มากกว่า ทั้งหลีกเลี่ยงฟอลคล็อกโหนดและลำดับชั้นที่สั้น (ถ้าเป็นไปได้) จึงทำให้มีตัวเลือกน้อย โหนดแม่ที่ตรงตามเงื่อนไขจึงมักมีโหนดลูกเลือกมาเกาะด้วยมากกว่าโพรโทคอลอื่น



รูปที่ 4.27 จำนวนโหนดลูกเฉลี่ยต่อโหนดแม่ในแต่ละโครงสร้างลำดับชั้น

รูปที่ 4.28 แสดงจำนวนโหนดที่เป็นตัวเลือกของแต่ละโครงสร้างลำดับชั้น ซึ่งนับจากจำนวนโหนดที่เป็นตัวเลือกทั้งหมดโดยไม่ซ้ำกัน ซึ่งได้ยินจากโหนดแม่ในชั้นก่อนหน้าประกาศมา ตัวอย่างเช่นในชั้นที่ 0 กราฟมีค่าเป็น 1 เพราะมีโหนดรูตเพียงโหนดเดียวที่เป็นโหนดที่เป็นตัวเลือกได้ ต่อมาโหนดแม่ที่อยู่ชั้นที่ 1 มีจำนวนเฉลี่ย 12, 12 และ 8 โหนด ดังผลก่อนหน้านี้ (รูปที่ 4.27) เมื่อโหนดเหล่านี้สร้างตนเองเข้าไปในโครงสร้างลำดับชั้นเสร็จก็จะประกาศข้อมูลโครงสร้างให้กับโหนดข้างเคียงต่อไป จึงนับจำนวนโหนดที่เป็นตัวเลือกได้สอดคล้องกับจำนวนโหนดแม่ในชั้นก่อนหน้า และจำนวนโหนดลูกซึ่งจะเป็นโหนดที่เป็นตัวเลือกในชั้นต่อๆ ไปจึงมีจำนวนสูงขึ้นตามลักษณะการแพร่กระจายออกไปของโครงสร้างต้นไม้

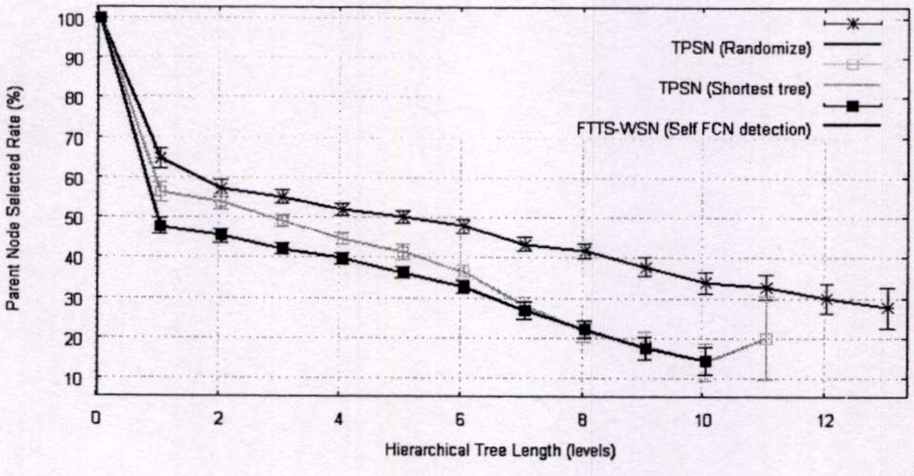
ความหนาแน่นที่สูงในช่วงต้นของโครงสร้างต้นไม้ในโพรโทคอลที่นำเสนอ และโพรโทคอล TPSN-Shortest tree ทำให้กราฟมีความชันมากในช่วงต้นและได้กราฟรูประฆังสูง โดยเฉพาะโพรโทคอล TPSN-Shortest tree ที่เน้นให้โครงสร้างสั้นที่สุดจึงมีความชันกราฟที่สูงกว่าเล็กน้อย ซึ่งแตกต่างอย่างชัดเจนกับ โพรโทคอล TPSN-Randomize ที่มีลักษณะการกระจายตัวที่สมดุลทั้งช่วงต้นและช่วงปลายจึงได้กราฟเป็นลักษณะระฆังคว่ำที่มีความชันกราฟน้อยกว่า



รูปที่ 4.28 จำนวนโหนดที่เป็นตัวเลือกในแต่ละโครงสร้างลำดับชั้น

รูปที่ 4.29 แสดงร้อยละของโหนดแม่ที่ถูกเลือกจากจำนวนโหนดที่เป็นตัวเลือกในแต่ละโครงสร้างลำดับชั้น ดังเช่น ชั้นที่ 0 มีโหนดรูตเป็นโหนดที่เป็นตัวเลือกเพียงโหนดเดียว จึงถูกเลือกให้เป็นโหนดแม่ในร้อยละ 100 ขณะที่ชั้นที่ 1 แต่ละโพรโทคอลมีจำนวนโหนดที่เป็นตัวเลือกเฉลี่ย 8, 12 และ 12 โหนดตามลำดับ ซึ่งถูกเลือกเป็นโหนดแม่ในร้อยละ 64, 56 และ 47 ตามลำดับ

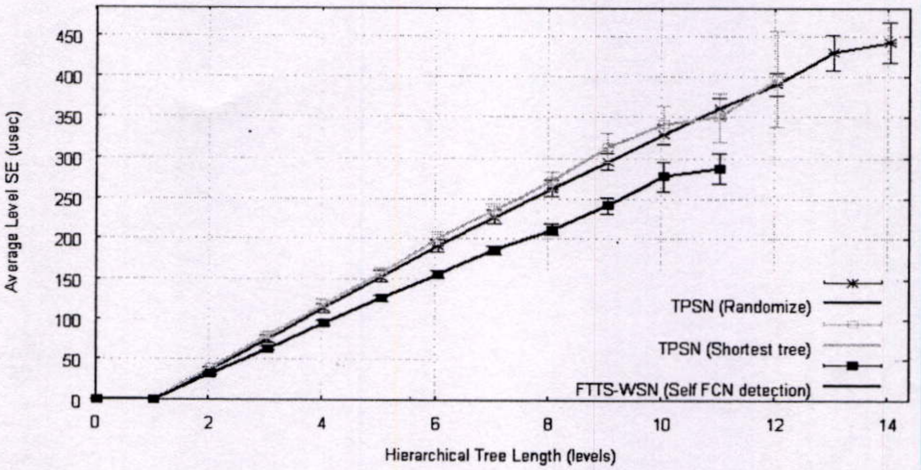
กราฟนี้แสดงให้เห็นว่าโพรโทคอลที่นำเสนอ แม้จะมีค่าเฉลี่ยจำนวนโหนดที่เป็นตัวเลือกใกล้เคียงกับโพรโทคอล TPSN-Shortest tree (ดังรูปที่ 4.28) แต่ก็มีการเลือกโหนดแม่ที่จำกัดกว่ามาก ขณะที่ โพรโทคอล TPSN-Randomize มีการเฉลี่ยเลือกโหนดแม่ที่สูงที่สุดหรือมีการกระจายของโหนดแม่มากที่สุดเมื่อเทียบกับโพรโทคอลอื่นๆ



รูปที่ 4.29 อัตราการกระจายโหนดแม่ในโหนดที่เป็นตัวเลือกในแต่ละโครงสร้างลำดับชั้น

รูปที่ 4.30 แสดงค่าเฉลี่ยเอสอีของโหนดในแต่ละโครงสร้างลำดับชั้น ที่มีค่าสูงขึ้นและกระจายตัวมากขึ้นตามโครงสร้างที่ยาวขึ้น และจะเห็นว่าการหลีกเลี่ยงฟอลคล็อกโหนด พร้อม

ทั้งพยายามให้โครงสร้างมีลำดับสั้นของโพรโทคอลที่นำเสนอ แสดงสมรรถนะที่ดีกว่าโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize) และสอดคล้องกับค่าเอเอสอีของโพรโทคอลที่มีสมรรถนะดีกว่า

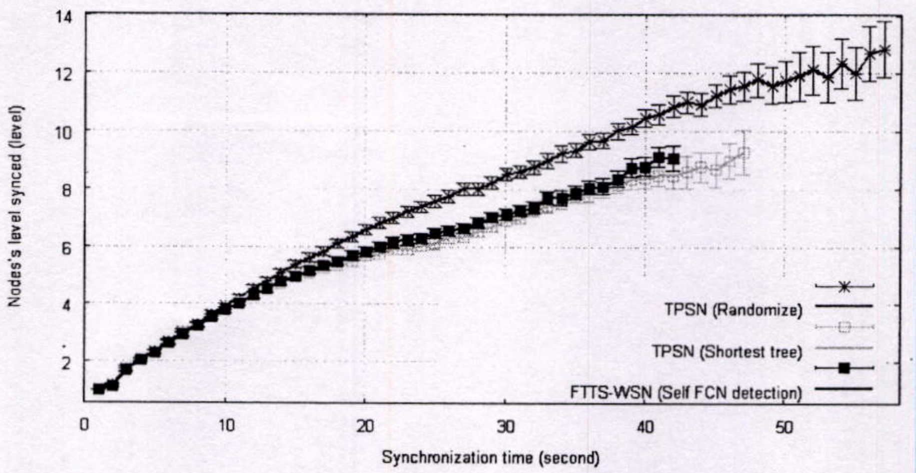


รูปที่ 4.30 ค่าเฉลี่ยเอสอีของโหนดในแต่ละโครงสร้างลำดับชั้น

จากผลรูปที่ 4.27 – 4.30 แสดงถึง หากความหนาแน่นของโครงข่ายลำดับชั้นมีค่าสูง โดยเฉพาะช่วงต้นของโครงข่าย ผลที่ตามมาจะมีทั้งผลดีและผลเสีย ผลดีคือช่วยลดความยาวโครงสร้างซึ่งจะทำให้ค่าผิดพลาดสะสมลดลง แต่ผลเสียจากความหนาแน่นของโครงข่ายก็คือการแก่งแย่งสื่อเพื่อใช้สื่อสารจะมีมากขึ้น และตามมาด้วยโอกาสที่ข้อมูลที่ถูกส่งไปจะชนกันแล้วส่งไม่สำเร็จ ต้องทำการพยายามส่งใหม่ซึ่งเปลืองเวลาและพลังงาน โชคดีที่ผลดีให้ผลลัพธ์ในแง่ความถูกต้องจากการประสานเวลาที่มากกว่าผลเสีย แต่การปรับปรุงโพรโทคอลเพื่อให้มีประสิทธิภาพและสอดคล้องกับการใช้สื่อร่วมกันยังคงควรได้รับการพัฒนาวิจัยต่อไป

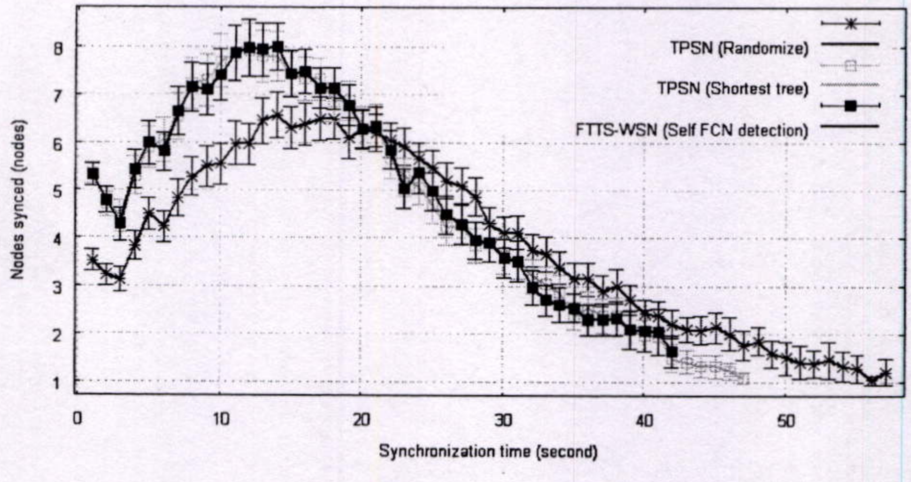
ผลการทดลองในมิติของเวลา จะช่วยให้ทราบและยืนยันสิ่งที่เกิดขึ้นในช่วงเวลาที่ผ่านไป ก่อนหน้านี้เราทราบว่าแต่ละโพรโทคอลมีการใช้ระยะเวลาในการประสานเวลาที่ไม่เท่ากัน (ดังรูปที่ 4.11, 4.17) แต่ยังไม่ทราบว่าช่วงเวลาดังกล่าวมีอะไรเกิดขึ้นบ้าง ในการทดลองนี้ทำการวัดผลทุกๆ วินาที ตลอดช่วงระยะการประสานเวลาของแต่ละโพรโทคอล (โดยยังคงใช้สถานการณ์การทดลองเดียวกันกับรูปที่ 4.23 – 4.26) แล้วนำผลมาเปรียบเทียบและแสดงดังรูปต่อไปนี้

รูปที่ 4.31 แสดงชั้นของโหนดที่ได้รับการประสานเวลาที่เวลาต่างๆ ผลที่ได้สอดคล้องกับโพรโทคอลที่นำเสนอ และโพรโทคอล TPSN-Shortest tree ที่มีโครงสร้างที่สั้นกว่าโพรโทคอล TPSN-Randomize ดังนั้น ณ เวลาเดียวกัน โหนดที่ประสานเวลาจึงอยู่ในชั้นที่ต่ำกว่า และโพรโทคอล TPSN-Shortest tree ก็มีค่าชั้นโดยเฉลี่ยที่ต่ำที่สุด



รูปที่ 4.31 ค่าชั้นโดยเฉลี่ยของโหนดที่ได้รับการประสานเวลา ณ เวลาต่างๆ

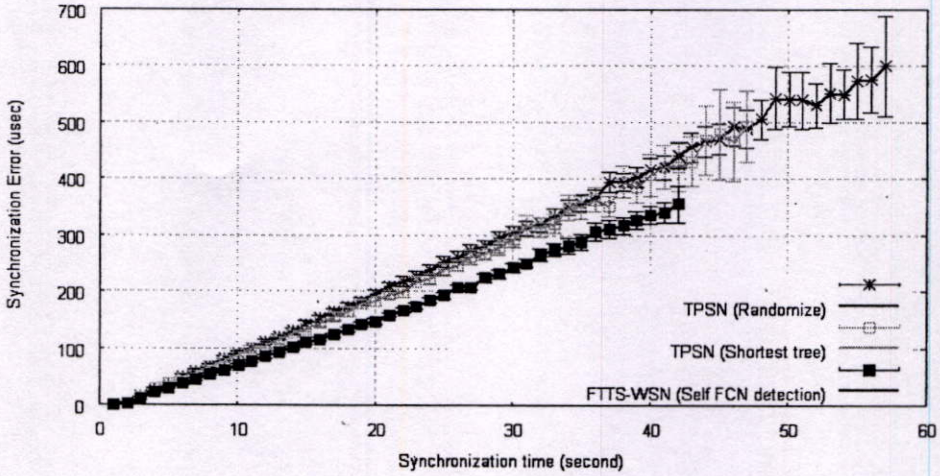
รูปที่ 4.32 แสดงจำนวนโหนดโดยเฉลี่ยที่ได้รับการประสานเวลา ณ เวลาต่างๆ จะเห็นว่าที่ช่วงต้นนั้น จำนวนโหนดที่ได้รับการประสานเวลามีปริมาณที่ลดลง สาเหตุก็เนื่องมาจากการแก่งแย่งกันที่มีปริมาณมากจากความหนาแน่นของโครงข่ายในช่วงแรก ทำให้ความสำเร็จในการประสานเวลามีปริมาณที่ต่ำลง แต่เมื่อการประสานเวลาเริ่มกระจายออกไปตามโครงสร้างลำดับชั้น นั่นคือรัศมีการสื่อสารไม่คาบเกี่ยวกัน ปริมาณความสำเร็จก็สูงขึ้น และเริ่มตกลงอีกครั้งเมื่อโครงข่ายเข้าสู่ช่วงปลาย ที่ความหนาแน่นต่ำจึงมีจำนวนโหนดที่ประสานเวลาสำเร็จลดลง



รูปที่ 4.32 ค่าจำนวนโหนดโดยเฉลี่ยที่ได้รับการประสานเวลา ณ เวลาต่างๆ

รูปที่ 4.33 แสดงค่าเฉลี่ยเอสอีของโหนดที่ได้รับการประสานเวลา ณ เวลาที่ต่างๆ โดยจะเห็นว่านอกจากค่าเฉลี่ยเอสอีจะสูงขึ้น สำหรับทุกโพรโทคอลประสานเวลาตามเวลาที่ผ่านไป พร้อมกันนี้ค่าเฉลี่ยเอสอียังมีลักษณะการกระจายตัวที่สูงขึ้นตามเวลาที่ผ่านไป ซึ่งการกระจายตัวนี้สื่อถึงเวลาของแต่ละโหนดแตกต่างกันมากขึ้นเมื่อเวลาผ่านไป แต่การหลีกเลี่ยงฟอลคล็อกโหนด

พร้อมทั้งพยายามให้โครงสร้างมีลำดับสั้นของโพรโทคอลที่นำเสนอ ยังคงแสดงสมรรถนะที่ดีกว่าโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize) ตลอดระยะเวลาการประสานเวลา



รูปที่ 4.33 ค่าเฉลี่ยเอสอีของโหนด ณ เวลาที่ผ่านไป

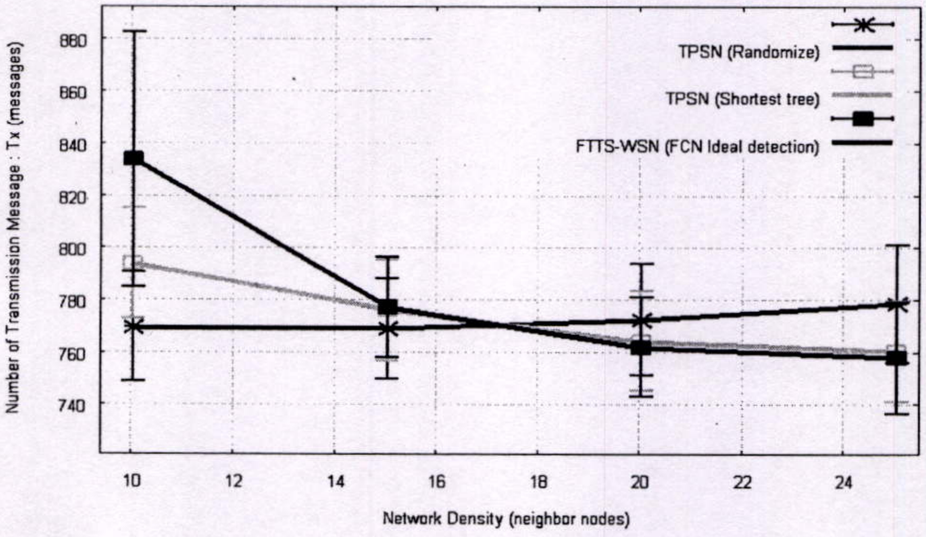
4.5.7 สมรรถนะปริมาณข้อมูลการประสานเวลา

เนื่องด้วยโพรโทคอลที่นำเสนอ (FTTS-WSN) มีการปรับปรุงที่สำคัญในส่วนการระบุและหลีกเลี่ยงฟอลคล็อกโหนด ขณะที่การประสานเวลายังคงเหมือนกับโพรโทคอลทีพีเอสเอ็น ซึ่งใช้การแลกเปลี่ยนข้อมูลเวลา (ทีดับบิวเอ็มอี) ตามโครงสร้างลำดับชั้นเป็นทอดๆ ดังนั้นปริมาณข้อมูลที่ถูกเพิ่มขึ้นมาในโพรโทคอลที่นำเสนอนี้ จึงมาจากการแลกเปลี่ยนข้อมูลเวลาเพื่อระบุฟอลคล็อกโหนดเพียงส่วนเดียว ซึ่งแสดงด้วยผลดังต่อไปนี้

สถานการณ์การทดลองนี้คือโหนดเซนเซอร์ 200 โหนด ความหนาแน่นเครือข่ายระดับต่างๆ กำหนดให้มีฟอลคล็อกโหนดที่มีค่าความคลาดเคลื่อนสัมพัทธ์ 6 เท่าจากปกติในจำนวนร้อยละ 10 ของจำนวนโหนดทั้งหมด ระยะเวลาหลังจากประสานเวลา 180 วินาทีแล้วเก็บข้อมูลเวลาที่แตกต่างกับโหนดข้างเคียงเพื่อใช้ในการพิจารณาระบุฟอลคล็อกโหนด

รูปที่ 4.34 ปริมาณข้อมูลทั้งหมดที่มีการส่งในระบบตั้งแต่สร้างโครงสร้างลำดับชั้นและประสานเวลา แต่กรณีนี้จะยังไม่นับปริมาณข้อมูลที่โหนดแลกเปลี่ยนข้อมูลเวลาเพื่อใช้ระบุฟอลคล็อกโหนด จะเห็นว่าตั้งแต่ความหนาแน่นของเครือข่ายที่โหนดข้างเคียงเฉลี่ย 10 โหนดเป็นต้นไป ข้อมูลที่ส่งในระบบมีจำนวนที่ใกล้เคียงกัน หรือไม่แตกต่างในทุกโพรโทคอล ขณะที่ความหนาแน่นของเครือข่ายเบาบาง โพรโทคอลทั้งสามมีจำนวนข้อมูลที่ส่งแตกต่างกัน ซึ่งมีสาเหตุมาจากการทำงานของอัลกอลิทึมของโพรโทคอล ที่พยายามเปลี่ยนแปลงโครงสร้างเพื่อหาโหนดแม่ที่ดีที่สุดตามเงื่อนไขของแต่ละโพรโทคอล แต่ด้วยโหนดที่เป็นตัวเลือกมีจำนวนน้อย จึงมีความพยายามส่งคำร้องขอซ้ำๆ หลายครั้ง

ความหนาแน่นที่มีโหนดข้างเคียงเฉลี่ย 10 โหนดขึ้นไป ทุกโพรโทคอลมีปริมาณการส่งข้อมูลที่ใกล้เคียงกัน โดยจำนวนข้อมูลดังกล่าวสามารถวิเคราะห์ได้ดังนี้ เมื่อ *N* คือจำนวนโหนดทั้งหมดในโครงสร้างลำดับชั้น หรือจำนวนโหนดทั้งหมดที่ได้รับการประสานเวลา ข้อมูลชนิด *LEVEL_DISCOVERY* (3.3.1) ที่เมื่อทุกโหนดปรับตนเองเข้าโครงสร้างลำดับชั้นเสร็จแล้วต้องส่งออกไปอย่างน้อย 1 ครั้ง นั่นคือ *N* x 1 ข้อมูล บวกกับข้อมูลชนิด *NODE_SYNC_REQ* (3.3.2) ที่ทุกโหนดในโครงสร้างลำดับชั้นจะส่งไปถามเวลาของโหนดแม่ของตนเพื่อประสานเวลา นั่นคือ *N* x 1 ข้อมูล บวกกับข้อมูลชนิด *NODE_SYNC_REPLY* (3.3.2) ที่โหนดแม่ต้องตอบกลับตามจำนวนคำขอจากโหนดลูกๆ ของตน นั่นคือ ถ้ามี *N* คำขอก็จะมี *N* x 1 ข้อมูลตอบกลับ บวกกับจำนวนข้อมูลที่ต้องส่งซ้ำ เนื่องจากข้อมูลที่ส่งไปนั้นเกิดการชนกันซึ่งแทนด้วยจำนวน *M* ดังนั้นปริมาณข้อมูลทั้งหมดจึงเท่ากับ (*N* x 1) + (*N* x 1) + (*N* x 1) + *M* เมื่อสมมุติว่า *N* เป็น 200 โหนด จึงมีปริมาณข้อมูลที่ส่งทั้งหมดมากกว่า 600 ข้อมูล

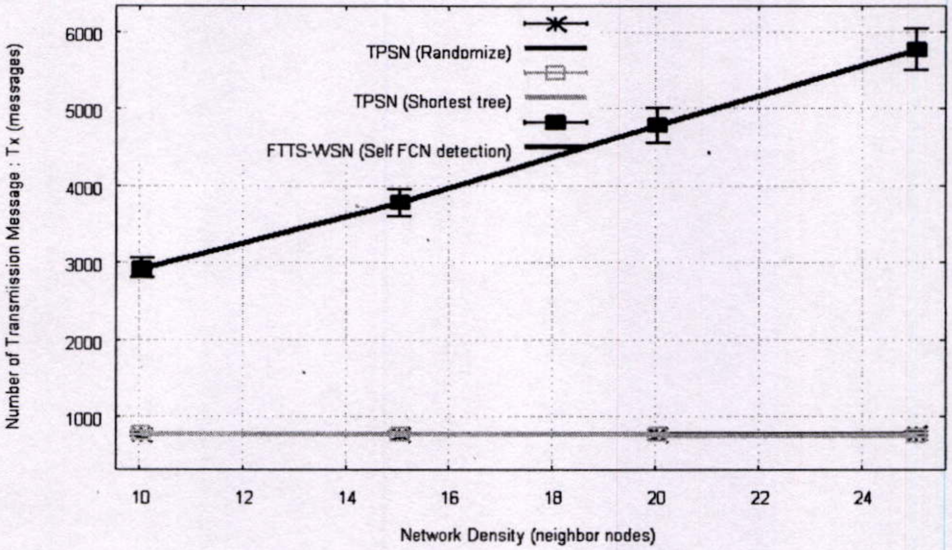


รูปที่ 4.34 ปริมาณข้อมูลที่ถูกส่ง หากไม่นับข้อมูลที่ใช้ระบุฟอลคล็อกโหนด

สำหรับการหลีกเลี่ยงฟอลคล็อกโหนดในโพรโทคอลที่นำเสนอนี้ เราได้ใช้การเพิ่มฟิลด์ข้อมูลลงในข้อมูลชนิด *LEVEL_DISCOVERY* ซึ่งไม่เป็นการเพิ่มข้อมูลชิ้นใหม่ในระบบ แต่ในการจะคำนวณหรือประเมินโหนดที่จะเป็นฟอลคล็อกโหนด โพรโทคอลที่นำเสนอนี้เป็นวิธีที่เพิ่มข้อมูลให้กับระบบโดยการแลกเปลี่ยนข้อมูลเวลาระหว่างโหนด เพราะได้ข้อมูลที่แม่นยำและทันสมัยที่สุด

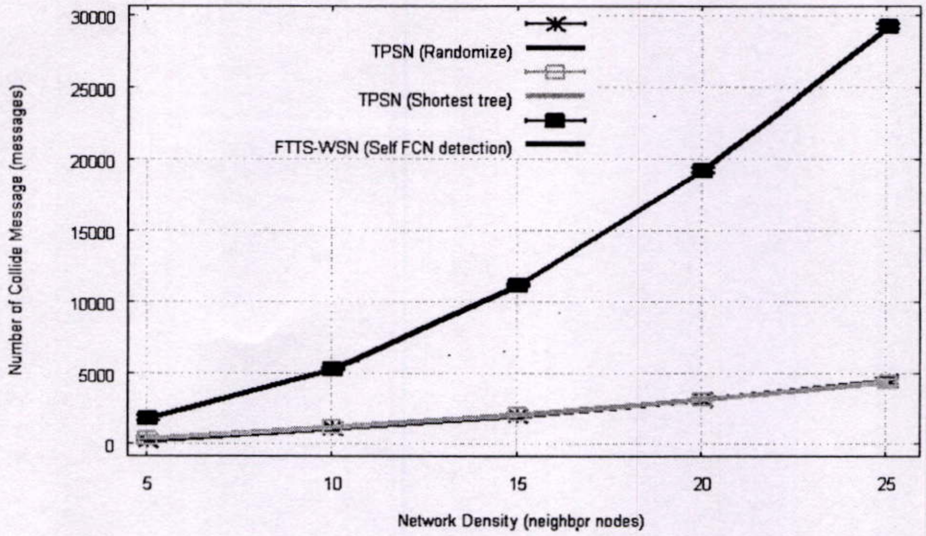
รูปที่ 4.35 ปริมาณข้อมูลทั้งหมดที่มีการส่งในระบบตั้งแต่ขั้นตอนการแลกเปลี่ยนข้อมูลเพื่อหาเวลาที่แตกต่างเฉลี่ยกับโหนดข้างเคียง ขั้นตอนสร้างโครงสร้างลำดับชั้นและขั้นตอนประสานเวลา ซึ่งกรณีนี้จะรวมปริมาณข้อมูลที่โหนดต้องแลกเปลี่ยนข้อมูลเวลาเพื่อใช้ระบุฟอลคล็อกโหนด โดยวิธีการที่เรานำเสนอในวิทยานิพนธ์นี้มีวัตถุประสงค์หลักเพื่อระบุฟอลคล็อก

โหนดเป็นหลัก จึงไม่ได้คำนึงถึงประสิทธิภาพอันเกิดจากปริมาณข้อมูลที่ต้องใช้จากการแลกเปลี่ยนข้อมูลเวลาเท่าที่ควร ผลที่แสดงออกมาจึงมีปริมาณข้อมูลที่ต้องส่งจำนวนสูงมาก และเมื่อเครือข่ายหนาแน่นมากขึ้นจะยิ่งมีปริมาณข้อมูลสูงมากขึ้นด้วย สาเหตุก็เนื่องมาจากการแลกเปลี่ยนข้อมูลเวลาระหว่างโหนดข้างเคียง นั้นเป็นแบบถามหนึ่งตอบหนึ่ง ดังนั้นปริมาณข้อมูลจึงขึ้นอยู่กับจำนวนข้อมูลที่ถามและจำนวนโหนดที่ต้องการแลกเปลี่ยนข้อมูลเวลา ที่เครือข่ายหนาแน่นมากจะต้องส่งคำขอทีละหนึ่งครั้งไปยังทุกโหนดข้างเคียง ปริมาณข้อมูลจึงมากขึ้นเมื่อเครือข่ายหนาแน่นสูงขึ้น



รูปที่ 4.35 ปริมาณข้อมูลที่ถูกส่ง เมื่อนับรวมข้อมูลที่ใช้ระบุฟอลคล็อกโหนด

รูปที่ 4.36 แสดงปริมาณข้อมูลที่มีการชนกัน ซึ่งสัมพันธ์จากจำนวนโหนดและปริมาณข้อมูลที่ส่งออกมา (ดังรูป 4.35) ที่ยิ่งส่งมากยิ่งมีโอกาสเกิดการชนกันของข้อมูลมาก และยิ่งการสื่อสารที่อาศัยสื่อร่วมกันนี้ ข้อมูลที่ชนกันจะเสียหายไม่สามารถใช้การได้เลย จึงเป็นสิ่งที่จะส่งผลต่อสมรรถนะและพลังงานที่สูญเสียไปอีกด้วย



รูปที่ 4.36 ปริมาณข้อมูลที่ชนกัน กรณีรวมข้อมูลที่ใช้ระบุฟอลคล็อกโหนด

สำหรับปัญหาเรื่องปริมาณข้อมูลนี้เราขอเสนอวิธีการอื่นๆ ทดแทน ดังเช่น การหาเวลาที่แตกต่างโดยการแจ้งเวลาของโหนดให้โหนดข้างเคียงแบบทางเดียว ทุกๆ ระยะเวลาหนึ่งแทนการแลกเปลี่ยนข้อมูลเวลา, การรับฟังข้อมูลเวลาเพียงอย่างเดียว (หรือที่เรียกว่า Passive) แทนวิธีการเพิ่มข้อมูลใหม่เข้าไป (หรือที่เรียกว่า Active), ตลอดจนการแลกเปลี่ยนข้อมูลเวลาแบบรวมส่งทีเดียว ซึ่งมองในมุมของจำนวนข้อมูลจะลดลงอย่างมาก แต่หากมองที่ขนาดของข้อมูลแล้วอาจจะแตกต่างกันไม่มากกับวิธีปัจจุบัน เป็นต้น ซึ่งวิธีการเหล่านี้ควรได้รับการพัฒนาและดำเนินการพิสูจน์สมรรถนะในโอกาสต่อไป และน่าจะช่วยรักษาสมรรถนะให้ดีเหมือนปัจจุบันหรือได้ดีกว่า อีกทั้งสามารถลดภาระให้กับระบบได้อีกด้วย

บทที่ 5
สรุปผลการวิจัย และข้อเสนอแนะ

5.1 สรุปผลการวิจัย

แอพพลิเคชันโครงข่ายเซนเซอร์ไร้สายที่ต้องการความถูกต้อง และความละเอียดของเวลาระดับนัยสำคัญหน่วยไมโครวินาที ดังเช่นระบบระบุตำแหน่งมือปืน [17] ซึ่งคำนวณตำแหน่งมือปืนและทิศทางของกระสุน จากเวลาที่เซนเซอร์หลายๆ โหนดได้ยินคลื่นเสียงจากลูกกระสุนที่วิ่งผ่านอากาศ เวลาของโหนดเซนเซอร์จึงต้องมีความถูกต้องมากกว่าปกติ

โพรโทคอลทีพีเอสเอ็นเป็นโพรโทคอลหนึ่งที่ได้รับการยอมรับเป็นอย่างมาก เนื่องจากเสนอให้ใช้การแลกเปลี่ยนข้อมูลเวลาที่เขียนเวลาในชั้นล่างของโพรโทคอลเครือข่ายเพื่อลดความผิดพลาดระหว่างการประสานเวลา อีกทั้งเสนอวิธีการประสานเวลาแบบต่อกันเป็นทอดๆ ตามโครงสร้างลำดับชั้น ซึ่งเหมาะสมกับเซนเซอร์ที่มีทรัพยากรจำกัด แต่ด้วยการประสานเวลาเป็นทอดๆ นี้จึงเกิดการสะสมของค่าผิดพลาดจากค่าคลาดเคลื่อนของนาฬิกาโหนด อีกทั้งคุณภาพและผลกระทบของสิ่งแวดล้อมทำให้นาฬิกาของโหนดเซนเซอร์ใดๆ เกิดความไม่เที่ยงตรงมากกว่าปกติหรือเราเรียกว่าฟอลคล็อกโหนด โพรโทคอลทีพีเอสเอ็นและตลอดจนงานวิจัยอื่นๆ ที่มีพื้นฐานจากโพรโทคอลทีพีเอสเอ็น ต่างได้รับผลกระทบจากการมีฟอลคล็อกโหนดทำให้สมรรถนะการประสานเวลาด้อยประสิทธิภาพลงไม่มากก็น้อย

โพรโทคอลเรานำเสนอวิธีระบุฟอลคล็อกโหนดด้วยวิธีการทางสถิติแบบง่าย และเสนอวิธีการหลีกเลี่ยงโดยให้ตัดสินจากค่าตัวแปร *BIL* ที่มีค่าต่ำที่สุดก่อนพิจารณาค่าลำดับชั้นต่อไป เราทำการทดลองโดยปรับปรุงวิธีการที่นำเสนอกับโพรโทคอลทีพีเอสเอ็น แสดงผลสมรรถนะค่าเอเอสอีที่ดีกว่าโพรโทคอลทีพีเอสเอ็นแบบเน้นโครงสร้างลำดับชั้นที่สั้น 1-2%, 4-6% และ 4-6% เมื่อระบบมีฟอลคล็อกโหนดที่มีอัตราค่าความคลาดเคลื่อนสัมพัทธ์ 3, 6, 10 เท่าตามลำดับและมีฟอลคล็อกโหนดปะปนอยู่ร้อยละ 5-30 ของโหนดทั้งหมดในระบบ และดีกว่า 18-19%, 20-23% และ 21-23% ตามลำดับเมื่อเปรียบเทียบกับโพรโทคอลทีพีเอสเอ็นดังเดิมที่เลือกโหนดแม่แบบสุ่ม

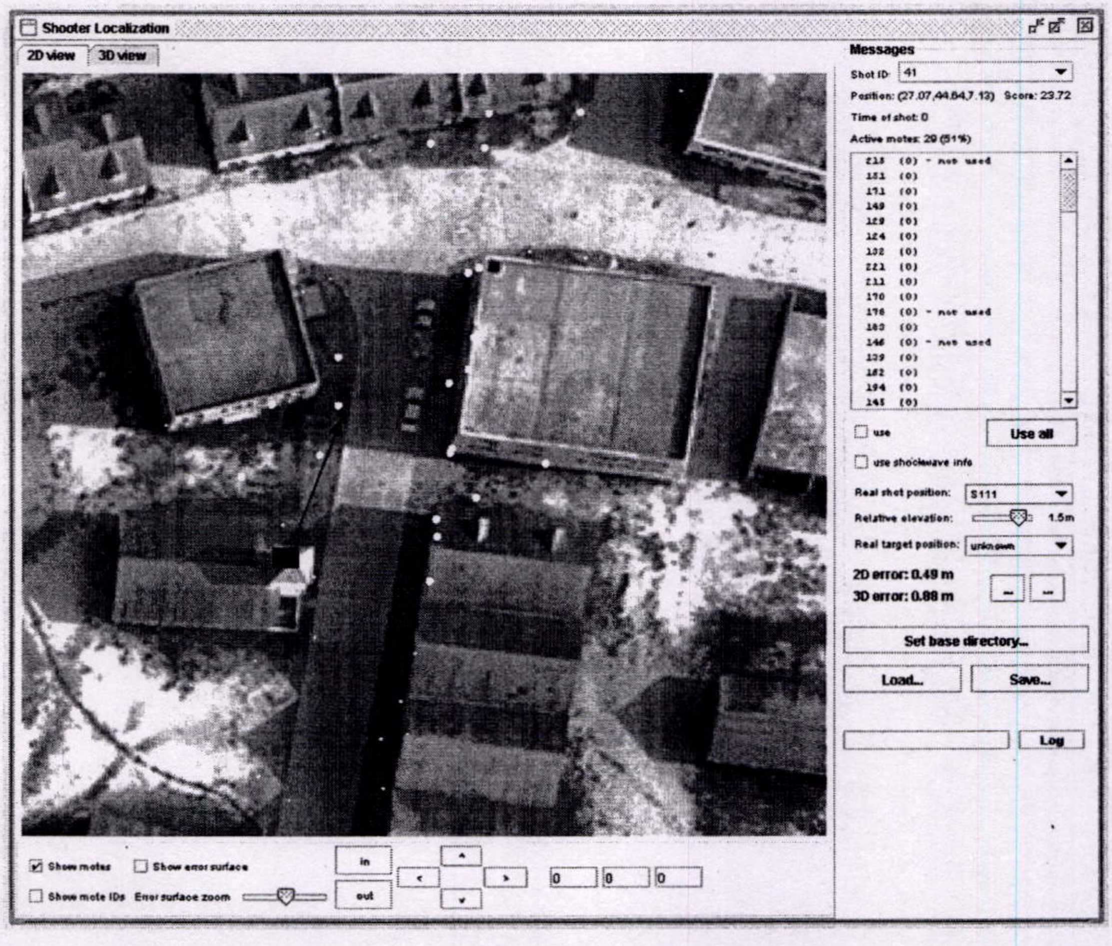
โพรโทคอลที่นำเสนอจะทำงานได้ดีในโครงข่ายที่มีความหนาแน่น (จำนวนโหนดข้างเคียงเฉลี่ย) ตั้งแต่ 10 โหนดเป็นต้นไป ซึ่งจะมีสมรรถนะการประสานเวลาดีเท่าเทียมหรือดีกว่าโพรโทคอลทีพีเอสเอ็นแบบเน้นโครงสร้างลำดับชั้นที่สั้น ในกรณีมีฟอลคล็อกโหนดปริมาณและอัตราค่าความคลาดเคลื่อนสัมพัทธ์ระดับต่างๆ

จากผลการทดลอง โพรโทคอลที่นำเสนอนี้ (FTTS-WSN) จึงมีส่วนสำคัญให้โพรโทคอลทีพีเอสเอ็นมีความทนทานเมื่อมีฟอลคล็อกโหนดปะปนในระบบ

5.2 ข้อเสนอแนะ และแนวทางที่ควรพัฒนาต่อ

5.2.1 ตัวอย่างแอพพลิเคชันที่มีความต้องการความแม่นยำของเวลาสูง

การระบุตำแหน่งมือปืนจากเซนเซอร์รับเสียงราคาถูก [17] ที่โปรยไว้ ดังรูปที่ 5.1 [18] มีโหนดเซนเซอร์ (แทนด้วยวงกลมสีเขียว) กระจายอยู่รอบๆ มือปืน (แทนด้วยสี่เหลี่ยมสีแดง) ที่มีวิถีกระสุนปืนแสดงด้วยเส้นลูกศรสีแดง สำหรับโพรโทคอลประสานเวลาที่แอพพลิเคชันนี้เลือกใช้เป็นการส่งเวลาต่อเป็นทอดๆ ผ่านกลุ่ม ซึ่งหัวหน้ากลุ่มจะถูกเลือกขึ้นมาด้วยวิธีอย่างง่าย โหนดรูตเขียนเวลาที่จะส่งที่ Radio layer ก่อนส่งไปยังหัวหน้ากลุ่มต่างๆ หัวหน้ากลุ่มแก้ไขเวลาตามเวลาที่ได้รับแล้วส่งต่อให้สมาชิกภายในกลุ่มต่อไป และให้ทำการประสานเวลาทุกๆ 1 นาที เวลาที่ได้รับ 10 ครั้งจะถูกใช้คำนวณหาค่าความถดถอย (Linear regression) เพื่อใช้ปรับเวลาไปพร้อมๆ กันอีกวิธีหนึ่ง พบว่าค่าผิดพลาดโดยเฉลี่ยในแต่ละชั้นมีค่า ±60 ไมโครวินาที ซึ่งค่าผิดพลาดเฉลี่ยจากการประสานเวลาทั้งระบบอยู่ในระดับที่งานนี้ยอมรับได้คือต่ำกว่า 1 มิลลิวินาที ซึ่งผลการทดลองสามารถรวบรวมและคำนวณข้อมูลที่ส่วนกลางแล้วระบุตำแหน่งมือปืนได้ภายใน 2 วินาที โดยมีค่าผิดพลาด 0.64 เมตรใน 2 มิติ และ 1.5 เมตรใน 3 มิติ



รูปที่ 5.1 ตำแหน่งของโหนดเซนเซอร์และมือปืน

5.2.2 วิธีการระบุฟอลคล็อกโหนดวิธีอื่น

เทคนิคการระบุฟอลคล็อกโหนดในโพรโทคอลที่นำเสนอ มีข้อจำกัดจากวิธีการคำนวณค่าเฉลี่ย ซึ่งไม่สามารถกำจัดผลกระทบจากฟอลคล็อกโหนดให้หมดไปได้ อีกทั้งยังต้องมีการตั้งค่าขอบเขต (Threshold) เพื่อเป็นเกณฑ์พิจารณาฟอลคล็อกโหนดซึ่งไม่สามารถจะยืนยันว่าครอบคลุมฟอลคล็อกโหนดได้ในทุกกรณี

5.2.2.1 Grubbs Test

การกำจัดข้อมูลที่อยู่นอกกลุ่มหรือ Grubbs test [19] เป็นเทคนิคหนึ่งที่ใช้หาว่าข้อมูลใดในกลุ่มข้อมูลที่ควรกำจัดหรือไม่ต้องสนใจ เพื่อให้ค่าเฉลี่ยและค่าความคลาดเคลื่อนดีขึ้น ครั้งหนึ่งเราเคยประยุกต์ใช้เทคนิคนี้เพื่อระบุฟอลคล็อกโหนด โดยมีสมมุติฐานว่าถ้าเวลาของฟอลคล็อกโหนดแตกต่างจากเวลาโหนดปกติมาก Grubbs test จะสามารถช่วยตัดค่าข้อมูลของฟอลคล็อกโหนดออกมา ซึ่งจะเป็นการระบุฟอลคล็อกโหนดไปในตัว แต่จากการทดลองที่กลุ่มข้อมูลมีค่าฟอลคล็อกโหนดปะปนอยู่นั้น การคำนวณ G_{calc} ดังสมการ (5.1) นั้นมักไม่สามารถแยกแยะ $X_{outlier}$ ที่เป็นฟอลคล็อกโหนดได้ เนื่องมาจากค่าเฉลี่ยจากการคำนวณรอบแรกถูกกระทบจากค่าเวลาฟอลคล็อกโหนด ผลที่ตามมาคือบางครั้งไม่มีการตัดข้อมูลเวลาของฟอลคล็อกโหนดออก และยิ่งไปกว่านั้นพบว่ามีตัดข้อมูลเวลาของโหนดปกติออกแทน เมื่อเป็นเช่นนั้นเราจึงไม่ได้นำ Grubbs test มาใช้ต่อ

$$G_{calc} = \frac{\left| x_{outlier} - \overline{x} \right|}{s} \tag{5.1}$$

แต่กระนั้น อาจมีการลองใช้ Grubbs test เพื่อกรองชุดข้อมูลแม้จะไม่สามารถตัดค่าข้อมูลของฟอลคล็อกโหนดได้ ในอัลกอลิทึมการระบุฟอลคล็อกโหนดที่จะนำเสนอในโอกาสต่อไป

5.2.2.2 Voting Algorithm

ประสิทธิภาพของอัลกอลิทึมการออกเสียง (voting algorithm) อยู่ที่ความสามารถจัดกลุ่มของชุดข้อมูลออกมาได้ชัดเจนเพียงใด ซึ่งหลายงานที่ใช้อัลกอลิทึมการออกเสียงในการตัดสินใจเลือกสิ่งที่ดีที่สุดหรือกำจัดสิ่งที่แย่ที่สุด โดยความแตกต่างและความสำเร็จของอัลกอลิทึมการออกเสียงส่วนใหญ่ขึ้นอยู่กับการให้คะแนนและเงื่อนไขที่ใช้ตัดสินใจ บางงานอาศัยเพียงเงื่อนไขเดียว และบางงานถึงกับผสมเงื่อนไขร่วมกัน ซึ่งมีตัวอย่างดังเช่น

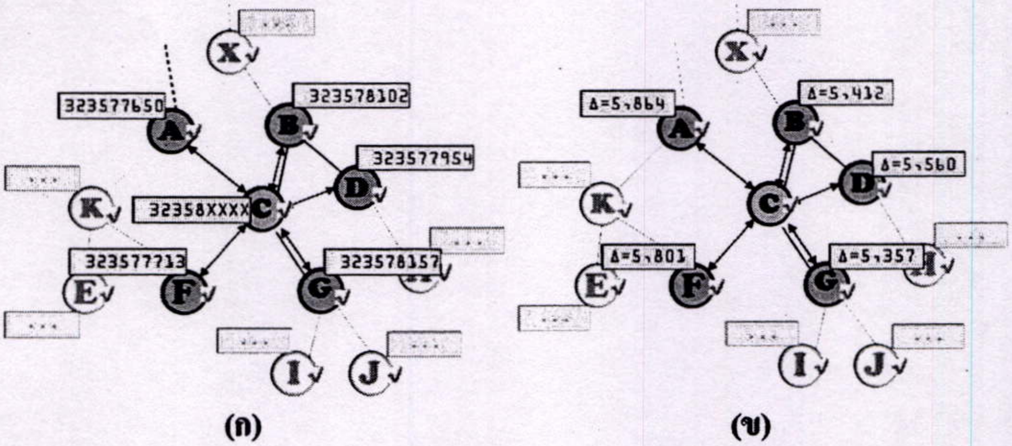
Leung และ Parker [20] กล่าวไว้ว่าแม้การโหวตคือการออกเสียงและการยอมรับเสียงส่วนใหญ่ แต่บางครั้งผู้ชนะก็ไม่ใช่ผู้ที่ได้รับเสียงโหวตสูงสุดเสมอไป เพราะขึ้นอยู่กับว่าเราต้องการอะไรจากผลการออกเสียง ถ้าผู้ชนะคือผู้ที่ได้รับเสียงส่วนใหญ่เรียกว่า The plurality

ถ้าผู้ชนะคือผู้ที่ได้รับเสียงต่ำสุดเรียกว่า The anti-plurality บางครั้งการหาผู้ชนะ (หรือเงื่อนไข) จะผ่านวิธีการหลายรอบโดยแต่ละรอบจะตัดข้อมูลที่ต่ำสุดในกลุ่มสูงสุดออก แล้วทำเช่นนี้อีกหลายรอบจนได้เงื่อนไขผู้ชนะคือกลุ่มที่เหลืออยู่ (เป็นเหมือนการลดอำนาจผู้ที่ได้รับเสียงมากลงมาให้เท่าเทียมกับผู้สมัครคนอื่น) เรียกว่า The plurality-elimination ในกรณีที่ผู้ (หรือเงื่อนไข) ได้รับเสียงสนับสนุนมีหลายคน ผลรวมคะแนนแต่ละคนจะถูกเรียงลำดับตั้งแต่ 0..n คนที่ได้ลำดับที่ n จะเป็นผู้ชนะ (ซึ่งก็ทำให้ดูเหมือนไม่สนใจคะแนนรวมที่แตกต่างในแต่ละคน แต่ผู้ชนะก็คือผู้รับคะแนนเสียงสนับสนุนสูงสุดนั่นเอง) เรียกว่า The Borda count ในการแข่งขันแบบจับคู่พบกันหมด ผู้ชนะได้ 1 คะแนน เสมอแบ่งคะแนนกันคนละครึ่ง (0.5 คะแนน) ส่วนผู้แพ้ไม่ได้คะแนน แล้วรวมคะแนนใครได้สูงสุดผู้นั้นชนะ เรียกว่า The pairwise comparison

ในการประยุกต์ใช้อัลกอลิทึมการออกเสียงเพื่อให้แต่ละโหนดตรวจสอบตนเองว่าเป็นฟอลคล็อกโหนดนั้นมีความเป็นไปได้ โดยใช้วิธีที่เรียกว่า The plurality เมื่อพิจารณาว่าค่าสัมบูรณ์เวลาที่โหนดตนแตกต่างกับโหนดข้างเคียงนั้นสูงกว่าค่าคลาดเคลื่อนสัมพัทธ์ปกติ (ซึ่งไม่ต้องสนใจค่าคลาดเคลื่อนสำหรับฟอลคล็อกโหนดที่ยากจะหาค่าที่เหมาะสม) จะให้คะแนนเป็น 1 แต่ถ้าต่ำกว่าจะให้คะแนน -1 แล้วเมื่อรวมคะแนนหากเป็นบวก จะหมายถึงโหนดดังกล่าวเป็นฟอลคล็อกโหนด แต่ถ้ามีค่าเป็นลบหรือศูนย์โหนดดังกล่าวเป็นโหนดปกติ

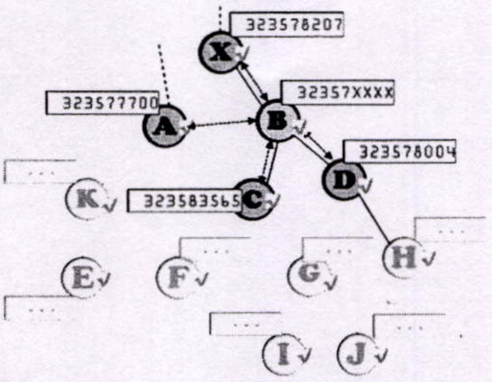
รูป 5.2 (ก) เมื่อเวลาของแต่ละโหนดหลังผ่านการประสานเวลามา 180 วินาที รูป 5.2 (ข) เวลาที่คาดว่าแตกต่างระหว่างโหนดข้างเคียงกับโหนด C และค่าความคลาดเคลื่อนของโหนดปกติสำหรับเวลา 180 วินาที คำนวณได้ดังสมการ (5.2) มีค่าเท่ากับ 990 ไมโครวินาที ผลรวมของคะแนนของทุกโหนดจึงเป็น +5 นั่นแสดงว่าโหนด C คือฟอลคล็อกโหนด

$$\textit{Normal Clock Drift (NCD)} = \textit{Drift_Rate} \times \textit{Synced_Time} \quad (5.2)$$



รูปที่ 5.2 ค่าเวลาที่โหนด C แตกต่างจากโหนดข้างเคียง

เช่นเดียวกับ รูป 5.3 (ก) เมื่อเวลาของแต่ละโหนดหลังผ่านการประสานเวลามา 180 วินาที รูป 5.2 (ข) เวลาที่คาดว่าแตกต่างระหว่างโหนดข้างเคียงกับโหนด B และค่าความคลาดเคลื่อนของโหนดปกติสำหรับเวลา 180 วินาที คำนวณได้ดังสมการ (5.2) มีค่าเท่ากับ 990 ไมโครวินาที ผลรวมของคะแนนของทุกโหนดจึงเป็น -2 นั่นแสดงว่าโหนด B คือโหนดปกติ



รูปที่ 5.3 ค่าเวลาที่โหนด B แตกต่างจากโหนดข้างเคียง

5.2.3 วิธีการอาศัยการสื่อสารที่มีอยู่เพื่อแลกเปลี่ยนข้อมูลเวลา

เนื่องจากวิธีการแลกเปลี่ยนข้อมูลเวลาระหว่างโหนดข้างเคียงทำให้เกิดการสื่อสารจำนวนมาก ซึ่งควรได้รับการปรับปรุงให้ลดจำนวนลง หรือลดให้หมดไป

การศึกษา Piggy Back ในเรื่องการสื่อสารระดับชั้นเครือข่าย (Network layer) ดังเช่น AODV [21] และ DSDV [22] ซึ่งเป็น Routing Protocol ที่มีศึกษาและทดลองนำมาใช้ในระบบเซนเซอร์ไร้สาย [23] พบว่า DSDV (เป็น Routing Protocol ชนิด Table-driven หรือ Proactive) ซึ่งมีการเตรียมเส้นทางตลอดเวลามีความเหมาะสมต่อการผนวกข้อมูลเวลาเข้าไปในแพ็กเก็ตมากกว่า AODV ซึ่งจะค้นหาและดูแลเส้นทางให้ใช้งานได้ก็ต่อเมื่อมีความต้องการ เนื่องจากโพรโทคอลที่นำเสนอนี้ต้องการข้อมูลเวลาโดยสม่ำเสมอ จึงไม่สามารถประยุกต์ได้กับ Routing Protocol ชนิด On-demand หรือ Reactive

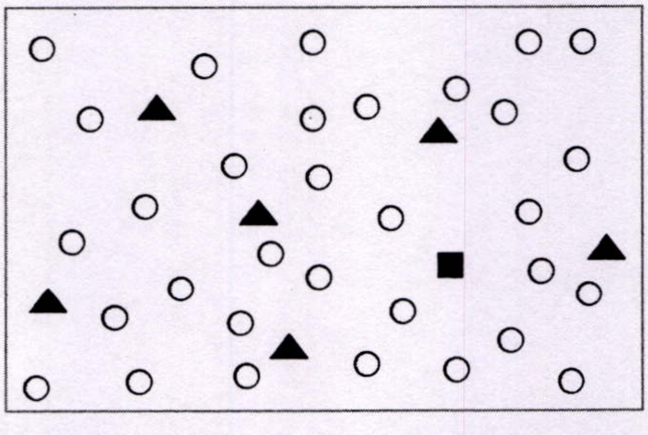
ทั้งนี้ DSDV ได้รับการพัฒนาโดยต้องการลดปริมาณข้อมูล จึงมีส่งข้อมูลเส้นทางอยู่สองประเภทคือ ทั้งตารางเส้นทาง (Full dump) และ เฉพาะเส้นทางที่เปลี่ยนแปลง (Incremental) ซึ่งทั้งสองชนิดนี้มีความสัมพันธ์กับการเปลี่ยนแปลงโครงสร้างเครือข่าย หากในโครงข่ายเซนเซอร์ไร้สาย ไม่มีการเปลี่ยนแปลงโครงสร้างเครือข่ายภายหลังจากทำการเรียนรู้เส้นทางแล้ว การส่งข้อมูลเส้นทางอาจจะไม่เกิดขึ้นหรือเกิดขึ้นเพียงบางส่วน ซึ่งจะทำให้ข้อมูลเวลาที่จะฝากไปด้วยนั้นขาดการทำให้ทันสมัย ก็จะส่งผลต่อปัญหากับอัลกอลิทึมการระบุฟอลคล็อกโหนดด้วยเช่นกัน

โชคดีที่ Routing Protocol ชนิด Table-driven หรือ Proactive นั้นต้องมีข้อมูลจากชั้น Data link layer ที่เรียก Hello Message ส่งอยู่ทุกๆ ช่วงเวลาหนึ่ง เพื่อบอกสถานะโหนดให้กับโหนดข้างเคียง ข้อมูล Hello Message นี้จึงน่าจะเหมาะสมกับการใส่ข้อมูลเวลาไปด้วย (Piggy Back) มากกว่าข้อมูลเส้นทางจากชั้นเครือข่าย แต่อย่างไรก็ตาม DSDV ไม่ได้ให้รายละเอียดเกี่ยวกับโครงสร้างข้อมูล Hello Message มากนัก ซึ่งแตกต่างกับ OLSR2 [24] ที่มีรายละเอียดระบุโครงสร้างข้อมูล NeighborHood Discovery Protocol (NHDP) ตาม RFC5444 [25] ซึ่งยินยอมให้ใส่ข้อมูลอื่น เช่น ข้อมูลเวลา หรือข้อมูลควบคุมโครงข่าย เป็นต้น ร่วมไปได้

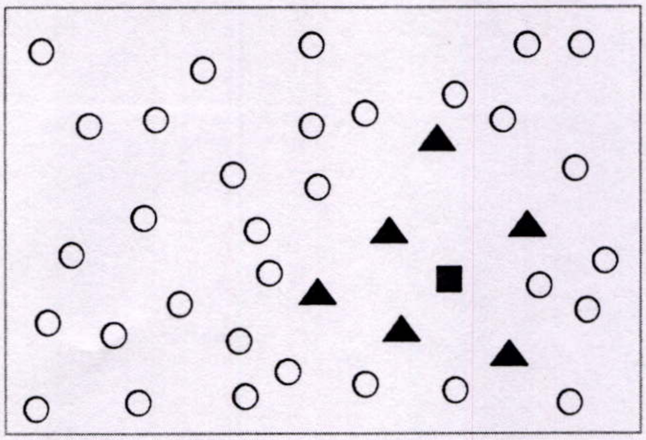
ดังนั้นจึงมีความเป็นไปได้ที่จะส่งข้อมูลเวลา (Piggy Back) ร่วมไปกับ Hello Message หรือ NHDP ที่ใช้ใน DSDV และ OLSR2 ตามลำดับ ซึ่งทุกโหนดจำเป็นต้องส่งอยู่สม่ำเสมอ เพื่อช่วยลดปริมาณข้อมูลเวลาที่โพรโทคอลที่นำเสนอนี้สร้างขึ้น โดยสามารถลงได้มากจนถึงอาจจะไม่ต้องส่งเพิ่มเติมอีก ทำให้สมรรถนะด้านปริมาณข้อมูลที่ใช้ในโพรโทคอลที่นำเสนอไม่แตกต่างจากโพรโทคอล TPSN (Shortest tree) และ TPSN (Randomize)

5.2.4 ตำแหน่งและลักษณะการกระจายตัวของฟอลคล็อกโหนด

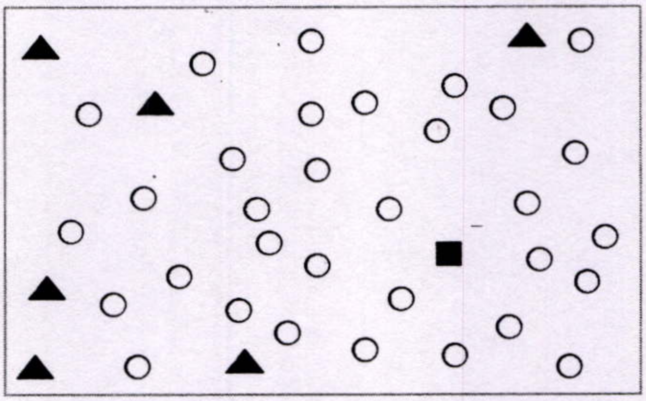
เนื่องจากโพรโทคอลที่นำเสนอนี้มีการทดลองการระบุและหลีกเลี่ยงฟอลคล็อกโหนดที่กระจายตัวอย่างอิสระในเครือข่าย ดังรูปที่ 5.4 เป็นกรณีสมมุติของตำแหน่งของโหนดชนิดต่างๆ ได้แก่ โหนดรูต (แทนด้วยสี่เหลี่ยมสีดำ) โหนดปกติ (แทนด้วยวงกลมสีขาว) และฟอลคล็อกโหนด (แทนด้วยสามเหลี่ยมสีดำ) ซึ่งมีการตั้งข้อสมมุติว่าหากลักษณะการกระจายตัวเปลี่ยนเป็นดังรูปที่ 5.5 และ 5.6 โพรโทคอลที่นำเสนอจะยังคงให้ผลลัพธ์สมรรถนะในลักษณะเดิมหรือไม่ เพราะถ้าฟอลคล็อกโหนดจับกลุ่มกันอยู่ใกล้โหนดรูต โพรโตคอลที่นำเสนออาจจะมีผลไม่แตกต่างหรือแย่กว่าโพรโทคอล TPSN (Shortest tree) ก็เป็นไปได้ ซึ่งข้อสมมุติฐานนี้ควรได้รับการทดลองในโอกาสต่อไป



รูปที่ 5.4 ตำแหน่งของโหนดเซนเซอร์ที่สุ่มให้มีการกระจายตัวอิสระ



รูปที่ 5.5 ตำแหน่งของฟอลคล็อกโหนดที่สุ่มให้กระจายตัวใกล้โหนดรูต



รูปที่ 5.6 ตำแหน่งของฟอลคล็อกโหนดที่สุ่มให้กระจายตัวไกลจากโหนดรูต

บรรณานุกรม

[1] Crossbow Technology.[Online]. Available : http://www.xbow.com.

[2] D. Culler, D. Estrin and M. Srivastava, "Overview of Sensor Networks," IEEE Computer Society 2004 , pp. 41-49, Aug 2004.

[3] D. L. Mills, "Internet time synchronization: The Network Time Protocol" In Ż. Yang and T.A. Marsland, editors, Global States and Time in Distributed Systems. IEEE Computer Society Press, 1994.

[4] E. D. Kaplan, editor. Understanding GPS: Principles and Applications. Artech House, 1996.

[5] B. Sundararaman, U. Buy and A. D. Kshemkalyani, "Clock synchronization for wireless sensor networks: a survey," Elsevier, Ad Hoc Networks, 2005.

[6] J. R. Vig. Introduction to Quartz Frequency Standards. Technical Report SLCET-TR-92-1, Army Research Laboratory, Electronics and Power Sources Directorate, October 1992. Available at http://www.ieee-uffc.org/freqcontrol/quartz/vig/vigtoc.htm.

[7] S. Ganeriwal, R. Kumar, and M.B. Srivastava, "Timing-Sync Protocol for Sensor Networks," Proc. First Int'l Conf Embedded Networked Sensor Systems (SenSys), 2003.

[8] J. Elson, L. Girod and D. Estrin, "Fine-Grained Network Time Synchronization using Reference Broadcasts," In the proceedings of the fifth symposium on Operating System Design and Implementation (OSDI 2002), December 2002.

[9] H. Kim, D. Kim and S.Yoo, "Cluster-based Hierarchical Time Synchronization for Multi-hop Wireless Sensor Networks," Proc. of the 20th Int'l Conf Advanced Information Networking and Applications (AINA'06), 2006.

[10] S. Hwang and Y. Baek, "Reliable Time Synchronization Protocol for Wireless Sensor Networks," EUC 2005, LNCS 3824, pp. 663–672, 2005.

[11] P. Ubolkosold, S. Knedlik and O. Loffeld, "Clock synchronization protocol for distributed satellite networks," Geoscience and Remote Sensing Symposium, 2005. IGARSS '05. Proceedings. 2005 IEEE International, July 2005.

[12] A.Perrig, R.Szewczyk, V.Wen, D.Culler and J.D.Tygar. "SPINS: Security protocols for sensor networks," ACM MobiCom'01. 2001.

[13] M.Yu, H.Mokhtar, and M.Merabti. "A Survey on Fault Management in Wireless Sensor Networks," The 8th Annual Postgraduate Symposium. 2007.

[14] S. Seareesavetrat, C. Pornavalai and R. Varakulsiripunth, “A Light-weight Fault-Tolerant Time Synchronization for Wireless Sensor Networks,” The 8th International Conference on ITS Telecommunications, September 2008.

[15] S. Ganeriwal, V. Tsiatsis, C. Schurgers and M. B. Srivastava, “NESLsim: A parsec based simulation platform for sensor networks, ” NESL, 2002.

[16] PARSEC User Manual, Available :http://pcl.cs.ucla.edu/projects/parsec, 1999.

[17] P. Volgyesi, G. Balogh, A. Nadas, C. B. Nash and A. Ledeczi, "Shooter localization and weapon classification with soldier-wearable networked sensors," In Proceedings of the 5th international Conference on Mobile Systems, Applications and Services (San Juan, Puerto Rico, June 11 - 13, 2007). MobiSys '07. ACM, New York, NY, 113-126.

[18] A. Ledeczi, K. Frampton, G. Simon, and M. Maroti, "Shooter localization problem in urban warfare," Available : http://www.isis.vanderbilt.edu/projects/nest/ documentation/Vanderbilt_NEST_Shooter_SanDiego.ppt.

[19] Grubbs Test.[Online]. Available : http://en.wikipedia.org/wiki/ Grubbs'_test_for_outliers

[20] Kelvin T. Leung, D. Stott Parker, Empirical Comparisons of Various Voting Methods in Bagging, SIGKDD 03, 2003

[21] C. E. Perkins, E. M. Royer and S. R. Das, "Ad Hoc On-demand Distance Vector Routing", October 99 IETF Draft, 33 pages. Available : http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-04.txt

[22] C.E. Perkins, and P. Bhagwat, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, ACM SIG-COMM'94, pp.234-244, (1994).

[23] M. D. Yarvis, W. S. Conner, L. Krishnamurthy, A. Mainwaring, J. Chhabra, and B. Elliott. Real-World Experiences with an Interactive Ad Hoc Sensor Network. In International Conference on Parallel Processing Workshops, 2002.

[24] T. Clausen, C. Dearlove and P. Jacquet, "The Optimized Link State Routing Protocol version 2," Available : http://www.ietf.org/internet-drafts/draft-ietf-manet-olsrv2-08.txt

[25] T. Clausen, C. Dearlove, J. Dean and C. Adjih, "Generalized Mobile Ad Hoc Network (MANET) Packet/Message Format," Available : http://www.ietf.org/rfc/rfc5444.txt

ภาคผนวก

ภาคผนวก ก.

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

1. **S. Seareesavetrat**, C. Pornavalai, R. Varakulsiripunth, **"A Light-weight Fault-Tolerant Time Synchronization for Wireless Sensor Networks,"** The 8th International Conference on ITS Telecommunications, September 2008.

A Light-weight Fault-Tolerant Time Synchronization for Wireless Sensor Networks

Sakpong Seareesavetrat*, Chotipat Pornavalai* and Ruttikorn Varakulsiripunth†
Research Center for Communications and Information Technology (ReCCIT) and
*Faculty of Information Technology
E-mail: s7066150@kmitl.ac.th, chotipat@it.kmitl.ac.th
† Faculty of Engineering
E-mail: kvruttik@kmitl.ac.th
King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand

***Abstract-* Wireless sensor networks (WSN) have received a lot of attention recently due to their enormous potential in different applications. Many applications in WSN require time synchronization among all the sensor nodes. However, in practice, network do normally have some abnormal nodes (we called Fault Clock Node or FCN), which could cause high synchronization error. In this paper, we present a light-weight fault-tolerant time synchronization protocol (LiFTiS) which works in a way similar to NTP. The protocol is aware of FCN and is able to detect its children in hierarchical tree. We also propose a light-weight algorithm for selecting the best parent. Our simulation results show that the average synchronization error using the proposed protocol is 5-31% less than the existing algorithms e.g. TPSN etc.**

Keyword: wireless sensor networks, time synchronization, fault-tolerance.

I. INTRODUCTION

Time synchronization is a critical part of infrastructure for any distributed system. Especially, distributed wireless sensor networks make extensive use of synchronized time: for example, to integrate a time-series of proximity detection into velocity estimate; to measure the time-of-travel of sound for localizing its source; to distribute a beamforming array; or to suppress redundant message by recognizing that they are duplicates of the same event detected by difference sensors. In addition to these domain-specific requirements, sensor network applications often rely on synchronization as typical distributed systems do: for secure cryptographic schemes, coordination of future action, ordering logged events during system debugging, etc.

A. General clock definition [8]

Nearly all computing machines have a circuit for keeping track of time. Despite the widespread use of the word *clock* to refer to these devices, is *timer* more meaningful the way they work. An electronic timer is a precisely tooled quartz crystal. Typical crystal oscillators are accurate of the order of one part in 10^4 to 10^6 (±40 p.p.m) [4] – that is, two nodes' clocks will have a maximum drift 4-5 μs/s or approximately 20 ms/hr. The *skew* is the frequency difference between nodes. It will cause difference in actual time between them, which is called *offset*.

B. Terms in time synchronization [8]

Accuracy refers to how well the maintained time is true to a standard time. To *synchronize time* means to set their time at a particular epoch to be exactly the same. After time synchronization, remain a tiny drift error accumulates which is referred as synchronization error (*SE*).

C. Performance metrics for time synchronization

Two significant metrics to evaluate time synchronization protocol are average synchronization error (*ASE*) and transmission overhead (*TX* messages). *ASE* is an average of time synchronization errors that still remain after time synchronization of all nodes in the network. Most time synchronization protocols will try to improve *ASE* i.e. reducing it. The second metric is the communication messages created for time synchronization which causes an overhead. A lot of transmission messages cause reduction in node life time.

D. Sources of average synchronization error

Sources of *ASE* in wireless sensor networks can be divided into two. First source of error is the synchronization error (*SE*) between a pair of nodes. S. Ganeriwal et al. [1] proposed TPSN and proved that two factors towards the synchronization error are the uncertainty in communication delay and the drift between the local clocks of pair nodes, expressed as $S^{UC} + P^{UC} + R^{UC} + RD$. Here S^{UC}, R^{UC}, P^{UC} and RD stand for the uncertainty at sender, at receiver, in propagation time and the relative drift between two nodes respectively. The second source error is the *accumulated synchronization error* through the synchronization procedure, typically increase with more hop counts [2], [3].

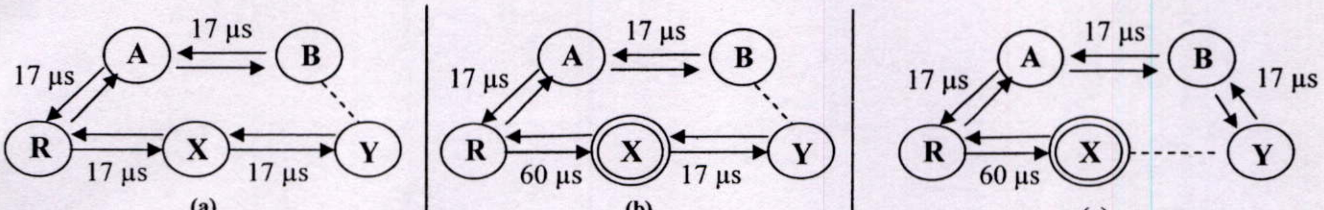
E. Sensor nodes with abnormal or fault clock

A wireless sensor networks is typically made up of numerous sensors which have different clock accuracies. Notes abnormal clock behavior having high skew error are referred as *fault clock node* (FCN). The high skew error in clock maybe caused with due to hardware error or harsh environment such as temperature, poor battery condition and shock at the time of installation etc.

The hierarchical infrastructure is composed of parent nodes and child nodes. When a child selects FCN to be its parent, we refer this parent node as *bad parent node* (BPN). Any child node synchronized with BPN is referred as *inherited bad parent node* (IBPN). Additionally, any child node that inherit from IBPN is also called IBPN.

F. Significance of being aware of BPN/IBPN

978-1-4244-2858-8/08/$25.00 ©2008 IEEE



(a)

$$ASE \approx \frac{(SE_{RA} + SE_{RB} + SE_{RX} + SE_{RY})}{\textit{Total node synchronized}}$$

$$\approx \frac{(SE_{RA} + (SE_{RA} + SE_{AB}) + SE_{RX} + (SE_{RX} + SE_{XY}))}{4}$$

$$\approx (17 + (17 + 17) + 17 + (17 + 17))/4$$

$$\approx \mathbf{25.5\ \mu s}$$

(b)

$$ASE \approx \frac{(SE_{RA} + SE_{RB} + SE_{RX} + SE_{RY})}{\textit{Total node synchronized}}$$

$$\approx \frac{(SE_{RA} + (SE_{RA} + SE_{AB}) + SE_{RX} + (SE_{RX} + SE_{XY}))}{4}$$

$$\approx (17 + (17 + 17) + 60 + (60 + 17))/4$$

$$\approx \mathbf{47\ \mu s}$$

(c)

$$ASE \approx \frac{(SE_{RA} + SE_{RB} + SE_{RX} + SE_{RY})}{\textit{Total node synchronized}}$$

$$\approx \frac{(SE_{RA} + (SE_{RA} + SE_{AB}) + SE_{RX} + (SE_{RA} + SE_{AB} + SE_{BY}))}{4}$$

$$\approx (17 + (17 + 17) + 60 + (17 + 17 + 17))/4$$

$$\approx \mathbf{40.5\ \mu s}$$

Figure 1: Significant of Awareness and avoidance BPN/IBPN. (a) General case. (b) BPN/IBPN is cause to increase *ASE*, compared to general case. (c) Aware and avoid BPN/IBPN can reduce *ASE*, eventually a new path may longer than old path.

First we will give two assumptions about FCN. (1) The FCN or BPN is identified by itself (2) All clock nodes have been drifting in the same direction and error was a deterministic quantity. Let us explain why we should avoid BPN and IBPN. Figure 1 shows a WSNs topology. Node R in this topology is the root node. A, B, X, and Y try to synchronize time with R or its own parent. Bidirectional arrows represent message exchanges for offset and propagation delay estimation. Besides, Dashed line represents possible alternate communication paths. We assign SE_{ij} as synchronization error between node *i* and node *j*. TPSN [1] was able to synchronize a pair of nodes to an average accuracy of 17 µs. The synchronization error would accumulate as hop count along the synchronization path increases in multi-hop wireless sensor network. As in figure 1(a), if $SE_{RX} \approx 17$ µs, then $SE_{RY} \approx (SE_{RX} + SE_{XY}) \approx (17 + 17) \approx 34$ µs.

Approximate of *ASE* for this network is the summation of all accumulated synchronization errors from every pair of nodes divided by the total node synchronized in the network. As shown in figure 1, *ASE* is $(SE_{RA} + SE_{RB} + SE_{RX} + SE_{RY}) / n\text{-}1$; when $n = 5$, figure 1 (a) show approximate *ASE* is 25.5 µsec. In figure 1 (b) node X was identified to be BPN and has synchronization error with its parent ≈ 60 µs. Although all another nodes still have synchronization error at ≈ 17 µs, the *ASE* in this case was increased to 47 µs, because of BPN synchronization errors and IBPN accumulated synchronization errors. In the worst case, when all the child nodes synchronize with BPN or IBPN, the approximate *ASE* would be very high. In figure 1 (c), we show that if Y simply chooses B as parent instead of X, the *ASE* will be reduced from 47 µs to 40.5 µs.

In this paper, we propose a new time synchronization algorithm and protocol that can reduce *ASE* when FCN exists in the network. Our method avoid BPN/IBPN by addition of a light-weight best parent selection algorithm, without increasing overhead to the system. Only a small retransmission overhead is needed when collision or lost reply occur. The results from simulation show that our proposed protocol has *ASE* lower than several existing time-synchronization protocols.

The rest of this paper is organized as follows. Section 2 discusses the motivation and related research in this area. In section 3, we describe our contributions and explain the proposed algorithm. Section 4 includes the performance evaluation of the proposed method. Finally, we conclude this paper and describe our future work in section 5.

II. RELATED WORKS

J. Elson et al. [5] propose Reference Broadcast Synchronization (RBS). It is a 'Receiver to Receiver' model, which synchronizes a group of receivers. Main idea is to remove the uncertainty from sender side with explicit broadcast manner synchronization error and to estimate the clock drift of nodes with linear regression. S. Ganeriwal et al. [1], [9] propose Timing-sync Protocol for Sensor Networks (TPSN) which is 'Sender to Receiver' model. A pair of nodes exchange time message for calculating its offset. TPSN proved that its accuracy from time synchronization is half compared to RBS [5], but it must do timestamp at MAC layer when sending a message. H. Kim et al. [2] propose a Cluster-based Hierarchical Time Synchronization protocol (CHTS) by clustering the network to decrease the hop count along the synchronization path. S. Hwang et al. [3] proposed a Reliable Time Synchronization Protocol (RTSP) by maintaining and updating information of candidate parent nodes. These two papers propose methodologies to create hierarchical tree with lower depth. Their experimental results show reduction of average synchronization error when tree depth is reduced.

III. IDEA & PROPOSED ALGORITHM

A. Idea

We supposed that if the wireless sensor network have the knowledge of FCN or BPN, the system can maintain BPN and IBPN states along the network. Consequently, nodes can choose its best parent from its candidate parent nodes for constructing hierarchical tree. Eventually, if the system can not completely avoid IBPN, selecting the IBPN at the lowest level

is still better. By maintaining the nodes' states and trying to avoid such bad nodes (BPN), large *ASE* caused by BPN or IBPN can be reduced.

We propose a light-weight parent selection algorithm by modifying the traditional TPSN level discovery and panic level discovery processes. Our proposed algorithm almost neither increases overhead nor has any negative effects to other actions of traditional TPSN.

B. Algorithm

Our protocol and algorithm is similar to the TPSN. Every node in the network, either good node or bad node (BPN and IBPN) are required to be synchronized to the same clock standard i.e. same root node. The modifications to the TPSN are at the level discovery phase and at panic level discovery phases. We extend the process of parent identification and a parent selection for optimization in the hierarchical tree. We summarize our methodology into 2 parts as follows.

B.1) *BPN and IBPN identifications* - In this paper, we use a simple algorithm and assume a trust model. Each node verify its high error clock' skew compared with drift error since last time it has synchronized (normally, clock has drift approximate 20 ms/hr. [4]). If the difference is over the threshold (for example, threshold is set at 50 ms/hr., a value much higher compared to the average), then this node will update its state as a BPN, and then broadcast this information to its neighbors when the network do level discovery. We define this information as 'bad_inherit_level' (*BIL*). The *BIL* information includes 'own_level' (*LV*) information, which is the level in the tree maintained at each node. The *BIL* will be maintained corresponding to the BPN or IBPN state along the path to the root. In case that a node cannot avoid BPN or IBPN to be its parent, it can choose one that has lowest *BIL* value. The child of BPN (or IBPN as well) will update its' *BIL* by increasing its *BIL* value from parent's *BIL* by one. A example of how to update *BIL* and *LV* is shown in Figure 2 (a).

B.2) *Light-weight best parent selection algorithm* is a decision algorithm that extended the TPSN' level discovery and panic level discovery phases. As in figure 2 (b), while Y is waiting for broadcast of level discovery information from upper level, Y will receive two level discovery messages, from X and from B (then *N_REPLY* = 2). As a result of information (*BIL* and *LV*) received from each parent candidates (X and B), Y can determine which candidate is a BPN or IBPN. If *BIL* value equal or more than 1 received from the parent candidates, then any child node can aware of BPN/IBPN and avoid it. Therefore in this example, Y will choose B as its parent node. Although in situations where BPN/IBPN cannot be avoided, the shortest path to root node or lowest *LV/BIL* value would still be a better choice. Finally every child node will identify its parent node (*Parent_ID*) and construct the tree. Figure 3 is the pseudo code for light-weight best parent selection algorithm.

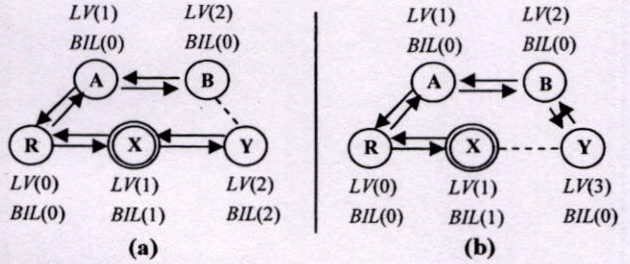


Figure 2: Level discovery and maintenance. (a) X is BPN and Y is IBPN. (b) Y avoid the BPN by select another candidate parent that *BIL* is 0 (or lowest first). In this case, Y will select B instead of X.

```
For i=1 To N_REPLY (number of parent candidate
received) Do
  If NOT ((candidate[i].state = = BPN) OR
(candidate[i].state = = IBPN))
    If (candidate[i].level is lowest)
      first_choice ← candidate[i];
      /* This is a best candidate parent,
         also do shortest path to root node. */
    Else
      second_choice ← candidate[i];
      /* This parent may do not shortest path to root node. */
    Endif
  Else If (candidate[i].state is lowest inherit level)
    third_choice ← candidate[i];
    /* This parent is a best of BPN or IBPN. */
  Else
    last_choice ← candidate[i]; is last choice to be the
parent;
    /* This parent is very bad parent, because its do
       longest path to root and a huge of cumulative error. */
  Endif
Endfor
/* Then set Parent_ID follow as priority choice
   were found from previous conditions */
  Parent_ID ← first_choice or second_choice or
third_choice or last_choice respectively;
```

Figure 3: A light-weight select best parent algorithm (LiFTiS).

IV. EVALUATIONS

In order to evaluate the performance of the proposed method, we established a simulation model in the NESLsim based on the PARSEC platform. PARSEC (PARallel Simulation Environment for Complex systems) is a C-based discrete event simulation language. It adopts the process interaction approach to discrete event simulation [7]. In NESLsim, a sensor network is modeled as a collection of sensor nodes, a channel, and a supervising entity to create the nodes [6]. NESL simulator was used in TPSN [9], [1] and RTSP [3].

A. Metrics

In this paper, we use the average synchronization error (*ASE*) and number of transmitted messages from all nodes to be the performance metrics. We then evaluate the efficiency of the proposed algorithm with the traditional TPSN (randomly

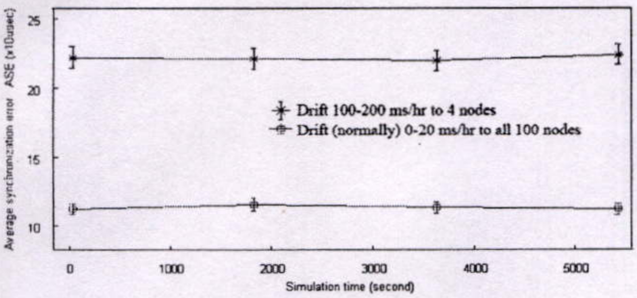


Figure 4: Effect of BPN(FCN) and IBPN on *ASE*.

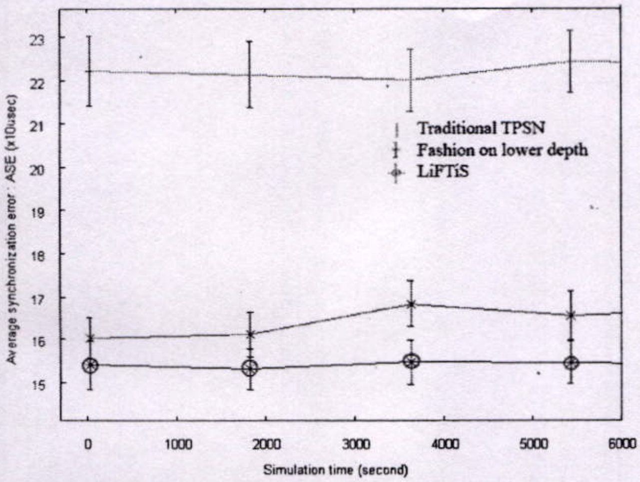


Figure 5: Reduction of *ASE* values for 3 difference parent selection algorithms.

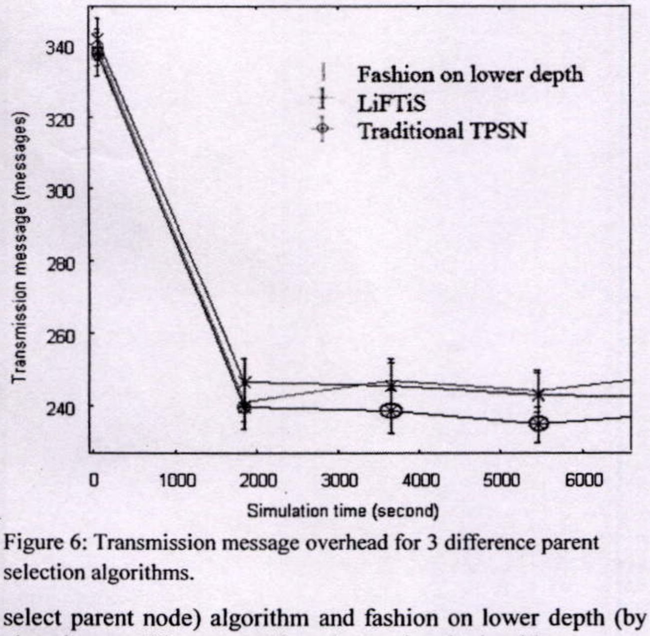


Figure 6: Transmission message overhead for 3 difference parent selection algorithms.

select parent node) algorithm and fashion on lower depth (by choosing candidate parent from lowest level) algorithm.

B. Parameters

Networks with 100 nodes are deployed in a uniform random fashion where each node has 0-9 node neighbors. The setup includes a CSMA MAC (no RTS/CTS/ACK and no radio layer buffer). The bit error ratio is zero, i.e., every transmission is either success or failure only by collision. The radio speed is 19.2kb/s, similar to the UC Berkeley MICA Motes. Every packet has a fixed size of 128 bits. A root node is stationary at the middle of network area. The granularity of the node clocks, which is the minimum accuracy attainable, is 10 μs. The clock model used in simulations has been derived from the characteristics of the oscillators used in sensor nodes. The frequency drift is varied randomly with time, normally within the specified range at 20 ms/hr. (or 4 μs/s) [4], to model the temporal variations with temperature. All sensor node clock drifts independent of each other. There is an initial random offset, uniformly distributed, at the boot up time. Each synchronization period (round) is graph show more than 30 minutes. Every scenarios run 300 times and graph results show 99% confidence interval. All other parameters have the same values as in the TPSN simulation [1], [9].

C. Results

Figure 4 shows how average synchronization error was affected by BPN/IBPN. In TPSN parent node was selected randomly. After adding various numbers of error nodes (BPN), and their error rates into network, *ASE* was increased.

Figure 5 compares *ASE* of LiFTiS with traditional TPSN and fashion on lower depth hierarchical tree. In this scenario there are 4 BPN nodes that have 100-200 ms/hr drift time error. LiFTiS has *ASE* lower than traditional TPSN and fashion on lower depth algorithms by about 31%, 5% respectively.

Figure 6 shows number of overhead messages for synchronization and level discovery processing. The result shows that LiFTiS and fashion on lower depth use a small or number of messages compared to the traditional TPSN by approximate 4%. This is caused by the number of retransmission after collision and panic level discovery which occurs more often in LiFTiS and fashion on lower depth compared to the traditional TPSN. By trying to choose best parents in LiFTiS, some parents will have more number of children. Then many child nodes will try to synchronize to the parent node almost at the same time. This leads to collision and requires level discovery process again. So some children will try to find a new parent with panic request for rebuild a new parent connection.

V. CONCLUSIONS AND FUTURE WORKS

In this paper, we proposed a time synchronization protocol and algorithm which is aware of FCN and could avoid of bad parent nodes and inherited bad parent nodes. It is based on TPSN protocol. The proposed algorithm has lower average synchronization error compared to the existing algorithms, at the cost of very small increase in overhead messages.

In the future, we will perform more simulations on various scenarios. We also consider modifying our algorithm to work

even when parent do not identify itself as FCN, so that time synchronization works even if it is under attack.

ACKNOWLEDGMENT

Authors wish to express gratitude Prof. Dr. Goutam Chakraborty who was abundantly helpful and offered invaluable assistance and guidance.

REFERENCES

[1] S. Ganeriwal, R. Kumar, and M.B. Srivastava, "Timing-Sync Protocol for Sensor Networks," Proc. First Int'l Conf Embedded Networked Sensor Systems (SenSys), 2003.

[2] H. Kim, D. Kim and S.Yoo, "Cluster-based Hierarchical Time Synchronization for Multi-hop Wireless Sensor Networks," Proc. of the 20th Int'l Conf Advanced Information Networking and Applications (AINA'06), 2006.

[3] S. Hwang and Y. Baek, "Reliable Time Synchronization Protocol for Wireless Sensor Networks," EUC 2005, LNCS 3824, pp. 663–672, 2005.

[4] R. John. Vig. Introduction to Quartz Frequency Standards. Technical Report SLCET-TR-92-1, Army Research Laboratory, Electronics and Power Sources Directorate, October 1992. Available at http://www.ieee-uffc.org/freqcontrol/quartz/vig/vigtoc.htm.

[5] J. Elson, L. Girod and D. Estrin, "Fine-Grained Network Time Synchronization using Reference Broadcasts," In the proceedings of the fifth symposium on Operating System Design and Implementation (OSDI 2002), December 2002.

[6] S. Ganeriwal, V. Tsiatsis, C. Schurgers, M. B. Srivastava, "NESLsim: A parsec based simulation platform for sensor networks, " NESL, 2002.

[7] PARSEC User Manual, http://pcl.cs.ucla.edu/projects/parsec, 1999.

[8] P. Ubolkosold, S. Knedlik, O. Loffeld, "Clock synchronization protocol for distributed satellite networks," Geoscience and Remote Sensing Symposium, 2005. IGARSS '05. Proceedings. 2005 IEEE International, July 2005.

[9] S. Ganeriwal, R. Kumar, S. Adlakha, M. Srivastava, "Network-wide time synchronization in sensor networks, NESL Technical Report, 2003.

ประวัติผู้เขียน

นายศักดิ์พงษ์ เสรีเศวตรัตน์ เกิดเมื่อวันที่ 9 กันยายน พ.ศ. 2520 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาระดับปริญญาตรี มหาวิทยาลัยรามคำแหง คณะวิทยาการคอมพิวเตอร์ ในปีการศึกษา 2544 หลังจากสำเร็จการศึกษาระดับปริญญาตรี ได้ทำงานตำแหน่งนักพัฒนาเว็บเพจและผู้ประสานงานกับลูกค้าต่างประเทศ เป็นเวลาประมาณ 1 ปี ระหว่างนั้นได้เป็นอาจารย์พิเศษสอนคอมพิวเตอร์ระดับเด็กและผู้ใหญ่ ต่อมาได้รับตำแหน่งนักพัฒนาโปรแกรมและเว็บ พร้อมกับทำหน้าที่ผู้จัดการสาขาให้บริษัท ฟลายบาย จำกัด ก่อนมาศึกษาต่อในระดับปริญญาโท หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาการสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2547 โดยในปี พ.ศ. 2549 ได้เข้าทำงานในตำแหน่งผู้ช่วยนักวิจัย คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง