

การเลือกคลัสเตอร์เฮดเพื่อสมดุลภาระบนเครือข่ายเซนเซอร์ไร้สาย

**Cluster Head Election for Load Balancing  
over Wireless Sensor Networks**

อนิรุธ ทองกลิ่น

**Anirooth Thongklin**

วิทยานิพนธ์นี้สำหรับการศึกษาตามหลักสูตรปริญญา  
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์  
มหาวิทยาลัยสงขลานครินทร์

**A Thesis Submitted in Fulfillment of the Requirements for the Degree of  
Master of Engineering in Computer Engineering  
Prince of Songkla University**

2554

ลิขสิทธิ์ของมหาวิทยาลัยสงขลานครินทร์

ชื่อวิทยานิพนธ์ การเลือกคลัสเตอร์เซตเพื่อสมมูลภาวะบนเครือข่ายเซนเซอร์ไร้สาย

ผู้เขียน นายอนิรุทธ ทองกลิ่น

สาขาวิชา วิศวกรรมคอมพิวเตอร์

---

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

คณะกรรมการสอบ

.....  
(ผู้ช่วยศาสตราจารย์ ดร.วรรณรัช สันติอมรทัต)

.....ประธานกรรมการ  
(ดร.ศกุนา เจริญปัญญาศักดิ์)

.....กรรมการ  
(ผู้ช่วยศาสตราจารย์ ดร.วรรณรัช สันติอมรทัต)

.....กรรมการ  
(ดร.กมล เขมระรังษี)

บัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์ อนุมัติให้บัณฑิตวิทยาลัย  
สำหรับการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรม  
คอมพิวเตอร์

.....  
(ศาสตราจารย์ ดร.อมรรัตน์ พงศ์ดารา)  
คณบดีบัณฑิตวิทยาลัย

ชื่อวิทยานิพนธ์	การเลือกคลัสเตอร์เฮดเพื่อสมดุลภาระบนเครือข่ายเซนเซอร์ไร้สาย
ผู้เขียน	นายอนิรุทธ ทองกลิ่น
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ปีการศึกษา	2553

## บทคัดย่อ

การจัดคลัสเตอร์ในเครือข่ายเซนเซอร์ไร้สายขนาดใหญ่เป็นวิธีที่นิยมนำมาประยุกต์ใช้งานเพื่อช่วยลดความซับซ้อนในการจัดการเครือข่ายและให้การใช้งานทรัพยากรมีประสิทธิภาพมากขึ้น แต่การจัดคลัสเตอร์ทั่วไปไม่ได้คำนึงถึงปริมาณภาระงานที่รับส่งภายในคลัสเตอร์ ทำให้แต่ละคลัสเตอร์มีอัตราการใช้พลังงานที่ไม่สมดุลกัน

งานวิจัยนี้จึงนำเสนออัลกอริทึมการกระจายภาระงานอย่างสมดุลและการเลือกคลัสเตอร์เฮดแบบกริดเซกเตอร์ (Grid Sectoring) ด้วยวิธีการแบ่งพื้นที่สี่เหลี่ยมจตุรัสออกเป็นเซกเตอร์ ซึ่งแทนพื้นที่ของคลัสเตอร์ที่เท่าๆ กัน เมื่อแต่ละคลัสเตอร์มีพื้นที่เท่ากันจำนวนโหนดสมาชิกก็จะมีโอกาสเท่ากัน และจะส่งผลให้ปริมาณภาระงานแต่ละคลัสเตอร์สมดุลกัน งานวิจัยได้ทำการทดสอบประสิทธิภาพของการจัดคลัสเตอร์ด้วยโปรแกรมจำลองกราฟ Networkx และทดสอบประสิทธิภาพการทำงานของอัลกอริทึมร่วมกับโพรโทคอล AODV ด้วยโปรแกรมจำลองเครือข่าย NS-2 บนสภาพของเครือข่ายที่มีการกระจายของโหนดแบบสม่ำเสมอและไม่สม่ำเสมอ การทดลองได้เปรียบเทียบผลที่ได้กับอัลกอริทึมอื่น ได้แก่ LEACH ACHS และ EACHS

จากการทดลองพบว่าอัลกอริทึมที่นำเสนอสามารถกระจายภาระงานในแต่ละคลัสเตอร์ การใช้พลังงานในแต่ละคลัสเตอร์ และการใช้พลังงานของคลัสเตอร์เฮดได้สมดุลกว่าอัลกอริทึมชนิดอื่น มีภาระงานในแต่ละคลัสเตอร์แตกต่างกันไม่เกิน 1.2 กิโลบิตต่อวินาทีซึ่งดีกว่าอัลกอริทึมอื่น 52.0 เปอร์เซ็นต์ มีการใช้พลังงานรวมของแต่ละคลัสเตอร์แตกต่างกันไม่เกิน 28.7 จูลซึ่งดีกว่าอัลกอริทึมอื่น 50.3 เปอร์เซ็นต์ และมีการใช้พลังงานของแต่ละคลัสเตอร์เฮดแตกต่างกันไม่เกิน 1.8 จูลซึ่งดีกว่าอัลกอริทึมอื่น 36.5 เปอร์เซ็นต์ ทั้งนี้การจัดคลัสเตอร์ของโหนดด้วยอัลกอริทึม Grid Sectoring ช่วยส่งผลให้ปริมาณการส่งข้อมูลและอัตราการรับต่อการส่งข้อมูลสูงขึ้น

**คำสำคัญ:** เครือข่ายเซนเซอร์ไร้สาย, การเลือกคลัสเตอร์เฮด, สมดุลภาระงาน, การกระจายตัวของโหนดในเครือข่ายด้วยความหนาแน่นสม่ำเสมอและไม่สม่ำเสมอ, การใช้พลังงานอย่างมีประสิทธิภาพ

**Thesis Title** Cluster Head Election for Load Balancing over Wireless Sensor Networks  
**Author** Mr. Anirooth Thongklin  
**Major Program** Computer Engineering  
**Academic Year** 2010

## ABSTRACT

Clustering has been widely applied in the large-scale wireless sensor networks in order to reduce a complication about network management and to enhance an efficiency of the resource usage. However, the traditional clustering algorithms do not consider the load balancing over networks. Thus, each cluster head will dissipate imbalanced energy consumption.

Consequently, the algorithm for energy-efficient clustering to balance the workloads and grid sectoring has been proposed in this work. Grid sectoring will divide the interesting area into sectors with an equal size. Therefore, the number of node member will be possibly equal. This leads to have the equal workloads. The study used Networkx and NS-2, which are graph and network simulations respectively. The proposed algorithm will be tested together with AODV routing protocol. We have investigated the performance of the proposed algorithm over both uniform and non-uniform deployments. The other three clustering algorithms, LEACH, ACHS, and EACHS have been chosen to compare the results with our algorithm.

We found that the proposed algorithm can has load balancing, energy usage balancing and energy usage balancing in cluster head better than the other algorithms. Each cluster has a different workload not more than 1.2 kbps which is better than the others about 52.0 percent. While the total energy in each cluster consumes a different power not more than 28.7 joules which is better than the others about 50.3 percent. Finally, each cluster head has a different energy usage not greater than 1.8 joules which is also better than the others about 36.5 percent. Additionally, the proposed clustering algorithm based on grid sectoring can also increase the throughput and the packet delivery ratio.

**KEYWORDS:** Wireless Sensor Networks, cluster head election, load balancing, uniform and non-uniform deployment, energy efficiency

## กิตติกรรมประกาศ

ขอแสดงคำขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.วรรณรัช สันติอมรทัต ประธานกรรมการที่ปรึกษางานวิจัย ที่ได้กรุณาให้การสนับสนุนและฝึกฝนการทำวิจัย กรุณาอุทิศเวลาในการให้คำปรึกษา แนะนำและให้แนวคิดในการทำวิจัย เอกสาร ข้อมูลต่างๆ เป็นอย่างดี รวมทั้งการขัดเกลากระบวนการคิดและให้กำลังใจในการแก้ไขปัญหาที่เกี่ยวกับการวิจัย ตลอดจนตรวจสอบและแก้ไขวิทยานิพนธ์ให้ดำเนินไปอย่างคล่องตัวสมบูรณ์

ขอขอบพระคุณ ดร. สุกุณา เจริญปัญญาศักดิ์ ประธานกรรมการสอบวิทยานิพนธ์ และกรรมการสอบโครงร่าง ที่ได้กรุณาให้คำปรึกษา คำแนะนำ ที่เป็นประโยชน์ต่อการทำวิจัยเสมอมา ตลอดจนช่วยตรวจทานแก้ไขวิทยานิพนธ์ให้ดำเนินไปอย่างสมบูรณ์

ขอขอบพระคุณ ดร.กมล เขมะรังษี (ผู้ทรงคุณวุฒิ) ที่กรุณาอุทิศเวลาเป็นกรรมการสอบวิทยานิพนธ์ ให้คำแนะนำที่มีประโยชน์ต่อการทำงานวิจัย อีกทั้งตรวจทานและแก้ไขวิทยานิพนธ์ให้มีความสมบูรณ์ยิ่งขึ้น

ขอขอบพระคุณสำนักงานคณะกรรมการกิจการโทรคมนาคมแห่งชาติ (สกทช.) ที่ให้การสนับสนุนทุนการศึกษาและทุนสนับสนุนการเผยแพร่ผลงานวิจัยแก่ข้าพเจ้า

ขอขอบพระคุณ บัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่ ที่ให้การสนับสนุนทุนในการทำวิจัยและให้ความช่วยเหลือด้านการประสานงานต่างๆ

ขอขอบพระคุณ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์ ที่กรุณาให้ทุนสนับสนุนการเผยแพร่ผลงานวิจัยแก่ข้าพเจ้า

ขอขอบพระคุณ คณาจารย์ บุคลากร และนักศึกษาปริญญาโทภาควิชาวิศวกรรมคอมพิวเตอร์ทุกคนที่ได้ช่วยเหลือ ให้คำปรึกษา และเป็นกำลังใจในการทำงานเป็นอย่างดีเสมอมา

และสุดท้าย ข้าพเจ้าน้อมรำลึกถึงพระคุณของ บิดามารดา และครอบครัว ที่ส่งเสริม สนับสนุน ให้คำแนะนำ ให้คำปรึกษา ให้กำลังใจที่ดีเสมอมาแก่ข้าพเจ้าจนสำเร็จการศึกษา

อนิรุธ ทองกลิ่น

## สารบัญ

	หน้า
สารบัญ.....	6
รายการตาราง.....	9
รายการภาพประกอบ.....	10
สัญลักษณ์คำย่อและตัวย่อ.....	12
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญและที่มาของการวิจัย.....	1
1.2 บทความและงานวิจัยที่เกี่ยวข้องกับประเด็นวิจัย.....	2
1.2.1 การเลือกคลัสเตอร์เฮดด้วยอัลกอริทึม LEACH และ LEACH-C.....	2
1.2.2 การเลือกคลัสเตอร์เฮดด้วยอัลกอริทึม ACHS และ EACHS.....	3
1.2.3 อัลกอริทึมการจัดคลัสเตอร์แบบกริด.....	3
1.2.4 อัลกอริทึมการจัดคลัสเตอร์แบบหกเหลี่ยม.....	4
1.3 ประเด็นวิจัย.....	5
1.4 วัตถุประสงค์การวิจัย.....	5
1.5 ขอบเขตของการวิจัย.....	5
1.6 ขั้นตอนและวิธีดำเนินงานวิจัย.....	6
1.7 โครงสร้างของรายงานวิทยานิพนธ์.....	6
บทที่ 2 ทฤษฎีและหลักการ.....	7
2.1 โพรโทคอลมาตรฐาน IEEE 802.15.4.....	7
2.1.1 ชั้นกายภาพมาตรฐาน IEEE 802.15.4 (Physical Layer).....	8
2.1.2 ชั้นย่อยการควบคุมการเข้าใช้สื่อกลางมาตรฐาน IEEE 802.15.4.....	10
2.2 การจัดเครือข่ายแบบคลัสเตอร์.....	13
2.3 โพรโทคอลการจัดเส้นทาง AODV.....	14

2.3.1 โพรโทคอลการจัดเส้นทาง AODV แบบดั้งเดิม .....	14
2.3.2 โพรโทคอลการจัดเส้นทาง AODV แบบ Clustering .....	15
2.4 การจัดคลัสเตอร์ด้วยอัลกอริทึม ACHS และ EACHS .....	16
บทที่ 3 การวิเคราะห์การเลือกคลัสเตอร์เฮด .....	17
3.1 การเลือกคลัสเตอร์เฮดด้วยอัลกอริทึมที่ใช้ทฤษฎีความน่าจะเป็น .....	17
3.2 อัลกอริทึมการจัดคลัสเตอร์แบบกริด .....	18
3.3 อัลกอริทึมการจัดคลัสเตอร์แบบหกเหลี่ยม .....	19
3.4 บทสรุป .....	20
บทที่ 4 อัลกอริทึมการเลือกคลัสเตอร์เฮดเพื่อสมมูลภาระบนเครือข่ายเซนเซอร์ไร้สาย .....	21
4.1 แนวคิดการจัดคลัสเตอร์แบบกริดเชกเตอร์ .....	21
4.1.1 การแบ่งพื้นที่สนใจเป็นกริดย่อย .....	23
4.1.2 การแบ่งพื้นที่กริดออกเป็นเชกเตอร์ .....	23
4.1.3 การเลือกคลัสเตอร์เฮด .....	24
4.1.4 การจัดคลัสเตอร์แบบกริดเชกเตอร์ .....	26
4.2 การทำงานร่วมกันระหว่างอัลกอริทึมกริดเชกเตอร์กับโพรโทคอล AODV .....	27
4.3 บทสรุป .....	28
บทที่ 5 ผลการจำลองเครือข่าย .....	29
5.1 ตัวชี้วัดที่สนใจ .....	29
5.1.1 การจำลองกราฟ .....	29
5.1.2 การจำลองเครือข่าย .....	30
5.2 การจำลองกราฟ .....	32
5.2.1 สภาพแวดล้อมการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายแบบสม่ำเสมอ .....	32
5.2.2 ผลการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายแบบสม่ำเสมอ .....	35
5.2.3 สภาพแวดล้อมการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายแบบไม่สม่ำเสมอ .....	36

5.2.4 ผลการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายแบบไม่สม่ำเสมอ .....	37
5.3 การจำลองเครือข่าย.....	37
5.3.1 การจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบสม่ำเสมอ.....	37
5.3.2 ผลการจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบสม่ำเสมอ .....	39
5.3.3 การจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบไม่สม่ำเสมอ .....	40
5.3.4 ผลการจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบไม่สม่ำเสมอ .....	42
5.4 ผลการทดลองจำนวนฮอปของอัลกอริทึมกริดเชกเตอร์บน โพรโทคอล AODV .....	44
5.5 บทสรุปและการวิเคราะห์เพิ่มเติม .....	46
5.5.1 สรุปผลการจำลองกราฟ .....	46
5.5.2 สรุปผลการจำลองเครือข่าย .....	46
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ .....	47
6.1 สรุปผลการวิจัย.....	47
6.2 ปัญหาและข้อเสนอแนะ .....	48
6.2.1 ปัญหาจากการดำเนินงานวิจัย .....	48
6.2.2 ข้อเสนอแนะ.....	49
บรรณานุกรม .....	50
ภาคผนวก .....	53
ภาคผนวก ก ผลการทดสอบเพิ่มเติม.....	54
ภาคผนวก ข ผลการวิเคราะห์โพรโทคอล AODV บนมาตรฐาน IEEE 802.15.4 .....	68
ภาคผนวก ค ผลงานตีพิมพ์เผยแพร่จากวิทยานิพนธ์.....	76



## รายการตาราง

	หน้า
ตารางที่ 3-1 จำนวนโหนดสมาชิกเฉลี่ยการแบ่งแบบกริดย่อยกับใช้คลัสเตอร์เซด 5 เปอร์เซ็นต์ .....19	
ตารางที่ 5-1 การแบ่งพื้นที่ของอัลกอริทึมกริดเซกเตอร์ตามจำนวนโหนดในเครือข่าย.....33	
ตารางที่ 5-2 การแบ่งพื้นที่ของอัลกอริทึมกริดเซกเตอร์ตามจำนวนโหนดในเครือข่าย.....37	

## รายการภาพประกอบ

	หน้า
ภาพประกอบ 1-1 การจัดคลัสเตอร์แบบกริด.....	4
ภาพประกอบ 1-2 การจัดคลัสเตอร์แบบหกเหลี่ยม.....	4
ภาพประกอบ 2-1 มาตรฐานเครือข่ายไร้สาย .....	8
ภาพประกอบ 2-2 ย่านความถี่ขึ้นกายภาพมาตรฐาน IEEE 802.15.4 .....	9
ภาพประกอบ 2-3 โครงสร้างชั้นกายภาพและชั้นย่อย MAC มาตรฐาน IEEE 802.15.4 .....	10
ภาพประกอบ 2-4 โทโพลยีที่ชั้นย่อย MAC สนับสนุนบนมาตรฐาน IEEE 802.15.4.....	11
ภาพประกอบ 2-5 การทำงานของโพรโทคอล AODV (a) การส่งข้อความ RREQ และ .....	15
ภาพประกอบ 2-6 การเลือกคลัสเตอร์เฮดของอัลกอริทึม ACHS.....	16
ภาพประกอบ 3-1 ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของจำนวนคลัสเตอร์เฮด .....	18
ภาพประกอบ 4-1 การแบ่งพื้นที่สนใจเป็นสี่เหลี่ยมจัตุรัสย่อย.....	21
ภาพประกอบ 4-2 (a) การแบ่งพื้นที่สนใจเป็นกริดย่อยในลำดับที่ 1 และ 2 ตามลำดับ (b) การแบ่งกริดย่อยออกเป็นเซกเตอร์ .....	22
ภาพประกอบ 4-3 การแบ่งพื้นที่สนใจเป็นกริดย่อยและแทนจำนวน โหนดลงในเมตริกซ์ .....	23
ภาพประกอบ 4-4 การแบ่งพื้นที่กริดออกเป็นเซกเตอร์.....	24
ภาพประกอบ 4-5 ผลลัพธ์การแบ่งพื้นที่กริดออกเป็น 6 เซกเตอร์ด้วยโปรแกรม Networkx.....	25
ภาพประกอบ 4-6 (a) ผลรวมของมุมเท่ากับ $2\pi$ เมื่อ โหนดอยู่ในขอบเขตของคลัสเตอร์ $a_4$ .....	27
ภาพประกอบ 4-7 การจำกัดจำนวนฮอปในการส่งแพ็กเก็ตควบคุมของโพรโทคอล AODV .....	28
ภาพประกอบ 5-1 ผลการแบ่งพื้นที่ออกเป็น 5 เซกเตอร์ เมื่อ โหนดในเครือข่ายมีจำนวน 100 โหนด .....	33
ภาพประกอบ 5-2 ผลการแบ่งพื้นที่ออกเป็น 10 เซกเตอร์ เมื่อ โหนดในเครือข่ายมีจำนวน 200 โหนด .....	34
ภาพประกอบ 5-3 ผลการแบ่งพื้นที่ออกเป็น 4 กริด กริดละ 4 เซกเตอร์ เมื่อ โหนดในเครือข่ายมีจำนวน 300 โหนด.....	34
ภาพประกอบ 5-4 ผลการแบ่งพื้นที่ออกเป็น 4 กริด กริดละ 5 เซกเตอร์ เมื่อ โหนดในเครือข่ายมีจำนวน 400 โหนด.....	34
ภาพประกอบ 5-5 ผลการแบ่งพื้นที่ออกเป็น 4 กริด กริดละ 6 เซกเตอร์ เมื่อ โหนดในเครือข่ายมีจำนวน 500 โหนด.....	35

ภาพประกอบ 5-6 ผลการทดลองค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย.....	35
ภาพประกอบ 5-7 การกระจายความหนาแน่นของโหนดที่แตกต่างกันในพื้นที่ 4 ส่วน .....	36
ภาพประกอบ 5-8 ผลการแบ่งกริดเป็นเซกเตอร์ของอัลกอริทึมกริดเซกเตอร์บนเครือข่ายที่โหนดมีการกระจายด้วยความหนาแน่นไม่สม่ำเสมอ .....	36
ภาพประกอบ 5-9 ผลการทดลองค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย.....	37
ภาพประกอบ 5-10 ผลการจำลองเครือข่ายที่ใช้อัลกอริทึมกริดเซกเตอร์ เมื่อโหนดในเครือข่ายมีจำนวน 100 โหนด.....	38
ภาพประกอบ 5-11 ผลการทดลองค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของภาระงาน .....	39
ภาพประกอบ 5-12 ผลการทดลองการใช้พลังงานเฉลี่ยต่อคลัสเตอร์และค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์ .....	39
ภาพประกอบ 5-13 ผลการทดลองการใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด.....	40
ภาพประกอบ 5-14 การกระจายความหนาแน่นของโหนดในแต่ละพื้นที่ 4 ส่วน .....	41
ภาพประกอบ 5-15 ผลการจำลองเครือข่ายตามสภาพแวดล้อมที่ 2 ด้วยอัลกอริทึม LEACH.....	41
ภาพประกอบ 5-16 ผลการจำลองเครือข่ายตามสภาพแวดล้อมที่ 2 ด้วยอัลกอริทึม ACHS.....	41
ภาพประกอบ 5-17 ผลการจำลองเครือข่ายตามสภาพแวดล้อมที่ 2 ด้วยอัลกอริทึม Grid Sectoring.....	42
ภาพประกอบ 5-18 ผลการทดลองค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของภาระงาน .....	42
ภาพประกอบ 5-19 ผลการทดลองการใช้พลังงานเฉลี่ยต่อคลัสเตอร์และค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์ .....	43
ภาพประกอบ 5-20 ผลการทดลองการใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด.....	43
ภาพประกอบ 5-21 ผลการทดลองการใช้พลังงานเฉลี่ยต่อคลัสเตอร์และค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์ .....	45
ภาพประกอบ 5-22 ผลการทดลองการใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด.....	45

## สัญลักษณ์คำย่อและตัวย่อ

ACHS	Adaptive Cluster Head Selection
AODV	Ad-hoc On-demand Distance Vector
BPSK	Binary Phase Shift Keying
BS	Base Station
CCA	Clear Chanel Assessment
CH	Cluster Head
CSMA-CA	Carrier Sense Multiple Access with Collision Avoidance
DSSS	Direct Sequence Spread Spectrum
EACHS	Extended Adaptive Cluster Head Selection
ED	Energy Detection
FFD	Full Function Device
GS	Grid Sectoring
GS-HLF	Grid Sectoring with Hop Limited Flooding
LEACH	Low-Energy Adaptive Clustering Hierachy
LLC	Logical Link Control
LR-WPAN	Low-Rate Wireless Personal Area Network
MAC	Media Access Control
MN	Member Node
NS-2	Network Simulator 2
OQPSK	Offset-Quadrature Phase Shift Keying
PAN	Personal Area Network
PHY	Physical Layer
RFD	Reduced Function Device
WPAN	Wireless Personal Area Network
WSNs	Wireless Sensor Networks

# บทที่ 1

## บทนำ

### 1.1 ความสำคัญและที่มาของการวิจัย

เครือข่ายเซนเซอร์ไร้สาย (Wireless Sensor Networks หรือ WSNs) ได้รับความนิยมและถูกนำไปใช้ในงานประยุกต์ที่หลากหลาย ได้แก่ งานทางด้านเกษตรกรรม ด้านการแพทย์ สำหรับเฝ้าติดตามสัญญาณชีพที่สำคัญของผู้ป่วย ด้านการเฝ้าระวังหรือเตือนภัยธรรมชาติ เป็นต้น เครือข่ายเซนเซอร์ไร้สายประกอบด้วยอุปกรณ์ขนาดเล็กจำนวนมาก ที่เรียกว่าโหนด (Sensor node) มีความสามารถในการประมวลผลข้อมูล ตรวจวัด และสื่อสารบนเครือข่ายแบบไร้สาย เครือข่ายเซนเซอร์ไร้สายมีความสามารถในการปรับเปลี่ยนรูปแบบการเชื่อมต่อและสามารถซ่อมแซมเครือข่ายด้วยตัวเองได้ แต่ด้วยแนวคิดที่ต้องการใช้โหนดจำนวนมากในการใช้งานดังนั้นจึงเป็นข้อจำกัดที่โหนดจะต้องมีขนาดเล็กและราคาถูกลง ส่งผลให้โหนดมีข้อจำกัดในเรื่องของแหล่งจ่ายพลังงาน หน่วยประมวลผลความเร็วต่ำ และมีหน่วยความจำน้อย

เนื่องจากเครือข่ายเซนเซอร์ไร้สายส่วนใหญ่จะประยุกต์ใช้ในพื้นที่ยานขนาดใหญ่และมีโหนดจำนวนมาก ทำให้โหนดโดยรอบโหนดปลายทางจะใช้พลังงานสูงเนื่องจากต้องทำหน้าที่ส่งผ่านข้อมูลให้โหนดอื่นในเครือข่าย ด้วยเหตุนี้จึงมีการนำเสนอให้มีการแบ่งเครือข่ายเป็นกลุ่มย่อย เพื่อให้ตัวแทนของกลุ่มหรือเรียกว่าคลัสเตอร์เฮด (Cluster head) ทำหน้าที่รวบรวมข้อมูลจากโหนดสมาชิกก่อนจะส่งต่อไปยังสถานีฐาน (Base station) ซึ่งกลุ่มย่อยที่ได้จากการจัดแบ่งเครือข่ายเรียกว่าคลัสเตอร์ (Cluster)

เมื่อเครือข่ายถูกแบ่งเป็นคลัสเตอร์จะทำให้โอเวอร์เฮดของการสื่อสารในเครือข่ายลดลง การใช้พลังงานโดยรวมจะมีประสิทธิภาพมากขึ้นเนื่องจากโหนดที่อยู่ใกล้สถานีฐานไม่ต้องสูญเสียพลังงานในการส่งผ่านข้อมูลของโหนดอื่น นอกจากนี้วิธีการจัดคลัสเตอร์จะช่วยให้ขนาดของเครือข่ายขยายออกไปได้ (Scalability) เนื่องจากการขยายขนาดของเครือข่ายไม่ได้ส่งผลต่อขนาดของเครือข่ายย่อย แต่เป็นการเพิ่มจำนวนเครือข่ายย่อยเท่านั้น ซึ่งแต่ละเครือข่ายย่อยยังสามารถทำงานได้โดยปกติ การจัดคลัสเตอร์ยังช่วยทำให้เครือข่ายมีประสิทธิภาพ เนื่องจากการเปลี่ยนแปลงที่เกิดขึ้นภายในคลัสเตอร์หนึ่งจะไม่กระทบต่อกิจกรรมภายในคลัสเตอร์อื่น

งานวิจัยนี้แนะนำเสนออัลกอริทึมการกระจายภาระงานอย่างสมดุลและการเลือกคลัสเตอร์เฮดแบบกริดเซกเตอร์ (Grid Sectoring) เนื่องจากการเลือกคลัสเตอร์ทั่วไปไม่ได้คำนึงถึงปริมาณภาระงานที่รับส่งภายในคลัสเตอร์ ทำให้แต่ละคลัสเตอร์มีอัตราการใช้พลังงานที่ไม่สมดุล งานวิจัยจะได้ทำการทดสอบประสิทธิภาพของการจัดคลัสเตอร์ด้วยโปรแกรมจำลองกราฟ

Networkx และทดสอบประสิทธิภาพการทำงานของอัลกอริทึมร่วมกับโปรโตคอล AODV ด้วยโปรแกรมจำลองเครือข่าย NS-2 บนสภาพของเครือข่ายที่มีการกระจายของโหนดแบบสม่ำเสมอและไม่สม่ำเสมอ

## 1.2 บทความและงานวิจัยที่เกี่ยวข้องกับประเด็นวิจัย

### 1.2.1 การเลือกคลัสเตอร์เฮดด้วยอัลกอริทึม LEACH และ LEACH-C

บทความ [1] ได้นำเสนอโปรโตคอล LEACH ที่ใช้อัลกอริทึมการจัดคลัสเตอร์ด้วยวิธีการเลือกคลัสเตอร์เฮดจากทฤษฎีความน่าจะเป็น โหนดที่เป็นคลัสเตอร์เฮดจะใช้พลังงานสูงกว่าโหนดทั่วไปทำให้หมดพลังงานก่อนโหนดอื่น ดังนั้นจึงมีการกำหนดให้ทุกโหนดมีการผลัดเปลี่ยนกันทำหน้าที่เป็นคลัสเตอร์เฮด ด้วยวิธีนี้ทำให้เกิดการกระจายการใช้พลังงานในเครือข่ายมากขึ้น LEACH แบ่งระยะการทำงานของเครือข่ายออกเป็น 2 ระยะ คือ ระยะเริ่มต้นและระยะคงตัว

ระยะเริ่มต้นจะเกิดขึ้นเป็นระยะแรกของการจัดคลัสเตอร์ใหม่ ในระยะนี้เครือข่ายจะมีการเลือกคลัสเตอร์เฮดเฉพาะโหนดที่ยังไม่เคยทำหน้าที่เป็นคลัสเตอร์เฮดมาก่อน แต่ละโหนดจะสุ่มหมายเลขและทำการเปรียบเทียบกับค่า Threshold ตามสมการที่ 1-1 ถ้าหากค่าที่โหนดสุ่มมีค่าสูงกว่าค่า threshold โหนดดังกล่าวจะเปลี่ยนสถานะเป็นคลัสเตอร์เฮด ในทางตรงกันข้ามถ้าค่าที่สุ่มได้ต่ำกว่าค่า threshold โหนดดังกล่าวจะเปลี่ยนสถานะเป็นโหนดสมาชิก

$$T(n) = \begin{cases} \frac{P}{1 - P \left( r \bmod \left[ \frac{1}{P} \right] \right)} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases} \quad (1-1)$$

เมื่อ  $T$  คือค่า threshold ที่คำนวณได้  $P$  คือความน่าจะเป็นของจำนวนคลัสเตอร์เฮดที่ต้องการ โดยคิดเป็นเปอร์เซ็นต์  $r$  คือลำดับที่ของรอบที่ทำการจัดคลัสเตอร์ใหม่  $n$  คือหมายเลขโหนด และ  $G$  คือเซตของโหนดที่ยังไม่เคยทำหน้าที่เป็นคลัสเตอร์เฮดมาก่อน

ระยะคงตัว เป็นการเก็บข้อมูลจากการตรวจวัดของโหนดสมาชิกและส่งต่อไปรวบรวมและประมวลผลที่คลัสเตอร์เฮด หลังจากนั้นข้อมูลจะถูกส่งต่อไปยังสถานีฐาน เมื่อระบบทำงานจนถึงระยะเวลาที่กำหนด เครือข่ายจะเข้าสู่ระยะเริ่มต้นเพื่อทำการเลือกคลัสเตอร์เฮดใหม่อีกครั้ง บทความนี้ได้เสนอแนะว่าจำนวนคลัสเตอร์เฮดที่เหมาะสมที่จะทำให้เครือข่ายมีการใช้พลังงานอย่างมีประสิทธิภาพคือ 5 เปอร์เซ็นต์ของจำนวนโหนดทั้งหมด โดยทดสอบในเครือข่ายที่มี 100 โหนดในพื้นที่ 100x100 ตารางเมตร

ต่อมาบทความ [2] ได้ทำการพัฒนา LEACH เป็น LEACH-C ที่มีการควบคุมการจัดคลัสเตอร์โดยสถานีฐาน ทำให้ประหยัดพลังงานในการประมวลผลของเครือข่าย ซึ่งสถานีฐานจะรับข้อมูลตำแหน่งและข้อมูลพลังงานของทุกโหนดในเครือข่ายเพื่อมาคำนวณค่าพลังงานเฉลี่ย จากนั้นจะพิจารณาให้โหนดที่มีพลังงานที่เหลือมากที่สุดเป็นคลัสเตอร์เฮด การเลือกคลัสเตอร์เฮดด้วยวิธีนี้ยังมีการคำนวณตำแหน่งคลัสเตอร์เฮดที่เหมาะสมด้วยการคัดเลือกคลัสเตอร์เฮดที่อยู่ตรงกลางของคลัสเตอร์ ซึ่งจะช่วยลดจำนวนฮอปในการสื่อสาร ส่งผลให้ประหยัดพลังงานลงได้

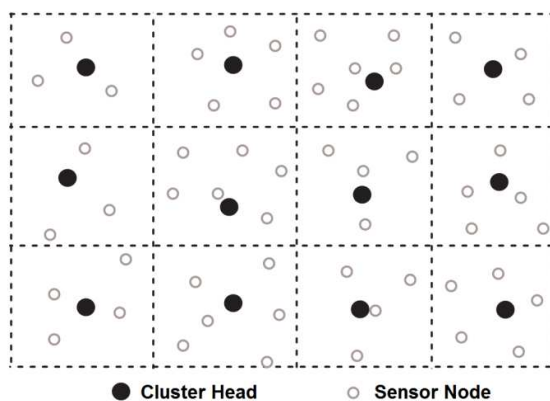
### 1.2.2 การเลือกคลัสเตอร์เฮดด้วยอัลกอริทึม ACHS และ EACHS

บทความ [3, 4] นำเสนอโพรโทคอล ACHS (Adaptive Cluster Head Selection) และ EACHS (Extended ACHS) ตามลำดับ ซึ่งใช้อัลกอริทึมในการเลือกคลัสเตอร์เฮดที่ได้รับการปรับปรุงมาจาก LEACH โดยเพิ่มกระบวนการเลือกซ้ำ (Reselection) เพื่อกำหนดหาคลัสเตอร์เฮดใหม่ที่ทำให้มีความสมดุลของจำนวนสมาชิกในแต่ละคลัสเตอร์มากขึ้น เพราะเมื่อสมาชิกในแต่ละคลัสเตอร์มีจำนวนใกล้เคียงกัน จะทำให้ปริมาณภาระงานภายในคลัสเตอร์ใกล้เคียงกันด้วย ดังนั้นการใช้พลังงานในแต่ละคลัสเตอร์จึงมีความใกล้เคียงกัน นอกจากนี้วิธีการดังกล่าวยังสามารถทำให้อายุ (Lifetime) ของแต่ละคลัสเตอร์มีความใกล้เคียงกัน อัลกอริทึม EACHS ได้รับการปรับปรุงจากอัลกอริทึม ACHS ให้มีการใช้พลังงานอย่างมีประสิทธิภาพมากขึ้น การเลือกคลัสเตอร์เฮดด้วยอัลกอริทึม ACHS และ EACHS นอกจากจะมีความซับซ้อนของกระบวนการน้อยและมีความจำนวนโหนดสมาชิกในแต่ละคลัสเตอร์ที่สมดุลกันแล้ว ผลการเลือกคลัสเตอร์เฮดด้วยวิธีนี้ยังสามารถหลีกเลี่ยงปัญหาที่เกิดจากคลัสเตอร์เฮดอยู่ติดกัน ซึ่งเป็นสาเหตุให้ตำแหน่งของคลัสเตอร์เฮดอยู่บริเวณขอบของคลัสเตอร์หลังจากแบ่งพื้นที่เรียบร้อยแล้ว

### 1.2.3 อัลกอริทึมการจัดคลัสเตอร์แบบกริด

การจัดคลัสเตอร์แบบกริด (Grid clustering) [5-7] เป็นการแบ่งพื้นที่สนใจออกเป็นสี่เหลี่ยมจัตุรัสย่อยขนาดเท่ากันที่เรียกว่ากริด แต่ละกริดคือขอบเขตของคลัสเตอร์ดังกล่าวประกอบที่ 1-1 การแบ่งพื้นที่เป็นกริดสามารถสร้างโครงสร้างเครือข่ายได้สะดวก ลดความซับซ้อนในกระบวนการจัดคลัสเตอร์และยังทำให้การประสานงานของเครือข่ายดีขึ้น การจัดคลัสเตอร์ด้วยวิธีนี้สามารถกระจายคลัสเตอร์เฮดและจำนวนโหนดสมาชิกในแต่ละคลัสเตอร์ได้อย่างสมดุล ทำให้ปริมาณภาระงานและการใช้พลังงานในทุกคลัสเตอร์สมดุลกัน การเลือกคลัสเตอร์เฮดจะพิจารณาจากพลังงานที่เหลือของโหนดที่มากที่สุด บทความ [6, 7] ยังพบว่าวิธีการจัดคลัสเตอร์แบบกริดสามารถสมดุลการใช้พลังงานได้ดีกว่าอัลกอริทึมอื่นและสามารถยืดอายุของเครือข่ายได้มากขึ้น

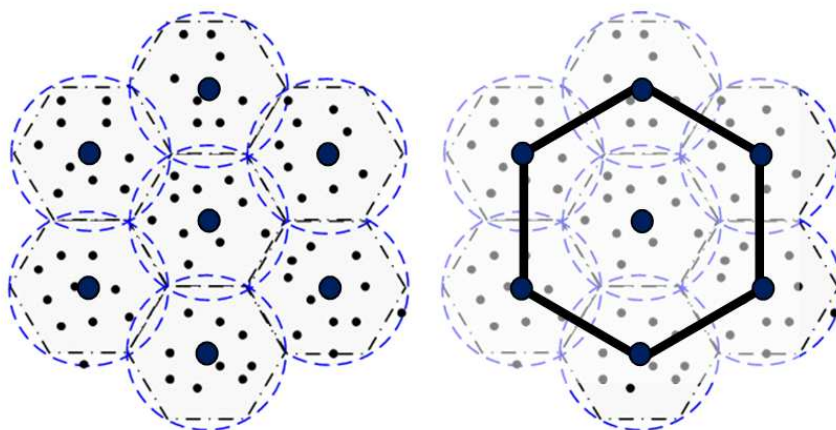
บทความที่ [5] ได้นำเสนอขนาดกริดที่เหมาะสมซึ่งสอดคล้องกับระยะเวลาสื่อสารของโหนด นอกจากนี้บทความ [8] ได้นำเสนอวิธีจัดเส้นทางระหว่างกริดไปยังสถานีฐานโดยใช้ประโยชน์จากข้อมูลตำแหน่งของโหนดในการจัดเส้นทาง



ภาพประกอบ 1-1 การจัดคลัสเตอร์แบบกริด

#### 1.2.4 อัลกอริทึมการจัดคลัสเตอร์แบบหกเหลี่ยม

การจัดคลัสเตอร์แบบหกเหลี่ยม (Hexagonal clustering) [9, 10] เป็นการจัดคลัสเตอร์ให้มีพื้นที่ในลักษณะหกเหลี่ยม การจัดคลัสเตอร์ด้วยวิธีนี้สามารถกระจายคลัสเตอร์ได้อย่างสม่ำเสมอทั้งเครือข่าย และแต่ละคลัสเตอร์ยังมีพื้นที่เท่ากัน จึงมีการกระจายภาระงานและการใช้พลังงานอย่างสมดุล การจัดคลัสเตอร์ด้วยวิธีนี้จะใช้ระยะส่งสัญญาณวิทยุของโหนดมาพิจารณาขนาดและขอบเขตของคลัสเตอร์ นอกจากนี้การจัดคลัสเตอร์แบบหกเหลี่ยมยังสามารถลดการเหลื่อมล้ำของขอบเขตคลัสเตอร์ ทำให้ลดการรบกวนกันของสัญญาณวิทยุ รูปแบบการจัดคลัสเตอร์แบบหกเหลี่ยมแสดงดังภาพประกอบที่ 1-2



ภาพประกอบ 1-2 การจัดคลัสเตอร์แบบหกเหลี่ยม



### 1.3 ประเด็นวิจัย

งานวิจัยนี้เป็นการออกแบบและพัฒนาอัลกอริทึมในการเลือกคลัสเตอร์เฮดสำหรับเครือข่ายเซนเซอร์ไร้สายขนาดใหญ่เพื่อสมมูลภาระงานและการใช้พลังงานของคลัสเตอร์บนเครือข่ายที่มีการกระจายตัวของโหนดแบบสม่ำเสมอและไม่สม่ำเสมอ โดยดำเนินการทดสอบด้วยโปรแกรมจำลองกราฟ Networkx และจำลองการทำงานร่วมกับโพรโทคอล AODV เพื่อทดสอบประสิทธิภาพบนโปรแกรมจำลองเครือข่าย NS-2 ในประเด็นความสมมูลของปริมาณภาระงานและการใช้พลังงานในแต่ละคลัสเตอร์

### 1.4 วัตถุประสงค์การวิจัย

ออกแบบ พัฒนาและนำเสนออัลกอริทึมในการเลือกคลัสเตอร์เฮดสำหรับเครือข่ายเซนเซอร์ไร้สาย โดยมีเป้าหมายหลักดังนี้

1. เพื่อออกแบบ และพัฒนาอัลกอริทึมในการเลือกคลัสเตอร์เฮดสำหรับเครือข่ายเซนเซอร์ไร้สาย ให้มีการกระจายภาระงานและการใช้พลังงานอย่างสมมูลในแต่ละคลัสเตอร์
2. ผลของการเลือกคลัสเตอร์เฮดสามารถสนับสนุนและนำมาใช้งานร่วมกับโพรโทคอล AODV ได้อย่างประหยัดพลังงานมากยิ่งขึ้น

### 1.5 ขอบเขตของการวิจัย

1. เครือข่ายเซนเซอร์ไร้สายที่สนใจศึกษามีโหนดในเครือข่ายกระจายตัวด้วยความหนาแน่นสม่ำเสมอและไม่สม่ำเสมอ
2. ดำเนินการวิจัยด้วยโปรแกรมการจำลองกราฟ Networkx และโปรแกรมจำลองเครือข่าย NS-2 (Network Simulator 2)
3. เครือข่ายที่ใช้ทดสอบประสิทธิภาพไม่คำนึงถึงสถานการณ์ เครือข่ายจึงประกอบด้วยโหนดและคลัสเตอร์เฮดเท่านั้น และทุกโหนดในเครือข่ายไม่มีการเคลื่อนที่
4. การดำเนินการวิจัยการจัดคลัสเตอร์เฮดด้วยอัลกอริทึมที่นำเสนอไม่ได้ครอบคลุมถึงเรื่อง การพิจารณาเลือกคลัสเตอร์เฮดใหม่

## 1.6 ขั้นตอนและวิธีดำเนินงานวิจัย

ขั้นที่ 1: การศึกษาก่อนดำเนินงานวิจัย

ขั้นที่ 1.1: ศึกษาผลการเลือกคลัสเตอร์เสดด้วยอัลกอริทึมที่ใช้ทฤษฎีความน่าจะเป็น ได้แก่ LEACH ACHS และ EACHS ศึกษาและวิเคราะห์วิธีการจัดคลัสเตอร์แบบกริดและการจัดคลัสเตอร์แบบหกเหลี่ยม

ขั้นที่ 1.2: ศึกษาประสิทธิภาพของเครือข่ายที่ทำงานร่วมกับโพรโทคอล AODV บนมาตรฐาน IEEE 802.15.4 โดยการจำลองเครือข่ายด้วยโปรแกรม NS-2

ขั้นที่ 2: ออกแบบ ปรับปรุง พัฒนา และนำเสนออัลกอริทึมในการเลือกคลัสเตอร์เสด เพื่อให้มีลักษณะเฉพาะตามวัตถุประสงค์โครงการ

ขั้นที่ 3: ทดสอบสมรรถนะของเครือข่ายที่ทำงานร่วมกับอัลกอริทึมในการจัดคลัสเตอร์ที่นำเสนอด้วยโปรแกรมจำลองเครือข่าย NS-2 โดยแบ่งขั้นตอนการทดสอบดังนี้

ขั้นที่ 3.1: ทดสอบเครือข่ายด้วยการจำลองกราฟเมื่อ โหนดมีการกระจายตัวด้วยความหนาแน่นสม่ำเสมอและไม่สม่ำเสมอ

ขั้นที่ 3.2: ทดสอบด้วยการจำลองเครือข่ายเมื่อ โหนดมีการกระจายตัวด้วยความหนาแน่นสม่ำเสมอและไม่สม่ำเสมอ

ขั้นที่ 4: ปรับปรุงอัลกอริทึมในการเลือกคลัสเตอร์เสดที่นำเสนอให้สามารถใช้งานร่วมกับโพรโทคอลการจัดเส้นทาง AODV ให้มีการใช้พลังงานอย่างมีประสิทธิภาพ

ขั้นที่ 5: จัดทำรายงานฉบับสมบูรณ์

## 1.7 โครงสร้างของรายงานวิทยานิพนธ์

ในบทที่ 2 จะกล่าวถึงทฤษฎีและหลักการที่จำเป็นเกี่ยวกับงานวิจัยขั้นนี้ บทที่ 3 เป็นผลการเลือกคลัสเตอร์เสดด้วยอัลกอริทึมที่ใช้ทฤษฎีความน่าจะเป็น การศึกษาอัลกอริทึมการจัดคลัสเตอร์เสดแบบกริดและแบบหกเหลี่ยม ตลอดจนการศึกษสมรรถนะของเครือข่ายที่ทำงานร่วมกับโพรโทคอล AODV บนมาตรฐาน IEEE 802.15.4 บทที่ 4 จะกล่าวถึงรายละเอียดการออกแบบ ปรับปรุงและพัฒนาอัลกอริทึมในการเลือกคลัสเตอร์เสดเพื่อให้เกิดสมมูลภาระงาน บทที่ 5 เป็นการวิเคราะห์ผลที่ได้จากการทดสอบโพรโทคอล AODV ที่ทำงานร่วมกับอัลกอริทึมที่ได้นำเสนอ และบทสุดท้ายจะกล่าวสรุปผลที่ได้จากการวิจัยพร้อมทั้งข้อเสนอแนะเกี่ยวกับงานวิจัยนี้

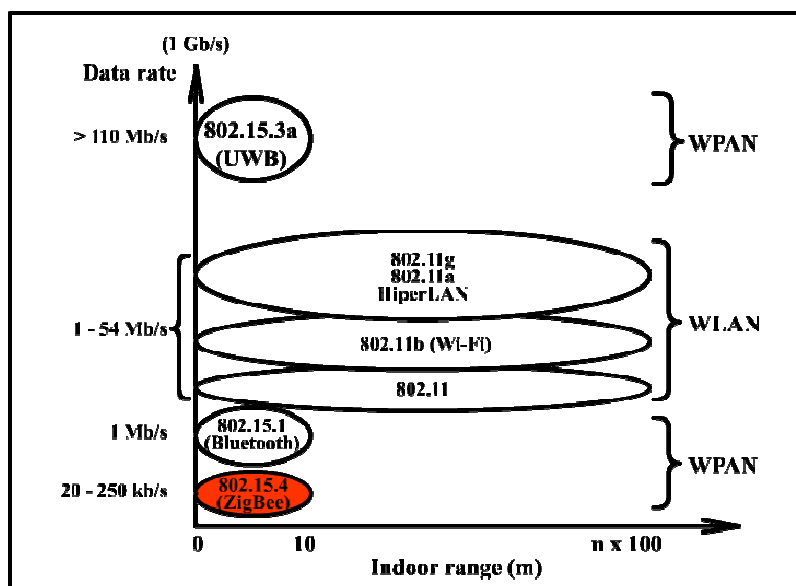
## บทที่ 2

### ทฤษฎีและหลักการ

ในบทนี้กล่าวถึงทฤษฎีและหลักการที่เกี่ยวข้องกับงานวิจัยนี้ ซึ่งประกอบด้วย โพรโทคอลมาตรฐาน IEEE 802.15.4 การจัดเครือข่ายแบบคลัสเตอร์ โพรโทคอลการจัดเส้นทาง ชนิด AODV แบบดั้งเดิม AODV แบบ Clustering และการจัดคลัสเตอร์ด้วยอัลกอริทึม ACHS และ EACHS

#### 2.1 โพรโทคอลมาตรฐาน IEEE 802.15.4

โพรโทคอลมาตรฐาน IEEE 802.15.4 [11] เป็นโพรโทคอลสำหรับเครือข่ายส่วนบุคคลที่มีอัตราการรับส่งข้อมูลต่ำ (Low Rate Wireless Personal Area Network: LR-WPAN) มีต้นทุนต่ำ (Low cost) มีการใช้พลังงานน้อย (Low power) และมีระยะการสื่อสารสั้น (Short range communication) สามารถเปรียบเทียบมาตรฐานนี้กับมาตรฐานเครือข่ายไร้สายอื่นๆ ได้ดังภาพประกอบที่ 2-1 [12] โพรโทคอลมาตรฐาน IEEE 802.15.4 มีการกำหนดกลไกการเข้าใช้ช่องสัญญาณ (Channel access mechanism) การส่งเฟรมตอบรับ (Acknowledge frame delivery) การเชื่อมต่อและการยกเลิกการเชื่อมต่อ (Network association and disassociation) เป็นต้น มาตรฐานนี้มีการสนับสนุน DSSS (Direct Sequence Spread Spectrum) ในชั้นกายภาพ (Physical layer) 2 แบบ คือ ที่ย่านความถี่ต่ำ (Low band) จะทำงานด้วยความถี่ 868 และ 915 เมกะเฮิร์ต ด้วยอัตราการส่งข้อมูล 20 และ 40 กิโลบิตต่อวินาทีตามลำดับ และที่ย่านความถี่สูง (High band) จะทำงานด้วยความถี่ 2.4 กิกะเฮิร์ต ด้วยอัตราการส่งข้อมูล 250 กิโลบิตต่อวินาที โดยที่ DSSS เป็นวิธีระบุมความถี่ที่ใช้บนคลื่นพาหะเพื่อให้เครือข่ายสามารถรับส่งข้อมูลได้มากกว่าแบบย่านความถี่แคบ (Narrow band) และเพื่อความเหมาะสมสำหรับสภาพแวดล้อมที่มีการแทรกสอดจากคลื่นวิทยุชนิดอื่นสูง ที่ย่านความถี่ 2.4 กิกะเฮิร์ตจะเป็นช่วงความถี่สาธารณะที่สามารถใช้งานได้อย่างแพร่หลาย



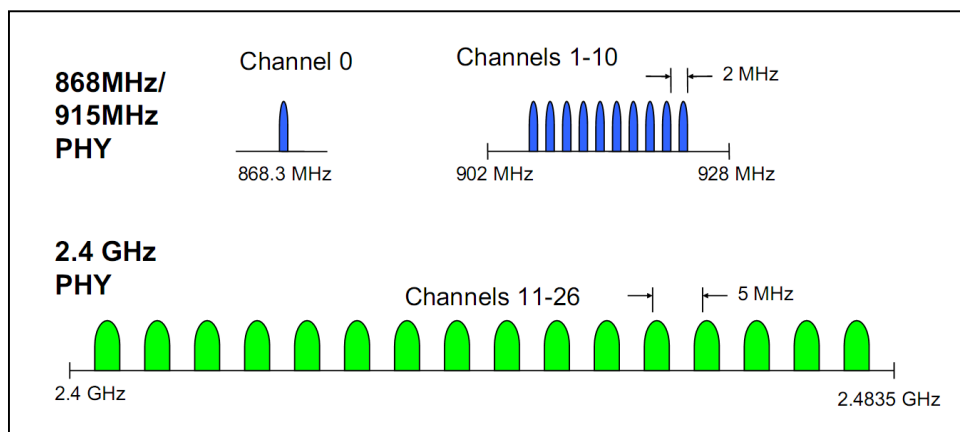
ภาพประกอบ 2-1 มาตรฐานเครือข่ายไร้สาย

### 2.1.1 ชั้นกายภาพมาตรฐาน IEEE 802.15.4 (Physical Layer)

ชั้นกายภาพ (Physical Layer: PHY) เป็นชั้นที่นำเสนออินเตอร์เฟซการเชื่อมต่อระหว่างชั้นการควบคุมการเข้าใช้สื่อกลาง (Media Access Control: MAC) และช่องสัญญาณวิทยุทางกายภาพ (Physical radio channel) ในชั้นนี้จะรับผิดชอบการรับส่งข้อมูลโดยใช้ช่องสัญญาณวิทยุที่แน่นอน ชั้นกายภาพมีการทำงาน 2 แบบคือ บริการข้อมูล (Physical data service) และบริการการจัดการ (Physical management service)

#### 2.1.1.1 การมอดูเลชันและย่านความถี่ที่ใช้

มาตรฐาน IEEE 802.15.4 อนุญาตให้ใช้ความถี่ 3 ย่านความถี่ คือ 868, 915 และ 2400 MHz ดังภาพประกอบที่ 2-2 [13] ในแต่ละย่านความถี่จะมีวิธีการมอดูเลชัน (Modulation) และวิธีการ Spread spectrum ที่ต่างกัน ทำให้อัตราการรับส่งข้อมูลในแต่ละย่านต่างกัน ที่ย่านความถี่ 868 และ 915 MHz จะมีช่องสัญญาณให้ใช้ 1 และ 10 ช่องสัญญาณตามลำดับ ที่ย่านความถี่นี้จะใช้วิธีการ DSSS ในชั้นกายภาพอย่างง่ายและมีวิธีการมอดูเลชันแบบ BPSK (Binary Phase Shift Keying) ส่วนย่านความถี่ 2.4 GHz จะมีช่องสัญญาณให้ใช้ 16 ช่องสัญญาณและมีวิธีการมอดูเลชันแบบ OQPSK (Offset Quadrature Phase Shift Keying)



ภาพประกอบ 2-2 ย่านความถี่ชั้นกายภาพมาตรฐาน IEEE 802.15.4

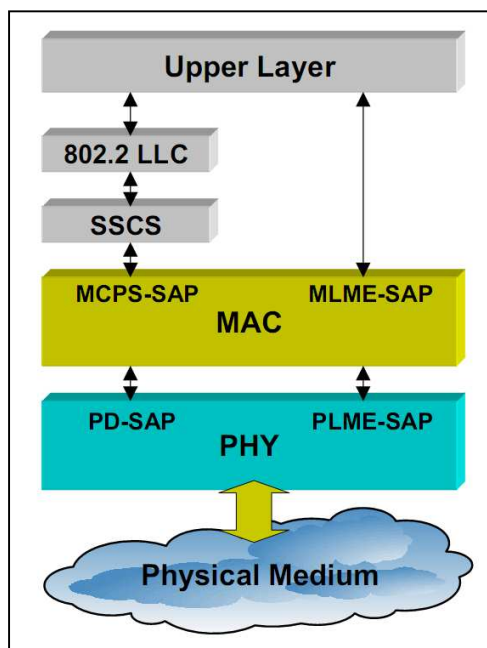
#### 2.1.1.1 ฟังก์ชันการทำงานในชั้นกายภาพมาตรฐาน IEEE 802.15.4

- เปิดปิดภาครับส่งสัญญาณวิทยุ (Activation and deactivation of the radio transceiver) เป็นฟังก์ชันที่ใช้ควบคุมสัญญาณวิทยุให้มีสถานะใดสถานะหนึ่งจาก 3 สถานะได้แก่ รับ ส่ง และสถานะหลับ (Sleeping) ฟังก์ชันชนิดนี้จะทำงานก็ต่อเมื่อมีการร้องขอจากชั้น MAC
- การตรวจวัดพลังงาน (Energy Detection: ED) เป็นฟังก์ชันที่ใช้ประมาณกำลังส่งสัญญาณวิทยุที่รับได้ภายในแบนด์วิดท์ของมาตรฐาน IEEE 802.15.4 ฟังก์ชันนี้ไม่ได้มีการถอดรหัสบนช่องสัญญาณ การวัดระดับกำลังส่งของสัญญาณวิทยุที่รับได้จะใช้สนับสนุนอัลกอริทึมชั้นเครือข่าย เช่น CCA (Clear Channel Assessment) เพื่อประเมินว่าช่องสัญญาณว่างหรือไม่
- การชี้วัดคุณภาพการเชื่อมต่อ (Link Quality Indication: LQI) เป็นฟังก์ชันที่ใช้วัดความแรง (Strength) และคุณภาพของสัญญาณที่ได้รับ โดยการใช้ภาครับ ED (Receiver ED) หรืออาจจะใช้วิธีประมาณค่าสัญญาณข้อมูลต่อสัญญาณรบกวน (Signal to noise ratio) หรืออาจจะใช้ทั้งสองวิธีร่วมกัน
- การประเมินช่องสัญญาณว่าง (Clear Channel Assessment: CCA) เป็นฟังก์ชันที่ใช้ประเมินกิจกรรมของสื่อกลางว่ามีสถานะว่างหรือไม่ ฟังก์ชันนี้มีการทำงาน 3 โหมด คือ Energy Detection Mode ซึ่งเป็นโหมดที่ CCA ใช้รายงานชั้นที่อยู่สูงกว่าว่าสื่อไม่ว่างถ้าค่า ED ที่วัดได้เกินกว่าค่าที่กำหนดไว้ ต่อมาคือ Carrier Sense Mode ซึ่งเป็นโหมดที่ CCA ใช้รายงานว่าสื่อไม่ว่างถ้าการทำมอดูเลชันและ Spread spectrum ของมาตรฐาน IEEE 802.15.4 สูงกว่าหรือต่ำกว่าค่าที่กำหนด โหมดสุดท้ายคือ Carrier Sense with Energy Detection Mode ซึ่งเป็นโหมดที่รวมวิธีการพิจารณาจากทั้งสองโหมดที่กล่าวมา

- การเลือกความถี่ช่องสัญญาณ (Channel Frequency Selection) เป็นฟังก์ชันที่ใช้เลือกช่องสัญญาณเพื่อใช้สื่อสารเพียง 1 ช่องจาก 27 ช่องสัญญาณของมาตรฐานนี้ให้แก่ภาครับส่งสัญญาณวิทยุ โดยฟังก์ชันนี้จะทำงานก็ต่อเมื่อมีการร้องขอจากชั้นที่สูงกว่า

### 2.1.2 ชั้นย่อยการควบคุมการเข้าใช้สื่อกลางมาตรฐาน IEEE 802.15.4

ชั้นย่อยการควบคุมการเข้าใช้สื่อกลาง (MAC sublayer) เป็นชั้นที่นำเสนออินเตอร์เฟซการเชื่อมต่อระหว่างชั้นย่อย SSCS (Service Specific Convergence Sublayer) และชั้นกายภาพ [14] ดังภาพประกอบที่ 2-3 ในชั้นนี้มีบริการ 2 ชนิดเช่นเดียวกับชั้นกายภาพคือ MAC data unit และ MAC management unit ชั้นย่อย MAC จะทำหน้าที่อนุญาตว่าสื่อกลางควรได้รับการเข้าใช้งานอย่างไรจากอุปกรณ์ ชั้นที่อยู่สูงกว่าชั้นนี้คือชั้น IEEE 802.2 LLC (Logical Link Control) ซึ่งเป็นชั้นที่รับผิดชอบเกี่ยวกับลอจิกัลแอดเดรส การควบคุมความผิดพลาด และเป็นชั้นที่สามารถเข้าถึงชั้น MAC ผ่านชั้น SSCS สำหรับชั้น 802.2 LLC นี้ได้รับการออกแบบมาเพื่อสนับสนุนการควบคุมงานประเภทอุตสาหกรรมและอุปกรณ์ภายในบ้านจำนวนมาก ซึ่งงานประเภทนี้ต้องการอัตราการส่งข้อมูลต่ำและมีการประกันความล่าช้าในการส่ง (Delay) ที่ยืดหยุ่นได้

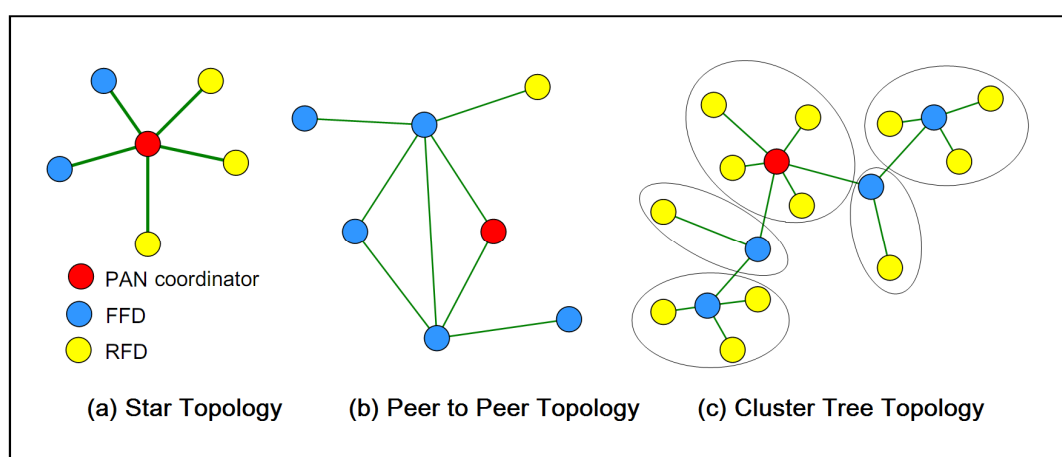


ภาพประกอบ 2-3 โครงสร้างชั้นกายภาพและชั้นย่อย MAC มาตรฐาน IEEE 802.15.4

### 2.1.2.1 ประเภทอุปกรณ์ในเครือข่ายและโทโพโลยีของ IEEE 802.15.4

- PAN Coordinator เป็นโหนดที่ใช้เริ่มต้นสร้างเครือข่ายและยังเป็นโหนดควบคุม (Controller) อันดับแรกๆ ของเครือข่าย โหนดชนิดนี้สามารถทำหน้าที่เป็นเกตเวย์เพื่อเชื่อมต่อไปยังเครือข่ายอื่น ในแต่ละ PAN (Personal Area Network) จะประกอบไปด้วย PAN coordinator ได้เพียง 1 โหนดเท่านั้น โหนดชนิดนี้สามารถส่งเบคอน (Beacon) และสื่อสารไปยังอุปกรณ์ในเครือข่ายได้โดยตรงภายในระยะเวลาการสื่อสารของมันเอง นอกจากนี้ PAN coordinator ยังทำหน้าที่เสมือนแหล่งจัดเก็บข้อมูลเครือข่าย (Repository)
- Coordinator เป็นโหนดประเภท FFD (Full Function Device) ที่มีความสามารถและฟังก์ชันการทำงานมากกว่าโหนดปกติ คือ มีการสนับสนุนข้อมูลของฟังก์ชันจัดเส้นทาง โหนดชนิดนี้ทำหน้าที่เป็นโหนดกลางการส่งต่อข้อมูลภายในเครือข่าย หรืออาจกล่าวสรุปได้ว่าโหนดชนิดนี้เป็นเราเตอร์ที่ใช้สื่อสารระหว่างเราเตอร์อื่นกับอุปกรณ์ในเครือข่าย
- End device เป็นโหนดประเภท RFD (Reduced Function Device) ที่มีความสามารถและฟังก์ชันการทำงานจำกัด มีฟังก์ชันเพียงพอสำหรับการสื่อสารโดยตรงไปยัง PAN coordinator หรือ Coordinator เท่านั้น ไม่มีความสามารถที่จะส่งต่อข้อมูลระหว่างอุปกรณ์

ชั้นย่อย MAC บนมาตรฐาน IEEE 802.15.4 สนับสนุนโทโพโลยี 2 ชนิด คือ โทโพโลยีแบบดาว (Star topology) และโทโพโลยีแบบเพียร์ทูเพียร์ นอกจากนี้ยังสามารถรวมโทโพโลยีทั้งสองชนิดเข้าด้วยกันเรียกว่า โทโพโลยีแบบคลัสเตอร์ทรี (Cluster tree topology) ดังภาพประกอบที่ 2-4 [13]



ภาพประกอบ 2-4 โทโพโลยีที่ชั้นย่อย MAC สนับสนุนบนมาตรฐาน IEEE 802.15.4

- โทโพโลยีแบบดาว ดังภาพประกอบที่ 2-4 (a) เป็นรูปแบบโทโพโลยีพื้นฐาน อุปกรณ์ทุกตัวหรือโหนดทุกตัวในเครือข่ายจะสื่อสารโดยตรงกับ PAN coordinator เท่านั้น
- โทโพโลยีแบบเพียร์ทูเพียร์ ดังภาพประกอบที่ 2-4 (b) อุปกรณ์ในเครือข่ายสามารถเชื่อมต่อสื่อสารซึ่งกันและกันได้ถ้าหากอยู่ในระยะสื่อสาร อุปกรณ์ประเภท FFD สามารถทำงานเสมือน PAN coordinator และยังทำหน้าที่ในการส่งต่อข้อมูล ส่วนอุปกรณ์ประเภท RFD จะเป็นส่วนหนึ่งของเครือข่ายด้วยการเชื่อมต่อโดยตรงกับ FFD หรือ PAN coordinator
- โทโพโลยีแบบคลัสเตอร์ทรี ดังภาพประกอบที่ 2-4 (c) เป็นอีกรูปแบบหนึ่งที่ได้จากการปรับเปลี่ยนรูปแบบของโทโพโลยีแบบเพียร์ทูเพียร์ โทโพโลยีแบบคลัสเตอร์ทรีจะเริ่มสร้างเครือข่ายด้วยอุปกรณ์ PAN coordinator จากนั้นอุปกรณ์ประเภท FFD จะสร้างเครือข่ายย่อยที่มีโทโพโลยีแบบดาวโดยมีลูกข่ายเป็นอุปกรณ์ RFD โทโพโลยีชนิดนี้จะส่งต่อข้อมูลผ่านอุปกรณ์ FFD ทั้งหมด

#### 2.1.2.2 ฟังก์ชันในชั้น MAC ของมาตรฐาน IEEE 802.15.4

- สร้างเบคอนเครือข่าย โคออดิเนเตอร์จะตัดสินใจว่าจะทำงานในโหมดเบคอนหรือไม่ (Beacon enabled mode) โดยการใช้ซูเปอร์เฟรม (Superframe) ปกติซูเปอร์เฟรมจะเต็มไปด้วยเบคอนเครือข่ายที่แบ่งเป็น 16 ช่อง (slot) เท่ากัน ประกอบด้วยคาบแอคทีฟและคาบอินแอคทีฟ (Active and inactive period) คาบแอคทีฟจะมีการแบ่งย่อยเป็น 2 ส่วนคือ Contention Access Period (CAP) เพื่อขอเข้าใช้สื่อกลางด้วยอัลกอริทึม CSMA-CA และ Contention Free Period (CFP) ที่มีกลไก Guaranteed Time Slot (GTS) ในการประกันความน่าเชื่อถือของการสื่อสาร ส่วนคาบอินแอคทีฟ โคออดิเนเตอร์จะไม่สามารถโต้ตอบกับเครือข่ายและใช้โหมดประหยัดพลังงานได้ โดยปกติโคออดิเนเตอร์จะส่งเบคอนด้วยคาบเวลาที่กำหนดเพื่อเทียบจังหวะสัญญาณนาฬิกาให้ตรงกัน (Synchronization)
- การเทียบจังหวะสัญญาณนาฬิกาให้ตรงกับเบคอน (Synchronizing to the beacons) อุปกรณ์ที่เชื่อมต่อกับโคออดิเนเตอร์จะทำงานตามเบคอนที่ได้ซิงโครไนซ์กับโคออดิเนเตอร์ การทำซิงโครไนซ์จะมีความสำคัญในกระบวนการทำโพลลิง (Polling) ซึ่งเป็นรูปแบบการทำงานตามคิว
- สนับสนุนการเชื่อมต่อและยกเลิกการเชื่อมต่อเครือข่ายส่วนบุคคล (PAN) ชั้น MAC จะมีฟังก์ชันการเชื่อมต่อและยกเลิกการเชื่อมต่อสำหรับการเริ่มระบบเครือข่ายด้วยตัวเอง (Self-configuration) โคออดิเนเตอร์จะส่งเบคอนเป็นระยะตามที่ได้รับคำสั่งจาก PAN coordinator สำหรับกระบวนการขอเชื่อมต่อ อุปกรณ์จะส่งข้อความขอเชื่อมต่อไปยัง PAN



coordinator ซึ่งจะได้รับข้อความตอบกลับขึ้นอยู่กับว่าทรัพยากรมีเพียงพอหรือไม่ ส่วนการยกเลิกการเชื่อมต่อนั้นจะทำได้โดยอุปกรณ์เองหรือทำโดย PAN coordinator

- ฟังก์ชันการขอเข้าใช้สื่อกลางด้วยอัลกอริทึม CSMA-CA อัลกอริทึมการขอเข้าใช้สื่อกลางชนิดนี้เหมือนกับอัลกอริทึมการขอเข้าใช้สื่อกลางสำหรับเครือข่ายไร้สายอื่น แต่จะไม่มีกลไก Request-To-Send (RTS) และ Clear-To-Send (CTS) ที่ใช้พิจารณาว่าอุปกรณ์จะทำงานแบบ Slotted CSMA-CA หรือ Unslotted CSMA-CA ซึ่งขึ้นอยู่กับว่า PAN coordinator มีการทำงานในโหมดเบคอนหรือไม่ตามลำดับ

## 2.2 การจัดเครือข่ายแบบคลัสเตอร์

การจัดเครือข่ายแบบคลัสเตอร์ (Clustering) เป็นโครงสร้างแบบลำดับชั้น (Hierarchical structure) เพื่อให้โหนดในเครือข่ายสามารถใช้พลังงานอย่างมีประสิทธิภาพ การจัดคลัสเตอร์ช่วยให้จัดการเครือข่ายได้สะดวกขึ้น ทำให้เครือข่ายประกอบขึ้นด้วยโหนดที่มีสถาปัตยกรรมและความสามารถที่แตกต่างกันได้ เช่น ระยะเวลาสื่อสาร แหล่งพลังงาน หรือเซนเซอร์ที่แตกต่างกัน ชนิดของโหนดในเครือข่ายประกอบด้วย [15]

- **คลัสเตอร์เฮด (cluster head)** ทำหน้าที่ควบคุมดูแลเกือบทุกกิจกรรมทั้งภายในและภายนอกคลัสเตอร์ คลัสเตอร์เฮดอาจถูกเลือกโดยสถานีฐาน หรือโดยวิธีการร่วมกันพิจารณาของโหนดในเครือข่าย หรืออาจตัดสินใจด้วยตัวเองของโหนด
- **เกตเวย์ (gateways)** ทำหน้าที่ในการส่งต่อข้อมูลจากคลัสเตอร์หนึ่งไปยังคลัสเตอร์อื่นหรือไปยังสถานีฐาน เกตเวย์ส่วนใหญ่จะอยู่บริเวณขอบของคลัสเตอร์ เกตเวย์มีความสำคัญมาก เพราะถ้าหากโหนดชนิดนี้สูญหายจะทำให้สูญเสียเส้นทางที่ใช้สื่อสารระหว่างคลัสเตอร์ การทำหน้าที่เกตเวย์ถูกกำหนดโดยโปรโตคอลจัดเส้นทาง
- **โหนดสมาชิก (member nodes)** เป็นโหนดทั่วไปอยู่ภายในคลัสเตอร์ ทำหน้าที่ตรวจวัดสิ่งแวดล้อมตามชนิดของเซนเซอร์ที่ติดตั้ง และส่งข้อมูลไปยังคลัสเตอร์เฮด

การจัดคลัสเตอร์จะแบ่งออกเป็น 2 ระยะ ได้แก่ ระยะเริ่มระบบ (Set-up phase) และระยะเสถียรภาพ (Steady state phase) ระยะเริ่มระบบซึ่งเป็นระยะที่มีการสร้างคลัสเตอร์ในเครือข่ายจนเสร็จสมบูรณ์ จากนั้นจะเข้าสู่ระยะเสถียรภาพ ซึ่งเป็นระยะที่มีการจัดเส้นทาง การซ่อมแซมโทโพโลยีเส้นทาง ระยะนี้จะสิ้นสุดเมื่อเครือข่ายไม่สามารถรักษาสภาพการทำงานให้เป็นปกติได้ หรือเมื่อถึงเวลาการจัดคลัสเตอร์ใหม่ จากนั้นจะวนเข้าสู่ระยะเริ่มระบบใหม่อีกครั้ง

## 2.3 โพรโทคอลการจัดเส้นทาง AODV

### 2.3.1 โพรโทคอลการจัดเส้นทาง AODV แบบดั้งเดิม

AODV [16] เป็นโพรโทคอลจัดเส้นทางชนิด reactive คือจะทำงานเมื่อโหนดต้องการค้นหาเส้นทางใหม่หรือทำการปรับปรุงเส้นทางเดิม ซึ่งเรียกการทำงานแบบนี้ว่า On-demand โพรโทคอลจัดเส้นทางชนิดนี้แบ่งการทำงานออกเป็น 2 ระยะด้วยกันคือ

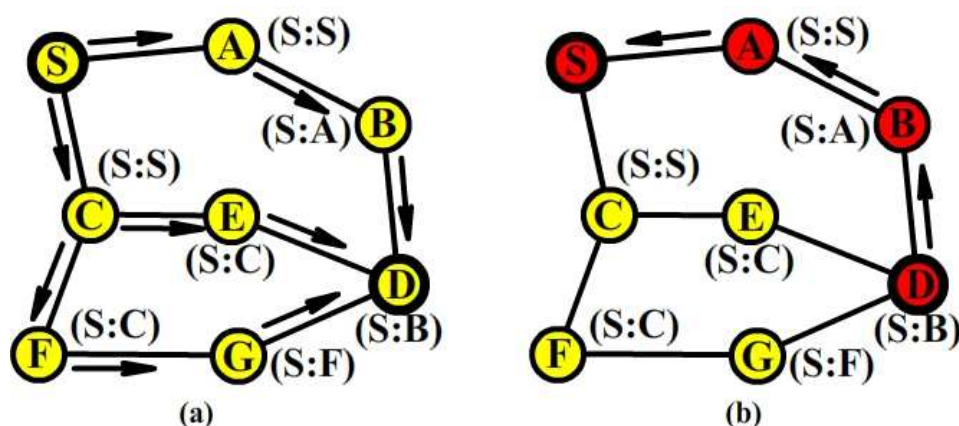
1. ระยะเริ่มจัดเส้นทาง ระยะนี้เริ่มต้นเมื่อโหนดในเครือข่ายต้องการค้นหาเส้นทางเพื่อสื่อสารไปยังโหนดอื่น การค้นหาเส้นทางจากโหนดต้นทางไปยังโหนดปลายทาง เครือข่ายจะใช้วิธีการกระจายแพ็กเก็ตเกิดควบคุมผ่านโหนดใกล้เคียงไปทั่วเครือข่าย จนกระทั่งถึงโหนดปลายทาง หรือเมื่อค่า TTL (Time to live) ลดลงจนมีค่าเท่ากับศูนย์ จึงระงับการส่งต่อ เมื่อได้เส้นทาง ข้อมูลที่จำเป็นของเส้นทางจะถูกบันทึกไว้ในตารางจัดเก็บข้อมูลเส้นทาง (Routing table)
2. ระยะซ่อมบำรุง ระยะนี้ใช้ซ่อมบำรุงเส้นทางในเครือข่ายเมื่อไม่สามารถสื่อสารไปยังโหนดปลายทางเดิมได้ เนื่องจากการสูญเสียเส้นทางเพราะโหนดในเส้นทางหรือปลายทางสูญหาย

AODV จะใช้ข้อความ 3 ชนิดในการจัดเส้นทางและซ่อมบำรุงเส้นทาง คือ RREQ (Route REQ) RREP (Route REP) และ RERR (Route ERR) การจัดเส้นทางจะเริ่มต้นเมื่อโหนดต้นทางต้องการเส้นทางสื่อสารไปยังโหนดปลายทาง จึงเริ่มแพร่กระจาย RREQ ไปยังโหนดใกล้เคียง ข้อความ RREQ จะเก็บหมายเลขเส้นทางล่าสุด เพื่อป้องกันไม่ให้โหนดที่เคยได้รับข้อความนี้ส่งต่อข้อความต่อไป โหนดที่ได้รับ RREQ จะแพร่กระจายข้อความดังกล่าวต่อไปยังโหนดที่อยู่ใกล้เคียง จนกระทั่งถึงโหนดปลายทางหรือเมื่อค่าฮอปสูงสุด (Max hop count) ในการแพร่กระจายลดลงเป็นศูนย์ เมื่อข้อความ RREQ ถึงปลายทางข้อมูลเส้นทางจะถูกเรียงกลับเพื่อใช้ส่ง RREP ตอบกลับไปยังโหนดต้นทางตามเส้นทางเดิม [17] ถ้าหากโหนดต้นทางได้รับ RREP ที่มีหมายเลขโหนดปลายทางซ้ำกัน เส้นทางที่มีหมายเลขลำดับมากที่สุดซึ่งเป็นเส้นทางที่ใหม่ที่สุด และมีจำนวนฮอปไปยังปลายทางน้อยที่สุดจะได้รับการพิจารณาเป็นเส้นทางที่ดีที่สุด หลังจากที่โหนดต้นทางได้รับ RREP เรียบร้อยแล้ว โหนดต้นทางจะเริ่มต้นส่งแพ็กเก็ตข้อมูล

เส้นทางจะมีสถานะใช้งาน (Active) เมื่อยังคงมีแพ็กเก็ตข้อมูลถูกส่งในเส้นทางดังกล่าว แต่ถ้าเส้นทางเกิดสูญหาย โหนดที่อยู่ปลายสุดของเส้นทางที่เสียหายจะทำหน้าที่สร้าง RERR เพื่อแจ้งให้โหนดต้นทางทราบ จากนั้นกลไกการซ่อมแซมเส้นทางจะเริ่มทำงานเพื่อค้นหาเส้นทางใหม่ ในกรณีที่เส้นทางไม่ได้ใช้งานเป็นระยะเวลาหนึ่งจะมีการลบเส้นทางดังกล่าวทิ้ง เพื่อประหยัดเนื้อที่หน่วยความจำของตารางจัดเก็บข้อมูลเส้นทาง นอกจากนี้การปรับปรุงข้อมูลเส้นทาง

อยู่เป็นระยะๆ จะทำได้โดยการส่งข้อความ HELLO จากโหนดในเครือข่ายเพื่อปรับปรุงข้อมูลในตารางจัดเก็บข้อมูลของตัวเอง [18]

ตัวอย่างการส่ง RREQ และ RREP สำหรับกระบวนการค้นหาเส้นทางแสดงได้ดังภาพประกอบที่ 2-5 เริ่มต้นจากโหนด S เป็นโหนดต้นทางที่ต้องการค้นหาเส้นทางไปยังโหนดปลายทาง D ดังภาพประกอบที่ 2-5 (a) จึงทำการแพร่กระจาย RREQ ไปยังโหนดใกล้เคียง เมื่อโหนดใกล้เคียงได้รับข้อความดังกล่าว โหนดจะปรับปรุงข้อมูลในตารางจัดเก็บข้อมูลเส้นทางตัวเอง โดยจะจัดเก็บในรูปแบบ (Src:Nxt) เมื่อ Src คือหมายเลขโหนดต้นทางและ Nxt คือโหนดในฮอปถัดไป เมื่อโหนด D ได้รับ RREQ ก็จะส่ง RREP ตามเส้นทางที่เดิมแบบเรียงกลับไปยังโหนดต้นทาง ซึ่งเส้นทางที่ใช้ส่ง RREP และใช้เป็นเส้นทางสื่อสารระหว่างโหนด S และ D แสดงดังภาพประกอบที่ 2-5 (b)



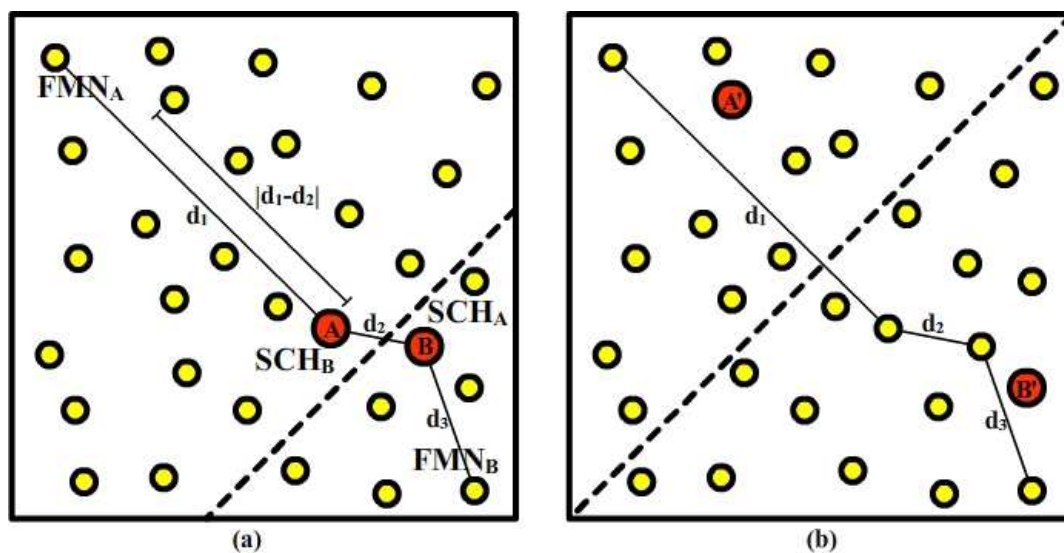
ภาพประกอบ 2-5 การทำงานของโพรโทคอล AODV (a) การส่งข้อความ RREQ และ (b) การส่งข้อความ RREP

### 2.3.2 โพรโทคอลการจัดเส้นทาง AODV แบบ Clustering

บทความ [19-21] นำเสนอโพรโทคอลจัดเส้นทางชนิด AODV clustering และผลการทดสอบ พบว่าการจัดเส้นทางชนิดนี้สามารถสนับสนุนการจัดการเครือข่ายที่มีความหนาแน่นสูง เพราะโดยปกติการจัดเส้นทางด้วยวิธี AODV จะมีโอเวอร์เฮดเพิ่มขึ้นเมื่อเครือข่ายมีขนาดใหญ่ขึ้น ทำให้สมรรถนะของเครือข่ายลดลง จึงมีการปรับปรุง AODV เป็น AODV Clustering ที่สามารถลดโอเวอร์เฮด โดยเฉพาะเมื่อมีการเปลี่ยนแปลงโทโพโลยี AODV clustering จะมีระยะเวลาในการส่งข้อมูลน้อย มีการใช้พลังงานอย่างมีประสิทธิภาพ มีอายุของเครือข่ายเพิ่มขึ้น และยังสามารถเคลื่อนที่

## 2.4 การจัดคลัสเตอร์ด้วยอัลกอริทึม ACHS และ EACHS

ACHS [3, 4] จะเริ่มต้นด้วยการเลือกคลัสเตอร์เฮดตามวิธีของ LEACH ซึ่งผลการเลือกคลัสเตอร์เฮดแสดงดังภาพประกอบที่ 2-6 (a) หลังจากได้คลัสเตอร์เฮด A และ B จะมีการพิจารณาเลือกคลัสเตอร์เฮดใหม่ โดยการเลือกโหนดสมาชิกที่ไกลที่สุดในคลัสเตอร์ซึ่งเรียกว่า FMN (Furthest member node) และคลัสเตอร์เฮดอื่นที่ใกล้ที่สุดซึ่งเรียกว่า SCH (Shortest cluster head) ตัวอย่างการเลือกคลัสเตอร์เฮดใหม่ของคลัสเตอร์เฮด A จะแสดงในภาพประกอบที่ 2-6 (a) ระยะทางจากคลัสเตอร์เฮด A ไปยัง FMN<sub>A</sub> แทนด้วยระยะ  $d_1$  และระยะทางจากคลัสเตอร์เฮด A ไปยัง SCH<sub>A</sub> จะแทนด้วยระยะ  $d_2$  จากนั้นนำผลต่างระยะทาง  $d_1$  และ  $d_2$  มาพิจารณาเลือกตำแหน่งของคลัสเตอร์เฮดใหม่ โดยพิจารณาว่าระยะทางจากคลัสเตอร์เฮด A ไปยัง FMN<sub>A</sub> หรือไปยัง SCH<sub>A</sub> ระยะใดยาวกว่า จากนั้นจึงขยับตำแหน่งคลัสเตอร์เฮดใหม่ไปทางที่ยาวกว่าด้วยระยะทางเท่ากับผลต่างดังกล่าว ซึ่งในที่นี้ระยะจากคลัสเตอร์เฮด A ไปยัง FMN<sub>A</sub> ยาวกว่า ดังนั้นตำแหน่งคลัสเตอร์เฮดใหม่ของคลัสเตอร์เฮด A คือ A' ซึ่งแสดงดังภาพประกอบ 2-6 (b) ส่วนอัลกอริทึม EACHS ได้ปรับปรุงการเลือกคลัสเตอร์เฮดใหม่เฉพาะคลัสเตอร์ที่มีจำนวนโหนดสมาชิกแตกต่างจากจำนวนโหนดสมาชิกเฉลี่ย 10 เปอร์เซ็นต์ ทำให้สามารถประหยัดทั้งเวลาและพลังงานได้



ภาพประกอบ 2-6 การเลือกคลัสเตอร์เฮดของอัลกอริทึม ACHS

### บทที่ 3

#### การวิเคราะห์การเลือกคลัสเตอร์เฮด

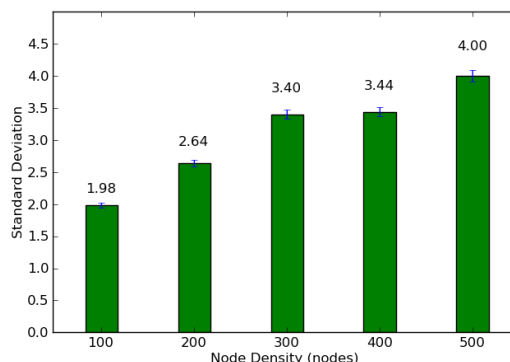
ในบทนี้จะกล่าวถึงผลการเลือกคลัสเตอร์เฮดด้วยอัลกอริทึมต่างๆ ได้แก่ การใช้ความน่าจะเป็น การจัดคลัสเตอร์แบบกริด และการจัดคลัสเตอร์แบบหกเหลี่ยม

##### 3.1 การเลือกคลัสเตอร์เฮดด้วยอัลกอริทึมที่ใช้ทฤษฎีความน่าจะเป็น

ในหัวข้อนี้เป็นการศึกษาความคลาดเคลื่อนของจำนวนคลัสเตอร์เฮดที่ได้จากวิธีการสุ่มตัวเลขของอัลกอริทึมที่ใช้ทฤษฎีความน่าจะเป็น เช่น LEACH LEACH-C ACHS และ EACHS [1-4] เนื่องจากอัลกอริทึมดังกล่าวให้จำนวนคลัสเตอร์เฮดไม่เท่ากับจำนวนที่ต้องการ ซึ่งเกิดจากโหนดแต่ละตัวสุ่มตัวเลขและเปรียบเทียบกับค่า Threshold ตามสมการ (2-1) ในกรณีที่เลขสุ่มมีค่ามากกว่า threshold โหนดดังกล่าวจะเป็นคลัสเตอร์เฮด ในทางกลับกัน โหนดดังกล่าวจะเป็นโหนดสมาชิก ด้วยวิธีการสุ่มทำให้มีโหนดที่มีเลขสุ่มเกิน threshold มีจำนวนไม่แน่นอน

$$T(n) = \begin{cases} \frac{P}{1-P \left( r \bmod \frac{1}{P} \right)} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases} \quad (2-1)$$

เมื่อทดลองด้วยโหนดจำนวน 100, 200, 300, 400 และ 500 โหนดในพื้นที่ขนาด 100x100 ตารางเมตร ด้วยโปรแกรมจำลองกราฟ Networkx ทำซ้ำ 50 ครั้ง โดยกำหนดให้เครือข่ายมีคลัสเตอร์เฮดจำนวน 5 เปอร์เซ็นต์ของโหนดทั้งหมด หลังจากนั้นวัดความคลาดเคลื่อนจากค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย (Average absolute standard deviation) ของจำนวนคลัสเตอร์เฮด โดยเปรียบเทียบกับจำนวนคลัสเตอร์เฮด 5 เปอร์เซ็นต์ดังกล่าว เช่น ในโหนดจำนวน 100 โหนด จะใช้จำนวนคลัสเตอร์เฮด 5 โหนดเป็นค่ามาตรฐาน จากนั้นจึงนำผลต่างจากค่ามาตรฐานมาคำนวณเป็นค่าเฉลี่ยเพื่อพิจารณาว่ามีความคลาดเคลื่อนมากน้อยเท่าไร ผลลัพธ์การทดลองนี้จะแสดงให้เห็นถึงจำนวนคลัสเตอร์เฮดที่ได้จากวิธีการสุ่มตัวเลขของอัลกอริทึมที่ใช้ทฤษฎีความน่าจะเป็น ว่ามีความน่าเชื่อถือมากน้อยเท่าไร เมื่อเปรียบเทียบกับจำนวนคลัสเตอร์เฮดที่ต้องการ ผลการทดลองแสดงดังภาพประกอบที่ 3-1



ภาพประกอบ 3-1 ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของจำนวนคลัสเตอร์เฮด

จากกรณีทดสอบแรกกำหนดให้เครือข่ายประกอบด้วยโหนด 100 โหนด จำนวนคลัสเตอร์เฮดที่เหมาะสมคือ 5 เปอร์เซนต์ของจำนวนโหนดทั้งหมด [1] แต่จากการทดลองด้วยจำนวน 100 โหนด จะมีค่าเบี่ยงเบนมาตรฐานที่ประมาณ 2 โหนด หมายถึงสามารถมีจำนวนคลัสเตอร์เฮด 3 ถึง 7 โหนด และเมื่อพิจารณากรณีทดสอบที่ 500 โหนด จำนวนคลัสเตอร์เฮดที่ได้คลาดเคลื่อนเพิ่มขึ้นเป็น 2 เท่า คือมีคลัสเตอร์เฮดจำนวน  $25 \pm 4$  โหนด (21 ถึง 29 โหนด)

จากผลการทดลองสรุปได้ว่าจำนวนคลัสเตอร์เฮดที่ได้จากการสุ่มตัวเลขของอัลกอริทึม LEACH มีความคลาดเคลื่อนจากจำนวนที่ต้องการ หรือกล่าวได้ว่ามีความไม่แน่นอนของจำนวนคลัสเตอร์เฮด ทำให้การเลือกคลัสเตอร์เฮดด้วยวิธีนี้มีความน่าเชื่อถือลดลง นอกจากนี้ยังทำให้จำนวนโหนดสมาชิกเฉลี่ยในแต่ละคลัสเตอร์คลาดเคลื่อนอีกด้วย กล่าวคือโดยปกติแล้วที่ 100 โหนดและมีคลัสเตอร์เฮดจำนวน 5 โหนดจะมีโหนดสมาชิกเฉลี่ยในแต่ละคลัสเตอร์จำนวน 19 โหนด แต่ผลการทดสอบแรกกลับได้จำนวนคลัสเตอร์เฮด  $5 \pm 2$  โหนด (3 ถึง 7 โหนด) ดังนั้นจะมีจำนวนโหนดสมาชิกเฉลี่ยต่อคลัสเตอร์ 32 โหนด (คำนวณจาก  $(100-3) \div 3$ ) ถึง 13 โหนด (คำนวณจาก  $(100-7) \div 7$ ) ตามลำดับ เมื่อจำนวนโหนดสมาชิกเฉลี่ยมากกว่าหรือน้อยกว่า 19 โหนด จะทำให้ประสิทธิภาพการใช้พลังงานของคลัสเตอร์เฮดลดลง ส่งผลให้ปริมาณภาระงานและการใช้พลังงานของแต่ละคลัสเตอร์ไม่สมดุลกัน

### 3.2 อัลกอริทึมการจัดคลัสเตอร์แบบกริด

การจัดคลัสเตอร์แบบกริด [5-8] มีข้อจำกัดคือ จำนวนคลัสเตอร์ของเครือข่ายในกรณีที่พื้นที่สนใจเป็นสี่เหลี่ยมจัตุรัสจะแบ่งย่อยได้เพียงบางจำนวนเท่านั้น ได้แก่ 4, 9, 16, 25, 36, ... ตามลำดับ ดังนั้นในหัวข้อนี้จึงเป็นการศึกษาเปรียบเทียบการกระจายจำนวนโหนดสมาชิกของ

อัลกอริทึมการจัดคลัสเตอร์แบบกริดและอัลกอริทึมที่ใช้คลัสเตอร์เฮคจำนวน 5 เพอร์เซ็นต์ของ โหนดในเครือข่าย ซึ่งผลการทดลองที่ได้จากการคำนวณ แสดงไว้ในตาราง 3-1

ตารางที่ 3-1 จำนวนโหนดสมาชิกเฉลี่ยการแบ่งแบบกริดย่อยกับใช้คลัสเตอร์เฮค 5 เพอร์เซ็นต์

จำนวนโหนด ในเครือข่าย	วิธีคลัสเตอร์เฮค 5 เพอร์เซ็นต์		วิธีแบ่งกริดย่อย	
	จำนวนคลัสเตอร์เฮค	จำนวนโหนด สมาชิกเฉลี่ย	จำนวนกริดย่อย (เท่ากับจำนวน คลัสเตอร์เฮค)	จำนวนโหนด สมาชิกเฉลี่ย
100	5	19	4	24
200	10	19	9	~21
300	15	19	16	~18
400	20	19	16	24
500	25	19	25	19

จากตาราง 3-1 จำนวนโหนดสมาชิกเฉลี่ยที่ได้จากวิธีแบ่งเครือข่ายออกเป็นกริดย่อยสำหรับในกรณีที่โหนดในเครือข่ายมี 500 โหนด การแบ่งเครือข่ายเป็นกริดย่อยจะได้จำนวนโหนดสมาชิกเฉลี่ยเท่ากับวิธีการใช้คลัสเตอร์เฮค 5 เพอร์เซ็นต์ แต่ในกรณีที่โหนดในเครือข่ายมีจำนวนน้อยกว่านี้ จะทำให้จำนวนโหนดสมาชิกเฉลี่ยแตกต่างจากวิธีการใช้คลัสเตอร์เฮค 5 เพอร์เซ็นต์ค่อนข้างมาก โดยเฉพาะในเครือข่ายที่มี 100 และ 400 โหนด จะมีโหนดสมาชิกเฉลี่ย 24 โหนดโดยประมาณ แตกต่างจากวิธีการใช้คลัสเตอร์เฮค 5 เพอร์เซ็นต์ถึง 5 โหนดต่อคลัสเตอร์ จำนวนโหนดสมาชิกที่มีความแตกต่างกันเพียง 5 โหนด สามารถทำให้ปริมาณภาระงานและการใช้พลังงานในแต่ละคลัสเตอร์มีความแตกต่างกันเพิ่มขึ้น

นอกจากนี้อัลกอริทึมการจัดคลัสเตอร์แบบกริดยังมีความยืดหยุ่นต่ำสำหรับการประยุกต์ใช้ในเครือข่ายที่มีการกระจายตัวของโหนดแบบไม่สม่ำเสมอ เนื่องจากแต่ละกริดจะได้รับการแบ่งให้มีพื้นที่เท่าๆ กัน ทำให้ไม่สามารถปรับเปลี่ยนขนาดพื้นที่กริดให้มีขนาดแตกต่างกันตามความหนาแน่นของโหนดที่กระจายอย่างไม่เท่ากันในแต่ละพื้นที่ได้

### 3.3 อัลกอริทึมการจัดคลัสเตอร์แบบหกเหลี่ยม

การจัดคลัสเตอร์แบบหกเหลี่ยม [9-10] สามารถลดการเหลื่อมล้ำกันของขอบเขตของแต่ละคลัสเตอร์ โดยลดการรบกวนกันของสัญญาณวิทยุระหว่างคลัสเตอร์เฮคและทำให้ลดการ

สูญหายของข้อมูล การจัดคลัสเตอร์แบบนี้เหมาะสมกับเครือข่ายที่มีการกระจายตัวด้วยความหนาแน่นสม่ำเสมอ ส่งผลให้แต่ละคลัสเตอร์มีจำนวนโหนดสมาชิกที่ใกล้เคียงกัน ทำให้เกิดสมดุลของภาระงานและการใช้พลังงาน คลัสเตอร์เฮดที่ถูกเลือกด้วยวิธีนี้จะอยู่ตรงกลางคลัสเตอร์ ส่งผลให้สามารถกระจายเส้นทางการส่งต่อข้อมูลจากโหนดสมาชิกไปยังคลัสเตอร์เฮดได้มากขึ้น ซึ่งเป็นการลดโอกาสการเกิดปัญหาคอขวด นอกจากนี้ผลการจัดคลัสเตอร์แบบหกเหลี่ยมยังสามารถทำให้เครือข่ายสามารถขยายขนาดได้ ด้วยการเชื่อมต่อทุกคลัสเตอร์เข้าด้วยกันเป็นโครงสร้างรูปร่างสี่

แต่การจัดคลัสเตอร์ด้วยวิธีนี้จะประยุกต์ใช้ในเครือข่ายที่มีการกระจายตัวของโหนดสมาชิกแบบไม่สม่ำเสมอให้สามารถสมดุลภาระงานและการใช้พลังงานได้ยาก อีกทั้งไม่สามารถจัดคลัสเตอร์ให้มีจำนวนคลัสเตอร์เฮดเท่ากับ 5 เปอร์เซ็นต์ของจำนวนโหนดในเครือข่าย เนื่องจากการจัดคลัสเตอร์แบบหกเหลี่ยมจะคำนึงถึงเรื่องระยะส่งสัญญาณวิทยุของคลัสเตอร์เฮดมากกว่าเรื่องสมดุลภาระงานและการใช้พลังงาน นอกจากนี้การเชื่อมต่อคลัสเตอร์ดังกล่าวให้เป็นโครงสร้างรูปร่างสี่ก่อนข้างเป็นไปได้อาจสำหรับเครือข่ายชนิดนี้

### 3.4 บทสรุป

จากการศึกษา ทดลองและวิเคราะห์ผลในเรื่องผลการจัดคลัสเตอร์ด้วยอัลกอริทึมต่างๆ ในหัวข้อที่ 3.1 ถึง 3.3 พบว่าการจัดคลัสเตอร์ด้วยอัลกอริทึมที่ใช้ทฤษฎีความน่าจะเป็นทำให้จำนวนคลัสเตอร์เฮดและจำนวนโหนดสมาชิกคลาดเคลื่อนจากที่ต้องการ ส่งผลต่อประสิทธิภาพและการใช้พลังงานของเครือข่าย การจัดคลัสเตอร์แบบกริดและแบบหกเหลี่ยมมีการกระจายคลัสเตอร์เฮดสมดุลทั้งเครือข่าย ตลอดจนจำนวนสมาชิกในแต่ละคลัสเตอร์มีความสมดุลกันทำให้ปริมาณภาระงานและการใช้พลังงานมีความสมดุล แต่จำนวนคลัสเตอร์เฮดที่สามารถกำหนดได้ของอัลกอริทึมการจัดคลัสเตอร์แบบกริดมีจำนวนจำกัด ทั้งการจัดคลัสเตอร์แบบกริดและแบบหกเหลี่ยมไม่สามารถนำมาประยุกต์ใช้ในเครือข่ายที่โหนดมีการกระจายแบบไม่สม่ำเสมอให้มีการกระจายภาระงานและการใช้พลังงานได้อย่างสมดุล อย่างไรก็ตามจากการศึกษาทั้งข้อดีและข้อจำกัดของการจัดคลัสเตอร์ในแต่ละวิธี ทำให้สามารถนำไปออกแบบและพัฒนาอัลกอริทึมที่สามารถกระจายภาระงานและการใช้พลังงานอย่างสมดุลในเครือข่ายที่มีการกระจายของโหนดสม่ำเสมอและไม่สม่ำเสมอในหัวข้อที่ 4 ได้



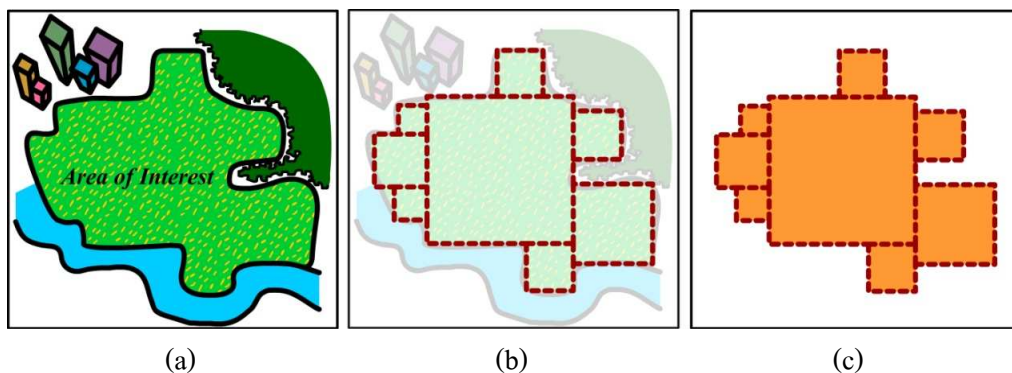
## บทที่ 4

### อัลกอริทึมการเลือกคลัสเตอร์เฮดเพื่อสมดุลภาระบนเครือข่ายเซนเซอร์ไร้สาย

เมื่อพิจารณาข้อดีและข้อจำกัดของการจัดคลัสเตอร์ในแต่ละวิธี งานวิจัยนี้จึงนำจุดเด่นมาพิจารณาออกแบบ ปรับปรุงและพัฒนาการจัดคลัสเตอร์ที่มีจำนวนคลัสเตอร์เฮดที่แน่นอน อยู่ในตำแหน่งตรงกลางของคลัสเตอร์ สามารถกระจายสมดุลภาระงานในแต่ละคลัสเตอร์ โดยใช้หลักการแบ่งพื้นที่สนใจออกเป็นหลายคลัสเตอร์ที่มีขนาดพื้นที่เท่ากัน จะได้คลัสเตอร์ที่มีจำนวนสมาชิกใกล้เคียงกัน และเมื่อแต่ละคลัสเตอร์มีความสมดุลด้านจำนวนสมาชิกจะทำให้มีความเป็นไปได้สูงที่ภาระงานและการใช้พลังงานใกล้เคียงกันในทุกคลัสเตอร์ อัลกอริทึมการจัดคลัสเตอร์ที่นำเสนอมีความยืดหยุ่นสามารถประยุกต์ใช้งานได้ทั้งในเครือข่ายที่โหนดกระจายตัวแบบสม่ำเสมอและไม่สม่ำเสมอ และสามารถรองรับการใช้งานร่วมกับโพรโทคอล AODV อย่างมีประสิทธิภาพ โดยเรียกการจัดคลัสเตอร์ชนิดนี้ว่าการจัดคลัสเตอร์แบบกริดเซกเตอร์ (Grid Sectoring) รายละเอียดของวิธีการนี้ประกอบด้วย แนวคิดการจัดคลัสเตอร์แบบกริดเซกเตอร์และการทำงานร่วมกันระหว่างอัลกอริทึมที่นำเสนอกับโพรโทคอล AODV

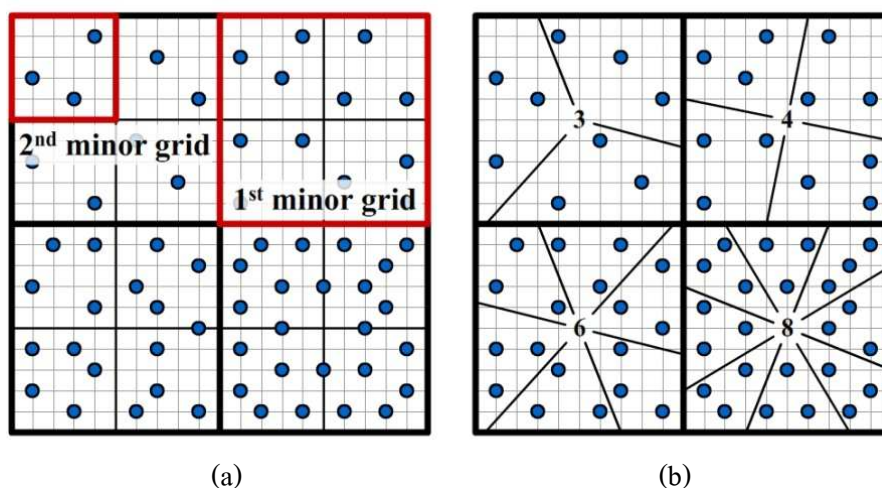
#### 4.1 แนวคิดการจัดคลัสเตอร์แบบกริดเซกเตอร์

การจัดคลัสเตอร์แบบกริดเซกเตอร์ (Grid Sectoring) ทุกโหนดในเครือข่ายจำเป็นต้องส่งข้อมูลตำแหน่งทางภูมิศาสตร์ของตัวเองไปให้สถานีฐาน ซึ่งตำแหน่งดังกล่าวมาจากเซนเซอร์วัดตำแหน่งทางภูมิศาสตร์ เช่น GPS ที่อยู่บนโหนด และพื้นที่สนใจจะต้องแบ่งเป็นลักษณะสี่เหลี่ยมจตุรัสได้ดังเช่นภาพประกอบที่ 4-1



ภาพประกอบ 4-1 การแบ่งพื้นที่สนใจเป็นสี่เหลี่ยมจตุรัสย่อย

เมื่อโหนดกระจายแบบไม่สม่ำเสมอในพื้นที่สนใจ เครือข่ายจะถูกแบ่งเป็นกริดย่อยในครั้งแรก (1<sup>st</sup> minor grid) ผลที่ได้คือ โหนดในแต่ละกริดมีจำนวนไม่เท่ากัน ดังภาพประกอบที่ 4-2 (a) เมื่อมีการแบ่งเป็นกริดย่อยในครั้งที่ 2 (2<sup>nd</sup> minor grid) แต่ละคลัสเตอร์ยังคงมีจำนวนโหนดสมาชิกในปริมาณไม่เท่ากันเช่นเดิม เนื่องจากจำนวนกริดย่อยที่สามารถแบ่งได้มีจำกัด ในทางกลับกันหากใช้วิธีแบ่งกริดย่อยจากที่แบ่งได้ในครั้งแรกออกเป็นเซกเตอร์ ดังภาพประกอบที่ 4-2 (b) จะเห็นได้ว่าผลลัพธ์จากการแบ่งมีความยืดหยุ่นสูงกว่า วิธีการนี้สามารถแบ่งกริดออกเป็นเซกเตอร์ย่อยที่มีพื้นที่เท่ากันได้ดีกว่า เมื่อ 1 เซกเตอร์คือ 1 คลัสเตอร์ เมื่อแบ่งโหนดด้วยวิธีการนี้แล้วแต่ละคลัสเตอร์มีโอกาสมิที่จะมีจำนวนโหนดสมาชิกเท่ากันสูงกว่าหรือมีความสมดุลด้านจำนวนสมาชิกสูงกว่าวิธีแบ่งแบบกริดย่อยเพียงอย่างเดียว



ภาพประกอบ 4-2 (a) การแบ่งพื้นที่สนใจเป็นกริดย่อยในลำดับที่ 1 และ 2 ตามลำดับ (b) การแบ่งกริดย่อยออกเป็นเซกเตอร์

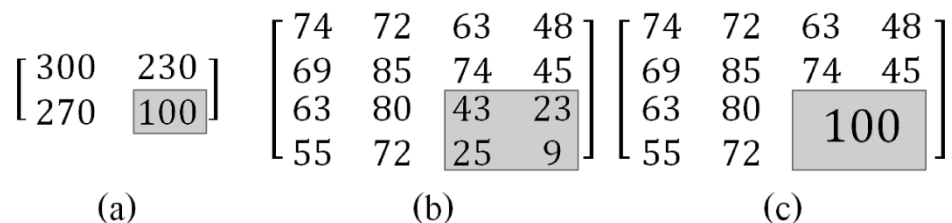
การจัดคลัสเตอร์แบบกริดเซกเตอร์แบ่งออกเป็นขั้นตอนย่อย 4 ขั้นตอน คือ

1. การแบ่งพื้นที่สนใจเป็นกริดย่อย
2. การแบ่งพื้นที่กริดออกเป็นเซกเตอร์
3. การเลือกคลัสเตอร์เฮด
4. การจัดคลัสเตอร์แบบกริดเซกเตอร์

โดย 3 ขั้นตอนแรก จะดำเนินการโดยสถานีฐาน ส่วนขั้นตอนการจัดคลัสเตอร์จะเป็นการทำงานร่วมกันระหว่างสถานีฐาน คลัสเตอร์เฮด และ โหนด ขั้นตอนการจัดคลัสเตอร์แบบกริดเซกเตอร์แสดงในหัวข้อที่ 4.1.1 ถึง 4.1.4 ดังนี้

#### 4.1.1 การแบ่งพื้นที่สนใจเป็นกริดย่อย

สถานีฐานจะคำนวณความหนาแน่นของโหนดในพื้นที่สนใจเพื่อพิจารณาว่าจะแบ่งเป็นกี่กริดย่อย โดยปกติการแบ่งเป็นกริดย่อย แต่ละกริดควรมีจำนวนโหนดระหว่าง 20-220 โหนด เพราะเมื่อคำนวณคลัสเตอร์เสดที่ 5 เปอร์เซนต์แล้วจะได้คลัสเตอร์เสดจำนวน 1-11 โหนดต่อกริด หากมีจำนวนคลัสเตอร์เสดในกริดมากเกินไป เช่น มากกว่า 11 โหนด จะทำให้กริดดังกล่าวถูกแบ่งเป็นเซกเตอร์ที่มีพื้นที่น้อยเกินไป ทำให้คลัสเตอร์เสดที่ถูกเลือกมีโอกาสอยู่ใกล้ขอบของคลัสเตอร์ ซึ่งเป็นสาเหตุให้การกระจายเส้นทางการสื่อสารในคลัสเตอร์ไม่มีประสิทธิภาพ ตัวอย่างเช่น พื้นที่สนใจประกอบด้วยโหนด 1000 โหนดกระจายตัวแบบไม่สม่ำเสมอ หากสถานีฐานแบ่งพื้นที่ดังกล่าวออกเป็น 4 กริดย่อยดังภาพประกอบที่ 4-3 (a) จะทำให้มีจำนวนโหนดเกิน 220 โหนดจำนวน 3 กริด จากนั้นสถานีฐานจะแบ่งแต่ละกริดที่แบ่งได้ในตอนแรกออกเป็น 4 กริดย่อย ซึ่งขณะนี้จะมีทั้งหมด 16 กริดดังภาพประกอบที่ 4-3 (b) จะทำให้จำนวนโหนดต่อกริดลดลง แต่ขณะนี้จะเห็นได้ว่าบางกริดย่อยในพื้นที่แรงงามีโหนดน้อยกว่า 20 โหนด ดังนั้นกริดในพื้นที่แรงงาจึงไม่จำเป็นต้องแบ่งเป็นกริดย่อยดังภาพประกอบที่ 4-3 (c)



ภาพประกอบ 4-3 การแบ่งพื้นที่สนใจเป็นกริดย่อยและแทนจำนวนโหนดลงในเมตริกซ์

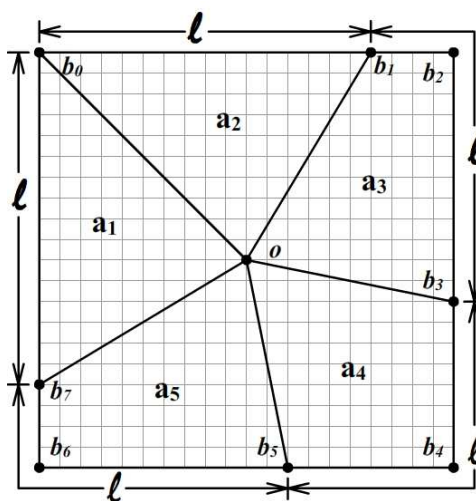
เมื่อผลลัพธ์การแบ่งพื้นที่สนใจเป็นกริดได้รับการคำนวณและเก็บในรูปแบบตารางเมตริกซ์แล้ว สถานีฐานจะดำเนินการแบ่งแต่ละกริดย่อยออกเป็นเซกเตอร์ดังในหัวข้อ 4.1.2

#### 4.1.2 การแบ่งพื้นที่กริดออกเป็นเซกเตอร์

ขั้นตอนนี้เป็นการแบ่งกริดออกเป็นคลัสเตอร์ในลักษณะเซกเตอร์ วิธีนี้จะทำให้แต่ละกริดมีจำนวนคลัสเตอร์ที่มีความยืดหยุ่นกว่าวิธีแบ่งเป็นกริดย่อย คือสามารถแบ่งออกเป็น 1,2,3,... คลัสเตอร์ได้ แต่ไม่ควรเกิน 8 คลัสเตอร์ต่อกริด ทุกคลัสเตอร์ที่ถูกแบ่งในกริดหนึ่งๆ จะมีขนาดพื้นที่เท่ากัน การแบ่งกริดออกเป็นเซกเตอร์ที่เท่ากันทำได้โดยนำความยาวเส้นรอบรูปของพื้นที่กริด  $p$  (Perimeter) มาหารด้วยจำนวนเซกเตอร์ที่ต้องการแบ่ง  $n$  (Number of sectors) ซึ่งจะได้ความยาวด้านของขอบด้านนอกของเซกเตอร์  $l$  (Length of outer side of sector) ที่มีขนาดเท่ากันได้ ตามสมการที่ 4-1 ตัวอย่างการแบ่งกริดเป็นเซกเตอร์ที่เท่ากันแสดงได้ดังภาพประกอบที่ 4-4

$$l = \frac{p}{n} \quad (4-1)$$

จากนั้นทุกๆ จุดสิ้นสุดของความยาว  $l$  บนเส้นรอบรูปของพื้นที่กริด เมื่อลากจากจุดเหล่านี้ไปยังจุดศูนย์กลางของพื้นที่กริด จะเป็นการแบ่งพื้นที่กริดออกเป็นคลัสเตอร์ในลักษณะเชกเตอร์ตามจำนวน  $n$  ที่ต้องการ ผลลัพธ์การแบ่งที่ได้จะทำให้พื้นที่แต่ละเชกเตอร์มีขนาดเท่ากัน



ภาพประกอบ 4-4 การแบ่งพื้นที่กริดออกเป็นเชกเตอร์

เมื่อแต่ละกริดได้รับการแบ่งเป็นเชกเตอร์เรียบร้อยแล้ว สถานีฐานจะบันทึกจุดพิกัดขอบเขตของแต่ละคลัสเตอร์เพื่อใช้ในการคำนวณหาจุดศูนย์กลางของคลัสเตอร์ ด้วยวิธีการหาจุดเซนทรอยด์ของรูปหลายเหลี่ยม (Centroid of polygon) ซึ่งจะแสดงรายละเอียดในหัวข้อที่ 4.1.3

#### 4.1.3 การเลือกคลัสเตอร์เฮด

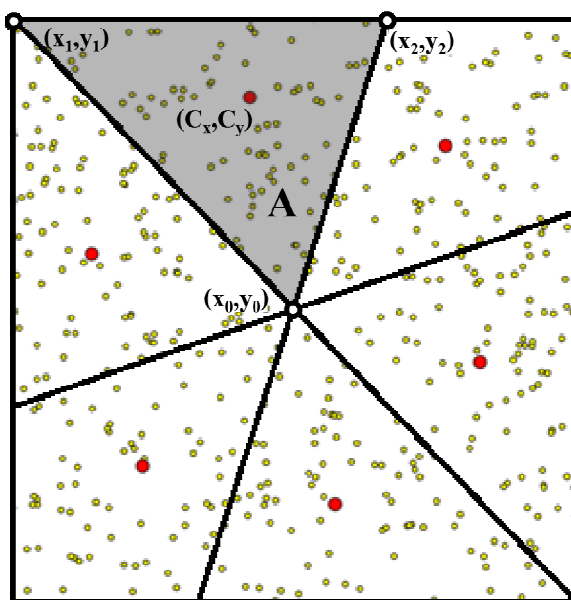
บริเวณตรงกลางพื้นที่ของคลัสเตอร์จะเป็นตำแหน่งที่เหมาะสมของคลัสเตอร์เฮด [2] เพราะจะทำให้คลัสเตอร์เฮดล้อมรอบไปด้วยโหนดสมาชิกทุกด้าน ดังนั้นเส้นทางในการสื่อสารระหว่างโหนดสมาชิกกับคลัสเตอร์เฮดจึงเพิ่มขึ้น ส่งผลให้ความคับคั่งในการสื่อสารลดลง วิธีหาจุดศูนย์กลางคลัสเตอร์เพื่อใช้เลือกโหนดที่อยู่ใกล้เคียงจุดดังกล่าวมาเป็นคลัสเตอร์เฮด ทำได้โดยการประยุกต์ใช้สมการเซนทรอยด์ของรูปหลายเหลี่ยม (Centroid formulas of polygon) ดังสมการที่ 4-1 ถึง 4-3 [22] เพื่อหาจุดศูนย์กลางพื้นที่ในแต่ละคลัสเตอร์ ซึ่งปกติแล้วการจัดคลัสเตอร์ด้วยวิธีที่นำเสนอนี้ แต่ละคลัสเตอร์จะมีรูปทรงเรขาคณิตเป็นรูปสามเหลี่ยม สี่เหลี่ยม หรือห้าเหลี่ยมเท่านั้น

$$C_x = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (4-2)$$

$$C_y = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (4-3)$$

$$A = \frac{1}{2} \sum_{i=0}^{n-1} (x_i y_{i+1} - x_{i+1} y_i) \quad (4-4)$$

จากสมการที่ 4-2 ถึง 4-4 กำหนดให้พิกัด  $(C_x, C_y)$  คือ จุดเซนทรอยด์หรือศูนย์กลางของคลัสเตอร์  $A$  คือ ขนาดพื้นที่ของคลัสเตอร์  $(x_i, y_i)$  คือ พิกัดของจุดของรูปหลายเหลี่ยมที่  $i$  ซึ่งพิกัดทั้งหมดดังที่กล่าวมานี้แสดงได้ดังภาพประกอบที่ 4-5 หลังจากที่ได้คำนวณจุดศูนย์กลางของแต่ละคลัสเตอร์เรียบร้อยแล้ว สถานะฐานจะเลือกโหนดที่อยู่ใกล้กับจุดศูนย์กลางของแต่ละคลัสเตอร์มากที่สุดเป็นคลัสเตอร์เฮดโดยใช้วิธีการคำนวณระยะทางของออยเลอร์ (Euler distance) ซึ่งผลการคำนวณที่ได้จะเป็นระยะทางที่สั้นที่สุด ผลลัพธ์ที่ได้จากขั้นตอนนี้แสดงดังภาพประกอบที่ 4-5



ภาพประกอบ 4-5 ผลลัพธ์การแบ่งพื้นที่กริดออกเป็น 6 เซกเตอร์ด้วยโปรแกรม Networkx

จากภาพประกอบที่ 4-5 แสดงให้เห็นว่าผลลัพธ์การแบ่งพื้นที่กริดออกเป็น เซกเตอร์และผลการเลือกคลัสเตอร์เฮดที่จุดศูนย์กลางของคลัสเตอร์ทำให้ได้ผลลัพธ์ตำแหน่งของคลัสเตอร์เฮดคล้ายกับผลลัพธ์ของการจัดคลัสเตอร์แบบหกเหลี่ยม โดยเฉพาะอย่างยิ่งเมื่อแบ่ง

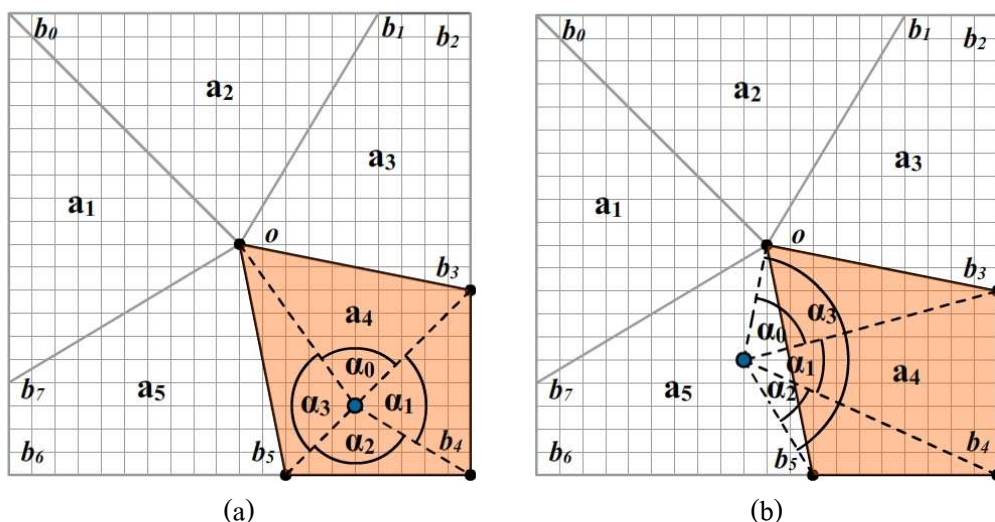
พื้นที่กริดออกเป็น 6 คลัสเตอร์ หลังจากสถานีฐานเลือกคลัสเตอร์เฮดเรียบร้อยแล้ว ขั้นตอนต่อมาจะเป็นขั้นตอนสุดท้ายของการจัดคลัสเตอร์ด้วยอัลกอริทึมชนิดนี้ ซึ่งจะเป็นการทำงานร่วมกันระหว่างสถานีฐาน คลัสเตอร์เฮดและโหนด โดยรายละเอียดจะแสดงไว้ในหัวข้อ 4.1.4

#### 4.1.4 การจัดคลัสเตอร์แบบกริดเชกเตอร์

เมื่อสถานีฐานเลือกคลัสเตอร์เฮดเรียบร้อยแล้ว สถานีฐานจะส่งข้อความการประกาศคลัสเตอร์เฮด (Cluster head notification message) ไปยังคลัสเตอร์เฮดที่ได้รับเลือกทุกโหนด ในข้อความจะบรรจุหมายเลขโหนด (Node ID) ของคลัสเตอร์เฮดและข้อมูลขอบเขตของคลัสเตอร์ไว้ เมื่อคลัสเตอร์เฮดได้รับข้อความนี้แล้วจะเปลี่ยนสถานะตัวเองเป็นคลัสเตอร์เฮดและเผยแพร่ (Broadcast) ข้อความเดิมไปยังโหนดใกล้เคียง เพื่อค้นหาโหนดสมาชิกที่อยู่ในขอบเขตของคลัสเตอร์ตัวเอง เมื่อโหนดที่อยู่ใกล้เคียงจะนำค่าขอบเขตของคลัสเตอร์ที่บรรจุอยู่ในข้อความที่ได้รับมาคำนวณว่าตัวมันเองอยู่ในคลัสเตอร์ดังกล่าวหรือไม่ ถ้าหากใช้โหนดที่ได้รับข้อความจะเปลี่ยนสถานะเป็นสมาชิกของคลัสเตอร์ทันที แต่ถ้าหากผลการคำนวณออกมาว่าโหนดที่ได้รับข้อความไม่ได้อยู่ในคลัสเตอร์ดังกล่าว โหนดที่ได้รับข้อความจะลบข้อความนี้ทิ้งทันที

วิธีการที่โหนดใช้ในการคำนวณว่าอยู่ในขอบเขตของคลัสเตอร์หรือไม่คือ การนำขอบเขตของคลัสเตอร์ที่ระบุไว้ในข้อความมาคำนวณหามุมรวม ถ้าหากคำนวณมุมรวมได้เท่ากับ  $2\pi$  แสดงว่าโหนดดังกล่าวอยู่ในขอบเขตของคลัสเตอร์ แต่ถ้าหากคำนวณมุมรวมได้น้อยกว่า  $2\pi$  แสดงว่าโหนดดังกล่าวไม่ได้อยู่ในขอบเขตของคลัสเตอร์ เช่น โหนดที่ได้รับข้อความอยู่ในขอบเขตของคลัสเตอร์  $a_4$  ดังภาพประกอบที่ 4-6 (a) ที่มีเซตข้อมูลของขอบเขตของคลัสเตอร์เป็น  $\{0, b_3, b_4, b_5\}$  เมื่อคำนวณมุมรวมรอบโหนด  $(\alpha_0 + \alpha_1 + \alpha_2 + \alpha_3)$  แล้วจะได้ผลลัพธ์เท่ากับ  $2\pi$  แต่ในกรณีที่โหนดดังกล่าวไม่ได้อยู่ในขอบเขตของคลัสเตอร์  $a_4$  ดังภาพประกอบที่ 4-6 (b) ผลการคำนวณมุมรวมรอบโหนดดังกล่าวจะมีค่าน้อยกว่า  $2\pi$

การคำนวณมุมดังกล่าวจะใช้วิธีการคำนวณผลรวมของมุมที่อยู่ระหว่างด้านของสามเหลี่ยมที่ลากจากคู่พิกัดขอบเขตของคลัสเตอร์ที่อยู่ติดกัน ไปยังโหนด เช่น มุม  $\alpha_3$  ดังภาพประกอบที่ 4-6 จะเป็นมุมที่อยู่ระหว่างด้านที่เกิดจากการลากเส้นตรงจากจุด  $O$  และ  $b_5$  มายังโหนด ส่วนวิธีการได้มาซึ่งขนาดมุมดังกล่าวนี้คือ การประยุกต์ใช้กฎของโคไซน์ (Law of cosine)

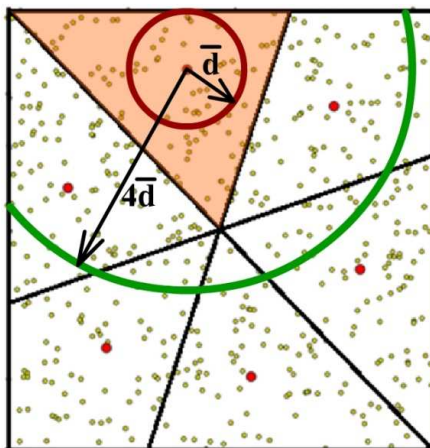


ภาพประกอบ 4-6 (a) ผลรวมของมุมเท่ากับ  $2\pi$  เมื่อโหนดอยู่ในขอบเขตของคลัสเตอร์  $a_4$   
 (b) ผลรวมของมมน้อยกว่า  $2\pi$  เมื่อโหนดไม่ได้อยู่ในขอบเขตของคลัสเตอร์  $a_4$

จากขั้นตอนการจัดคลัสเตอร์แบบกริดเชกเตอร์ดังที่กล่าวมาข้างต้น จะเห็นได้ว่า ผลการเลือกคลัสเตอร์เสดด้วยวิธีนี้ทำให้ได้คลัสเตอร์เสดทั้งหมดอยู่ในตำแหน่งใกล้เคียงจุดศูนย์กลางของคลัสเตอร์ ซึ่งนอกจากจะเพิ่มการกระจายเส้นทางการสื่อสารจากโหนดสมาชิกไปยังคลัสเตอร์เสดแล้ว ยังสามารถจำกัดจำนวนฮอปในการส่งแพ็กเก็ตควบคุม (Control packet) ของโพรโทคอลการจัดเส้นทางได้อีกด้วย ผลที่ได้จากการจำกัดจำนวนฮอปคือ สามารถลดจำนวนแพ็กเก็ตควบคุมที่แพร่กระจายในเครือข่าย ส่งผลให้ลดการใช้พลังงานของทั้งเครือข่ายได้เพิ่มขึ้น รายละเอียดการปรับปรุงดังกล่าวอธิบายไว้ในหัวข้อที่ 4.2

#### 4.2 การทำงานร่วมกันระหว่างอัลกอริทึมกริดเชกเตอร์กับโพรโทคอล AODV

เนื่องจากพื้นที่คลัสเตอร์ที่ได้จากอัลกอริทึมที่ได้นำเสนอมีขนาดแน่นอน และคลัสเตอร์เสดที่ถูกเลือกจะอยู่บริเวณศูนย์กลางของคลัสเตอร์ เมื่อทราบจำนวนฮอปเฉลี่ยจากโหนดสมาชิกไปยังคลัสเตอร์เสดด้วยการคำนวณ จะทำให้สามารถจำกัดจำนวนฮอปในการสื่อสารแพ็กเก็ตควบคุมของโพรโทคอลจัดเส้นทางได้ ด้วยวิธีการนี้นอกจากจะสามารถลดจำนวนแพ็กเก็ตควบคุมและลดการใช้พลังงานของเครือข่ายได้แล้ว ยังสามารถลดความคับคั่งของการสื่อสารทั้งภายในและภายนอกคลัสเตอร์ได้อีกด้วย



ภาพประกอบ 4-7 การจำกัดจำนวนฮอปในการส่งแพ็กเก็ตควบคุมของโพรโตคอล AODV

จากภาพประกอบที่ 4-7 เมื่อจำนวนฮอปเฉลี่ย ( $\bar{d}$ ) จากโหนดสมาชิกไปยังคลัสเตอร์เฮดได้รับการคำนวณ คลัสเตอร์เฮดจะนำค่าจำนวนฮอปเฉลี่ยนี้ส่งไปยังโหนดสมาชิกทุกโหนด เพื่อให้ควบคุมจำนวนฮอปสูงสุด (Maximum hop) ของแพ็กเก็ตควบคุมของโพรโตคอลการจัดเส้นทาง โดยอาจจะจำกัดให้สามารถส่งได้สูงสุดเท่ากับ 4 เท่าของค่าจำนวนฮอปเฉลี่ย ดังภาพประกอบ 4-7 ( $4\bar{d}$ ) การทดลองการจำกัดจำนวนฮอปของแพ็กเก็ตควบคุมของโพรโตคอลการจัดเส้นทางชนิด AODV ในหัวข้อที่ 5.4 ซึ่งผลการทดลองแสดงให้เห็นว่า การปรับปรุงอัลกอริทึมการจัดคลัสเตอร์แบบกริดเชกเตอร์ด้วยการจำกัดจำนวนฮอป สามารถลดการใช้พลังงานคลัสเตอร์และลดการใช้พลังงานของคลัสเตอร์เฮดแต่ละโหนดได้เพิ่มขึ้น

#### 4.3 บทสรุป

จากแนวคิดการจัดคลัสเตอร์ด้วยอัลกอริทึมที่นำเสนอทำให้สามารถแบ่งพื้นที่สนใจออกเป็นกริดและแต่ละกริดถูกแบ่งย่อยเป็นคลัสเตอร์ในลักษณะเชกเตอร์ด้วยจำนวนที่สัมพันธ์กับความหนาแน่นของโหนด วิธีการนี้นอกจากจะสามารถสมดุลพื้นที่และสมดุลจำนวนโหนดสมาชิกในแต่ละคลัสเตอร์ได้แล้ว ยังสามารถประยุกต์ใช้กับเครือข่ายที่โหนดมีการกระจายด้วยความหนาแน่นไม่สม่ำเสมอได้ วิธีการจัดคลัสเตอร์ในลักษณะเชกเตอร์ยังส่งผลดีต่อการจำกัดจำนวนฮอปของโพรโตคอลจัดเส้นทางชนิด AODV เนื่องจากแต่ละคลัสเตอร์มีขนาดเท่ากัน ทำให้จำนวนฮอปเฉลี่ยจากโหนดสมาชิกไปยังคลัสเตอร์เฮดใกล้เคียงกันทุกคลัสเตอร์ การทดสอบอัลกอริทึมการจัดคลัสเตอร์แบบกริดเชกเตอร์ในด้านสมดุลภาระงานและการใช้พลังงานจะใช้โปรแกรมจำลองกราฟและโปรแกรมจำลองเครือข่าย



## บทที่ 5

### ผลการจำลองเครือข่าย

หลังจากออกแบบและพัฒนาอัลกอริทึมการจัดคลัสเตอร์แบบกริดเชกเตอร์ตามรายละเอียดที่ได้นำเสนอในหัวข้อที่ 4 ในบทนี้กล่าวถึงประเด็นที่ต้องการศึกษา วิธีการทดลองตัวชี้วัดที่สนใจ ผลการทดลอง และบทสรุป สำหรับการทดลองจะแบ่งเป็น 2 การทดลอง ได้แก่ การจำลองกราฟและการจำลองเครือข่ายซึ่งจะใช้โปรแกรมจำลองกราฟ (Graph simulation) Networkx [23] และโปรแกรมจำลองเครือข่าย Network Simulator-2 (NS-2) [24] ตามลำดับ ในแต่ละการทดลองจะแบ่งย่อยออกเป็น 2 สภาพแวดล้อมคือ สภาพแวดล้อมที่ 1 การทดลองในเครือข่ายที่โหนดมีการกระจายแบบสม่ำเสมอ (Uniform deployment) และสภาพแวดล้อมที่ 2 การทดลองในเครือข่ายที่โหนดมีการกระจายตัวแบบไม่สม่ำเสมอ (Non-uniform deployment) แต่ละการทดลองทำซ้ำ 20 ครั้ง และผลการทดลองได้คำนวณช่วงความเชื่อมั่นที่มีการกระจายแบบ T หรือ T-distribution โดยใช้ค่าความเชื่อมั่น 95 เปอร์เซ็นต์ เครือข่ายที่ทดลองประกอบด้วยโหนด 2 ชนิดคือ โหนดสมาชิกและคลัสเตอร์เฮด สำหรับโหนดสมาชิก จะทำหน้าที่เป็นโหนดต้นทางในการสร้างแพ็กเก็ตข้อมูลและส่งข้อมูลผ่านโหนดประเภทเดียวกันไปยังคลัสเตอร์เฮด ส่วนคลัสเตอร์เฮดจะทำหน้าที่เป็นโหนดปลายทางในการรับข้อมูลจากโหนดสมาชิกภายในคลัสเตอร์ ส่วนสถานีฐานที่ทำหน้าที่หลักในการจัดคลัสเตอร์จะไม่คำนึงถึงและไม่ปรากฏในการทดลองของงานวิจัยชิ้นนี้ ผู้ดำเนินงานวิจัยใช้คอมพิวเตอร์ที่ใช้จำลองเครือข่ายประมวลผลอัลกอริทึมที่นำเสนอแทนสถานีฐาน

#### 5.1 ตัวชี้วัดที่สนใจ

##### 5.1.1 การจำลองกราฟ

###### 1. ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย (Average absolute standard deviation)

ค่าเบี่ยงเบนมาตรฐานเฉลี่ยของจำนวนโหนดสมาชิกในแต่ละคลัสเตอร์ เป็นดัชนีที่ใช้ประเมินการกระจายของจำนวนโหนดสมาชิกในแต่ละคลัสเตอร์โดยการเปรียบเทียบกับจำนวนโหนดสมาชิกเฉลี่ยว่ามีความแตกต่างกันมากน้อยเท่าไร หากผลการทดลองให้ค่าดัชนีชนิดนี้ต่ำ แสดงว่าอัลกอริทึมที่ทดลองสามารถกระจายจำนวนโหนดสมาชิกในแต่ละคลัสเตอร์ให้มีจำนวนเท่ากันได้ดีกว่า ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยคำนวณได้ดังสมการที่ 5-1 เมื่อ  $i$  คือหมายเลขคลัสเตอร์เฮดที่  $0, 1, 2, \dots, k-1$  ตัวแปร  $k$  คือ จำนวน คลัสเตอร์เฮด และตัวแปร  $m_i$  คือจำนวนสมาชิกในคลัสเตอร์ที่  $i$  จากสมการดังกล่าว การใช้จำนวนโหนดสมาชิกเฉลี่ย 19 โหนด เนื่องจากเมื่อลบจำนวนคลัสเตอร์เฮด 5 เปอร์เซ็นต์ออกจากจำนวนโหนดทั้งหมด จะเหลือโหนดที่

เป็นโหนดสมาชิก 95 เปอร์เซ็นต์ เมื่อนำโหนดสมาชิกจำนวน 95 เปอร์เซ็นต์ดังกล่าวมาหารด้วยจำนวนคลัสเตอร์เฮด (5 เปอร์เซ็นต์) จะทำให้ได้จำนวนโหนดสมาชิกในแต่ละคลัสเตอร์เท่ากันคือ 19 โหนดเสมอซึ่งเป็นจำนวนโหนดสมาชิกเฉลี่ย

$$\text{Average Absolute SD} = \frac{\sum_{i=0}^{k-1} |m_i - 19|}{k} \quad (5-1)$$

### 5.1.2 การจำลองเครือข่าย

#### 1. ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของภาระงานต่อโหนดปลายทาง 1 โหนด (Average absolute standard deviation of load per sink)

ดัชนีชนิดนี้ใช้ประเมินปริมาณข้อมูลที่รับได้ของคลัสเตอร์เฮดในแต่ละโหนดว่ามีความแตกต่างกันมากน้อยเท่าไร ถ้าหากผลการทดลองให้ค่าดัชนีชนิดนี้ต่ำ แสดงว่าอัลกอริทึมที่ทดลองสามารถกระจายภาระงานในแต่ละคลัสเตอร์ให้มีจำนวนเท่ากันได้ดีกว่า ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของภาระงานต่อโหนดปลายทาง 1 โหนด มีหน่วยเป็นบิตต่อวินาที (Bit per sec) คำนวณได้จากสมการที่ 5-2 เมื่อ  $i$  คือหมายเลขคลัสเตอร์เฮดที่  $0, 1, 2, \dots, k-1$  ตัวแปร  $k$  คือจำนวนคลัสเตอร์เฮด ตัวแปร  $TP_i$  คือ ปริมาณการส่งข้อมูลของคลัสเตอร์ที่  $i$  ตัวแปร  $TP_{avg}$  คือ ปริมาณการส่งข้อมูลเฉลี่ย ซึ่งได้จากสมการที่ 5-3 ตัวแปร *Received Data Packet* คือปริมาณข้อมูลที่รับสำเร็จในเครือข่าย และตัวแปร *Time* คือเวลาทั้งหมดที่ใช้ทดลอง

$$\text{Average Absolute SD of Load per Sink} = \frac{\sum_{i=0}^{k-1} |TP_{avg} - TP_i|}{k} \quad (5-2)$$

$$\text{Throughput (TP)} = \frac{\text{Received Data Packet}}{\text{Time}} \quad (5-3)$$

#### 2. การใช้พลังงานเฉลี่ยต่อคลัสเตอร์ (Average energy consumption per cluster)

ดัชนีชนิดนี้ใช้ประเมินการใช้พลังงานของคลัสเตอร์โดยเฉลี่ย มีหน่วยเป็นจูล (Joule) คำนวณได้ตามสมการที่ 5-4 เมื่อตัวแปร  $k$  คือ จำนวนคลัสเตอร์เฮด ตัวแปร *Total Energy Consumption* คือพลังงานทั้งหมดที่ใช้ในเครือข่าย ซึ่งได้จากสมการที่ 5-5 ตัวแปร  $E_i$  คือพลังงานที่ใช้ของโหนดที่  $i$  และตัวแปร  $n$  คือจำนวนโหนดสมาชิกทั้งหมดในเครือข่าย

$$\text{Average Energy Consumption per Cluster} = \frac{\text{Total Energy Consumption}}{k} \quad (5-4)$$

$$\text{Total Energy Consumption} = \frac{\sum_{i=0}^{n-1} E_i}{n} \quad (5-5)$$

### 3. ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์ (Average absolute standard deviation of energy consumption per cluster)

ดัชนีชนิดนี้ใช้ประเมินการใช้พลังงานในแต่ละคลัสเตอร์ว่ามีความแตกต่างกันมากน้อยเท่าไร ถ้าหากผลลัพธ์มีค่าน้อย แสดงว่าแต่ละคลัสเตอร์มีการกระจายการใช้พลังงานเท่ากัน ดีกว่า ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์มีหน่วยเป็นจูล (Joule) คำนวณได้จากสมการที่ 5-6 เมื่อ  $i$  คือหมายเลขคลัสเตอร์เฮดที่  $0, 1, 2, \dots, k-1$  ตัวแปร  $k$  คือจำนวนคลัสเตอร์เฮด ตัวแปร  $EC_i$  คือ การใช้พลังงานเฉลี่ยของคลัสเตอร์ที่  $i$  ตัวแปร  $EC_{avg}$  คือ การใช้พลังงานเฉลี่ยต่อคลัสเตอร์ ซึ่งได้จากสมการที่ 5-7 และ ตัวแปร  $\text{Total Energy Consumption}$  คือ พลังงานทั้งหมดที่ใช้ในเครือข่าย ซึ่งได้จากสมการที่ 5-5

$$\text{SD of Energy Consumption per Cluster} = \frac{\sum_{i=0}^{k-1} |EC_{avg} - EC_i|}{k} \quad (5-6)$$

$$\text{Energy Consumption per Cluster (EC)} = \frac{\text{Total Energy Consumption}}{k} \quad (5-7)$$

### 4. การใช้พลังงานเฉลี่ยของคลัสเตอร์เฮด (Average energy consumption of cluster head)

ดัชนีชนิดนี้ใช้ประเมินการใช้พลังงานของคลัสเตอร์เฮดโดยเฉลี่ย มีหน่วยเป็นจูล (joule) คำนวณได้จากสมการที่ 5-8 เมื่อ  $i$  คือหมายเลขคลัสเตอร์เฮดที่  $0, 1, 2, \dots, k-1$  ตัวแปร  $k$  คือจำนวนคลัสเตอร์เฮด ตัวแปร  $ECH_i$  คือ การใช้พลังงานของคลัสเตอร์เฮดที่  $i$

$$\text{Average Energy Consumption of CH} = \frac{\sum_{i=0}^{k-1} ECH_i}{k} \quad (5-8)$$

## 5. ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด (Average absolute standard deviation of energy consumption of cluster head)

ดัชนีชนิดนี้ใช้ประเมินการใช้พลังงานของคลัสเตอร์เฮดแต่ละโหนดว่ามีความแตกต่างกันมากน้อยเท่าไร ถ้าหากผลลัพธ์มีค่าน้อย แสดงว่าคลัสเตอร์เฮดแต่ละโหนดมีการกระจายการใช้พลังงานเท่ากันดีกว่า ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด มีหน่วยเป็นจูล (Joule) คำนวณได้จากสมการที่ 5-9 เมื่อ  $i$  คือหมายเลขคลัสเตอร์เฮดที่  $0, 1, 2, \dots, k-1$  ตัวแปร  $k$  คือ จำนวนคลัสเตอร์เฮด ตัวแปร  $ECH_{avg}$  คือ การใช้พลังงานเฉลี่ยของคลัสเตอร์เฮด ซึ่งได้จากสมการที่ 5-8 และตัวแปร  $ECH_i$  คือ การใช้พลังงานของคลัสเตอร์เฮดที่  $i$

$$SD \text{ of Energy Consumption of CH} = \frac{\sum_{i=0}^{k-1} |ECH_{avg} - ECH_i|}{k} \quad (5-9)$$

### 5.2 การจำลองกราฟ

การทดลองนี้จะเป็นการเปรียบเทียบสมรรถนะของอัลกอริทึม Grid Sectoring กับ LEACH, ACHS และ EACHS การทดลองแบ่งออกเป็น 2 ส่วนคือ สภาพแวดล้อมที่ 1 การทดลองในเครือข่ายที่โหนดมีการกระจายแบบสม่ำเสมอ และสภาพแวดล้อมที่ 2 การทดลองในเครือข่ายที่โหนดมีการกระจายแบบไม่สม่ำเสมอ โดยแต่ละสภาพแวดล้อมมีวิธีการทดลองและสภาพแวดล้อมหรือพารามิเตอร์แตกต่างกันดังนี้

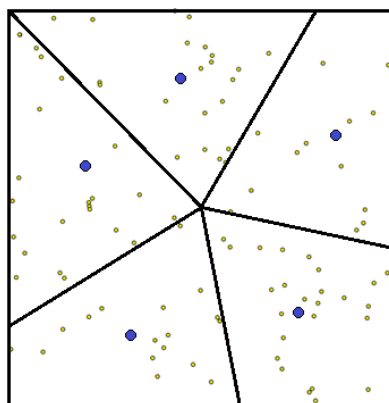
#### 5.2.1 สภาพแวดล้อมการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายแบบสม่ำเสมอ

การทดลองนี้เป็นการทดสอบความสมดุลของจำนวนสมาชิกในแต่ละคลัสเตอร์ โดยวัดจากค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย ทำการทดลองด้วยโปรแกรม Networkx กำหนดให้โหนดในเครือข่ายมีจำนวน 100, 200, 300, 400 และ 500 โหนดตามลำดับ โหนดกระจายด้วยความหนาแน่นสม่ำเสมอแบบสุ่มชนิดจีโอกราฟฟิก (Geographic random distribution) บนพื้นที่ขนาด  $100 \times 100$  ตารางเมตร จำนวนคลัสเตอร์เฮดที่ต้องการเท่ากับ 5 เปอร์เซนต์ของจำนวนโหนดทั้งหมดในเครือข่าย การแบ่งพื้นที่สนใจออกเป็นกริดย่อยและการแบ่งกริดย่อยเป็นเซกเตอร์ให้มีจำนวนคลัสเตอร์เฮดใกล้เคียง 5 เปอร์เซนต์แสดงไว้ในตาราง 5-1 เพื่อให้คลัสเตอร์เฮดในเครือข่ายของทุกอัลกอริทึมมีจำนวนเท่ากัน โดยจำนวนคลัสเตอร์เฮดของอัลกอริทึมที่นำเสนอคำนวณจากจำนวนกริดคูณด้วยจำนวนเซกเตอร์

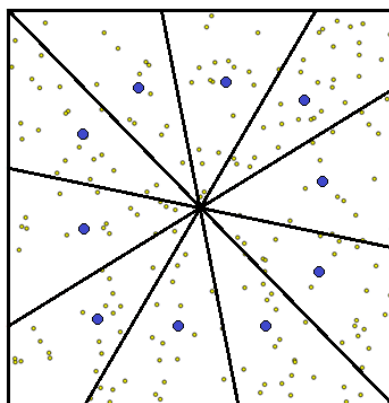
ตารางที่ 5-1 การแบ่งพื้นที่ของอัลกอริทึมกริดเซกเตอร์ตามจำนวนโหนดในเครือข่าย

จำนวนโหนด (โหนด)	การแบ่งพื้นที่สำหรับ อัลกอริทึม Grid Sectoring		จำนวนคลัสเตอร์เฮด		เปอร์เซ็นต์ ของคลัสต์ เตอร์เฮด
	จำนวนกริดย่อย	จำนวน เซกเตอร์	Grid Sectoring	อัลกอริทึม อื่นๆ	
100	1	5	5	5	5%
200	1	10	10	10	5%
300	4	4	16	16	5.33%
400	4	5	20	20	5%
500	4	6	24	24	4.80%

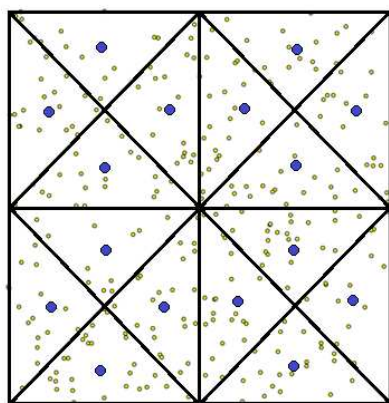
ตัวอย่างผลการแบ่งพื้นที่ของอัลกอริทึมกริดเซกเตอร์ ตามตาราง 5-1 ที่ทำการทดลองตามสภาพแวดล้อมที่ 1 ด้วยโปรแกรมจำลองกราฟ Networkx เมื่อโหนดในเครือข่ายมีขนาด 100, 200, 300, 400 และ 500 โหนด แสดงได้ดังภาพประกอบ 5-1, 5-2, 5-3, 5-4 และ 5-5 ตามลำดับ



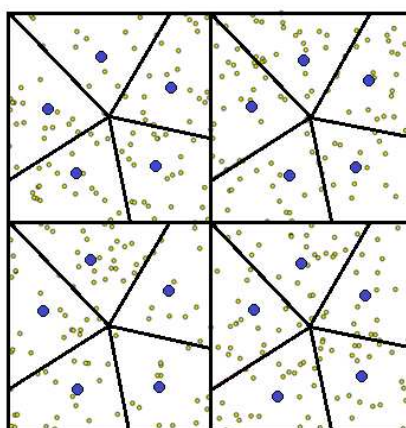
ภาพประกอบ 5-1 ผลการแบ่งพื้นที่ที่ออกเป็น 5 เซกเตอร์ เมื่อโหนดในเครือข่ายมีจำนวน 100 โหนด



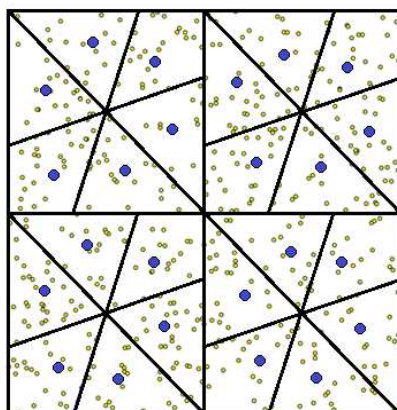
ภาพประกอบ 5-2 ผลการแบ่งพื้นที่ออกเป็น 10 เซกเตอร์ เมื่อโหนดในเครือข่ายมีจำนวน 200 โหนด



ภาพประกอบ 5-3 ผลการแบ่งพื้นที่ออกเป็น 4 กริด กริดละ 4 เซกเตอร์ เมื่อโหนดในเครือข่ายมี  
จำนวน 300 โหนด



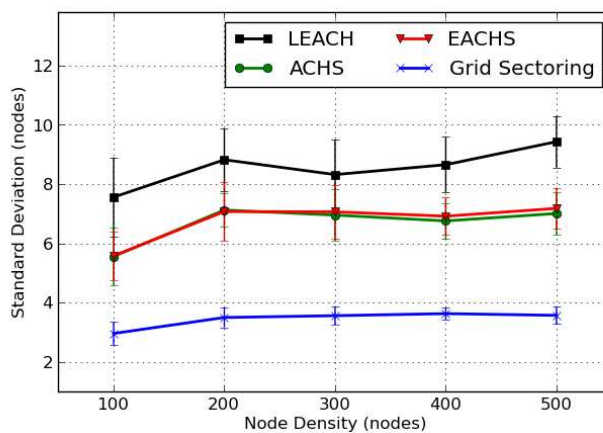
ภาพประกอบ 5-4 ผลการแบ่งพื้นที่ออกเป็น 4 กริด กริดละ 5 เซกเตอร์ เมื่อโหนดในเครือข่ายมี  
จำนวน 400 โหนด



ภาพประกอบ 5-5 ผลการแบ่งพื้นที่ออกเป็น 4 กริด กริดละ 6 เซกเตอร์ เมื่อโหนดในเครือข่ายมีจำนวน 500 โหนด

## 5.2.2 ผลการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายแบบสม่ำเสมอ

### 1. ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย

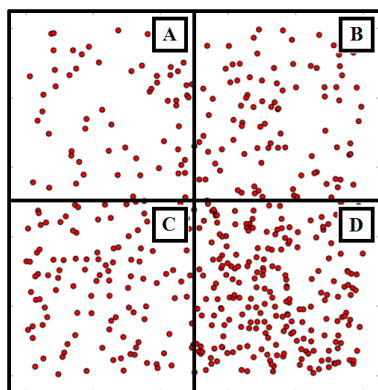


ภาพประกอบ 5-6 ผลการทดลองค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย

จากภาพประกอบที่ 5-6 จะเห็นได้ว่าค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของอัลกอริทึมกริดเซกเตอร์มีค่าต่ำที่สุดไม่ว่าจำนวนโหนดในเครือข่ายจะมีเท่าไร และค่าที่ได้จะมีค่าไม่เกิน 4 โหนด หมายความว่า การจัดคลัสเตอร์ด้วยอัลกอริทึมกริดเซกเตอร์ทำให้จำนวนโหนดสมาชิกในแต่ละคลัสเตอร์มีความแตกต่างกันไม่เกิน 4 โหนด โดยเฉลี่ย สำหรับอัลกอริทึม ACHS และ EACHS ให้ผลลัพธ์ที่ตรงลงมา คือให้ผลลัพธ์ค่าเบี่ยงเบนมาตรฐานที่ใกล้เคียงกันประมาณ 7 โหนด ส่วนอัลกอริทึม LEACH ให้ผลลัพธ์ค่าเบี่ยงเบนมาตรฐานสูงสุดประมาณ 9 โหนด

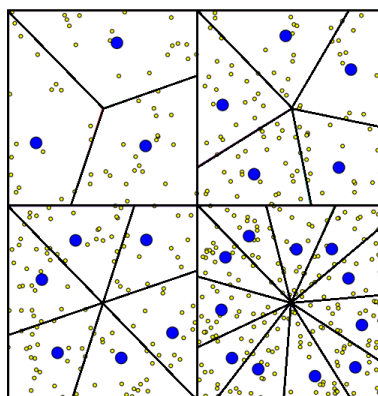
### 5.2.3 สภาพแวดล้อมการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายแบบไม่สม่ำเสมอ

การทดลองนี้เป็นการทดสอบความสมดุลของจำนวนสมาชิกในแต่ละคลัสเตอร์ โดยวัดจากค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย การทดลองใช้โปรแกรม Networkx เช่นเดียวกันกับหัวข้อที่ 5.2.1 แต่กำหนดให้โหนดในเครือข่ายมีจำนวน 500 โหนด เนื่องจากสามารถกำหนดให้โหนดมีการกระจายแบบไม่สม่ำเสมอได้อย่างชัดเจน พื้นที่การทดลองมีขนาด 100x100 ตารางเมตร จำนวนคลัสเตอร์ที่ต้องการเท่ากับ 5 เปอร์เซ็นต์ของจำนวนโหนดทั้งหมดในเครือข่าย เครือข่ายที่ทดสอบกำหนดให้มีความหนาแน่นแตกต่างกัน 4 ส่วน แสดงดังภาพประกอบที่ 5-9 โดยแต่ละส่วน A, B, C และ D จะมีโหนดจำนวน 60, 100, 120 และ 220 โหนด ตามลำดับ



ภาพประกอบ 5-7 การกระจายความหนาแน่นของโหนดที่แตกต่างกันในพื้นที่ 4 ส่วน

เมื่อพื้นที่แต่ละส่วนมีการกระจายของโหนดไม่เท่ากันทำให้จำนวนคลัสเตอร์เฮดในแต่ละพื้นที่มีไม่เท่ากัน เมื่อใช้คลัสเตอร์เฮดจำนวน 5 เปอร์เซ็นต์ การจัดคลัสเตอร์แบบกริดเชกเตอร์จึงมีคลัสเตอร์เฮดจำนวน 3, 5, 6 และ 11 ตามลำดับ ดังภาพประกอบที่ 5-10

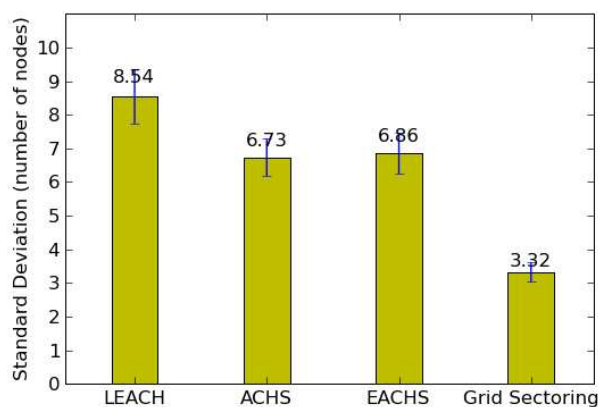


ภาพประกอบ 5-8 ผลการแบ่งกริดเป็นเชกเตอร์ของอัลกอริทึมกริดเชกเตอร์บนเครือข่ายที่โหนดมีการกระจายด้วยความหนาแน่นไม่สม่ำเสมอ



## 5.2.4 ผลการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายแบบไม่สม่ำเสมอ

### 1. ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย



ภาพประกอบ 5-9 ผลการทดลองค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย

จากภาพประกอบที่ 5-11 จะเห็นได้ว่าอัลกอริทึมที่นำเสนอให้ผลลัพธ์ค่าเบี่ยงเบนมาตรฐานต่ำที่สุดอยู่ที่ 3.32 โหนด ซึ่งผลลัพธ์ที่ได้น้อยกว่าอัลกอริทึมชนิดอื่นประมาณ 2 เท่า แสดงว่าอัลกอริทึมที่นำเสนอสามารถสมดุลจำนวนโหนดสมาชิกได้ดีกว่าแม้ว่าจะมีการกระจายตัวของโหนดในเครือข่ายแบบไม่สม่ำเสมอก็ตาม

## 5.3 การจำลองเครือข่าย

### 5.3.1 การจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบสม่ำเสมอ

การทดลองนี้เป็นการทดสอบความสมดุลของภาระงานและการใช้พลังงานในแต่ละคลัสเตอร์โดยวัดจากค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ย โหนดในเครือข่ายที่ทดสอบมีการกระจายตัวด้วยความหนาแน่นสม่ำเสมอ การจำลองการทำงานของเครือข่ายใช้โปรแกรม NS-2 และพารามิเตอร์ในการจำลองเครือข่ายแสดงได้ดังตาราง 5-2 พารามิเตอร์ที่กำหนดในการทดลองนี้ได้จากการทดสอบร่วมกับโปรโตคอล AODV ซึ่งแสดงไว้ในภาคผนวก ข ตารางที่ ข-4

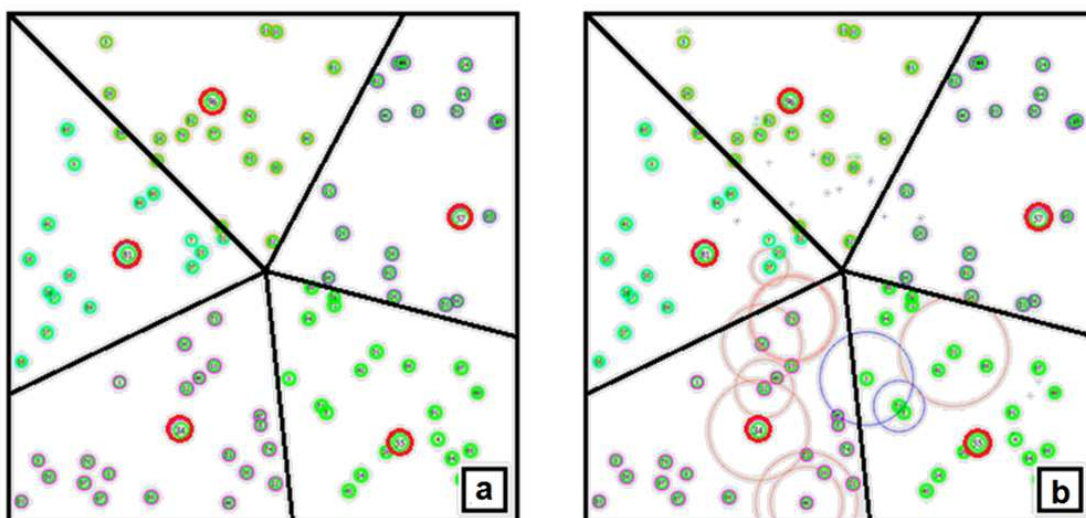
ตารางที่ 5-2 การแบ่งพื้นที่ของอัลกอริทึมกริดเซกเตอร์ตามจำนวนโหนดในเครือข่าย

พารามิเตอร์	ค่าที่กำหนด
ขนาดพื้นที่สนใจ (ตารางเมตร)	100x100
เวลาในการจำลองเครือข่าย (วินาที)	600 (10 นาที)
จำนวน โหนดในเครือข่าย (โหนด)	100

ตารางที่ 5-2 การแบ่งพื้นที่ของอัลกอริทึมกริดเชกเตอร์ตามจำนวนโหนดในเครือข่าย (ต่อ)

พารามิเตอร์	ค่าที่กำหนด
ชั้น MAC และ PHY ที่ใช้	IEEE 802.15.4
ความถี่ที่ใช้ในชั้น PHY (กิกะเฮิรต์)	2.4
แบนด์วิดท์ (กิโลบิตต่อวินาที)	250
ระยะการส่งสัญญาณวิทยุของโหนด (เมตร)	15
ชนิดของทราฟฟิก	CBR (constant bit rate)
ขนาดของแพ็กเก็ต (ไบต์)	50
ขนาดของบัฟเฟอร์ (ไบต์)	150
อัตราการส่งแพ็กเก็ตข้อมูล (แพ็กเก็ตต่อวินาที)	1
ชนิดของโพรโทคอลจัดเส้นทาง	AODV
คาบการส่งข้อความ HELLO ของโพรโทคอลจัดเส้นทาง (วินาที)	10
จำนวนฮอปสูงสุดในการส่งต่อแพ็กเก็ตควบคุมชนิด AODV (ฮอป)	30

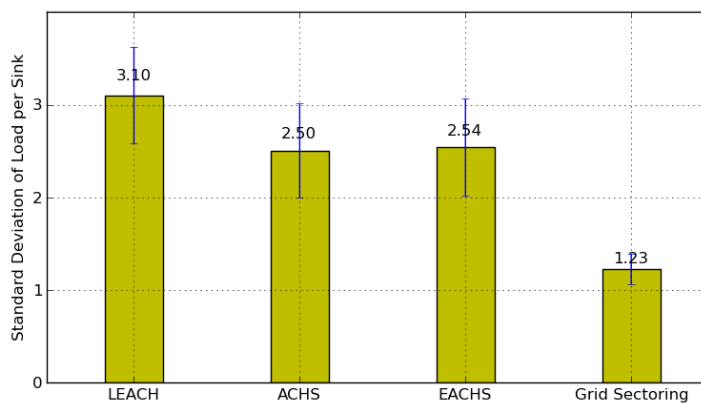
ผลการจำลองเครือข่ายในสภาพแวดล้อมที่ 1 ที่ทำงานร่วมกับอัลกอริทึมกริดเชกเตอร์ด้วยโปรแกรม NS-2 แสดงได้ดังภาพประกอบที่ 5-14 (a) และตัวอย่างผลการจำลองเครือข่ายแบบภาพเคลื่อนไหว (Animation) แสดงดังภาพประกอบที่ 5-14 (b)



ภาพประกอบ 5-10 ผลการจำลองเครือข่ายที่ใช้อัลกอริทึมกริดเชกเตอร์ เมื่อโหนดในเครือข่ายมีจำนวน 100 โหนด

### 5.3.2 ผลการจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบสม่ำเสมอ

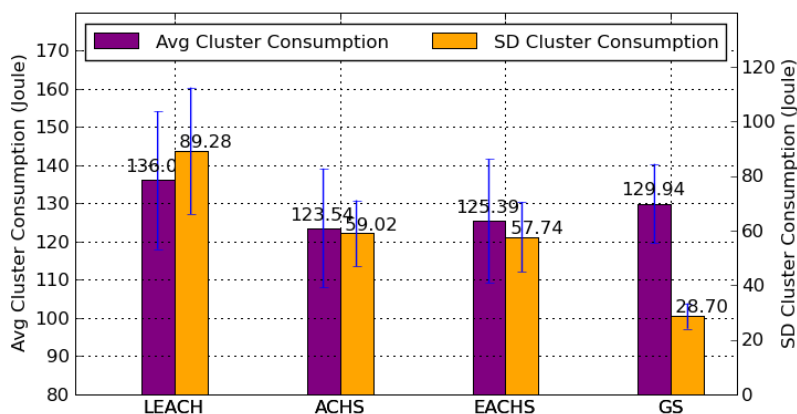
#### 1. ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของภาระงานต่อโหนดปลายทาง 1 โหนด



ภาพประกอบ 5-11 ผลการทดลองค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของภาระงาน

จากภาพประกอบที่ 5-15 จะเห็นได้ว่าผลลัพธ์ค่าเบี่ยงเบนมาตรฐานของอัลกอริทึมที่นำเสนอมีค่าต่ำสุด แสดงว่าแต่ละคลัสเตอร์สามารถกระจายภาระงานให้เท่ากัน ได้ดีกว่าอัลกอริทึมชนิดอื่น ซึ่งมีความแตกต่างของภาระงานไม่เกิน 1.23 กิโลบิตต่อวินาทีในแต่ละคลัสเตอร์

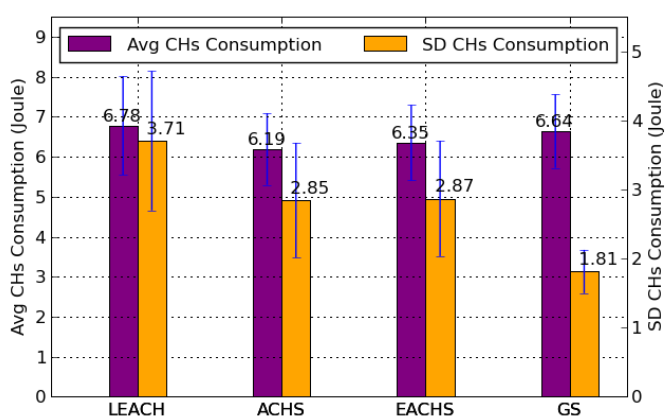
#### 2. การใช้พลังงานเฉลี่ยต่อคลัสเตอร์และค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์



ภาพประกอบ 5-12 ผลการทดลองการใช้พลังงานเฉลี่ยต่อคลัสเตอร์และค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์

ถึงแม้ว่าผลลัพธ์การใช้พลังงานเฉลี่ยในแต่ละคลัสเตอร์ของอัลกอริทึมที่นำเสนอ จะมีความใกล้เคียงกับอัลกอริทึมชนิดอื่น ดังภาพประกอบ 5-20 แต่ค่าเบี่ยงเบนมาตรฐานการใช้พลังงานในแต่ละคลัสเตอร์ของอัลกอริทึมที่นำเสนอกลับให้ผลลัพธ์ที่ดีกว่า คือให้ผลต่ำสุด แสดงว่า การใช้พลังงานในแต่ละคลัสเตอร์ของอัลกอริทึมที่นำเสนอมีความใกล้เคียงกันมากกว่า

### 3. การใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด



ภาพประกอบ 5-13 ผลการทดลองการใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด

จากภาพประกอบที่ 5-21 ผลลัพธ์การใช้พลังงานของคลัสเตอร์เฮดเฉลี่ยของอัลกอริทึมที่นำเสนอมีแนวโน้มไปในทิศทางเดียวกันกับการใช้พลังงานเฉลี่ยต่อคลัสเตอร์ คือผลลัพธ์ค่าเบี่ยงเบนมาตรฐานในการใช้พลังงานของคลัสเตอร์เฮดของอัลกอริทึมที่นำเสนอมีค่าต่ำที่สุด และต่ำกว่าอัลกอริทึมชนิดอื่นประมาณ 1 จูล แสดงว่าคลัสเตอร์เฮดแต่ละโหนดของอัลกอริทึมที่นำเสนอมีการใช้พลังงานใกล้เคียงกันมากกว่า

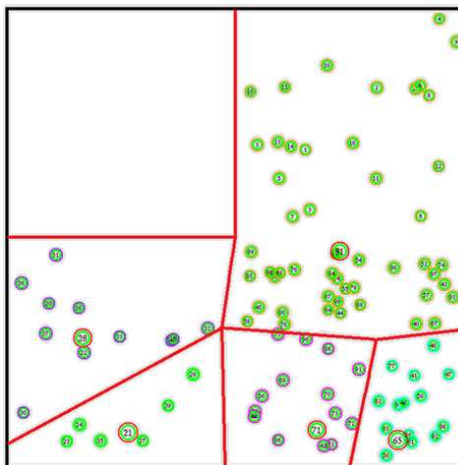
#### 5.3.3 การจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบไม่สม่ำเสมอ

การทดลองนี้เป็นการทดสอบความสมดุลของภาระงานและการใช้พลังงานในแต่ละคลัสเตอร์โดยวัดจากค่าเบี่ยงเบนมาตรฐานสัมบูรณ์บนสภาพแวดล้อมที่โหนดมีการกระจายแบบไม่สม่ำเสมอ โดยแบ่งเครือข่ายออกเป็น 4 กริดย่อยที่มีจำนวนโหนดแตกต่างกันแสดงดังภาพประกอบที่ 5-22 และพารามิเตอร์ต่างๆ ของการจำลองเครือข่ายด้วยโปรแกรม NS-2 ถูกกำหนดเช่นเดียวกันกับการทดลองในหัวข้อที่ 5.3.1 ดังตารางที่ 5-2

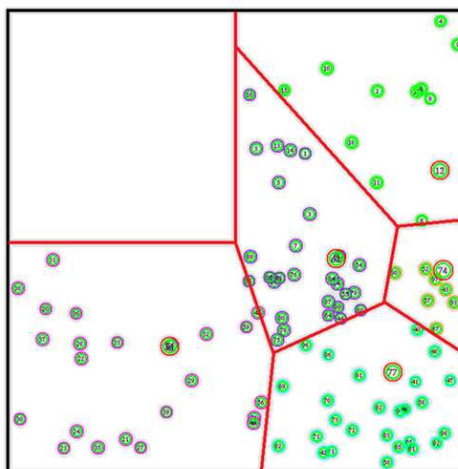
0 โหนด	20 โหนด
20 โหนด	60 โหนด

ภาพประกอบ 5-14 การกระจายความหนาแน่นของโหนดในแต่ละพื้นที่ 4 ส่วน

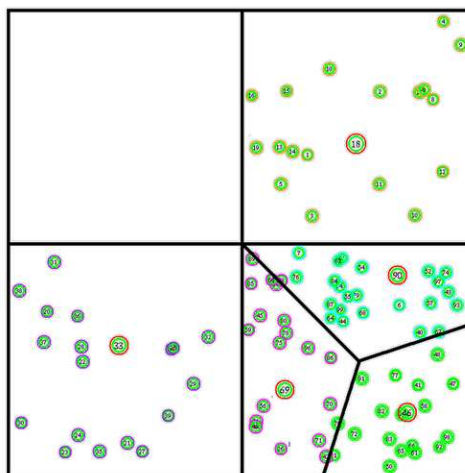
ผลการจำลองเครือข่ายด้วยโปรแกรมจำลองเครือข่าย NS-2 บนเครือข่ายตามสภาพแวดล้อมที่ 2 ที่ทำงานร่วมกับอัลกอริทึม LEACH, ACHS และอัลกอริทึมกริดเชกเตอร์ แสดงได้ดังภาพประกอบที่ 5-23 ถึง 5-25 ตามลำดับ



ภาพประกอบ 5-15 ผลการจำลองเครือข่ายตามสภาพแวดล้อมที่ 2 ด้วยอัลกอริทึม LEACH



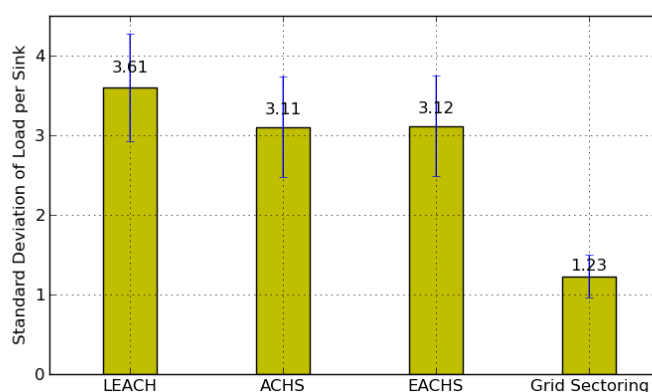
ภาพประกอบ 5-16 ผลการจำลองเครือข่ายตามสภาพแวดล้อมที่ 2 ด้วยอัลกอริทึม ACHS



ภาพประกอบ 5-17 ผลการจำลองเครือข่ายตามสภาพแวดล้อมที่ 2 ด้วยอัลกอริทึม Grid Sectoring

### 5.3.4 ผลการจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบไม่สม่ำเสมอ

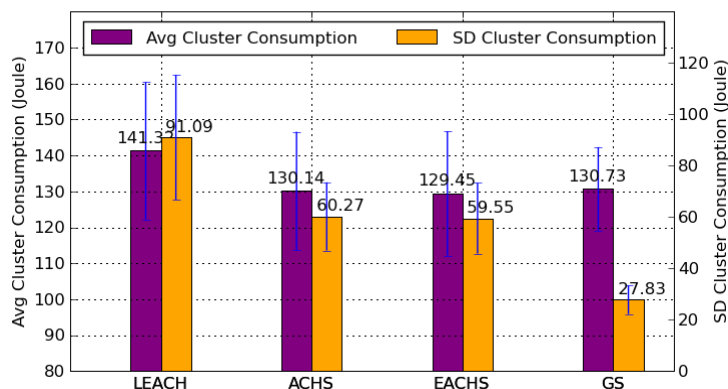
#### 1. ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของภาระงานต่อโหนดปลายทาง 1 โหนด



ภาพประกอบ 5-18 ผลการทดลองค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยของภาระงาน

จากภาพประกอบที่ 5-26 จะเห็นได้ว่าผลลัพธ์ค่าเบี่ยงเบนมาตรฐานของอัลกอริทึมที่นำเสนอมีค่าต่ำสุด หมายความว่าแต่ละคลัสเตอร์สามารถกระจายภาระงานให้เท่ากันได้ดีกว่าอัลกอริทึมชนิดอื่น โดยแต่ละคลัสเตอร์มีภาระงานแตกต่างกันประมาณ 1.23 กิโลบิตต่อวินาทีต่อคลัสเตอร์ ซึ่งน้อยกว่าอัลกอริทึมชนิดอื่นประมาณ 2 กิโลบิตต่อวินาทีต่อคลัสเตอร์

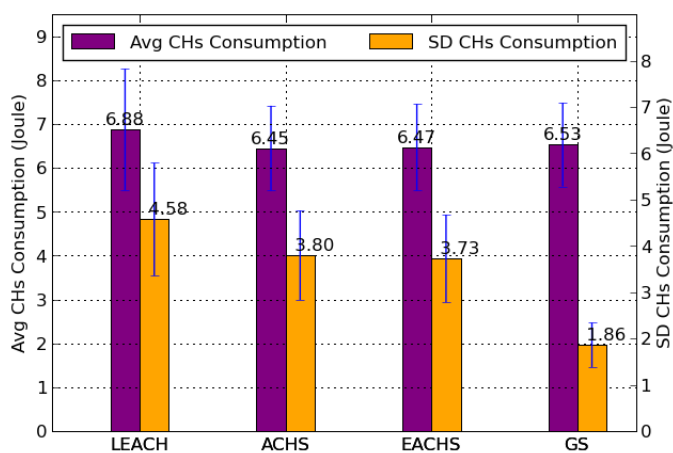
## 2. การใช้พลังงานเฉลี่ยต่อคลัสเตอร์และค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์



ภาพประกอบ 5-19 ผลการทดลองการใช้พลังงานเฉลี่ยต่อคลัสเตอร์และค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์

การใช้พลังงานเฉลี่ยต่อคลัสเตอร์ของอัลกอริทึมที่นำเสนอมีความใกล้เคียงกับอัลกอริทึมชนิดอื่น โดยให้ผลลัพธ์ประมาณ 130 จูลต่อคลัสเตอร์ ดังภาพประกอบ 5-28 แต่ค่าเบี่ยงเบนมาตรฐานการใช้พลังงานต่อคลัสเตอร์ อัลกอริทึมที่นำเสนอให้ผลลัพธ์ต่ำกว่า แสดงว่าอัลกอริทึมที่นำเสนอมีการใช้พลังงานในแต่ละคลัสเตอร์ใกล้เคียงกันมากกว่า

## 3. การใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด



ภาพประกอบ 5-20 ผลการทดลองการใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด

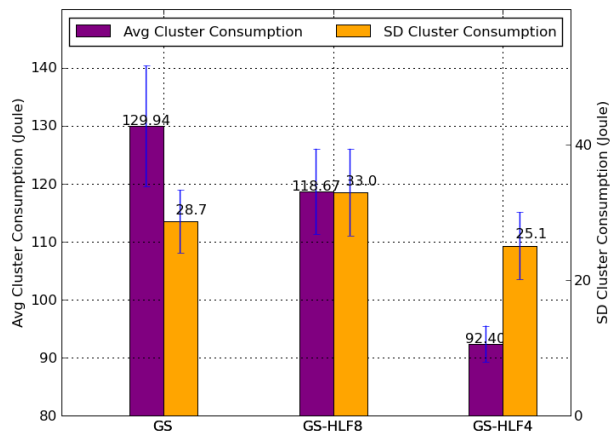
จากภาพประกอบที่ 5-29 ผลลัพธ์การใช้พลังงานของคลัสเตอร์เฮดเฉลี่ยของ อัลกอริทึมที่นำเสนอมีแนวโน้มไปในทิศทางเดียวกันกับการใช้พลังงานเฉลี่ยต่อคลัสเตอร์ อย่างไรก็ตามเมื่อพิจารณาผลลัพธ์ค่าเบี่ยงเบนมาตรฐานในการใช้พลังงานของคลัสเตอร์เฮดแล้ว พบว่า อัลกอริทึมที่นำเสนอให้ผลลัพธ์ต่ำกว่า แสดงว่าคลัสเตอร์เฮดแต่ละโหนดใช้พลังงานใกล้เคียงกันมากกว่าอัลกอริทึมชนิดอื่น

#### 5.4 ผลการทดลองจำนวนฮอปของอัลกอริทึมกริดเซกเตอร์บนโพรโทคอล AODV

การทดลองนี้เป็นการทดสอบประสิทธิภาพการใช้พลังงานของอัลกอริทึมการจัด คลัสเตอร์แบบกริดเซกเตอร์ที่มีการปรับปรุงให้จำกัดจำนวนฮอปในการส่งแพ็กเก็ตเกิดควบคุมของ โพรโทคอล AODV เพื่อให้ประหยัดพลังงานมากยิ่งขึ้น ซึ่งรายละเอียดการปรับปรุงแสดงไว้ใน หัวข้อที่ 4.2 อัลกอริทึมที่นำเสนอแบบดั้งเดิมจะใช้ชื่อ Grid Sectoring หรือ GS แต่สำหรับ อัลกอริทึมการจัดคลัสเตอร์แบบกริดเซกเตอร์ที่ได้รับการปรับปรุงให้จำกัดจำนวนฮอป จะใช้ชื่อ Grid Sectoring with Hop Limited Flooding หรือ GS-HLF ตัวอย่างเช่น GS-HLF8 คือ อัลกอริทึมที่มีการจำกัดจำนวนฮอปในการส่งแพ็กเก็ตเกิดควบคุม 8 ฮอป โดยปกติแล้วจำนวนฮอปของการส่งต่อ แพ็กเก็ตเกิดควบคุมของโพรโทคอลจัดเส้นทางชนิด AODV ที่ถูกกำหนดเริ่มต้นโดยโปรแกรม NS-2 จะจำกัดไว้ที่ 30 ฮอป แต่การจำลองเครือข่ายของการทดลองนี้จะเลือกการจำกัดจำนวนฮอปไว้ที่ 8 และ 4 ฮอป ซึ่งคิดเป็น 4 เท่าและ 2 เท่าของจำนวนฮอปเฉลี่ยจากโหนดสมาชิกไปยังคลัสเตอร์เฮดที่ได้จากการทดลองในภาคผนวก ก1.3 ซึ่งเป็นการทดสอบบนเครือข่าย 100 โหนด และจากการ ทดลองในภาคผนวก ก1.3 ทำให้ได้ผลลัพธ์จำนวนฮอปเฉลี่ยประมาณ 2 ฮอป สำหรับการจำลอง เครือข่ายในหัวข้อนี้จะใช้โหนดจำนวน 100 โหนดเช่นเดียวกัน และใช้พารามิเตอร์เดียวกันกับการ ทดลองในหัวข้อที่ 5.3.1 ดังตารางที่ 5-2 ผลการทดลองแสดงได้ดังต่อไปนี้



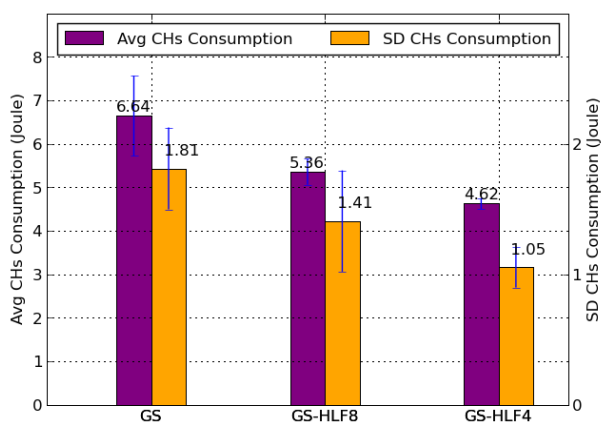
## 1. การใช้พลังงานเฉลี่ยต่อคลัสเตอร์และค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์



ภาพประกอบ 5-21 ผลการทดลองการใช้พลังงานเฉลี่ยต่อคลัสเตอร์และค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานต่อคลัสเตอร์

จากภาพประกอบ 5-30 การจำกัดจำนวนฮอป 8 ฮอปและ 4 ฮอปของอัลกอริทึมกริดเซกเตอร์ทำให้การใช้พลังงานต่อคลัสเตอร์ลดลง โดยผลลัพธ์ที่ได้ประมาณ 118 และ 92 จูลต่อคลัสเตอร์ตามลำดับ แต่ค่าเบี่ยงเบนมาตรฐานการใช้พลังงานแต่ละคลัสเตอร์ของอัลกอริทึมที่นำเสนอทั้งแบบดั้งเดิมและแบบที่จำกัดจำนวนฮอป ให้ผลลัพธ์ไม่แตกต่างกัน คือ 25 ถึง 33 จูล แสดงว่าการจำกัดจำนวนฮอปไม่ได้ส่งผลให้แต่ละคลัสเตอร์มีการใช้พลังงานใกล้เคียงกันมากขึ้น

## 2. การใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด



ภาพประกอบ 5-22 ผลการทดลองการใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยในการใช้พลังงานของคลัสเตอร์เฮด

จากภาพประกอบที่ 5-31 ทั้งผลลัพธ์การใช้พลังงานเฉลี่ยของคลัสเตอร์เฮดและค่าเบี่ยงเบนมาตรฐานในการใช้พลังงานของคลัสเตอร์เฮดของอัลกอริทึมที่นำเสนอทั้งแบบดั้งเดิมและแบบจำกัดจำนวนฮอปมีค่าลดลงเมื่อจำนวนฮอปที่จำกัดลดลง การที่ค่าเบี่ยงเบนมาตรฐานในการใช้พลังงานของคลัสเตอร์เฮดลดลง แสดงว่าเมื่อมีการจำกัดจำนวนฮอปให้สามารถส่งต่อแพ็กเก็ตได้น้อยลงจะทำให้การใช้พลังงานของคลัสเตอร์เฮดใกล้เคียงกันมากยิ่งขึ้น

## 5.5 บทสรุปและการวิเคราะห์เพิ่มเติม

### 5.5.1 สรุปผลการจำลองกราฟ

จากผลการทดลองทั้งในเครือข่ายที่โหนดมีการกระจายด้วยความหนาแน่นสม่ำเสมอและไม่สม่ำเสมอ อัลกอริทึมที่นำเสนอให้ผลลัพธ์ค่าเบี่ยงเบนมาตรฐานสัมบูรณ์เฉลี่ยน้อยที่สุด โดยแต่ละคลัสเตอร์มีจำนวนโหนดสมาชิกแตกต่างกันไม่เกิน 4 โหนด แสดงว่าอัลกอริทึมที่นำเสนอสามารถสมดุลจำนวนโหนดสมาชิกในแต่ละคลัสเตอร์ได้ดีกว่าอัลกอริทึมชนิดอื่น

### 5.5.2 สรุปผลการจำลองเครือข่าย

จากการจำลองเครือข่าย ผลลัพธ์ในเรื่องค่าเบี่ยงเบนมาตรฐานของภาระงานต่อ 1 โหนดปลายทางของอัลกอริทึมที่นำเสนอมีค่าต่ำกว่าอัลกอริทึมอื่น แสดงว่าปริมาณภาระงานในแต่ละคลัสเตอร์มีความใกล้เคียงกันมากกว่า โดยแต่ละคลัสเตอร์มีภาระงานแตกต่างกันไม่เกิน 1.2 กิโลบิตต่อวินาทีซึ่งดีกว่าอัลกอริทึมอื่น 52.0 เปอร์เซ็นต์ ผลที่ได้ดังกล่าวมีความสอดคล้องกับผลลัพธ์ในเรื่องค่าเบี่ยงเบนมาตรฐานการใช้พลังงานของคลัสเตอร์และการใช้พลังงานของคลัสเตอร์เฮดของอัลกอริทึมที่นำเสนอ คือให้ผลลัพธ์ต่ำกว่าอัลกอริทึมอื่นเช่นกัน โดยมีการใช้พลังงานในแต่ละคลัสเตอร์ที่แตกต่างกันไม่เกิน 28.7 จูลซึ่งดีกว่าอัลกอริทึมอื่น 50.3 เปอร์เซ็นต์ และคลัสเตอร์เฮดแต่ละโหนดมีการใช้พลังงานที่แตกต่างกันไม่เกิน 1.81 จูลซึ่งดีกว่าอัลกอริทึมอื่น 36.5 เปอร์เซ็นต์ จึงสรุปได้ว่า อัลกอริทึมการจัดคลัสเตอร์แบบกริดเชกเตอร์สามารถกระจายภาระงานและการใช้พลังงานในแต่ละคลัสเตอร์ได้สมดุลกว่าอัลกอริทึมอื่น

จากการปรับปรุงอัลกอริทึมการจัดคลัสเตอร์แบบกริดเชกเตอร์ให้สามารถจำกัดจำนวนฮอปเพื่อทำงานร่วมกับโปรโตคอลจัดเส้นทาง AODV ให้มีการใช้พลังงานอย่างมีประสิทธิภาพ เมื่อมีการจำกัดจำนวนฮอปให้สามารถส่งได้สูงสุด 8 ฮอป และ 4 ฮอป ผลลัพธ์ที่ได้ดีขึ้นตามลำดับ โดยเฉพาะการจำกัดที่ 4 ฮอป สามารถลดการใช้พลังงานเฉลี่ยในแต่ละคลัสเตอร์และลดการใช้พลังงานของคลัสเตอร์เฮดแต่ละโหนดมากกว่าวิธีที่ไม่ได้มีการปรับปรุงประมาณ 37.54 และ 2.02 จูลตามลำดับ ซึ่งคิดเป็น 28.9 และ 30.4 เปอร์เซ็นต์ตามลำดับ

## บทที่ 6

### สรุปผลการวิจัยและข้อเสนอแนะ

หลังจากได้ทำการศึกษาและพัฒนาอัลกอริทึมการจัดคลัสเตอร์แบบกริดเซกเตอร์ ดังรายละเอียดในหัวข้อที่ 3 ถึง 5 ในบทนี้จะกล่าวถึงการสรุปผลการวิจัย ปัญหาที่เกิดขึ้นขณะทำงานวิจัยและข้อเสนอแนะ เพื่อเป็นประโยชน์ต่อผู้ที่สนใจหรือต้องการจะนำงานวิจัยชิ้นนี้ไปศึกษาและเป็นแนวทางในการพัฒนางานวิจัยทางด้านเครือข่ายเซนเซอร์ไร้สาย

#### 6.1 สรุปผลการวิจัย

จากการศึกษา ผู้วิจัยพิจารณาเลือกประเด็นการจัดคลัสเตอร์เพื่อสมมูลภาระงานและการใช้พลังงานให้เท่ากันทุกคลัสเตอร์ที่สามารถประยุกต์ใช้งานได้ทั้งในเครือข่ายที่โหนดมีการกระจายตัวแบบสม่ำเสมอและไม่สม่ำเสมอมาเป็นประเด็นวิจัย โดยเริ่มจากการศึกษาอัลกอริทึมการจัดคลัสเตอร์เฮดที่มีวัตถุประสงค์เพื่อสมมูลภาระงานและการใช้พลังงานในทุกคลัสเตอร์ ได้แก่ อัลกอริทึม LEACH, ACHS และ EACHS จากการศึกษาอัลกอริทึมเหล่านี้พบว่า มีข้อจำกัดคือมีการเลือกคลัสเตอร์เฮดแบบสุ่ม แล้วนำค่าที่สุ่มได้มาตัดสินด้วยสมการความน่าจะเป็น ทำให้ไม่สามารถกระจายภาระงานและการใช้พลังงานให้สมมูลทุกคลัสเตอร์ได้อย่างมีประสิทธิภาพ นอกจากนี้ทำให้จำนวนคลัสเตอร์เฮดที่เลือกมีจำนวนคลาดเคลื่อนจากจำนวนที่ต้องการ ซึ่งส่งผลต่อประสิทธิภาพการใช้พลังงานของเครือข่าย ต่อมาได้ทำการศึกษาการจัดคลัสเตอร์ในรูปแบบกริดและแบบหกเหลี่ยม พบว่ามีข้อดีคือทำให้ได้คลัสเตอร์ที่มีขนาดพื้นที่เท่ากัน ส่งผลให้สามารถกระจายภาระงานและการใช้พลังงานในแต่ละคลัสเตอร์เท่ากันเมื่อโหนดมีการกระจายด้วยความหนาแน่นสม่ำเสมอ แต่การจัดคลัสเตอร์แบบกริดและแบบหกเหลี่ยมไม่สามารถกำหนดจำนวนคลัสเตอร์เฮดตามที่ต้องการได้อย่างอิสระ และไม่สามารถนำมาประยุกต์ใช้บนเครือข่ายที่โหนดมีการกระจายตัวด้วยความหนาแน่นไม่สม่ำเสมอได้อย่างมีประสิทธิภาพ นอกจากนี้ผู้วิจัยได้ทำการศึกษาหาพารามิเตอร์ของเครือข่ายที่ใช้โพรโทคอล AODV เพื่อใช้ออกแบบสภาพแวดล้อมในการจำลองเครือข่ายที่มีการทำงานร่วมกับอัลกอริทึมที่นำเสนอ โดยการศึกษาทั้งหมดที่กล่าวมาข้างต้นแสดงในหัวข้อที่ 3

ในหัวข้อที่ 4 เป็นรายละเอียดการออกแบบอัลกอริทึมการจัดคลัสเตอร์แบบกริดเซกเตอร์ (Grid Sectoring) ซึ่งเป็นอัลกอริทึมที่ควบคุมการทำงานโดยสถานีฐานเพื่อให้มีการใช้พลังงานเครือข่ายอย่างประหยัด สามารถกระจายภาระงานและการใช้พลังงานทุกคลัสเตอร์อย่างสมดุลบนเครือข่ายที่โหนดมีการกระจายตัวด้วยความหนาแน่นสม่ำเสมอและไม่สม่ำเสมอได้

นอกจากนี้ยังปรับปรุงอัลกอริทึมที่นำเสนอให้สามารถใช้งานร่วมกับโปรโตคอลจัดเส้นทาง AODV อย่างมีประสิทธิภาพด้านการใช้พลังงาน หลังจากนั้นจึงได้ทำการทดลองทั้งการจำลองกราฟและการจำลองเครือข่าย ตลอดจนการวิเคราะห์ผลซึ่งได้แสดงรายละเอียดไว้ในหัวข้อที่ 5

จากการจำลองกราฟและการจำลองเครือข่ายที่ทำงานร่วมกับอัลกอริทึมที่นำเสนอ โดยเปรียบเทียบกับอัลกอริทึม LEACH, ACHS และ EACHS พบว่าอัลกอริทึมที่นำเสนอมีการกระจายภาระงานและการใช้พลังงานได้สมดุลกว่าอัลกอริทึมชนิดอื่น คือแต่ละคลัสเตอร์มีความแตกต่างของจำนวนโหนดสมาชิกไม่เกิน 4 โหนด มีปริมาณภาระงานต่อคลัสเตอร์แตกต่างกันไม่เกิน 1.2 กิโลบิตต่อวินาทีต่อคลัสเตอร์ซึ่งดีกว่าอัลกอริทึมอื่น 52.0 เปอร์เซ็นต์ ส่วนการใช้พลังงานของแต่ละคลัสเตอร์แตกต่างกันไม่เกิน 28.7 จูลซึ่งดีกว่าอัลกอริทึมอื่น 50.3 เปอร์เซ็นต์ และมีการใช้พลังงานของคลัสเตอร์เสดแตกต่างกันไม่เกิน 1.81 จูลซึ่งดีกว่าอัลกอริทึมอื่น 36.5 เปอร์เซ็นต์ นอกจากนี้ผลการปรับปรุงอัลกอริทึมที่นำเสนอให้มีการจำกัดจำนวนฮอปในการส่งต่อแพ็กเก็ตควบคุมให้สามารถส่งได้สูงสุด 8 ฮอป และ 4 ฮอป ผลที่ได้ทำให้สามารถลดการใช้พลังงานที่ใช้ต่อคลัสเตอร์และลดการใช้พลังงานของคลัสเตอร์เสดแต่ละโหนดได้ประมาณ 37.54 และ 2.02 จูลตามลำดับ ซึ่งลดลงจากวิธีแบบดั้งเดิมคิดเป็น 28.9 และ 30.4 เปอร์เซ็นต์ตามลำดับ

## 6.2 ปัญหาและข้อเสนอแนะ

### 6.2.1 ปัญหาจากการดำเนินงานวิจัย

จากการศึกษาและดำเนินงานวิจัยในประเด็นเรื่องการเลือกคลัสเตอร์เสดเพื่อสมดุลภาระบนเครือข่ายเซนเซอร์ไร้สายพบว่า อัลกอริทึมที่นำเสนอจำเป็นต้องแบ่งพื้นที่สนใจเป็นลักษณะกริดย่อยก่อนทุกครั้ง ถ้าหากแบ่งพื้นที่เป็นกริดย่อยไม่ได้ จะทำให้ไม่สามารถดำเนินการจัดคลัสเตอร์ต่อได้ อย่างไรก็ตามการแบ่งเป็นกริดย่อยนั้นไม่จำเป็นต้องมีขนาดเท่ากันทุกกริด นอกจากนี้ อัลกอริทึมการจัดคลัสเตอร์แบบกริดเชกเตอร์จะไม่สามารถดำเนินการได้ถ้าโหนดไม่ได้รับการติดตั้งเซนเซอร์ที่สามารถรับรู้ตำแหน่งทางภูมิศาสตร์ของตัวเองหรือ GPS เนื่องจากสถานีฐานซึ่งเป็นโหนดที่ควบคุมการทำงานของอัลกอริทึมจำเป็นต้องใช้ข้อมูลตำแหน่งทางภูมิศาสตร์ของทุกโหนดมาประมวลผลการเลือกคลัสเตอร์เสดและการจัดคลัสเตอร์

### 6.2.2 ข้อเสนอแนะ

จากการทำวิจัยเรื่องอัลกอริทึมการจัดคลัสเตอร์แบบกริดเชกเตอร์ที่ได้นำเสนอ ผู้วิจัยพบว่า ยังมีประเด็นที่น่าสนใจและท้าทายในการศึกษา 2 ประเด็นดังนี้

1. ด้วยวิธีการแบ่งพื้นที่ของอัลกอริทึมที่ได้นำเสนอทำให้อัลกอริทึมการจัดคลัสเตอร์ชนิดนี้สามารถพัฒนาต่อยอดให้มีประสิทธิภาพและมีสมรรถนะมากขึ้น โดยเฉพาะในเรื่องสมดุลภาระงานและการใช้พลังงานของคลัสเตอร์เฮดสำหรับเครือข่ายที่มีการพิจารณาตำแหน่งของสถานีฐานเข้ามาเกี่ยวข้อง กล่าวคือปัญหาของเครือข่ายเซนเซอร์ไร้สายเมื่อทำงานไปเป็นระยะเวลาหนึ่ง คลัสเตอร์ที่อยู่ใกล้กับสถานีฐานจะหมดพลังงานเป็นอันดับแรก และเป็นบริเวณที่เกิดปัญหาคอขวดในการสื่อสารข้อมูลไปยังสถานีฐาน ดังนั้นหากมีการศึกษาถึงพารามิเตอร์ที่เกี่ยวข้องกับภาระงานภายในคลัสเตอร์และภาระงานที่เกิดจากการส่งต่อข้อมูลผ่านคลัสเตอร์เฮดจากคลัสเตอร์หนึ่งไปยังสถานีฐาน จะสามารถนำมาเป็นตัวแปรในการพิจารณาแบ่งจำนวนกริดย่อยตลอดจนจำนวนเชกเตอร์ให้มีจำนวนที่เหมาะสมตามตำแหน่งทางภูมิศาสตร์ กล่าวคือ ในบริเวณที่อยู่ใกล้สถานีฐานควรแบ่งพื้นที่ให้มีคลัสเตอร์จำนวนมาก แต่ให้มีโหนดสมาชิกในแต่ละคลัสเตอร์จำนวนน้อย เพื่อการกระจายการใช้พลังงานของคลัสเตอร์เฮดที่ทำหน้าที่ทั้งรับข้อมูลจากโหนดสมาชิกและส่งต่อข้อมูลจากคลัสเตอร์อื่นไปยังสถานีฐาน
2. การศึกษาประสิทธิภาพและสมรรถนะของเครือข่ายที่เป็นเครือข่ายเคลื่อนที่ (Mobile network) ที่มีอัตราเร็วและระยะการสื่อสารเข้ามาเกี่ยวข้อง ถ้าเป็นเครือข่ายเคลื่อนที่และมีโหนดกระจายตัวด้วยความหนาแน่นไม่สม่ำเสมอ จะทำให้การจัดคลัสเตอร์ในสภาพแวดล้อมแบบนี้มีความน่าสนใจมากขึ้น สำหรับอัลกอริทึมที่นำเสนอสามารถพัฒนาให้ใช้ในเครือข่ายเคลื่อนที่ที่มีโหนดในเครือข่ายกระจายตัวด้วยความหนาแน่นไม่สม่ำเสมอได้ ด้วยการปรับเปลี่ยนคาบเวลาในการสร้างคลัสเตอร์ใหม่ทั้งเครือข่าย (Cluster formation) ให้สัมพันธ์กับความเร็วในการเคลื่อนที่ของโหนดในเครือข่าย นอกจากนี้การเปลี่ยนแปลงความหนาแน่นของโหนดในพื้นที่ต่างๆ ที่เกิดจากการเคลื่อนที่ของโหนด อัลกอริทึมที่นำเสนอยังสามารถนำมาประยุกต์ใช้งานได้เนื่องจากสามารถปรับเปลี่ยนขนาดของคลัสเตอร์ตามความหนาแน่นของโหนดที่กระจายในบริเวณต่างๆ ได้อย่างมีประสิทธิภาพ

## บรรณานุกรม

- [1] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS), Maui, HI, Jan. 2000.
- [2] E.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," Transactions on Wireless Communications, IEEE, Vol.1, No.4, October 2002.
- [3] C. Nam, H. Jeong, and D. Shin, "The Adaptive Cluster Head Selection in Wireless Sensor Networks," International Workshop on Semantic Computing and Application, IWSCA'08, IEEE, pp. 147-149, July 2008.
- [4] C. Nam, Y. Ku, J. Yoon, and D. Shin, "Cluster Head Selection for Equal Cluster Size in Wireless Sensor Networks," Information Conference on New Trends in Information and Service Science, NISS, IEEE, pp. 618-623, July 2009.
- [5] Y. Zhuang, J. Pan, and G. Wu, "Energy-Optimal Grid-Based Clustering in Wireless Microsensor Networks with Data Aggregation," International Journal of Parallel, Emergent and Distributed Systems, Vol.25, No.6, pp. 531-550, December 2010.
- [6] W. Liu, Y. Sun, J. Yu, and B. Cao, "A Grid-based Distributed Multi-hop Routing Protocol for Wireless Sensor Networks," in Proceedings of the 5th International Conference on Embedded and Ubiquitous Computing, EUC'08, pp. 330-336, December 2008.
- [7] L. Li, X. Jiang, S. Zhong, and L. Hu, "Energy Balancing Clustering Algorithm for Wireless Sensor Network," in Proceedings of International conference on Networks Security, Wireless Communications and Trusted Computing, NSWCTC'09, IEEE, pp.61-64, May 2009.
- [8] M. Zhang, C. Gong, Y. Feng, and Y. Lu, "A Novel Energy-Efficient Dynamic Target Tracking Protocol for Wireless Sensor Networks," in Proceedings of International Conference on Networks Security, Wireless Communications and Trusted Computing: NSWCTC'09, IEEE, pp.649-652, May 2009.

- [9] G. Xin, W. YongXin and L. Fang, "An Energy-Efficient Clustering Technique for Wireless Sensor Networks," in Proceedings of International Conference on Networking, Architecture, and Storage, NAS'08, IEEE, pp. 248-252, July 2008.
- [10] R. Zhang, Lin Wang, Shicho Geng and Zhiping Jia, "A Balanced Cluster Routing Protocol of Wireless Sensor Network," in Proceedings of International Conference on Embedded Software and System Symposia, ICESS'08, IEEE, pp. 211-225, September 2008.
- [11] IEEE Computer Society, "IEEE Standard 802.15.4-2003 (draft), Part 15.4: Wireless Medium Access Control and Physical Layer Specifications for Low-Rate Wireless Personal Area Networks," IEEE Standards, 2006.
- [12] J. Zheng and Myung J. Lee, "Will IEEE 802.15.4 make ubiquitous networking a reality?: a discussion on a potential low power, low bit rate standard," IEEE Communication Magazine, Vol.42, No.6, pp. 140-146, June 2004.
- [13] IEEE 802.15.4, <http://www.researcher.nsc.gov.tw/public/8905780/Attachment/791017365-871.pdf>.
- [14] J. Zheng and M.J. Lee, "A Comprehensive Performance Study of 802.15.4," Sensor Network Operations Book, IEEE press, Wiley Interscience, Chapter 4, pp. 218-237, 2006.
- [15] F. Zhao and L. Guibas, "Wireless Sensor Networks An Information Processing Approach," Morgan Kaufmann Publishers, Elsevier, San Francisco, ISBN no. 81-8147-642-5, pp. 106-107, 2004.
- [16] C.E Perkins, E.M. Royer, and S. Das, "Ad hoc On-demand Distance Vector (AODV)," IETF RFC 3561, July 2003.
- [17] C.E. Perkins, "Ad Hoc On Demand Distance Vector (AODV) Routing," Internet draft, draft-ietf-manetaodv-01.txt, Work in progress, August 1998.
- [18] L. Klein-Bernd, "A Quick Guide to AODV Routing," Wireless Communication Technologies Group, NIST, [http://w3.antd.nist.gov/wctg/aodv\\_kernel/](http://w3.antd.nist.gov/wctg/aodv_kernel/).
- [19] M.K. Denko and H. Lu, "An AODV-Based Clustering and Routing Scheme for Mobile Ad Hoc Networks," IFIP International Federation for Information Processing, Springer Boston, ISSN No. 1571-5736, ISBN No. 978-0-387-34735-9, pp. 83-97, November 2006.

- [20] A. Bhatia and P. Kaushik, "A Cluster Based Minimum Battery Cost AODV Routing Using Multipath Route for Zigbee," in Proceedings of International Conference on Networks, ICON'08, pp. 1-7, December 2008.
- [21] Z. Kai, W. Neng and L. Ai-fang, "A New AODV Based Clustering Routing Protocol," in Proceedings of International Conference on Wireless Communications, Networking and Mobile Computing, pp. 728-731, September 2005.
- [22] P. Bourke, "Calculating the Area and Centroid of a Polygon," <http://paulbourke.net/geometry/polyarea/>, accessed December 2010.
- [23] Networkx Developer, "Networkx," <http://networkx.lanl.gov/>, accessed December 2010.
- [24] K. Fall and K. Varadhan, "The NS Manual (formerly ns Notes and Documentation)," VINT Project, A Collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC, May 2005.



**ภาคผนวก**

ภาคผนวก ก  
ผลการทดสอบเพิ่มเติม

ในหัวข้อนี้เป็นการทดลองเกี่ยวกับอัลกอริทึมการจัดคลัสเตอร์แบบกริดเชกเตอร์เพิ่มเติม เพื่อแสดงให้เห็นถึงประสิทธิภาพการทำงานของเครือข่ายในด้านอื่น โดยการทดลองจะกำหนดสภาพแวดล้อมเช่นเดียวกันกับการทดลองต่างๆ หัวข้อที่ 5 ซึ่งรายละเอียดแสดงดังต่อไปนี้

## ก1. ตัวชี้วัดที่สนใจ

### ก1.1 การจำลองกราฟ

#### 1. ค่าระยะทางเฉลี่ย (Average distance)

ค่าระยะทางเฉลี่ยจาก โหนดสมาชิกไปยังคลัสเตอร์เซตเป็นการวัดการกระจัด (Displacement) จากโหนดสมาชิกไปยังคลัสเตอร์เซต ซึ่งใช้ประเมินแนวโน้มการใช้พลังงานในการส่งข้อมูลจากโหนดสมาชิกไปยังคลัสเตอร์เซต ถ้าหากค่าระยะทางเฉลี่ยที่วัดออกมาได้มีค่าน้อย จะทำให้จำนวนฮอปในการส่งต่อข้อมูลลดลง ส่งผลให้แต่ละคลัสเตอร์ใช้พลังงานลดลงด้วย ค่าระยะทางเฉลี่ยคำนวณได้ดังสมการที่ ก-1 เมื่อ  $i$  คือหมายเลขคลัสเตอร์เซตที่  $0, 1, 2, \dots, k-1$  ตัวแปร  $k$  คือ จำนวนคลัสเตอร์เซต ตัวแปร  $m_i$  คือ จำนวนสมาชิกในคลัสเตอร์ที่  $i$  ตัวแปร  $j$  คือ โหนดสมาชิกที่  $j$  และตัวแปร  $d_{ij}$  คือการกระจัดจากโหนดสมาชิกที่  $j$  ไปยัง คลัสเตอร์ที่  $i$

$$\text{Average Hop Count} = \frac{\sum_{i=0}^{k-1} \left[ \frac{\sum_{j=0}^{m_i-1} d_{ij}}{m_i} \right]}{k} \quad (\text{ก-1})$$

#### 2. ค่าจำนวนฮอปเฉลี่ย (Average hop count)

ด้วยฟังก์ชันการคำนวณหาระยะทางที่สั้นที่สุด (Shortest path) ของโปรแกรมจำลองกราฟ Networkx ทำให้สามารถคำนวณหาจำนวนฮอปเฉลี่ยในการส่งต่อข้อมูลจากโหนดสมาชิกไปยังคลัสเตอร์เซตได้ คำนี้นั้นแตกต่างจากค่าระยะทางเฉลี่ยคือ ค่าจำนวนฮอปเฉลี่ยจะมีการพิจารณาระยะส่งสัญญาณวิทยุของแต่ละโหนดเข้ามาเกี่ยวข้อง หากจำนวนฮอปที่ได้จากการทดลองมีค่าน้อย แสดงว่า อัลกอริทึมชนิดที่ทำการทดลองสามารถทำให้เครือข่ายใช้พลังงานได้ต่ำกว่า ค่าจำนวนฮอปเฉลี่ยคำนวณได้จากสมการที่ ก-2 เมื่อ  $i$  คือหมายเลขคลัสเตอร์เซตที่  $0, 1, 2, \dots, k-1$  ตัวแปร  $k$  คือ จำนวนคลัสเตอร์เซต ตัวแปร  $m_i$  คือ จำนวนสมาชิกในคลัสเตอร์ที่  $i$  ตัวแปร  $j$  คือ โหนดสมาชิกที่  $j$  และตัวแปร  $h_{ij}$  คือจำนวนฮอปจากโหนดสมาชิกที่  $j$  ไปยังคลัสเตอร์ที่  $i$

$$Average\ Hop\ Count = \frac{\sum_{i=0}^{k-1} \left[ \frac{\sum_{j=0}^{m_i-1} h_{ij}}{m_i} \right]}{k} \quad (ก-2)$$

## ก1.2 การจำลองเครือข่าย

### 1. ปริมาณการส่งข้อมูล (Throughput)

ปริมาณการส่งข้อมูลหรือค่า Load per sink เป็นดัชนีที่ใช้วัดอัตราการส่งข้อมูลในช่วงเวลาหนึ่ง โดยวัดความสามารถในการส่งข้อมูลจากต้นทางไปยังปลายทาง ถ้าหากผลการทดลองออกมาให้ค่ามาก แสดงว่าอัลกอริทึมชนิดที่ทำการทดลองสามารถทำให้เครือข่ายส่งข้อมูลด้วยปริมาณที่สูง ปริมาณการส่งข้อมูลมีหน่วยเป็นบิตต่อวินาที (Bit per sec) คำนวณได้จากสมการที่ ก-3 เมื่อ *Received Data Packet* คือ ปริมาณข้อมูลที่รับได้สำเร็จมีหน่วยเป็นบิต และ *Time* คือเวลาทั้งหมดที่ใช้ในการสื่อสาร

$$Throughput = \frac{Received\ Data\ Packet}{Time} \quad (ก-3)$$

### 2. อัตราการรับต่อการส่งข้อมูล (Packet Delivery Ratio หรือ PDR)

ดัชนีชนิดนี้ใช้วัดความน่าเชื่อถือของเครือข่ายว่าสามารถส่งข้อมูลได้สำเร็จมากน้อยเท่าไร ถ้าหากผลการทดลองออกมาให้ค่ามาก แสดงว่าอัลกอริทึมชนิดที่ทำการทดลองสามารถทำให้เครือข่ายทำงานได้ด้วยความน่าเชื่อถือสูง มีอัตราการส่งข้อมูลสำเร็จสูง อัตราการรับข้อมูลต่อการส่งข้อมูลมีหน่วยเป็นเปอร์เซ็นต์ คำนวณได้จากสมการที่ ก-4 เมื่อ *No. of Received Data Packets* คือจำนวนแพ็กเก็ตข้อมูลที่รับได้สำเร็จ และ *No. of Sent Data Packets* คือจำนวนแพ็กเก็ตข้อมูลที่ส่งได้สำเร็จ

$$Packet\ Delivery\ Raio = \frac{No.\ of\ Received\ Data\ Packets}{No.\ of\ Sent\ Data\ Packets} \times 100 \quad (ก-4)$$

### 3. เวลาที่ใช้ในการส่งแพ็กเก็ตข้อมูลจากต้นทางสู่ปลายทาง (End-to-end delay)

ดัชนีชี้วัดนี้ใช้วัดความน่าเชื่อถือของการส่งแพ็กเก็ตข้อมูล ซึ่งจะเกี่ยวข้องกับประสิทธิภาพการทำงานของโพรโทคอลจัดเส้นทาง กล่าวคือถ้าหากโพรโทคอลการจัดเส้นทางสามารถจัดเส้นทางระหว่างต้นทางและปลายทางให้สั้นที่สุดได้ จะทำให้ค่าดัชนีชี้วัดนี้ออกมาต่ำ ซึ่งจะส่งผลทำให้ประหยัดเวลาและพลังงาน สำหรับการเลือกดัชนีชี้วัดนี้มาเกี่ยวข้องในการทดสอบสมรรถภาพของอัลกอริทึมการจัดคลัสเตอร์เพื่อเปรียบเทียบว่าอัลกอริทึมแต่ละชนิดสามารถจัดคลัสเตอร์แล้วใช้งานร่วมกับโพรโทคอลจัดเส้นทางได้อย่างไร เวลาที่ใช้ในการส่งแพ็กเก็ตข้อมูลมีหน่วยเป็นวินาที คำนวณได้จากสมการที่ ก-5 เมื่อ  $i$  คือหมายเลขแพ็กเก็ตที่ส่ง ตัวแปร  $t$  คือ เวลาที่ใช้ในการส่งแพ็กเก็ตที่  $i$  และตัวแปร  $n$  คือจำนวนแพ็กเก็ตที่ส่งสำเร็จ

$$End\ to\ End\ Delay = \frac{\sum_{i=0}^{n-1} t_i}{n} \quad (ก-5)$$

### 4. อัตราส่วนแพ็กเก็ตควบคุมต่อแพ็กเก็ตข้อมูล (Normalise Routing Load หรือ NRL)

ดัชนีชี้วัดนี้ใช้วัดประสิทธิภาพโพรโทคอลการจัดเส้นทาง กล่าวคือ เป็นการเปรียบเทียบว่าในการส่งแพ็กเก็ตข้อมูลหนึ่งแพ็กเก็ตจะต้องสิ้นเปลืองแพ็กเก็ตควบคุมจำนวนกี่แพ็กเก็ต ดังนั้นถ้าหากอัลกอริทึมการจัดคลัสเตอร์ชนิดใดสามารถทำให้ค่าดัชนีชี้วัดนี้ต่ำกว่า แสดงว่าผลการจัดคลัสเตอร์สามารถสนับสนุนการทำงานของโพรโทคอลจัดเส้นทางได้อย่างมีประสิทธิภาพมากกว่า อัตราส่วนแพ็กเก็ตควบคุมต่อแพ็กเก็ตข้อมูลคำนวณได้จากสมการที่ ก-6 เมื่อ *No. of Sent Routing Packets* คือจำนวนแพ็กเก็ตของโพรโทคอลจัดเส้นทางที่ส่งสำเร็จ และ *No. of Sent Data Packets* คือจำนวนแพ็กเก็ตข้อมูลที่ส่งสำเร็จ

$$Normalise\ Routing\ Load = \frac{No.\ of\ Sent\ Routing\ Packets}{No.\ of\ Sent\ Data\ Packets} \quad (ก-6)$$

### 5. โอเวอร์เฮด (Routing overhead)

ดัชนีชี้วัดนี้ใช้วัดปริมาณการส่งแพ็กเก็ตควบคุมของโพรโทคอลจัดเส้นทางชนิด AODV ตั้งแต่เริ่มการทดลองไปจนกระทั่งเสร็จสมบูรณ์ โดยทั่วไปจะใช้วัดประสิทธิภาพการทำงานของโพรโทคอลการจัดเส้นทาง ดังนั้นถ้าหากอัลกอริทึมการจัดคลัสเตอร์ชนิดใดสามารถให้ค่าดัชนี

ชนิดนี้ต่ำกว่า แสดงว่าผลการจัดคลัสเตอร์สามารถสนับสนุนการทำงานของโพรโทคอลการจัดเส้นทางได้อย่างมีประสิทธิภาพมากกว่า โอเวอร์เฮดมีหน่วยเป็นแพ็กเก็ต

#### 6. พลังงานทั้งหมดที่ใช้ในเครือข่าย (Total energy consumption)

ดัชนีชนิดนี้ใช้วัดการสิ้นเปลืองพลังงานของเครือข่ายที่ใช้งานร่วมกับอัลกอริทึมการจัดคลัสเตอร์ พลังงานทั้งหมดที่ใช้ในเครือข่ายมีหน่วยเป็นจูล (Joule) คำนวณได้จากสมการที่ ก-7 เมื่อ  $i$  คือหมายเลขโหนดสมาชิกในเครือข่าย ตัวแปร  $E_i$  คือ พลังงานที่ใช้ของโหนดที่  $i$  และตัวแปร  $n$  คือจำนวนโหนดสมาชิกทั้งหมดในเครือข่าย

$$Total\ Energy\ Consumption = \frac{\sum_{i=0}^{n-1} E_i}{n} \quad (ก-7)$$

#### 7. ประสิทธิภาพการใช้พลังงาน (Energy efficiency)

ดัชนีชนิดนี้เป็นการวัดอัตราส่วนของพลังงานที่ใช้ในเครือข่ายทั้งหมดต่อจำนวนแพ็กเก็ตที่ส่งสำเร็จ เพื่อประเมินว่าเครือข่ายใช้พลังงานต่อการส่งแพ็กเก็ตข้อมูล 1 แพ็กเก็ตเท่าไร ถ้าหากได้ผลลัพธ์ออกมาน้อยแสดงว่าอัลกอริทึมดังกล่าวใช้พลังงานอย่างมีประสิทธิภาพมากกว่า ประสิทธิภาพการใช้พลังงานมีหน่วยเป็นจูลต่อแพ็กเก็ต (Joule per packet) คำนวณได้จากสมการ ก-8 เมื่อ  $Total\ Energy\ Consumption$  คือพลังงานทั้งหมดที่ใช้ในเครือข่าย ซึ่งได้จากสมการที่ ก-7 และ  $No.\ of\ Received\ Data\ Packets$  คือจำนวนแพ็กเก็ตข้อมูลที่ส่งสำเร็จ

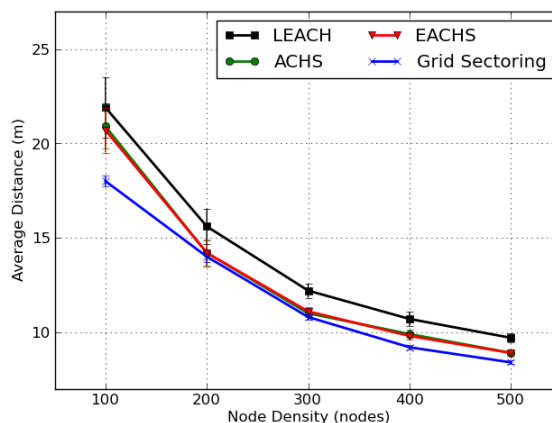
$$Energy\ Efficiency = \frac{Total\ Energy\ Consumption}{No.\ of\ Received\ Data\ Packets} \quad (ก-8)$$

#### 8. อัตราการใช้พลังงานของทั้งเครือข่าย

กำหนดให้พลังงานเริ่มต้นของทุกโหนดมีค่าเท่ากับ 1 จูล แล้ววัดค่าพลังงานที่ลดลงเทียบกับแกนเวลาจนถึงสิ้นสุดการจำลองเครือข่าย

### ก1.3 ผลการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายแบบสม่ำเสมอ

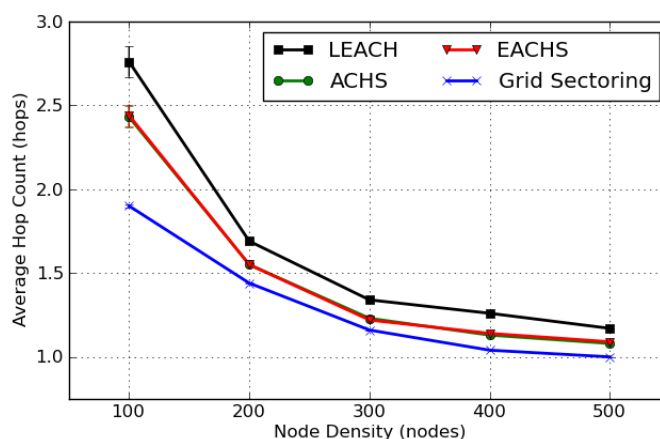
#### 1. ค่าระยะทางเฉลี่ย



ภาพประกอบ ก-1 ผลการทดลองค่าระยะทางเฉลี่ย

จากภาพประกอบที่ ก-1 ทุกอัลกอริทึมให้ผลลัพธ์ไปในทิศทางเดียวกัน แต่ อัลกอริทึมกริดเซกเตอร์สามารถให้ผลลัพธ์ระยะทางที่สั้นที่สุดไม่ว่าจำนวนโหนดในเครือข่ายมีขนาดเท่าไร ถึงแม้ว่าค่าผลลัพธ์ที่ได้ต่ำกว่าเพียงเล็กน้อยเมื่อเปรียบเทียบกับอัลกอริทึมชนิดอื่น แต่เมื่อพิจารณาถึงคุณภาพของการส่งสัญญาณวิทยุ โดยเฉพาะในเครือข่ายเซนเซอร์ไร้สายที่มีระยะการสื่อสารที่สั้นแล้ว ระยะทางที่แตกต่างกันในระดับครึ่งเมตร ก็สามารถกระทบต่อคุณภาพการส่งสัญญาณวิทยุของโหนดได้

#### 2. ค่าจำนวนฮอปเฉลี่ย

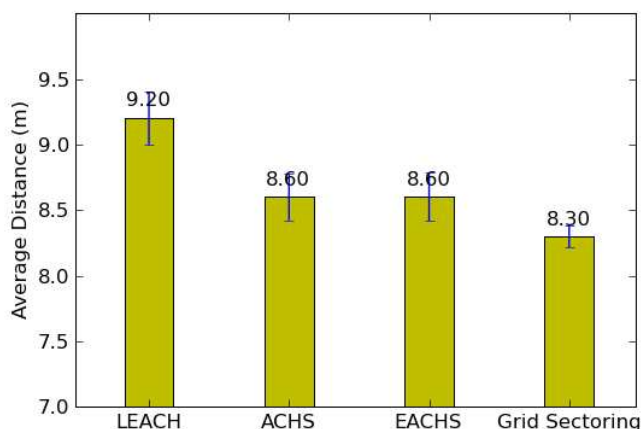


ภาพประกอบ ก-2 ผลการทดลองค่าจำนวนฮอปเฉลี่ย

สำหรับระยะรับส่งสัญญาณวิทยุของทุกโหนดที่กำหนดไว้ 15 เมตร ผลลัพธ์ที่ได้มีความสอดคล้องและเป็นไปในทิศทางเดียวกับผลลัพธ์ค่าระยะทางเฉลี่ย แต่ค่าจำนวนฮอปเฉลี่ยอาจมีค่าสูงกว่าค่าระยะทางเฉลี่ยเช่น ที่ 200 โหนด ค่าระยะทางเฉลี่ยมีค่า 15 เมตรโดยประมาณ ดังนั้นค่าจำนวนฮอปเฉลี่ยควรมีค่า 1 ฮอป เพราะกำหนดให้ระยะรับส่งสัญญาณวิทยุเท่ากับ 15 เมตร แต่ผลการทดลองกลับให้ค่าจำนวนฮอปเฉลี่ยเท่ากับ 1.5 ฮอปโดยประมาณ ทั้งนี้เนื่องจากค่าระยะทางคำนวณจากการกระจัด (Displacement) จากโหนดไปยังคลัสเตอร์เฮด แต่ค่าจำนวนฮอปเฉลี่ยคำนวณจากผลรวมระยะทางจากโหนดต้นทางไปยังโหนดระหว่างทางที่อยู่ในเส้นทางการสื่อสารและไปสิ้นสุดยังคลัสเตอร์เฮด ทำให้ค่าจำนวนฮอปเฉลี่ยที่คำนวณได้มีค่าสูงกว่าค่าระยะทางเฉลี่ย จากผลการทดลอง อัลกอริทึมที่นำเสนอยังคงให้ค่าน้อยที่สุดเมื่อเปรียบเทียบกับอัลกอริทึมชนิดอื่นซึ่งให้ผลลัพธ์ไม่เกิน 2 ฮอป ดังภาพประกอบที่ ก-2

#### ก1.4 ผลการจำลองกราฟในเครือข่ายที่โหนดมีการกระจายด้วยความหนาแน่นไม่สม่ำเสมอ

##### 1. ค่าระยะทางเฉลี่ย

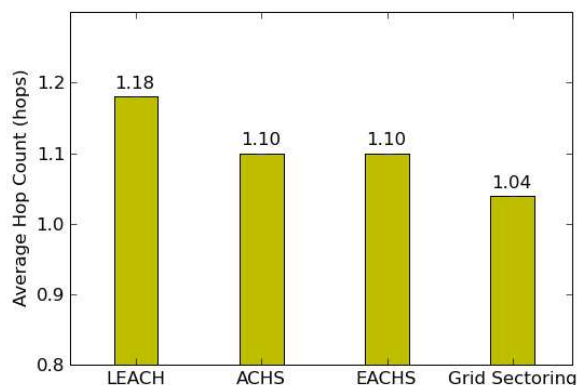


ภาพประกอบ ก-3 ผลการทดลองค่าระยะทางเฉลี่ย

จากภาพประกอบที่ ก-3 อัลกอริทึมกริดเซกเตอร์สามารถให้ผลลัพธ์ระยะทางที่สั้นที่สุด 8.30 เมตร แต่ไม่ได้แตกต่างจากผลลัพธ์ของอัลกอริทึมอื่นมาก



## 2. ค่าจำนวนฮอปเฉลี่ย

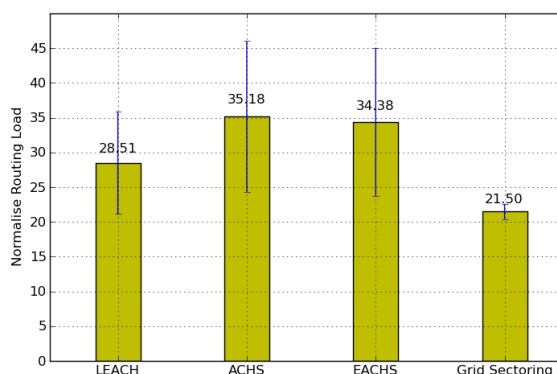


ภาพประกอบ ก-4 ผลการทดลองค่าจำนวนฮอปเฉลี่ย

สำหรับระยะรับส่งสัญญาณวิทยุของทุกโหนดที่กำหนดไว้ 15 เมตร ผลลัพธ์ที่ได้มีความสอดคล้องและเป็นไปในทิศทางเดียวกับผลลัพธ์ค่าระยะทางเฉลี่ยเช่นเดียวกันกับการทดลองในเครือข่ายที่โหนดมีการกระจายแบบสม่ำเสมอ โดยอัลกอริทึมที่นำเสนอก็ยังคงให้ค่าน้อยที่สุดดังภาพประกอบที่ ก-4

### ก1.5 ผลการจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบสม่ำเสมอ

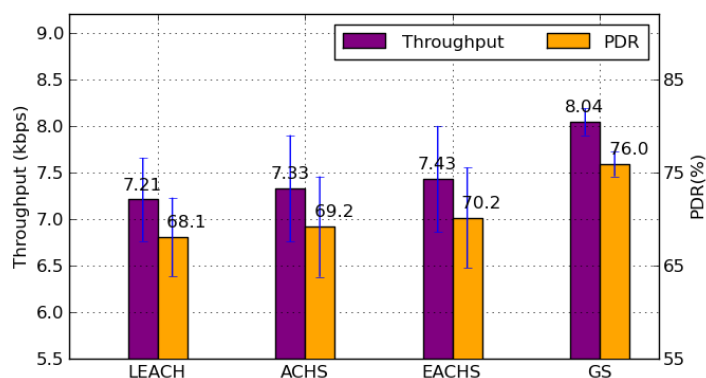
#### 1. อัตราส่วนแพ็กเก็ตเกิดควบคุมต่อแพ็กเก็ตข้อมูล



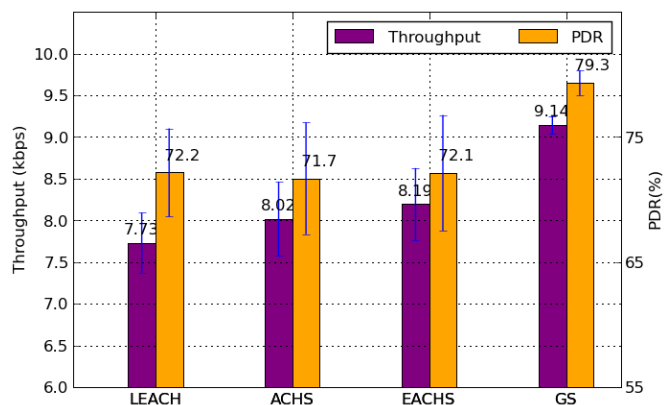
ภาพประกอบ ก-5 ผลการทดลองอัตราส่วนแพ็กเก็ตเกิดควบคุมต่อแพ็กเก็ตข้อมูล

จากภาพประกอบที่ ก-5 จะเห็นได้ว่าผลลัพธ์ในเรื่อง NRL อัลกอริทึมที่นำเสนอให้ผลลัพธ์ต่ำสุด ซึ่งสามารถวิเคราะห์ได้ว่า การจัดคลัสเตอร์ด้วยอัลกอริทึมชนิดที่นำเสนอสามารถทำให้เครือข่ายใช้แพ็กเก็ตควบคุมเป็นสัดส่วนที่น้อยที่สุดต่อการส่งแพ็กเก็ตข้อมูล 1 แพ็กเก็ต

## 2. ปริมาณการส่งข้อมูลและอัตราการรับต่อการส่งข้อมูล



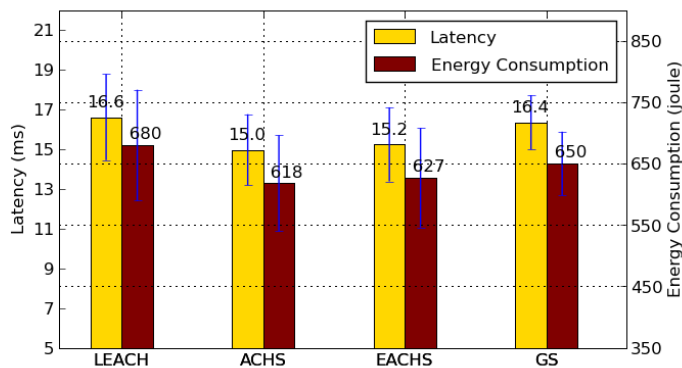
ภาพประกอบ ก-6 ผลการทดลองปริมาณการส่งข้อมูลและอัตราการรับต่อการส่งข้อมูล



ภาพประกอบ ก-7 ผลการทดลองปริมาณการส่งข้อมูลและอัตราการรับต่อการส่งข้อมูล

จากภาพประกอบที่ ก-6 อัลกอริทึมที่นำเสนอให้ผลลัพธ์ที่ดีที่สุดในเรื่องความสามารถในการสื่อสารข้อมูล คือให้ผลลัพธ์ปริมาณการส่งข้อมูลและ PDR ที่ใกล้เคียงกันประมาณ 8 กิโลบิตต่อวินาที และ 75 เปอร์เซ็นต์ตามลำดับ ซึ่งดีกว่าอัลกอริทึมชนิดอื่นประมาณ 0.5 กิโลบิตต่อวินาที และ 5 เปอร์เซ็นต์ตามลำดับ และเมื่อจำลองเครือข่ายด้วยระยะเวลาที่เพิ่มขึ้น คือ 3600 วินาที หรือ 1 ชั่วโมง ปรากฏว่าอัลกอริทึมที่นำเสนอให้ค่าปริมาณการส่งข้อมูลและ PDR เพิ่มขึ้นกว่าเดิม 1.1 กิโลบิตต่อวินาทีและ 3.3 เปอร์เซ็นต์ตามลำดับ ค่าปริมาณการส่งข้อมูลและอัตราการรับต่อการส่งข้อมูลของอัลกอริทึมที่นำมาเสนอมากกว่าอัลกอริทึมชนิดอื่นประมาณ 1 กิโลบิตต่อวินาที และ 7 เปอร์เซ็นต์ตามลำดับ ผลการทดลองแสดงดังภาพประกอบที่ ก-7

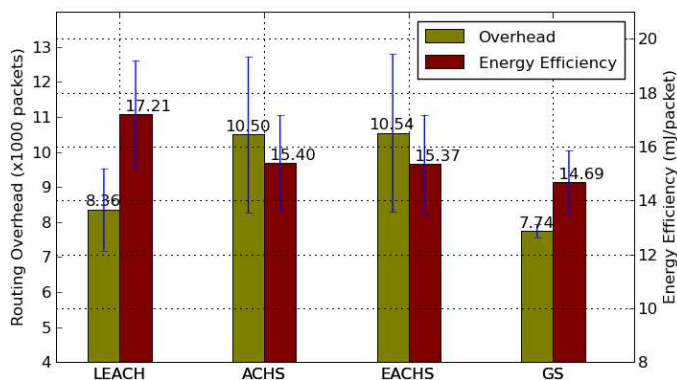
### 3. เวลาที่ใช้ในการส่งแพ็กเก็ตเกิดข้อมูลจากต้นทางสู่ปลายทางและพลังงานทั้งหมดที่ใช้ในเครือข่าย



ภาพประกอบ ก-8 ผลการทดลองเวลาที่ใช้ในการส่งแพ็กเก็ตเกิดข้อมูลจากต้นทางสู่ปลายทางและพลังงานทั้งหมดที่ใช้ในเครือข่าย

จากภาพประกอบที่ ก-8 จะเห็นได้ว่าผลลัพธ์ในเรื่องเวลาในการส่งแพ็กเก็ตเกิดข้อมูลของอัลกอริทึมที่นำเสนอมีค่าใกล้เคียงกับอัลกอริทึม LEACH และสูงกว่าอัลกอริทึม ACHS และ EACHS และเมื่อสังเกตผลลัพธ์เรื่องการใช้พลังงานแล้ว แนวโน้มของผลลัพธ์จะเป็นไปในทิศทางเดียวกันกับผลลัพธ์เรื่องระยะเวลาในการส่งแพ็กเก็ตเกิดข้อมูล อย่างไรก็ตามการใช้พลังงานของอัลกอริทึมที่นำเสนอ ให้ผลลัพธ์ต่ำกว่าอัลกอริทึม ACHS และ EACHS ทั้งนี้เนื่องจากจำนวนแพ็กเก็ตที่ส่งได้สำเร็จของอัลกอริทึมที่นำเสนอมีปริมาณมากกว่า โดยสังเกตจากค่า PDR และปริมาณข้อมูลที่ส่งมากกว่าสังเกตจากค่า Throughput ที่ส่งผลให้มีการใช้พลังงานสูงขึ้น

### 4. โอเวอร์เฮดและประสิทธิภาพการใช้พลังงาน

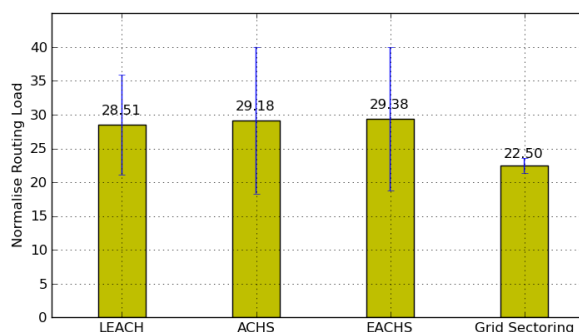


ภาพประกอบ ก-9 ผลการทดลองโอเวอร์เฮดและประสิทธิภาพการใช้พลังงาน

เมื่อพิจารณาจากภาพประกอบที่ ก-9 ผลลัพธ์โอเวอร์เฮดที่ได้ของอัลกอริทึมที่นำเสนอให้ผลลัพธ์ปริมาณแพ็กเก็ตเกิดควบคุมของโพรโทคอลจัดเส้นทางที่ส่งในเครือข่ายต่ำที่สุดซึ่งสอดคล้องกับผลลัพธ์ NRL ที่ต่ำที่สุดเช่นเดียวกัน ดังภาพประกอบที่ ก-5 ส่วนผลลัพธ์ประสิทธิภาพการใช้พลังงานของอัลกอริทึมที่นำเสนอก็ให้ผลลัพธ์ที่ต่ำที่สุด หมายความว่าอัลกอริทึมที่นำเสนอให้ประสิทธิภาพการใช้พลังงานดีกว่าอัลกอริทึมชนิดอื่นประมาณ 1 จุดต่อแพ็กเก็ต

## ก1.6 ผลการจำลองเครือข่ายในสภาพแวดล้อมที่โหนดมีการกระจายแบบไม่สม่ำเสมอ

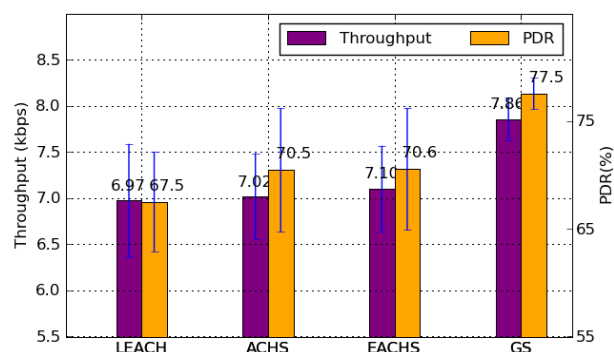
### 1. อัตราส่วนแพ็กเก็ตเกิดควบคุมต่อแพ็กเก็ตข้อมูล



ภาพประกอบ ก-10 ผลการทดลองอัตราส่วนแพ็กเก็ตเกิดควบคุมต่อแพ็กเก็ตข้อมูล

จากภาพประกอบที่ ก-10 จะเห็นได้ว่าผลลัพธ์ในเรื่อง NRL อัลกอริทึมที่นำเสนอให้ผลลัพธ์ต่ำสุด ซึ่งทำให้สามารถวิเคราะห์ได้ว่า การจัดคลัสเตอร์ด้วยอัลกอริทึมที่นำเสนอสามารถทำให้เครือข่ายใช้แพ็กเก็ตควบคุมเป็นสัดส่วนที่น้อยที่สุดต่อการส่งแพ็กเก็ตข้อมูล 1 แพ็กเก็ต

### 2. ปริมาณการส่งข้อมูลและอัตราการรับข้อมูลต่อการส่งข้อมูล

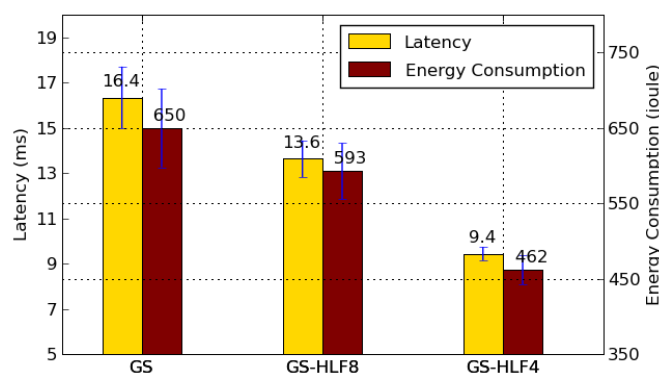


ภาพประกอบ ก-11 ผลการทดลองปริมาณการส่งข้อมูลและอัตราการรับต่อการส่งข้อมูล

จากภาพประกอบที่ ก-11 อัลกอริทึมที่นำเสนอให้ผลลัพธ์ที่ดีที่สุดในเรื่องความสามารถในการสื่อสารข้อมูล คือให้ผลลัพธ์ปริมาณการส่งข้อมูลและ PDR อยู่ที่ประมาณ 7.8 กิโลบิตต่อวินาที และ 77.5 เปอร์เซ็นต์ตามลำดับ ซึ่งดีกว่าอัลกอริทึมชนิดอื่นประมาณ 0.8 กิโลบิตต่อวินาที และ 7 เปอร์เซ็นต์ตามลำดับ

### ก1.7 ผลการทดลองจำนวนฮอปของอัลกอริทึมกริดเชกเตอร์บนโพรโทคอล AODV

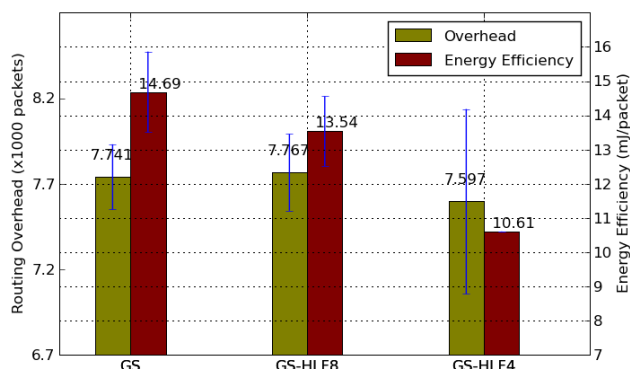
1. เวลาที่ใช้ในการส่งแพ็กเก็ตเกิดข้อมูลจากต้นทางสู่ปลายทางและพลังงานทั้งหมดที่ใช้ในเครือข่าย



ภาพประกอบ ก-12 ผลการทดลองเวลาที่ใช้ในการส่งแพ็กเก็ตเกิดข้อมูลจากต้นทางสู่ปลายทางและพลังงานทั้งหมดที่ใช้ในเครือข่าย

จากภาพประกอบที่ ก-12 จะเห็นได้ว่าผลลัพธ์ในเรื่องเวลาในการส่งแพ็กเก็ตเกิดข้อมูลของอัลกอริทึมที่นำเสนอเมื่อมีการปรับปรุงให้จำกัดจำนวนฮอปให้มีจำนวนน้อยลง จะสามารถลดระยะเวลาในการส่งแพ็กเก็ตส่งข้อมูลลงเช่นกัน ในขณะที่เดียวกันผลลัพธ์การใช้พลังงานของอัลกอริทึมที่นำเสนอทั้งแบบดั้งเดิมและแบบที่ปรับปรุงมีค่าลดลงเมื่อจำนวนฮอปที่จำกัดมีจำนวนลดลง แสดงว่าวิธีการปรับปรุงอัลกอริทึมที่นำเสนอให้มีการจำกัดจำนวนฮอปของแพ็กเก็ตควบคุมสามารถลดการใช้พลังงานของเครือข่ายได้ โดยเฉพาะการจำกัดที่ 4 ฮอป สามารถลดพลังงานจาก 650 จูล เหลือ 462 จูล ภายในระยะเวลาการจำลองเครือข่าย 600 วินาที หรือ 10 นาที

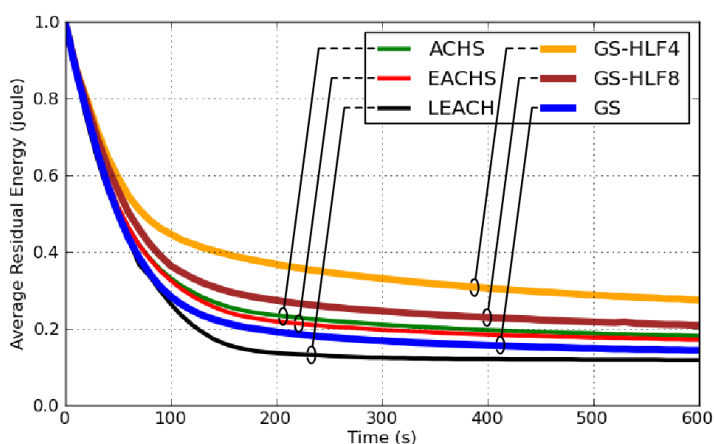
## 2. โอเวอร์เฮดและประสิทธิภาพการใช้พลังงาน



ภาพประกอบ ก-13 ผลการทดลองโอเวอร์เฮดและประสิทธิภาพการใช้พลังงาน

เมื่อพิจารณาจากภาพประกอบที่ ก-13 ผลลัพธ์โอเวอร์เฮดที่ได้ของอัลกอริทึมที่นำเสนอให้ปริมาณแพ็กเก็ตควบคุมของโพรโทคอลจัดเส้นทางที่ส่งในเครือข่ายลดต่ำลง เมื่อจำนวนฮอปที่จำกัดมีจำนวนลดลง โดยที่ 4 ฮอปสามารถลดปริมาณแพ็กเก็ตเกิดจาก 7741 แพ็กเก็ตลงเหลือ 7597 แพ็กเก็ตจากการจำลองเครือข่าย 600 วินาที นอกจากนี้ผลลัพธ์ประสิทธิภาพการใช้พลังงานของอัลกอริทึมที่นำเสนอยังดีขึ้น คือให้ผลต่ำลง แสดงว่าการลดจำนวนฮอปของอัลกอริทึมที่ได้รับการปรับปรุงทำให้ประสิทธิภาพการใช้พลังงานเพิ่มสูงขึ้น

## 3. อัตราการใช้พลังงานของทั้งเครือข่าย



ภาพประกอบ ก-14 ผลการทดลองอัตราการใช้พลังงานของทั้งเครือข่าย

เมื่อกำหนดค่าพลังงานเริ่มต้นให้กับทุกโหนดในเครือข่ายเท่ากับ 1 จูล และจำลองเครือข่ายด้วยระยะเวลาเดียวกันคือ 600 วินาที จะสามารถเห็นแนวโน้มการลดลงของพลังงานเฉลี่ยของโหนดในเครือข่ายของอัลกอริทึมต่างๆ ได้ดังภาพประกอบที่ ก-14 ซึ่งอัลกอริทึมที่นำเสนอแบบดั้งเดิมมีอัตราการใช้พลังงานสูงกว่าอัลกอริทึม ACHS และ EACHS แต่ต่ำกว่าอัลกอริทึม LEACH แต่เมื่อปรับปรุงอัลกอริทึมให้จำกัดจำนวนฮอปในการส่งแพ็กเก็ตข้อมูล จะทำให้อัตราการใช้พลังงานของเครือข่ายลดลง

### ก1.8 สรุปผลการทดลอง

จากผลการทดลองการจัดคลัสเตอร์แบบกริดเชกเตอร์ทั้งในเครือข่ายที่โหนดมีการกระจายด้วยความหนาแน่นสม่ำเสมอและไม่สม่ำเสมอ อัลกอริทึมที่นำเสนอให้ผลการทดลองในเรื่องระยะทางเฉลี่ยและจำนวนฮอปเฉลี่ยต่ำสุด ถึงแม้ว่าผลลัพธ์ที่ได้จะต่ำกว่าอัลกอริทึมอื่นเล็กน้อย แต่ด้วยข้อจำกัดทางการสื่อสารที่โหนดมีระยะการสื่อสารและกำลังส่งต่ำ จึงทำให้ระยะทางและจำนวนฮอปเฉลี่ยที่ลดลงเล็กน้อย สามารถเพิ่มคุณภาพการสื่อสารของเครือข่ายได้ดีขึ้น ทั้งยังช่วยให้แต่ละคลัสเตอร์มีการใช้พลังงานอย่างประหยัดมากยิ่งขึ้น

จากการจำลองเครือข่าย ผลลัพธ์ในด้าน Throughput และ PDR ให้ผลดีกว่าอัลกอริทึมชนิดอื่น คือให้ throughput และ PDR มากกว่าอัลกอริทึมอื่น 8.2 และ 8.3 เปอร์เซ็นต์ตามลำดับ และเมื่อจำลองเครือข่ายด้วยระยะเวลาที่เพิ่มขึ้น จาก 10 นาทีเป็น 1 ชั่วโมง กลับพบว่าอัลกอริทึมชนิดที่นำเสนอ ให้ throughput และ PDR ที่เพิ่มสูงขึ้น คือมากกว่าอัลกอริทึมอื่น 11.6 และ 10.0 เปอร์เซ็นต์ตามลำดับ

การปรับปรุงอัลกอริทึมการจัดคลัสเตอร์แบบกริดเชกเตอร์ให้จำกัดจำนวนฮอปเพื่อทำงานร่วมกับโพรโทคอลจัดเส้นทาง AODV อย่างมีประสิทธิภาพด้านการใช้พลังงาน เมื่อมีการจำกัดจำนวนฮอป 8 ฮอป และ 4 ฮอป ผลลัพธ์ที่ได้ดีขึ้นตามลำดับ คือให้ผลลัพธ์ในเรื่อง NRL และโอเวอร์เฮดที่ต่ำกว่าอัลกอริทึมอื่น ซึ่งสรุปได้ว่าการปรับปรุงด้วยวิธีดังกล่าวสามารถลดการแพร่กระจายแพ็กเก็ตเกิดควบคุมของโพรโทคอล AODV ส่งผลให้ลดความคับคั่งในการสื่อสาร นอกจากนี้เมื่อจำนวนฮอปทำให้ลดเวลาในการส่งแพ็กเก็ตข้อมูล โดยเฉพาะการจำกัดที่ 4 ฮอปสามารถลดเวลาในการส่งแพ็กเก็ตข้อมูลได้ 42.68 เปอร์เซ็นต์ เมื่อเปรียบเทียบกับอัลกอริทึมเดิม

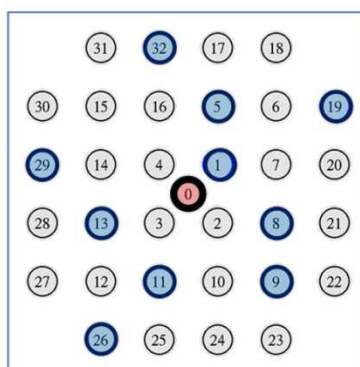
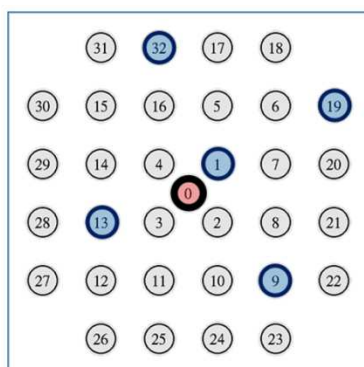
ภาคผนวก ข

ผลการวิเคราะห์โปรโตคอล AODV บนมาตรฐาน IEEE 802.15.4

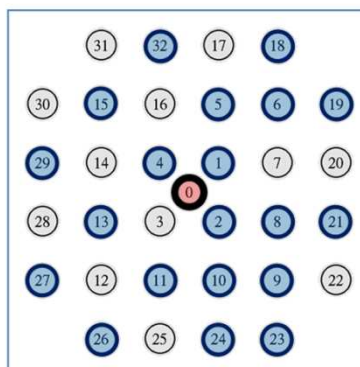
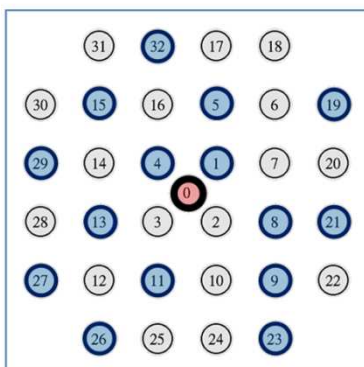


### ข1. การวิเคราะห์โพรโทคอล AODV บนมาตรฐาน IEEE 802.15.4

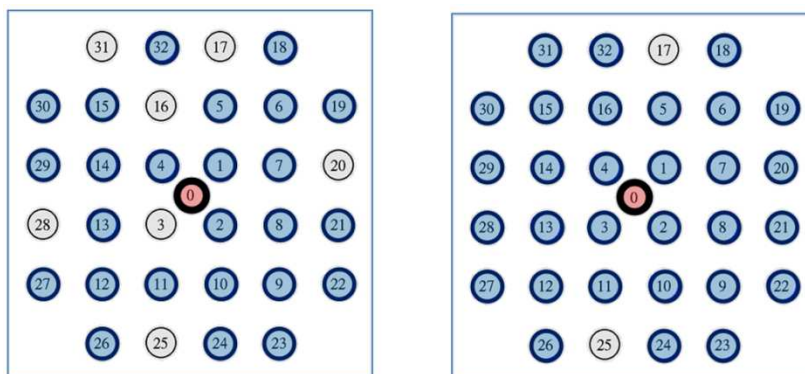
หัวข้อนี้เป็นการศึกษาเกี่ยวกับสภาพแวดล้อมหรือพารามิเตอร์ต่างๆ ที่ให้คุณภาพการสื่อสารของเครือข่ายที่ใช้โพรโทคอล AODV บนมาตรฐาน IEEE 802.15.4 เป็นที่น่าสนใจด้วยโปรแกรมจำลองเครือข่าย NS-2 [24] โดยพิจารณาจากพารามิเตอร์ที่ให้ผลการทดลองในเรื่องอัตราการรับต่อการส่งข้อมูล (Packet Delivery Ratio หรือ PDR) ที่มากกว่าหรือเท่ากับ 60 เปอร์เซ็นต์ การนำค่า PDR มาเป็นดัชนีในการตัดสินใจ เนื่องจากสามารถบ่งบอกถึงคุณภาพในการสื่อสารซึ่งเป็นประเด็นที่ใช้พิจารณาเป็นประการต้นๆ ของการสื่อสารไร้สายทุกชนิด พารามิเตอร์ที่สนใจได้แก่ ขนาดของแพ็กเก็ตข้อมูล ระยะการสื่อสารของโหนด จำนวนโหนดต้นทาง และอัตราการส่งแพ็กเก็ตข้อมูล ทดลองบนพื้นที่ขนาด 50x50 ตารางเมตร โหนดในเครือข่ายจำนวน 32 โหนด โดยมีคลัสเตอร์เฮด 1 โหนดทำหน้าที่เป็นโหนดปลายทาง (โหนดหมายเลข 0) โหนดต้นทางมีการกระจายทั่วพื้นที่ดังภาพประกอบที่ ข-1 ถึง ข-3 ทุกโหนดมีการสื่อสารแบบมัลติฮอปโดยใช้มาตรฐาน IEEE 802.15.4 ในชั้น MAC และกายภาพ และใช้โพรโทคอลจัดเส้นทางชนิด AODV



ภาพประกอบ ข-1 โหนดต้นทางจำนวน 5 และ 10 โหนด



ภาพประกอบ ข-2 โหนดต้นทางจำนวน 15 และ 20 โหนด



ภาพประกอบ ข-3 โหนดต้นทางจำนวน 25 และ 30 โหนด

การทดลองนี้ปรับเปลี่ยนพารามิเตอร์ตามเงื่อนไขดังต่อไปนี้คือ ขนาดของแพ็กเก็ตข้อมูลเป็น 20, 50 และ 80 ไบต์ ระยะการสื่อสารของโหนดกำหนดเป็น 10, 15, 20, 25 และ 30 จำนวนโหนดต้นทางเป็น 5, 10, 15, 20, 25 และ 30 โหนด โดยตำแหน่งของโหนดต้นทางจะกระจายดังภาพประกอบที่ 3-2 ถึง 3-4 ตามลำดับ คลัสเตอร์เฮดจะอยู่ตรงกลางพื้นที่ และความถี่ในการส่งแพ็กเก็ตข้อมูลเป็น 10, 5, 2, 1, 0.5, 0.2 และ 0.1 แพ็กเก็ตต่อวินาที ผลการทดลองแสดงดังตารางที่ ข-1 ถึง ข-3 ตามลำดับ

ตารางที่ ข-1 อัตราการรับข้อมูลต่อการส่งข้อมูลเมื่อแพ็กเก็ตข้อมูลมีขนาด 20 ไบต์

ระยะการสื่อสาร (เมตร)	โหนดต้นทาง	ความถี่ในการส่งแพ็กเก็ตข้อมูล (แพ็กเก็ตต่อวินาที)						
		10.0	5.0	2.0	1.0	0.5	0.2	0.1
10	5	79.85	99.78	80.01	100.00	100.00	100.00	100.00
	10	49.56	53.16	99.99	99.98	99.94	99.80	99.00
	15	20.50	35.88	91.72	86.67	93.18	93.14	93.08
	20	10.44	22.44	52.64	89.85	89.97	89.96	89.91
	25	12.56	17.93	24.19	82.25	87.87	87.64	87.61
	30	7.04	3.66	24.19	46.43	89.97	89.80	89.87

ตารางที่ ข-1 อัตราการรับข้อมูลต่อการส่งข้อมูลเมื่อแพ็กเก็ตข้อมูลมีขนาด 20 ไบต์ (ต่อ)

ระยะการ สื่อสาร (เมตร)	โหนด ต้นทาง	ความถี่ในการส่งแพ็กเก็ตข้อมูล (แพ็กเก็ตต่อวินาที)						
		10.0	5.0	2.0	1.0	0.5	0.2	0.1
15	5	79.97	100.00	99.98	99.98	99.96	99.90	99.80
	10	20.53	88.99	79.99	79.93	79.89	79.92	70.06
	15	4.02	23.61	79.88	86.54	79.97	79.95	79.96
	20	0.05	52.57	65.82	89.98	84.96	89.98	89.81
	25	0.05	28.32	48.16	79.90	91.93	91.96	91.89
	30	10.17	16.85	30.16	76.51	89.95	86.50	89.54
20	5	2.80	59.96	99.90	98.94	99.84	100.00	100.00
	10	29.76	29.76	89.85	79.93	89.88	89.61	88.42
	15	20.05	0.45	40.51	93.21	93.26	93.14	92.94
	20	0.84	21.22	55.07	79.87	89.93	89.81	84.37
	25	12.26	0.01	36.36	76.10	71.85	87.92	83.85
	30	7.42	13.52	21.32	40.46	84.76	89.84	82.74
25	5	0.12	59.97	99.88	79.97	79.88	80.04	79.68
	10	23.93	49.59	69.97	79.99	80.01	79.97	79.84
	15	0.06	8.61	46.64	79.94	79.98	79.91	66.58
	20	6.79	31.13	24.09	64.97	64.94	69.89	69.98
	25	16.92	19.74	21.11	53.01	63.93	67.91	63.95
	30	13.49	14.01	17.69	46.92	56.63	59.83	53.13
30	5	39.75	40.64	59.58	60.06	79.976	99.80	100.00
	10	20.97	39.94	69.93	79.65	69.792	89.76	89.72
	15	0.33	19.99	40.07	46.66	53.172	66.72	73.37
	20	15.67	18.26	40.07	64.83	39.852	54.92	59.54
	25	11.38	8.21	33.27	40.28	55.895	59.84	55.84
	30	0.41	0.41	0.02	27.78	53.2	66.58	66.46

ตารางที่ ข-2 อัตราการรับข้อมูลต่อการส่งข้อมูลเมื่อแพ็กเก็ตข้อมูลมีขนาด 50 ไบต์

ระยะการ สื่อสาร (เมตร)	โหนด ต้นทาง	ความถี่ในการส่งแพ็กเก็ตข้อมูล (แพ็กเก็ตต่อวินาที)						
		10.0	5.0	2.0	1.0	0.5	0.2	0.1
10	5	99.47	99.99	100.00	100.00	100.00	100.00	100.00
	10	49.79	69.84	100.00	100.00	100.00	100.00	100.00
	15	26.61	53.25	69.24	100.00	93.34	93.40	99.80
	20	10.41	30.83	68.80	94.24	95.00	95.00	94.86
	25	10.31	6.70	43.60	79.38	95.95	95.62	92.05
	30	12.69	24.55	14.96	65.34	96.17	96.17	96.44
15	5	59.91	79.98	99.97	99.92	99.80	99.90	99.60
	10	39.88	49.98	89.99	89.94	89.88	89.71	89.52
	15	18.90	41.68	74.21	86.65	86.63	86.58	79.69
	20	15.06	29.02	70.55	75.78	76.55	78.89	69.68
	25	12.30	20.96	40.56	75.02	83.97	83.95	83.45
	30	13.89	20.03	37.14	73.29	86.64	86.62	76.42
20	5	59.91	79.82	99.89	99.96	79.86	79.94	79.88
	10	20.04	40.00	79.97	89.86	99.82	99.70	99.30
	15	19.87	53.19	65.70	92.94	86.51	92.77	93.01
	20	0.15	11.37	52.48	84.79	84.88	89.88	84.67
	25	0.01	20.03	2.59	63.86	87.82	83.81	83.21
	30	3.91	0.04	12.63	63.14	79.88	73.89	76.28
25	5	40.02	79.83	79.85	79.99	99.40	80.04	79.68
	10	30.02	39.95	59.92	59.92	79.95	79.87	79.74
	15	21.97	29.06	59.90	66.64	79.82	79.91	65.91
	20	14.81	20.04	44.92	64.96	69.94	69.79	65.70
	25	11.82	20.04	32.10	52.01	75.83	79.81	71.14
	30	0.01	17.30	18.99	52.27	66.56	63.18	56.06

ตารางที่ ข-2 อัตราการรับข้อมูลต่อการส่งข้อมูลเมื่อแพ็กเก็ตข้อมูลมีขนาด 50 ไบต์ (ต่อ)

ระยะการ สื่อสาร (เมตร)	โหนด ต้นทาง	ความถี่ในการส่งแพ็กเก็ตข้อมูล (แพ็กเก็ตต่อวินาที)						
		10.0	5.0	2.0	1.0	0.5	0.2	0.1
30	5	29.53	79.88	79.70	79.87	79.90	99.60	99.80
	10	19.95	32.48	59.98	69.80	49.74	59.59	60.08
	15	0.01	20.99	39.75	46.51	66.54	66.09	45.94
	20	5.24	21.82	34.95	49.91	59.84	59.80	49.60
	25	16.03	15.97	31.64	32.05	43.91	47.92	43.76
	30	10.11	16.65	24.15	36.69	56.55	59.80	46.24

ตารางที่ ข-3 อัตราการรับข้อมูลต่อการส่งข้อมูลเมื่อแพ็กเก็ตข้อมูลมีขนาด 80 ไบต์

ระยะการ สื่อสาร (เมตร)	โหนด ต้นทาง	ความถี่ในการส่งแพ็กเก็ตข้อมูล (แพ็กเก็ตต่อวินาที)						
		10.0	5.0	2.0	1.0	0.5	0.2	0.1
10	5	59.85	99.91	99.93	99.92	100.00	100.00	100.00
	10	39.80	49.73	99.98	99.96	99.92	99.80	99.60
	15	20.31	40.00	58.87	99.96	99.92	99.80	99.47
	20	20.00	25.06	54.70	94.97	94.94	94.85	94.66
	25	8.36	12.77	48.53	68.52	95.95	95.88	95.20
	30	10.26	22.11	18.02	72.72	96.63	96.58	93.14
15	5	79.92	99.91	99.91	99.88	99.72	99.80	99.00
	10	29.96	59.98	60.00	69.99	69.87	69.93	70.06
	15	15.30	30.35	73.27	73.28	73.30	66.69	73.30
	20	10.22	30.22	74.88	69.97	74.94	70.01	64.89
	25	12.40	13.23	47.97	70.23	79.98	79.91	75.82
	30	8.66	9.28	31.38	63.81	83.29	83.29	83.14

ตารางที่ ข-3 อัตราการรับข้อมูลต่อการส่งข้อมูลเมื่อแพ็กเก็ตข้อมูลมีขนาด 80 ไบต์ (ต่อ)

ระยะการ สื่อสาร (เมตร)	โหนด ต้นทาง	ความถี่ในการส่งแพ็กเก็ตข้อมูล (แพ็กเก็ตต่อวินาที)						
		10.0	5.0	2.0	1.0	0.5	0.2	0.1
20	5	59.02	79.81	99.92	99.78	99.52	99.50	99.20
	10	24.30	39.89	89.91	89.96	89.72	99.50	89.72
	15	0.04	0.05	78.96	99.93	93.20	86.54	86.42
	20	0.02	24.99	34.93	89.91	79.92	94.93	74.78
	25	4.83	0.43	17.01	67.79	87.91	86.51	80.30
	30	8.82	12.95	33.44	56.35	73.22	79.86	50.27
25	5	54.11	79.78	99.80	99.62	99.84	60.00	79.68
	10	25.62	40.00	59.96	60.01	69.85	59.94	79.54
	15	0.06	7.00	46.57	66.60	53.33	59.96	59.99
	20	15.08	24.53	34.99	49.96	44.96	49.90	49.80
	25	9.12	28.14	35.91	43.96	55.98	59.92	55.88
	30	6.86	8.10	25.16	43.33	56.65	56.61	53.33
30	5	34.06	39.97	79.69	79.93	79.90	79.94	100.00
	10	0.00	49.85	59.77	49.88	69.53	49.90	59.48
	15	0.00	13.39	26.65	46.62	46.60	53.26	52.93
	20	0.00	0.25	24.97	44.94	44.90	44.90	49.60
	25	0.00	12.11	20.94	23.94	47.96	43.78	55.60
	30	0.00	13.49	23.84	18.60	46.54	49.75	49.63

จากผลการทดลองดังตารางที่ ข-1 ถึง ข-3 สามารถสังเกตได้ว่า อัตราการรับต่อการส่งข้อมูลหรือ PDR ได้รับผลกระทบจากการปรับเปลี่ยนพารามิเตอร์ทั้ง 4 ชนิดดังที่กล่าวมาข้างต้น ซึ่งสรุปไว้ในตารางที่ ข-4 ดังนี้

ตารางที่ ข-4 ช่วงของข้อมูลพารามิเตอร์ที่ให้ค่า PDR มากกว่าหรือเท่ากับ 60 เปอร์เซ็นต์

พารามิเตอร์	ช่วงของข้อมูล
ขนาดของแพ็กเก็ตข้อมูล (ไบต์)	20 - 80
ระยะเวลาสื่อสารของโหนด (เมตร)	10 - 20
จำนวนโหนดต้นทาง (โหนด)	5 - 20
ความถี่ในการส่งแพ็กเก็ตข้อมูล (แพ็กเก็ตต่อวินาที)	1 - 10

จากตารางที่ ข-1 ถึง ข-3 บริเวณที่แรเงาคือช่วงพารามิเตอร์ที่ให้ค่า PDR ที่มีค่ามากกว่าหรือเท่ากับ 60 เปอร์เซ็นต์เสมอ ซึ่งสรุปพารามิเตอร์ดังกล่าวไว้ในตารางที่ ข-4 การทดลองข้างต้นแสดงให้เห็นเมื่อเปลี่ยนแปลงขนาดของแพ็กเก็ตข้อมูล จะส่งผลกระทบต่อค่า PDR เล็กน้อย ดังนั้นจึงสามารถใช้ขนาดแพ็กเก็ตข้อมูลเท่าไรก็ได้โดยไม่เกิน 80 ไบต์ แต่ระยะเวลาสื่อสารของโหนด จำนวนโหนดต้นทาง และคาบเวลา จะส่งผลกระทบต่อค่า PDR อย่างมาก จากตารางที่ ข-1 ถึง ข-3 สามารถแสดงให้เห็นแนวโน้มการเปลี่ยนแปลงค่า PDR คือ เมื่อระยะเวลาสื่อสารของโหนดเพิ่มขึ้นจะทำให้ค่า PDR ลดลง เนื่องจากระยะเวลาสื่อสารที่เพิ่มขึ้น เกิดจากการเพิ่มกำลังส่งสัญญาณวิทยุ ทำให้สัญญาณวิทยุของแต่ละโหนดมีโอกาสรบกวนกันมากขึ้น ผลลัพธ์คือเกิดการชนกันของแพ็กเก็ตข้อมูล ทำให้ปริมาณข้อมูลที่รับได้สำเร็จลดลง ด้วยเหตุนี้จึงทำให้ค่า PDR ลดลง ต่อมาคือจำนวนโหนดต้นทางและคาบเวลาในการส่งแพ็กเก็ตข้อมูล จากผลการทดลองสังเกตได้ว่า เมื่อจำนวนโหนดต้นทางเพิ่มขึ้น ค่า PDR จะลดลง และถ้าคาบเวลาในการส่งแพ็กเก็ตข้อมูลลดลงหรืออัตราการส่งข้อมูลมีความถี่มากขึ้น ค่า PDR ก็ลดลงเช่นกัน สืบเนื่องจาก ทั้งสองพารามิเตอร์ส่งผลโดยตรงต่อปริมาณแพ็กเก็ตข้อมูลที่รับส่งในเครือข่าย ทำให้เกิดการแย่งชิงการเข้าใช้ช่องสัญญาณ เกิดความคับคั่งในการสื่อสารและเป็นการเพิ่มโอกาสในการชนกันของแพ็กเก็ตข้อมูล ผลลัพธ์คือปริมาณข้อมูลที่รับสำเร็จลดลง ด้วยเหตุนี้จึงทำให้ค่า PDR ลดลง

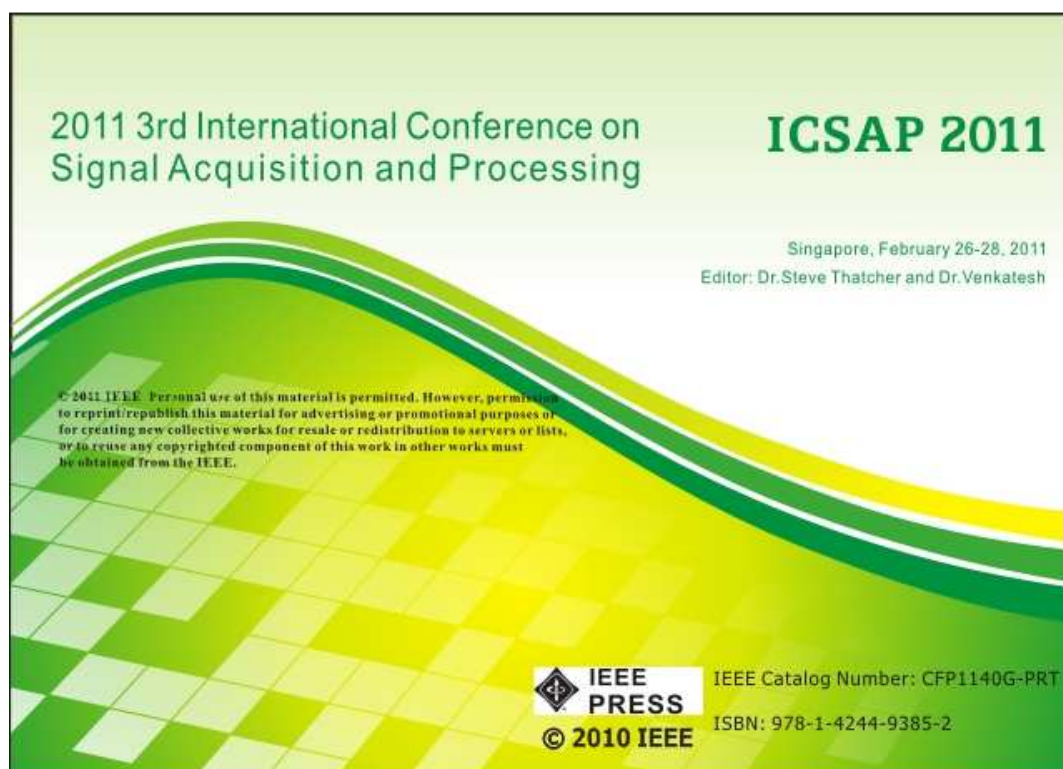
ภาคผนวก ค  
ผลงานตีพิมพ์เผยแพร่จากวิทยานิพนธ์





**2011 The 3rd International Conference on Signal Acquisition and Processing (ICSAP 2011)**

**February 26-28, 2011, Singapore**



## Grid Sectoring: A Cluster Head Election for Load Balancing over Wireless Sensor Networks

Anirooth Thonklin

Computer Engineering Department, Faculty of  
Engineering, Prince of Songkla University  
Hat Yai District, Songkhla Province, Thailand  
anirooth@gmail.com

W. Suntiamorntut

Computer Engineering Department, Faculty of  
Engineering, Prince of Songkla University  
Hat Yai District, Songkhla Province, Thailand  
wannarat@coe.psu.ac.th

**Abstract**—In order to handle with large-scale Wireless Sensor Networks and to provide efficient energy consumption, the network is divided into smaller clusters. Each cluster includes its cluster head responsible for major activities. However, this network still undergoes short-lived function due to several constraints. Consequently, this paper proposed a cluster head election called Grid Sectoring to distribute the load balancing over both uniform and non-uniform dispersion by dividing a grid network area into equal-sized clusters. In other words, there will be equally distribution of the number of member nodes and load balancing in each cluster. As a result, a cluster failure can be resisted because there will be no cluster head dies before the others. Moreover, the selected cluster heads are located nearest to a centre of clusters in order to distribute load balancing and to increase cluster bandwidth. Further, mechanisms of the cluster head election in this proposed can be done by base station to save network energy due to node correspondences. Furthermore, the results not only verified that this proposed algorithm could distribute load balancing better than LEACH, ACHS, and EACHS, but also implied that the proposed could extend the network lifetime even more.

**Keywords**— WSN; cluster head election; member balancing; network lifetime; and load balancing; non-uniform dispersion

### I. INTRODUCTION

Currently, a Wireless Sensor Network (WSN) is widely employable for various fields and purposes. This network composes of a large number of nodes collaborating with each other for the same objective. Lastly, information collected by sensor nodes is periodically transmitted toward a base station (BS). Since a large number of sensor nodes, the lowest cost for node invention makes node be extremely constrained such as poor communication range, inferior capacity, low computing capability, and short life of energy resource. Normally, the nodes can either be scattered or be placed by hand throughout in an area of interest called a monitored area. However, it is possible to place the nodes without random dispersion unless there will be such a large-volume nodes. Furthermore, in large-scale network, it is difficult to deal with dense nodes without complexity avoidance and high energy consumption. As a result, the whole network cannot stay alive to accomplish its mission for much longer due to running out of energy [1].

In the large-scale network, both network congestion and data collisions will waste network energy exceedingly and make the network become short-lived. Consequently, the clustering technique is introduced in order to cope with enormous data, to archive high energy efficiency, to provide more network scalability, and to prolong the network lifetime. In cluster-based network, the networks are partitioned into smaller clusters. There is a cluster heads (CH) in each cluster which is responsible for major activities. E.g., collecting sensed data from member nodes (MN), removing duplicated data, compressing data into smaller size, and transferring data to other CHs along the way as the shortest-path toward the BS. As long as there are alive nodes existing in the network, the nodes which have never been CH will each take turn to be CHs [2, 3]. In other words, the CH plays such an important role. Hence, the cluster head election will directly impact to the entire network performance as well as energy efficiency.

A study's point of view is, to equally distribute the load balancing in each cluster over both uniform and non-uniform dispersion. The study divides monitored area into equal-sized clusters as cake cutting. Intuitively, if the major nodes in network are dispersed uniformly, dividing into equal-sized clusters will give a number of MNs in each cluster evenly. Thereby, every CH will consume the energy evenly and there will be no CH running out of energy before the others. Furthermore, whichever node is located nearest to the center of cluster, it will probably be chosen to be CH. the mechanism of cluster head election can be done by BS to save energy. Therefore, it is high possible that every cluster can evenly distribute load balancing. In summary, the study's results in term of the standard deviation of the number of MNs and the hop count can reasonably be examined that the network lifetime can be extend even more

The rest of this paper is structured as follows. Section II briefly describes and discusses some relevant algorithms of clustering. Section III then introduces the proposed algorithm called Grid Sectoring. There is the key idea as well as the solution of the Grid Sectoring. Next, section IV presents the network performance evaluation. This section includes the network scenarios, experimental results, and the performance analysis as well. Finally, the study even concludes in section V and plan for the next study in section VI respectively.

## II. RELATED WORK

LEACH [4] has been a classical algorithm of cluster head election; the author wishes to prolong the network lifetime by distributing the energy load evenly among the nodes in the network. The nodes each form themselves into clusters and there is only one CH in each cluster at a current round. By the time that the nodes are new dispersed in the monitored area or cluster head cycle time ends. Each node will initially generate a random number between 0 and 1. Then, it will determine itself if it can be the CH or not by comparing the generated number to a threshold. If its number is less than the threshold, it can be a CH readily. Afterward, the elected nodes will broadcast a message to the other nodes located near to them to form new clusters.

The drastic problem of LEACH algorithm may sometimes be caused by the event that the elected CHs are located near an edge of the monitored area, so it can waste energy inefficient. Because there are fewer numbers of the nodes around those CHs than the event that the CHs are located near a center of the clusters. This problem can increase an average distance between CHs and their MNs which directly relates to hop count. The more hops a message traverses, the more resources it consumes harshly.

LEACH-C [7] improves LEACH to enhance the network lifetime even more. In LEACH-C, the CHs are selected by the BS which holds infinite computation-power and infinite energy in order to save the network energy. The BS cannot perform cluster head election properly unless it holds the location and energy of all nodes transmitted to the BS before. Thereby, according to the energy information of all nodes in each cluster, the BS can determine whichever node holds the strongest energy; it can probably to become CH. The outperforming of this algorithm are reducing an energy consumption and extending the network lifetime as well.

In LEACH and LEACH-C, the authors proposed the algorithms of cluster head election for distributing load balancing evenly. Likewise, ACHS [5] and EACHS [6] are proposed in another way which the number of MNs in each cluster are regarded. The key idea of these algorithms is, if the numbers of MNs in every cluster are equally, it will be feasible that both the network load and the energy dissipation will be probably equal highly. Initially, ACHS and EACHS start selection by LEACH algorithm followed by their re-selection algorithm. Each CH uses another nearest CH and the farthest MN in its cluster to determine a new suitable CH. In additionally, EACHS improves ACHS by calculating only the clusters which is different from the average number of MNs for 10 per cent; it can save time and energy for re-selection. The result showed that the network lifetime could be increased.

Nevertheless, ACHS and EACHS may not distribute the nodes in each cluster equally due to the random selection with probability. Furthermore, each round of selection will spend twice which it wastes much time and more energy. In additional, the drawback of selection by random with probability can be caused by disappeared CH regions when

selected CHs are scattered in the network unevenly. This problem resists distributing the load balancing efficiently.

The grid clustering algorithms [8-11] are proposed to avoid complication for cluster formation; these algorithms not only benefit in the large-scale network with the simple way, but further improve the load balancing of the entire network even more. The monitored area is basically divided into minor grids (smaller grid). Each minor grid is a cluster and contains the same size as the others. Whenever the nodes in the network are dispersed uniformly, it is possible that the number of MNs in each cluster is evenly as well. In [8], a proposed has been determined a suitable grid size by considering a transmission range. Furthermore, in [9], an author proposed how to relay the messages from one grid to another grid toward the sink by choosing the best optimal next h-hop routing path via multi-hop fashion.

However, the discussed issues each ignore the suitable location of CH in each cluster. The location of CH can avoid the cluster failure and can provide more bandwidth for intra-cluster communication. This is due to the fact that the nodes around the CH will highly consume the energy and will die first. Hence, the CH located nearest to the center of cluster, it will be able to be surrounded with many more nodes. Thereby, it not only distribute load balancing, but also resist cluster failure even more. Additionally, their functions have not been proven so far for the non-uniform dispersion yet. Moreover, the drawback of the grid clustering [8-11] is, the monitored area can be divided into minor grids with some number such as 1, 4, 9, 16, 25, ... or  $n^2$  only, where  $n$  is the positive integer. For this restrictive number, it is hardly possible to divide a 1<sup>st</sup> minor grid into 2<sup>nd</sup> minor grids with distributing the MNs evenly in the non-uniform dispersion. In other words, some grids contain a lot of nodes, but some contain a few number. Thus, the solution like grid clustering only will be not suitable for the non-uniform dispersion.

## III. GRID SECTORING

### A. The Key Idea of Grid Sectoring

To prolong the network lifetime, the algorithms [5-6, 10] consider distributing the number of MNs in each cluster evenly. Hence, the study proposed the Grid Sectoring which aims at the same purpose. In the uniform dispersion, whenever the monitored area is divided into clusters, and every cluster holds the same size; it is feasible that each cluster holds the same number of MNs as well. Thereby, the network will distribute load balancing evenly and none of the CHs will absent before the cluster head cycle time ends.

As mentioned in the objectives above, this study proposed such an algorithm to form equal-size clusters by dividing the squared-grid network area into equal-sized sectors. In this proposed, the Grid Sectoring algorithm, each sector is denoted a cluster divided by the simple way similar to cake cutting. Normally, in cake cutting, everyone is also experienced that it is always possible to fairly divide a cake among  $n$  people. Hence, in the squared-grid area, it can divide with the same way, a number of people mentioned in cake cutting represents to the desired number of clusters.

After the network is divided into sectors, the center of sector called centroid will be marked. Then, a node located nearest to the centroid in each cluster will be selected to be a new CH. As the CH is located nearest to the center of cluster, the network load and energy dissipation will be managed efficiently. Hence, the simple way for calculating the centroid and searching for the appropriate CHs is combined and details in the next sub-section B. 3) as well.

To perform the proposed algorithm for cluster formation efficiently, Both cluster formation and cluster head election should has been done by BS. It can save the network energy and reduces the complication of network coordination even more [7]. The BS holding the infinite energy and powerful computation is responsible overall activities of cluster formation. In the event that some selected CHs do not exist in the clusters, the BS can re-search for the new suitable CHs rapidly with little energy consumption.

In the non-uniform dispersion, the use of grid clustering is not possible to evenly distribute the number of MNs in each minor grid, due to fact that the number of divisions will be limited. In other words, there are  $n^2$  minor grids can be divided such as 1, 4, 9, 16, ... respectively. Therefore, this proposed algorithm provides better way in this case. The comparing figure between the typical Grid Sectoring and the general grid clustering are illustrated in the Fig. 1.

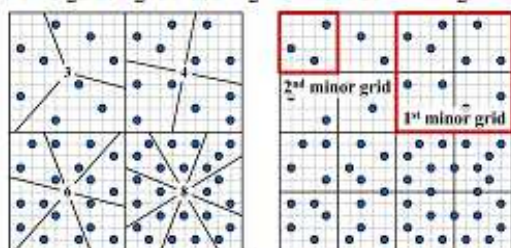


Figure. 1 The comparison between the Grid Sectoring (a) and the squared grid clustering (b) in the same non-uniform dispersion.

In large-scale non-uniform dispersion, before the monitored area is divided into sectors, it will be divided into several minor grids first. At this time, each minor grid may contain the different number of MNs. Hence, the BS then continues dividing each minor grid with a suitable number of sectors depending on node density. Whichever minor grid contains dense nodes, it will be divided into many sectors even more. On the contrary, sparse network, it will be divided into fewer numbers of sectors at once. Thereby, each cluster will contain such the number of MNs equally.

Furthermore, The Grid Sectoring does not depend on any routing and any communication fashion such as single hop or multi-hop routing. The popular routings for WSN can be used to accompany with this proposed algorithm and do not impact to its mechanism. Nevertheless, the Grid Sectoring will have been done satisfactorily when the network contains the location-base and energy-aware nodes.

## B. The Grid Sectoring Solution

1) *Network Partition*: Owing to the Grid Sectoring performs better on the square-grid network and the nodes are sometimes dispersed non-uniformly. Thus, the network should be divided into several minor grids firstly. The number of minor grids depends on the node density. The study provides the BS to hold the position of the entire nodes. It can determine the suitable number of minor grids by adapting an  $N \times N$  matrix to store the node distribution information, the values of the matrix elements are depended on the number of nodes that even fall into the grid [12]. It can be clearly revealed in Fig. 2, this matrix store the nodes presented in Fig. 1 (right) for  $1^{st}$  minor grid division.

$$\begin{bmatrix} 9 & 11 \\ 18 & 24 \end{bmatrix}$$

Figure. 2 A typical distribution of nodes falling into a  $2 \times 2$  matrix.

2) *Grid Sectoring Algorithm*: As the monitored area has been divided into minor grids, each grid will go on computing the Grid Sectoring with the suitable number of sectors. The perimeter ( $p$ ) of a grid of interest in the Fig. 3 will be divided into  $n$  equal length ( $l$ ); where  $n$  is a required number of sectors depending on node density. Hence, the perimeter ( $p$ ) is divided by  $n$  and the result is a length ( $l$ ). There is an equation to understand being set as (1).

$$p/n = l \quad (1)$$

On a contour line in Fig. 3, when the lines are drawn from  $b$ , which is the end point of each length ( $l$ ) toward the center of grid ( $o$ ), the  $n$  sectors will be established. Thereby, every sector ( $a_i$ ) will absolutely contain the area size equally. The Fig. 3 can clearly illustrate how the Grid Sectoring even divides any grid area into equal sectors.

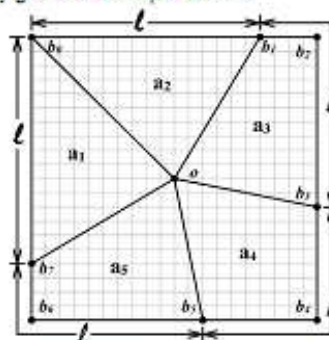


Figure. 3 The grid dividing with equal cluster size of the Grid Sectoring.

To prove the size of each sector as the mentioned before, the study supposed that the grid as seen in Fig. 3 has  $20 \times 20$   $m^2$  size, the perimeter ( $p$ ) is 80 meters long. The required number of sectors ( $n$ ) is 5, thus the length ( $l$ ) after  $p$  is divided by  $n$  is  $80/5 = 16$  meters long. In this supposed, each ideal sector-area should be  $(20 \times 20)/5 = 80$   $m^2$  size. Also, the

typical triangular area  $a_2$  in Fig. 3 has actually  $0.5 \times 16 \times 10 = 80 \text{ m}^2$  size and it is the same size as the other sectors.

3) *Cluster Centroid and Cluster Head Election:* Whenever the CHs are located near the center of their clusters, they will be surrounded with many more nodes. Hence, it is possible to increase bandwidth significantly and load balancing will be distributed efficiently. The study also takes into the center point of cluster that called centroid. Thus, the centroid of polygon-area can be calculated simply with the centroid formula of polygon which is set in (2-4).

$$C_x = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (2)$$

$$C_y = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (3)$$

$$A = \frac{1}{2} \sum_{i=0}^{n-1} (x_i y_{i+1} - x_{i+1} y_i) \quad (4)$$

It is necessary to calculate the area ( $A$ ) of the polygon area of interest by the formula (4) firstly. From this area, the study also gets the centroid position  $(C_x, C_y)$ . According to these formulas, the study can mark where the centroid point is by knowing the overall vertex point  $(x_i, y_i)$  of the polygon area. Therefore, the study applies such this algorithm in order to search for whichever node located nearest to the centroid will become the new suitable CHs. In other words, whichever node even holds such the shortest distance to the centroid, that node will sudden be a new CH of that cluster.

4) *Boundary-Considering:* By the time that the Grid Sectoring has already been selected and the CHs located nearest to the centroid in every clusters, each CH will broadcast a cluster boundary-notation toward the other nodes nearby. This notation contains the cluster head ID and the set of cluster boundary-information. The other nodes which receive this notation will consider themselves whether they are in that cluster area or not. If so it will sudden join in that cluster. The typical set,  $(a, b_1, b_2, b_3)$  of cluster boundary of the cluster  $a_3$  are shown in the Fig. 3.

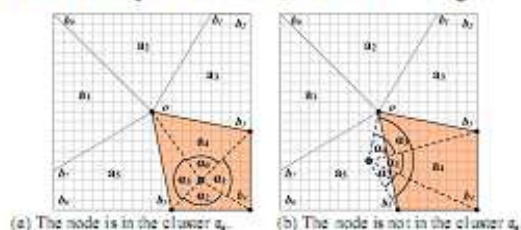


Figure. 4 The typical of boundary checking.

The node which receives the cluster notation will determine itself whether it is in such the cluster area or not by calculating the summation of every angle. These angles can be clearly revealed in the Fig. 4 such as the cluster  $a_4$ , there are four angles ( $a_1$  to  $a_4$ ) around the considering node.

These angles are also calculated by applying the law of cosine. If the summation of the all angles is close to  $2\pi$  as seen in Fig. 4 (a), this node certainly ensure that it is in that cluster and it will join in that cluster suddenly. On the other hand, if the summation is more than  $2\pi$  as seen in Fig. 4 (b), that node demonstrates itself that it is not in that cluster area.

IV. PERFORMANCE EVALUATION

This section, the study compared LEACH, ACHS, and EACHS to the proposed algorithm, Grid Sectoring. In order to investigate the load balancing, the study also evaluated the performance by using performance metrics such as the standard deviation of the number of MNs in cluster, as well as hop count estimated by the average distance between the CHs and their MNs as comparative factors respectively. The whole implementations are verified via Networkx which is a graph simulator. A snapshot of a result of Grid Sectoring is shown in Fig. 5. There are 120 nodes scattered over the uniform network. A grid area is divided into 6 sectors and obtains 6 CHs located nearest to the centroid.

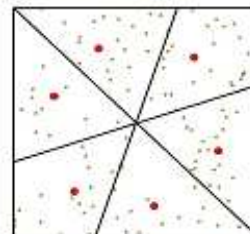


Figure. 5 The snapshot of Grid Sectoring algorithm.

A. Simulation Scenario

The network topology is organized by dispersing the nodes randomly with geometric distribution. The study simulates these algorithms with 100, 200, 300, 400, and 500 nodes in the area of interest on a  $100 \times 100 \text{ m}^2$  size. There are about 5 per cent of CHs in the entire nodes [4]. The number of CHs depends on node density as seen in Table I. The gray color in this table is an environment of the Grid Sectoring only. For each experiment, the study iterant simulated for 20 times.

TABLE I. THE NUMBER OF CHs DEPENDING ON NODE DENSITY

Node Density (nodes)	For Grid Sectoring		The Number of CHs		% of CHs
	The Number of Minor Cells	The Number of Sectors	Grid Sectoring	Other Algorithms	
100	1	5	5	~5	5%
200	1	10	10	~10	5%
300	4	4	16	~16	5.33%
400	4	5	20	~20	5%
500	4	6	24	~24	4.8%

To measure the performance of the proposed algorithm, the first scenario, the study counted the number of MNs in each cluster and evaluating the standard deviation. The number of CHs is 5 per cent of the total nodes. For instance,

there are 100 nodes dispersed in the network uniformly; there will be 5 nodes becoming CHs. Thus, the rest 95 nodes will be MNs. Therefore, each cluster should contain  $95/5 = 19$  MNs. The study used 19 nodes to compare to the result of the average number of MNs for each algorithm such as LEACH, ACHS, EACHS, and the proposed Grid Sectoring to get their standard deviations. Lastly, the study needs to evaluate hop count estimated by measuring the average distance between CHs and their MNs in each cluster.

### B. Performance Analysis

The first scenario, the study obtained the results of average standard deviation among four algorithms as shown in Fig. 5 and Fig. 6 which the experiments contain 100 nodes and varied nodes between 100 and 500 respectively.

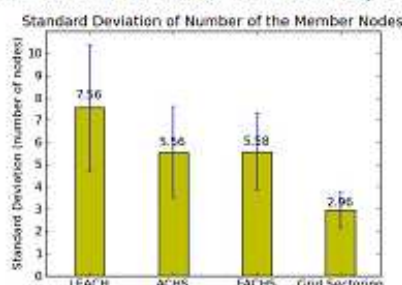


Figure 5 Standard deviation of the number of MNs with 100 nodes.

The chart on Fig. 5 shows the average standard deviation of the number of MNs in every cluster among four algorithms with 100 nodes. Though at varying values of standard deviation, it is projected that LEACH will have the highest value at 7.56 followed by ACHS and EACHS. The value of standard deviation for both algorithms is closely at 5.56 and 5.58 respectively. The Grid Sectoring, by contrast with the lowest projected value at 2.96. Furthermore, both the result indicated that the proposed algorithm provide more reliable in number of MNs equally. In additionally, the study then simulated in many more nodes varied between 100 and 500, the results are illustrated in Fig. 6.

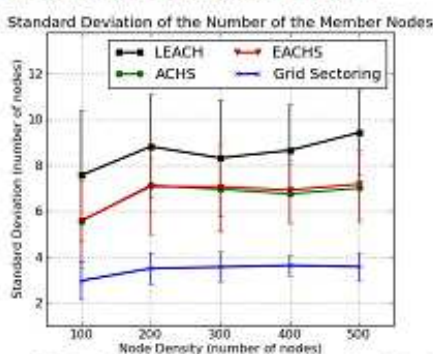


Figure 6 Standard deviation of the number of MNs with varied nodes between 100 and 500.

The graph on Fig. 6 provides information about the standard deviation of the number of MNs among four algorithms with the varied node density. The value of LEACH, ACHS, and EACHS go up by varying degree. LEACH fluctuated wildly between 7.56 and 8.82, but the trend was upward to 9.43 in 500 nodes. The similar pattern is repeated for ACHS and EACHS; their values both are upward from 5.56 to 7.13 between 100 nodes to 200, but after that the values fall steady and then climb gradually. It is also noticeable that EACHS' value is comparable to ACHS'. But in 500 nodes it overtakes ACHS' value to 7.18 a little. By contrast, the Grid Sectoring is the lowest degree; its value increases steadily from 2.96 to 3.50 between 100 nodes and 200 and then levels off about 3.56 to 500 nodes.

The last scenario, the study obtained the average distance between CHs and their MNs for each algorithm with 100 nodes in the network. The results are illustrated in Fig. 7.

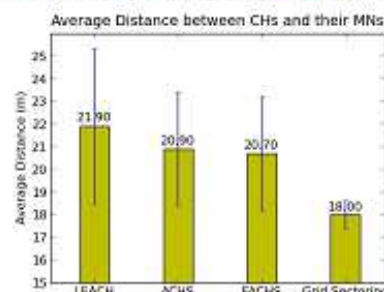


Figure 7 The average distance between CHs and their MNs with 100 nodes.

The chart on Fig. 7 provided information about the average distance between the CHs and their MNs. The data as seen in Fig. 7, LEACH has the furthest average distance at 21.90 meters long, whereas in comparison with the average distance of ACHS and EACHS is slightly lower. The average distance for both ACHS and EACHS is closely at 20.90 and 20.70 meters long respectively. The Grid Sectoring, by contrast, is the shortest average distance compared to the others at 18.00 meters long. Thereafter, the study continues simulating in varied nodes between 100 nodes and 500.

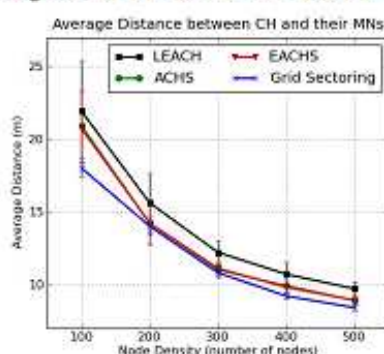


Figure 8 The average distance between CHs and their MNs with varied nodes between 100 and 500.

The chart on Fig. 8 provided information about the average distance between the CHs and their MNs among four algorithms with varied node density. For every algorithm, the distances fall dramatically between 100 nodes and 300 but after 300 nodes the rate of decrease is significantly slower than the previous density. As regards the first, LEACH, by comparison, experiences a continual drop in length between 100 nodes to 500, 21.9 meters long and 9.7 respectively. The same pattern followed by the distances of EACHS and ACHS, there are close degree in distances for every varied node but their degrees are lower. The Grid Sectoring, the figures show such a downward trend over the period 18.0 to 8.4, there is the lowest degree in distances. From the data in this chart, the distances go downward with the increase in node density. This is due to the fact that, in the denser network with the fixed size of the area of interest, the more number of CHs the network contains, the smaller the cluster size is. In summary, this proposed algorithm is also verified reasonably that it can give the shortest distance which directly relates to the smallest number of hop count.

#### V. CONCLUSIONS

The proposed algorithm, the Grid Sectoring is a simple concept for the cluster head election irrespective of how the nodes in network are dispersed. The first result even indicated that the proposed algorithm provided such the lowest value of the standard deviation, it will experience that the proposed algorithm can equally distribute the number of MNs better and can increase the load balancing even more. Then the second result of the average distance between the CHs which located nearest to the centroid and their MNs can overcome the other algorithms with such the shortest average distance. In general, this distance directly involves the hop count significantly. Consequently, the decreased distance of the proposed can reduce such the hop count and also decrease the resource consumption drastically. Moreover, the results of the proposed can perform significantly better in the large-scale network.

As both results mentioned above confirm that the proposed algorithm, Grid Sectoring, can distribute traffic load in such the network efficiently as well as even prolongs the greatest overall network lifetime actually. Furthermore, the Grid Sectoring mechanism which has been done by the BS with simple way will be able to save such network resource, both time and energy, even more.

#### VI. FUTURE WORK

The study will also implement the proposed algorithm via Network Simulator 2 and verify how the Grid Sectoring impacts to the network performance about the network lifetime over both large-scale and non-uniform network. In order to extremely extend the network lifetime, the study regards the residual energy to become CH and the distance between the CHs and BS in order to be decided how many sectors to be divided. This is because of this distance concerns about the efficiency of load and energy dissipation

reasonably in the network. In other word, the clusters located nearest to the BS will spend drastically energy due to relaying a large volume of messages. Therefore, the nearer the BS the clusters are, the more number of clusters the region around the BS is divided.

#### ACKNOWLEDGMENT

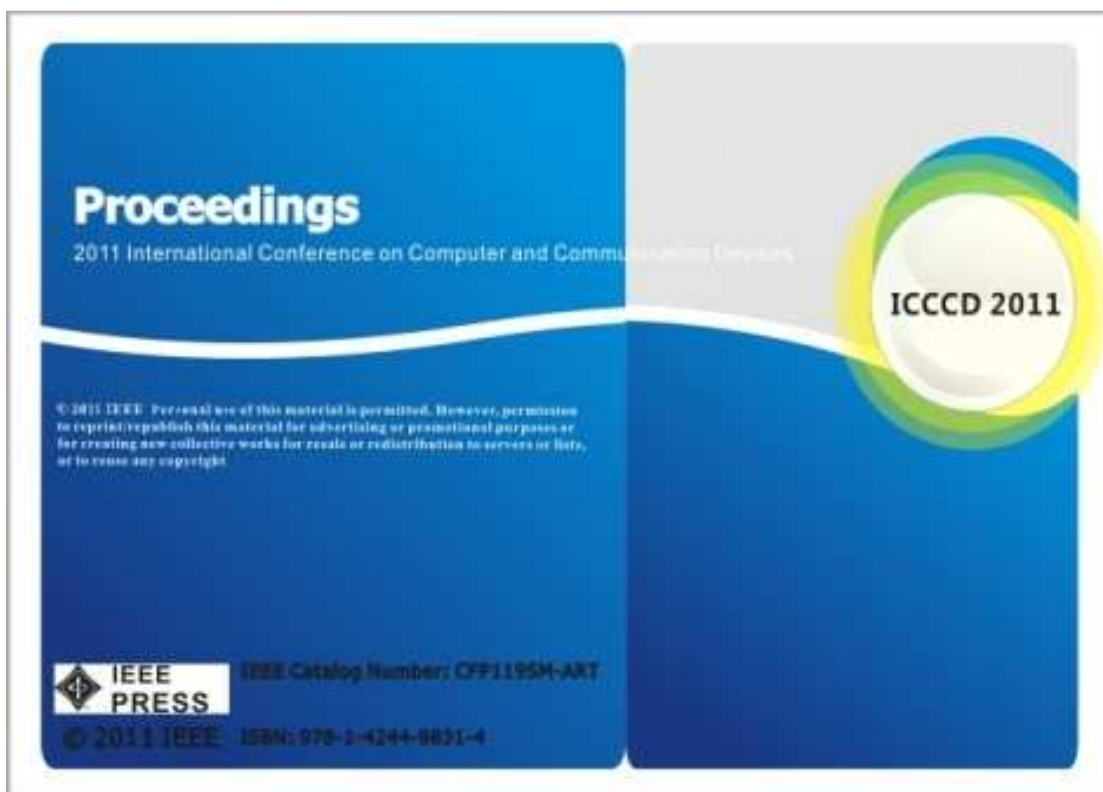
The research reported here was supported by a research grant from the Telecommunications Research and Industrial Development Institute (TRIDI), the office of National Telecommunications Commissions (The office of NTC), Thailand. The scholarship ID is NTC 008/2552.

#### REFERENCES

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramanian, and E. Cayirci, "A Survey on Sensor networks" *Communications Magazine*, IEEE, Volume: 40, Issue: 8, 2002, pp. 102 - 114, August 2002.
- [2] L. Marboleda C., and N. Nasser, "Comparison of Clustering Algorithms for Wireless Sensor Networks," *Electrical and Computer Engineering*, 2006. CCECE '06. Canadian Conference, 2006, pp. 1787 - 1792, May 2006.
- [3] P. Kumarawada, D.J. Dechene, M. Luccini, and A. Sauer, "Algorithm for node clustering in Wireless Sensor Networks: A Survey," *Information and Automation for Sustainability*, 2008. ICIAPS 2008. 4th International Conference, 2008, pp. 295 - 300, December 2008.
- [4] W. Heinzelman, A. Chandrakasa, and H. Balakrishnan, "Energy Efficient Communication Protocols for Wireless Microsensor Networks," *Proceedings of Hawaii International Conference on System Science*, January 2000.
- [5] C. Nam, H. Jeong, and D. Shin, "The Adaptive Cluster Head Selection in Wireless Sensor Networks," *IEEE International Workshop on Semantic Computing and Application*, pp. 147-149, July 2008.
- [6] C. Nam, Y. Ku, J. Yoon, and D. Shin, "Cluster Head Selection for Equal Cluster Size in Wireless Sensor Networks," in *Proceedings of IEEE - New Trends in Information and Service Science*, 2009. NISS, pp. 618-623, July 2009.
- [7] E. B. Heizerman, A. P. Chandrakasa, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," *IEEE Transactions on Wireless Communications*, Vol 1, No. 4, October 2002.
- [8] Y. Zang, J. Pan, and G. Wu, "Energy-Optimal Grid-Based Clustering in Wireless Microsensor Networks," *Distributed Computing Systems Workshops: ICDCS Workshop '09*, 29th IEEE International Conference, June 2009.
- [9] W. Liu, Y. Sun, J. Yu, and B. Cao, "A Grid-based Distributed Multi-hop Routing Protocol for Wireless Sensor Networks," *Embedded and Ubiquitous Computing 2008: EUC '08*, IEEE/IFIP international conference, pp. 330-336, December 2008.
- [10] L. Li, X. Jiang, S. Zhong, and L. Hu, "Energy Balancing Clustering Algorithm for Wireless Sensor Network," *Networks Security, Wireless Communications and Trusted Computing: NSWCTC '09*, pp.61-64, April 2009.
- [11] M. Zhang, C. Gong, Y. Feng, and Y. Lu, "An Novel Energy-Efficient Dynamic Target Tracking Protocol for Wireless Sensor Networks," *Networks Security, Wireless Communications and Trusted Computing: NSWCTC '09*, pp.649-652, April 2009.
- [12] L. Xiong, J. Wen, W. Huang, and M. Yu, "The Research of Mobile Fusion-nodes Routes for Data Collection in Multi-tier Mobile Wireless Sensor Network," *2009 First International Workshop on Education Technology and Computer Science: ETCS '09*, March 2009.



**2011 The International Conference on Computer Communication Devices (ICCCD 2011)  
April 1-3, 2011, Bali Island, Indonesia**





## A Load Balanced Cluster Head Election for Uniform/Non-uniform Deployment over Wireless Sensor Networks

Anirooth Thonklin

Department of Computer Engineering  
Faculty of Engineering, Prince of Songkla University  
Hat Yai District, Songkhla Province, Thailand  
anirooth@gmail.com

W. Suntiamorntut

Department of Computer Engineering  
Faculty of Engineering, Prince of Songkla University  
Hat Yai District, Songkhla Province, Thailand  
wannarat@coe.psu.ac.th

**Abstract**—Energy efficiency is the important key for Wireless Sensor Networks. To achieve long lifetime in large-scale network, the network is divided into several clusters; each cluster includes its cluster head responsible for every activities which impact to network performance. Consequently, this study proposes a cluster head election scheme called Grid Sectoring to increase distribution of load balancing over both uniform and non-uniform deployment by partitioning a grid network area into equal-sized clusters. In other words, each cluster will equally distribute number of node member and enhance load balancing. Moreover, the study provides the selected cluster heads which are located nearest to a center of each cluster in order to enhance distribution of load balancing as well. This position of selected cluster heads will be surrounded with many nodes to reduce the bottleneck problem. Furthermore, the results not only verified that the proposed algorithm could distribute load balancing better than LEACH, ACHS, and EACHS, but also implied that the proposed could extend the network performance and the network lifetime even more.

**Keywords**—WSN; cluster head election; member balancing; load balancing; uniform and non-uniform deployment

### I. INTRODUCTION

In a large-scale wireless sensor network (WSNs), the clustering play a significant role for reliable and energy efficient data dissemination. In cluster-based network, the networks are partitioned into smaller clusters. There is a cluster head (CH) within each cluster which is responsible for most activities. E.g., intra cluster communication, data aggregation, and inter cluster communication. Normally, the nodes which have never been CH will take turn to be CHs [1, 2] to balance the network energy. Hence, the cluster head election is such the main function to enhance the network performance as well as energy efficiency in advance.

A study's point of view is, to equally distribute the load balancing in every clusters over both uniform and non-uniform deployment. To achieve this goal, the study contribute a simple, load balanced, energy-efficient algorithm of cluster head election over WSN by dividing the area of interest into equal-sized clusters as cake cutting. Intuitively, if the most nodes in network are uniformly deployed and all clusters are similar in size, the number of member nodes (MNs) in each cluster can be more evenly distributed. Thereby, every cluster will distribute load and consume energy equally. Thus, there will be no CH running out of energy before the others. Furthermore, to distribute

more load balancing and energy efficiency, the study take the location of CHs into account by choosing the CHs located nearest to center of each cluster and provides the proposed mechanism done by BS to save energy. In summary, the study's results can reasonably be examined that the proposed scheme can overcome other algorithms such as LEACH, ACHS, and EACHS in term of load balancing.

The rest of this paper is organized as follows. Section II briefly discusses some relevant algorithms of clustering. Section III then introduces the proposed algorithm, Grid Sectoring. Next, section IV, the study presents the network performance evaluation. This section includes the network scenarios and experimental results. Finally, the study has been concluded the proposed algorithm in section V.

### II. RELATED WORKS

LEACH [3] has been a classical algorithm of cluster head election; to prolong the network lifetime by distributing the energy load evenly among the nodes in the network. The nodes each form themselves into clusters. In set-up phase, each node assigns itself a probability which is a number between 0 and 1. If this probability is less than a predefined threshold, then that node becomes a CH. However, there are some problems about this algorithm. It is sometimes caused by the elected CHs located near an edge of the area of interest, so it can wastes energy inefficient and it sometimes produces CHs closer to each other that makes the clusters imbalanced in term of the number of MNs and loads.

LEACH-C [6] improves LEACH to enhance the network lifetime. The selected CHs are chosen by the BS which holds infinite computation-power and infinite energy in order to save the network energy. The BS can determine nodes which hold the strongest energy; they can probably be CHs. The outperforming of this algorithm are reducing energy consumption and extending the network lifetime.

ACHS [4] and EACHS [5] are proposed to enhance load balanced clusters; the number of MNs within each cluster is taken into account. If the numbers of MNs in every cluster are equally, there will be feasible that both the network load and the energy consumption will be probably equal highly. Initially, ACHS and EACHS start selection by the same way as LEACH algorithm, and they then re-select the new suitable CHs by determine a nearest CH and a farthest MN for every cluster. EACHS improves ACHS by re-selecting only the clusters which differ from the average number of MNs for 10 per cent to save time and energy.

Nevertheless, LEACH, ACHS and EACHS may not distribute the nodes in each cluster equally due to the probabilistic random selection. Furthermore, LEACH can produce CHs closer to each other and ACHS and EACHS algorithms use twice times a round of selection that will spend much time and much more energy. In addition, the drawback of probabilistic random selection is caused by disappeared CH regions when selected CHs are scattered in the network unevenly. This problem even resists distributing of load balancing and energy efficiency drastically.

The algorithms of grid clustering [7-10] are proposed to avoid complication of cluster formation; these simple algorithms not only benefit in the large-scale network, but further improve the load balancing even more. The area of interest is basically divided into minor grids (smaller squared-grid). Each minor grid is a cluster area and holds the same size as the others. In uniform deployment, there will be possible that the number of MNs in each cluster is evenly. Zuang [7] has been determined suitable grid size by considering a transmission range. Furthermore, Liu [8] proposed how to relay the messages from one grid to another grid toward a sink by choosing the best optimal next h-hop routing path via multi-hop fashion with energy efficiency.

However, the discussed issues each ignore the suitable location of CHs where CHs can provide more bandwidth for intra-cluster communication. Normally, the nodes around the CHs will highly consume the energy and will die first. Hence, the CHs located nearest to the center of each cluster will be able to be surrounded with many more nodes. Thereby, the routing paths to the sinks (every CH) will be distributed to resist bottleneck problem. Additionally, their functions have not been proven so far for the non-uniform deployment yet. Lastly, the drawback of the grid clustering is, the area of interest can be divided into minor grids with limited number such as 1, 4, 9, 16, 25, ... or  $n^2$ .

### III. GRID SECTORING

#### A. The key Idea of Grid Sectoring

The study used the similar idea as [4-5, 9]. That is, if the cluster are similar in number of MNs, loads can be more evenly distributed within each cluster. Hence, the study proposed the Grid Sectoring at which aims this idea majorly. The study assumes that the nodes in networks contain location-aware sensors like Global Positioning Sensor (GPS) to sufficient archive the proposed algorithm. In the uniform deployment, to obtain the same size of all clusters, the area of interest should be divided into equal-size clusters, each cluster can also hold the same number of MNs as well.

Grid Sectoring, the squared-grid area will be divided into equal-sized sectors, each sector is denoted a cluster divided by the simple way similar to cake cutting. Intuitively, in cake cutting, everyone is also experienced that it is always possible to fairly divide a cake among  $n$  people. Further, Grid Sectoring produces CHs nearest to the center of their dusters to distribute load and energy dissipation efficiently. To achieve this goal with energy efficiency, both cluster formation and cluster head election should has been done by

the BS only in order to save the network energy and to reduce the complication of network coordination [6].

In non-uniform deployment, grid clustering cannot evenly distribute the number of MNs within each minor grid suitably, due to limited number of division such as 1, 4, 9, 16, ... etc. The proposed also provides a better way by diving the area of interest into several minor grids firstly. Each minor grid may contain different number of MNs. The BS then continues dividing each minor grid into sectors depending on node density. The more number of MNs the minor grid contains, the more number of sectors that minor grid is divided. Finally, there will be more equally number of MNs within each cluster. A typical of Grid Sectoring for non-uniform deployment is illustrated in the Figure. 1(a).

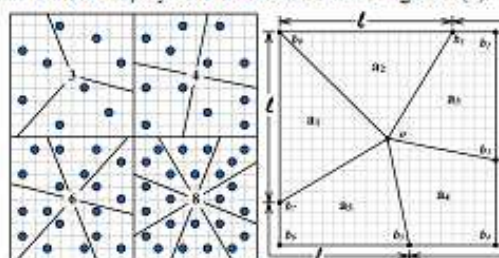


Figure 1. (a) Grid Sectoring for non-uniform deployment and (b) How Grid Sectoring divides the square-grid into sectors.

#### B. The Solution of Grid Sectoring

1) *Pre-Partition*: Grid Sectoring can be used in advance on the square-grid area, the network should also be divided into several minor grids as grid clustering firstly. The number of sectors, each minor grid will be divided, will be calculated by BS depending on the node density.

2) *An Algorithm of Grid Sectoring*: Each divided minor grid will go on re-dividing with a suitable number of sectors by an equation being set as (1).

$$p/n = l \quad (1)$$

The Figure. 1(b) can clearly be illustrated how the Grid Sectoring even divides any grid area into equal-sized sectors. Where  $p$  is a perimeter of a grid,  $n$  is a required number of sectors, and  $l$  is a length to divide minor grid into sectors. Thereby, every sector  $a_i$  will absolutely contain equal size.

3) *Centroid and Cluster Head Election*: To enhance cluster bandwidth, load balancing, and to resist bottleneck problem, the study also takes the location of CHs into account by choosing the nodes located nearest to a center of cluster to be CHs. In order to get the center point, the study applies centroid formulas of polygon which is set in (2-4).

$$C_x = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (2)$$

$$C_y = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (3)$$

$$A = \frac{1}{2} \sum_{i=0}^{n-1} (x_i y_{i+1} - x_{i+1} y_i) \quad (4)$$

After the centroid position (Cx,Cy) has been obtained, a node which is located nearest to the centroid will be chosen to be a new suitable CH of each cluster by the BS.

#### IV. PERFORMANCE EVALUATION

In term of the result of cluster head election, the study compared the proposed algorithm to LEACH, ACHS, and EACHS. In order to investigate the load balancing, the performance-metrics such as average absolute standard deviation (SD) of the number of MNs per cluster, average distance between CHs to their MNs, average hop count, average absolute standard deviation of load per sink (cluster head), network throughput, packet delivery ratio (PDR), and normalized routing load (NRL) were evaluated as comparative factors respectively.

##### A. Simulation Scenario

The network topology is organized into 2 sub-sections classified by node deployment in the network; in short, there are uniform distribution and non-uniform distribution. The study simulated via NetworkX and NS-2. The study used about 5 per cent of CHs in the entire nodes [3] and simulated in different 20 topologies a sub-section. The set up environment were shown in Table I.

TABLE I. THE ENVIRONMENT SET UP.

Parameters	Value
Simulation time	600 second (10 minutes)
Area of interest	100x100 m <sup>2</sup>
MAC protocol	IEEE 802.15.4
Frequency/Bandwidth	2.4GHz/250kbps
Transmission range	15 meters
Traffic type	CBR (constant bit rate)
Queue size	150
Packer size	50 bytes (at Application Layer)
Initial energy	1000 Joules/node
Traffic load (per node)	1 packet per second
Routing type	AODV (Ad hoc On-demand Distance Vector)
HELLO time interval	10 sec

1) *Uniform Deployment*: The whole nodes in network were deployed with uniform random distribution. The number of nodes and CHs would be defined as seen in Table II. This scenario, the study measured network performance such as the average absolute SD of the number of MNs per cluster, the average distance between CHs and their MNs, the average hop count, the average absolute SD of load per sink, the network throughput, PDR, and NRL.

TABLE II. THE NUMBER OF CHs FOR UNIFORM DEPLOYMENT.

Node Density (nodes)	For Grid Sectoring		The Number of CHs		% of CHs
	The Number of Minors	The Number of Sectors	Grid Sectoring	Other Algorithms	
100	1	5	5	5	5%
200	1	10	10	10	5%
300	4	4	16	16	5.33%
400	4	5	20	20	5%
500	4	6	24	24	4.8%

	Grids				
100	1	5	5	5	5%
200	1	10	10	10	5%
300	4	4	16	16	5.33%
400	4	5	20	20	5%
500	4	6	24	24	4.8%

2) *Non-Uniform Deployment*: The study simulated 500 nodes scattered over the network area, the nodes would be fallen into four regions as seen in Figure 2. The proposed divided each regions into sectors depending on node density. In short, there will be sectors or CHs about 5 per cent within each regions. The study evaluated the average absolute SD of MNs per cluster, the average distance between CHs and their MNs, and the average hop count.

~60 nodes 3 CHs	~400 nodes 5 CHs
~120 nodes 6 CHs	~220 nodes 10 CHs

Figure 2. The density of nodes in each region deployed over the network area for non-uniform random deployment.

##### B. Performance-Metrics and Results

1) *The Average Absolute Standard Deviation of MNs per Cluster*: In general, to measure balanced number of MNs in every cluster, this metric is utilized to demonstrate how different number of MNs among several clusters. The fewer number of SD each algorithm gives, the more load balancing it provides. Figure 3 and 6 (SD) showed the result of the average absolute SD for uniform deployment and non-uniform deployment respectively.

Standard Deviation of the Number of Member Nodes per Cluster

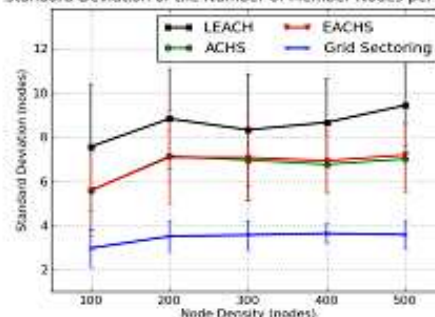


Figure 3. The average absolute SD of the number of MNs per cluster for uniform deployment.

The graph on Figure 3, the proposed algorithm gives the lowest degree with increasing steadily value from 2.96 to 3.50 between 100 and 200 nodes per sink and then levelling off about 3.56 to 500 nodes. The chart on Figure 6 (SD) provided for non-uniform deployment with 500 nodes. LEACH gives the furthest value at 8.54 nodes per cluster, whereas in comparison with the values of ACHS and EACHS are slightly lower and their values are closely at 6.73 and 6.86 nodes respectively. The proposed, by contrast, gives the lowest value at 3.32 nodes.

2) *The Average Distance between CHs and Their MNs:* This metric was described how long the MNs forward overwhelming data to the CHs; If there is shorter distance, the overall cluster energy will be consumed lower. Figure. 4 and 6 (Distance) showed the result of the average distance for uniform and non-uniform deployment respectively.

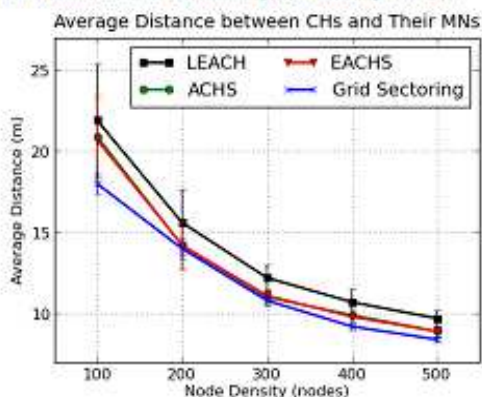


Figure 4. The average distance between CHs and their MNs for uniform deployment.

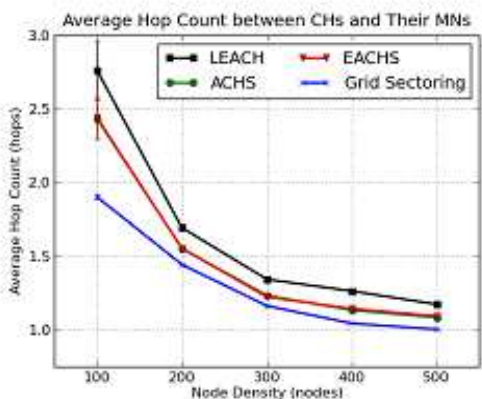


Figure 5. The average hop count for uniform deployment.

The result of the proposed on Figure. 4, the figures showed a downward trend over the period about 18.0 to 8.4, there is the lowest value in distances. The Similar results on Figure. 6 (Distance); the proposed algorithm still gives the shortest distance even though the nodes in the network were deployed with non-uniform random distribution.

3) *The Average Hop Count:* This metric related to the average distance in the previous sub-section. However, here is the result of the average hop count which the nodes' transmission range was set to be 15 meters long. The result

were shown on Figure. 5 and 6 (Hop Count) for uniform deployment and non-uniform deployment respectively.

The graph on Figure. 5 and the chart on Figure. 6 (Hop Count) were clearly visible that the proposed gives the lowest hop count when it was compared to the others.

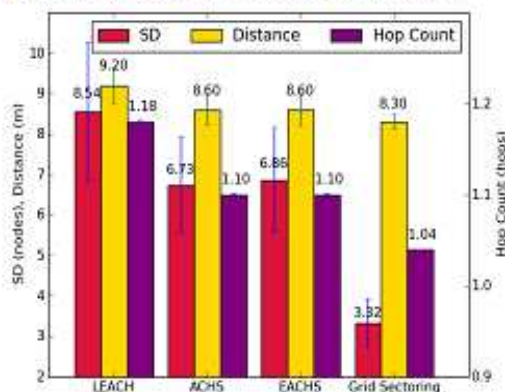


Figure 6. The results of the average absolute SD of the number of MNs per cluster, the average distance between CHs and their MNs, and the average hop count of 500 nodes for non-uniform deployment.

4) *The Average Absolute Standard Deviation of Load per Sink:* This metric is useful to test how effective the load balancing each algorithm balances the loads in network. The less value each algorithm gives, the more balanced that algorithm provides. The results of 100 nodes for uniform deployment was illustrated on Figure. 7 (SD of Load/Sink).

The results projected that the proposed gives the lowest value of the average absolute SD of load per sink at 1.22 kbps, its value lower than two time of the other values.

5) *Throughput:* The basical metric to measure network performance is a rate of successful delivered data in a certain time. Normally, the maximized value is the expected performance of any systems. The results of 100 nodes for uniform deployment was shown on Figure. 7 (Throughput).

The results on Figure. 7 (Throughput), the value of LEACH, ACHS, and EACHS are closely at 7.21, 7.33, and 7.43 kbps respectively. By contrast, the proposed algorithm projected the highest value at 8.04 kbps.

6) *Packet Delivery Ratio (PDR):* This is the measured fraction of the successful number of data received by sinks divided by the number of data sent by sources. The PDR is usually measured in per cent. The results of 100 nodes for uniform deployment was shown on Figure. 7 (PDR).

The results of PDR on Figure. 7 (PDR) showed the values of LEACH, ACHS, and EACHS are projected closely at 68.09, 69.20, and 70.16 per cent respectively. By contrast, the proposed projected the highest value at 75.95 per cent.

7) *Normalized Routing Load (NRL):* The ratio of the total number of routing packets transmitted to the total

number of data packets received at the sinks (CHs) at the network layer. This metric implies how the algorithm makes the network gives efficiency of the routing protocol and power consumption. The results of 100 nodes for uniform deployment was shown on Figure. 7 (NRL).

The results of NRL on Figure. 7 (NRL) illustrated that the proposed can reduce the routing loads by the projecting the lowest value at 21.50 per cent compared to the others.

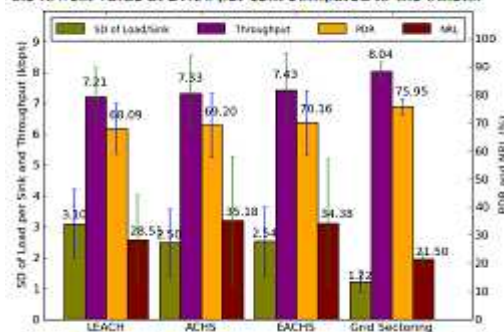


Figure 7. The results of the average absolute standard of load per sink, throughput, PDR, and NRL of 100 nodes for uniform deployment.

## V. CONCLUSIONS

The proposed algorithm, Grid Sectoring is a simple, flexible for any topologies (uniform or non-uniform deployment), load balanced, energy-efficient concept for the cluster head election. The results of the average absolute SD of the number of MNs per cluster and the average absolute of load per sink even indicated that the proposed provides such the lowest values, it will be able to experience that the proposed can equally distribute the load balanced better than the others. Additionally, the result of the average distance between CHs and their MNs, the average hop count, and NRL, the proposed can overcome the others with the shortest values. In general, these metrics will directly impact to the cost function of routing and relate to the energy efficiency. The last two metrics, throughput and PDR, can convince that not only the proposed provides the load balanced, but also gives the performance quite high as well as it can prolong the network lifetime even in the large-scale network as well.

## ACKNOWLEDGMENT

The research reported here was supported by a research grant from the Telecommunications Research and Industrial Development Institute (TRIDI), the office of National Telecommunications Commissions (NTC).

## REFERENCES

- [1] L. Marbolda C., and N. Nasser, "Comparison of Clustering Algorithms for Wireless Sensor Networks," Electrical and Computer Engineering, 2006. CCECE '06. Canadian Conference, 2006, pp. 1787 - 1792, May 2006.
- [2] P. Kumarawada, D.J. Dechene, M. Luccini, and A. Seuer, "Algorithm for node clustering in Wireless Sensor Networks: A Survey," Information and Automation for Sustainability, 2008. ICIAPS 2008. 4th International Conference, 2008, pp. 295 - 300, December 2008.
- [3] W. Heinzelman, A. Chandrakasa, and H. Balakrishnan, "Energy Efficient Communication Protocols for Wireless Microsensor Networks," Proceedings of Hawaii International Conference on System Science, January 2000.
- [4] C. Nam, H. Jeong, and D. Shin, "The Adaptive Cluster Head Selection in Wireless Sensor Networks," IEEE International Workshop on Semantic Computing and Application, pp. 147-149, July 2008.
- [5] C. Nam, Y. Ka, J. Yoon, and D. Shin, "Cluster Head Selection for Equal Cluster Size in Wireless Sensor Networks," in Proceedings of IEEE. New Trends in Information and Service Science, 2009. NISS, pp. 618-623, July 2009.
- [6] E. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," IEEE Transactions on Wireless Communications, Vol. 1, No. 4, October 2002.
- [7] Y. Zoung, J. Pan, and G. Wu, "Energy-Optimal Grid-Based Clustering in Wireless Microsensor Networks," Distributed Computing Systems Workshops: ICDCS Workshop '09, 29th IEEE International Conference, June 2009.
- [8] W. Liu, Y. Sun, J. Yu, and B. Cao, "A Grid-based Distributed Multi-hop Routing Protocol for Wireless Sensor Networks," Embedded and Ubiquitous Computing 2008: EUC '08, IEEE/IFIP international conference, pp. 330-336, December 2008.
- [9] L. Li, X. Jiang, S. Zhong, and L. Hu, "Energy Balancing Clustering Algorithm for Wireless Sensor Network," Networks Security, Wireless Communications and Trusted Computing: NSWCTC '09, pp.61-64, April 2009.
- [10] M. Zhang, C. Gong, Y. Feng, and Y. Lu, "An Novel Energy-Efficient Dynamic Target Tracking Protocol for Wireless Sensor Networks," Networks Security, Wireless Communications and Trusted Computing: NSWCTC '09, pp.649-652, April 2009.
- [11] L. Xiong, J. Wen, W. Huang, and M. Yu, "The Research of Mobile Fusion-nodes Routes for Data Collection in Multi-tier Mobile Wireless Sensor Network," 2009 First International Workshop on Education Technology and Computer Science: ETCS '09, March 2009.



**2011 The 8th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON 2011)**

**May 17-19, 2011, Khon Kaen, Thailand**

**ECTI-CON 2011**  
 KHON KAEN UNIVERSITY

**8<sup>th</sup>** Electrical Engineering/ Electronics,  
 Computer, Telecommunications and  
 Information Technology (ECTI) Association,  
 Thailand - Conference 2011

Khon Kaen, Thailand  
 May 17-19, 2011  
 Pullman Khon Kaen Raja Orchid Hotel

**ECTI** Association  
**KHON KAEN UNIVERSITY**  
**IEEE** THAILAND SESSION

## Load Balanced and Energy Efficient Cluster Head Election in Wireless Sensor Networks

Anirooth Thonklin

Department of Computer Engineering  
Faculty of Engineering, Prince of Songkla University  
Hat Yai District, Songkhla Province, Thailand  
anirooth@gmail.com

W. Suntiarnortut

Department of Computer Engineering  
Faculty of Engineering, Prince of Songkla University  
Hat Yai District, Songkhla Province, Thailand  
wannarat@coe.psu.ac.th

**Abstract**—Even though clustering algorithms provide energy efficiency into large-scale Wireless Sensor Networks (WSNs), but uneven load balancing still occurs in general. Not only does this problem will not distribute energy consumption and balancing traffic loads equally within network, but this problem further makes the network cannot stay alive much longer due to short-lived cluster heads. The cluster heads which are responsible for most activities including inter-cluster and intra-cluster communications will highly consume energy more than the other types of nodes. Consequently, this study presents a cluster head election called Grid Sectoring base-on distribution of load balancing and energy consumption over both uniform and non-uniform deployment. The study partitions a squared-grid area into equal-sized clusters in order to equal number of member nodes. Along with the selection of cluster heads located nearest to a center of their clusters will equally distribute load balancing even more. Further, not only the results verified that the proposed algorithm could distribute load balancing with energy efficiency better than LEACH, ACHS, and EACHS, but the results also implied that the proposed could extend the network performance and the network lifetime even more.

**Keywords**- WSN; cluster head election; load balancing; energy efficiency; uniform and non-uniform deployment

### I. INTRODUCTION

Wireless sensor networks (WSNs) compose of a large number of sensor nodes based on low cost, restrictive function and limited energy. To balance the network energy and prolong the network lifetime, clustering is introduced into this network by partitioning this network into smaller clusters. There is a cluster head (CH) within each cluster responsible for most activities. E.g., intra cluster communications, data aggregation, and inter cluster communications. Hence, the cluster head election is such the important function which directly impacts to the network performance and energy efficiency.

A study's point of view is, to equally distribute the load balancing in every clusters over both uniform and non-uniform deployment. To achieve this goal, the study contributes a simple, load balanced, energy-efficient algorithm of cluster head election in WSNs by dividing the squared-grid area of interest into equal-sized clusters as cake cutting. Intuitively, if the most nodes in network are

uniformly deployed and all clusters are similar in size, there will be high possible that the number of member nodes (MNs) in each cluster can be more evenly distributed as well. To enhance more distribution of load balancing and to provide more energy efficiency, the study takes the location of CHs into account by choosing the CHs located nearest to center of their clusters and provides the proposed algorithm done by BS in order to save energy. Furthermore, with the advantages of the proposed algorithm, the number of hops to forward routing messages can simply be limited to save energy as well. In summary, the study's results can reasonably be examined that the proposed can outperform the other algorithms such as LEACH, ACHS, and EACHS in term of load balancing and energy efficiency as well as can prolong the network lifetime even more.

### II. RELATED WORKS

LEACH [1] has been a classical algorithm of cluster head election; to prolong the network lifetime by distributing the energy load evenly among the nodes in the network. The nodes each form themselves into clusters. In set-up phase, each node assigns itself a probability which is a number between 0 and 1. If this probability is less than a predefined threshold, that node then becomes a new CH.

ACHS [2] and EACHS [3] are proposed to enhance load balanced clusters; the number of MNs within each cluster is taken into account. If the numbers of MNs in every cluster are equally, there will be feasible that both the network load and the energy consumption will be probably equal highly. Initially, ACHS and EACHS start selection by the same way as LEACH algorithm, and they then re-select the new suitable CHs by determine a nearest CH and a farthest MN for every cluster. EACHS improves ACHS by re-selecting only the clusters which differ from the average number of MNs for 10 per cent in order to save time and energy.

Nevertheless, LEACH, ACHS and EACHS sometimes produce the unsuitable selected CHs which are located near edges of the cluster area, disappeared over some regions, and located closer to each other. This is due to the probabilistic random selection which cannot distribute selected CHs evenly over the area of interest. Thus, it makes such the network wastes energy inefficient and may resist distributing of load balancing and energy consumption with

efficiency. Furthermore, ACHS and EACHS algorithms normally use twice times a round of selection that will spend much time and energy even more.

The algorithms of grid clustering [4-7] are proposed to avoid complication of cluster formation; not only these algorithms benefit in the large-scale network, but these algorithms further improve the load balancing with a simple way. The area of interest is basically divided into minor grids (smaller squared-grid). Each minor grid is represented as a cluster area and holds the same size and the same number of MNs if all nodes in the network are uniformly deployed.

The main drawback of the grid clustering is, the area of interest can be divided into minor grids with limited number such as 1, 4, 9, 16, 25, ... or  $n^2$ . Along with the limited number, a clearly typical trouble is, if the network contains 100 nodes and the suitable number of CHs becomes 5, the network area will not be able to be divided into 5 minor grids certainly. Further, their functions have not been proven so far to perform with the non-uniform deployment.

### III. GRID SECTORING

#### A. The key Idea of Grid Sectoring

The study proposed the algorithm called Grid Sectoring (GS) which is used an idea that is, if all clusters are similar in number of MNs and similar in size, loads can be more evenly distributed within each cluster. The study assumes that the nodes in networks contain location-aware sensors like Global Positioning Sensor (GPS) to sufficiently archive the proposed algorithm. In the uniform deployment, to obtain the same size of all clusters, the area of interest should be divided into equal-size clusters, each cluster will also be able to hold the same number of MNs.

With the function of GS, the squared-grid area will be divided into equal-sized sectors, each sector is denoted as a cluster divided by the simple way similar to cake cutting. In cake cutting, everyone is also experienced that it is always possible to fairly divide a cake among  $n$  people. Further, GS produces CHs nearest to the center of their clusters to distribute load and energy dissipation efficiently. Along with the proposed, as the clusters are similar in dimension, the number of hops to forward overwhelming routing messages can also be limited in order to save energy as well.

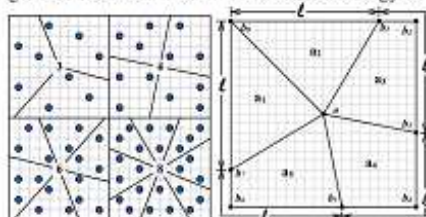


Figure 1. (a) Grid Sectoring for non-uniform deployment and (b) How Grid Sectoring divides the squared-grid into sectors.

In non-uniform deployment, grid clustering cannot evenly distribute the number of MNs within each minor grid suitably as mentioned above. The proposed also provides a better way by diving the area of interest into minor grids

firstly. Each minor grid may contain different number of MNs. The BS then continues dividing each minor grid into sectors depending on node density, the suitable number of cluster is 5 per cent of number of nodes [1]. The more number of MNs the minor grid contains, the more number of sectors that minor grid is divided. Finally, there will be more equally number of MNs within each cluster. A typical of GS for non-uniform deployment is illustrated in the Figure. 1(a).

#### B. The Solution of Grid Sectoring

Grid Sectoring can be used on the squared-grid area only, the network should also be divided into several minor grids as grid clustering firstly. The number of dividing is depending on node density.

1) *An Algorithm of Grid Sectoring*: Each divided minor grid will go on re-dividing into sectors with a suitable number of clusters, 5 per cent of nodes falling into each grid, by an equation being set as (1).

$$p/n = l \quad (1)$$

The Figure. 1(b) can clearly be illustrated how the GS evenly divides the grid area into equal-sized sectors. Where  $p$  is a perimeter of a grid,  $n$  is a required number of the sectors, and  $l$  is a length to divide minor grid into sectors. Thereby, every sector  $a_i$  will absolutely contain equal size.

2) *Centroid and Cluster Head Election*: To enhance cluster bandwidth, load balancing, and to reduce bottleneck problem, the selected CHs should be located nearest to a center of their clusters. With this way, the CHs will be surrounded with many more nodes to distribute routing paths. The proposed also applies centroid formulas of polygon [8] which is set in (2-4).

$$C_x = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (2)$$

$$C_y = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (3)$$

$$A = \frac{1}{2} \sum_{i=0}^{n-1} (x_i y_{i+1} - x_{i+1} y_i) \quad (4)$$

After the centroid position  $(C_x, C_y)$  has been obtained, a selected CHs will be located nearest to the centroid.

### IV. PERFORMANCE EVALUATION

#### A. Simulation Scenario

The network topology is organized into 2 sub-sections classified by node deployment; in short, there are uniform distribution and non-uniform deployment. The study simulated via NetworkX and NS-2. The study used about 5 per cent of nodes to be CHs [1] and simulated in different 20 topologies. The parameters were shown in Table. I.

TABLE I. THE ENVIRONMENT SET UP.

Parameters	Value
Simulation time	600 second (10 minutes)
Area of interest	100x100 m <sup>2</sup>
MAC protocol	IEEE 802.15.4



Parameters	Value
Transmission range	15 meters
Traffic type	CBR (constant bit rate)
Packet size	50 bytes (at Application Layer)
Traffic load (per node)	1 packet per second
Routing type	AODV (Ad hoc On-demand Distant Vector)

**B. Performance-Metrics and Results**

1) *Uniform Deployment:* The whole nodes in network were uniformly deployed. This scenario, the study measured network performance such as an average absolute SD of the number of MNs per cluster to measure balanced number of MNs in every cluster. The fewer number of SD each algorithm gives, the more number of MNs every clusters contains and the more load balancing the algorithm provides. Next, an average hop count which is described how long the MNs forward overwhelming routing data to their CHs was taken into account.

To measure the network performance, the study used throughput per sink and packet delivery ratio (PDR) to indicate the quality of service of the network. To be clearly illustrated whether every cluster is able to distribute load evenly or not, the study took an average absolute SD of load per sink into account as well. Next metric, normalized routing load (NRL) which is the ratio of the number of routing messages to the number of data messages is used to show the efficiency of routing protocol the algorithm provides. End-to-end delay (latency) and total energy consumption of network was used to evaluated the network performance accompanied with the other metrics. Finally, network lifetime which is determined by an average residual energy was taken into account as well. The study set the initial energy of all nodes in the network to be 1 joule.

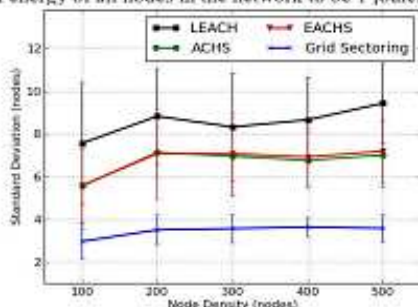


Figure 2. The average absolute SD of the number of MNs per cluster for uniform deployment.

The graph on Figure, 2, the proposed algorithm, GS gives the lowest degree of the average absolute SD with increasing steadily value from 2.96 to 3.50 between 100 and 200 nodes per area (100x100 m<sup>2</sup>) and then levelling off about 3.56 to 500 nodes. The proposed results of hop count which is shown on Figure, 3 gives such the lowest value when it was compared to the others.

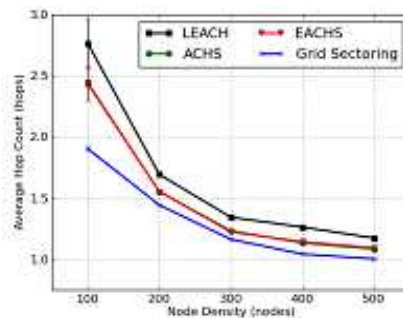


Figure 3. The average hop count for uniform deployment.

From the results about the average hop count with 100 nodes, the obtained average hop count about 2 hops is used to improve the traditional GS by reducing the number of hops to forward routing messages to be 2 and 4 times of obtained number of hops in order to save cluster energy. This improved GS algorithm is called Grid Sectoring-Hop limited Flooding with 4 hops (2x2 hops) and with 8 hops (4x2 hops) denoted as GS-HLF4 and GS-HLF8 respectively

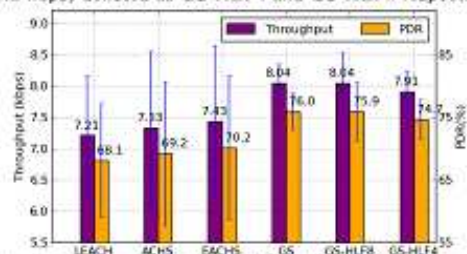


Figure 4. The throughput and PDR of 100 nodes for uniform deployment.

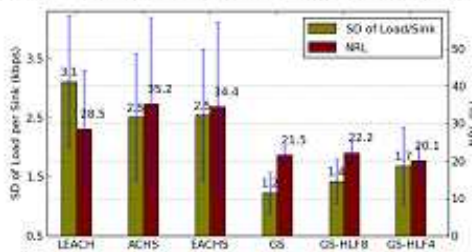


Figure 5. The average absolute standard deviation of load per sink and NRL of 100 nodes for uniform deployment.

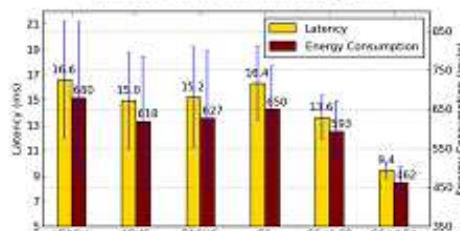


Figure 6. Latency and energy consumption of 100 nodes for uniform deployment.

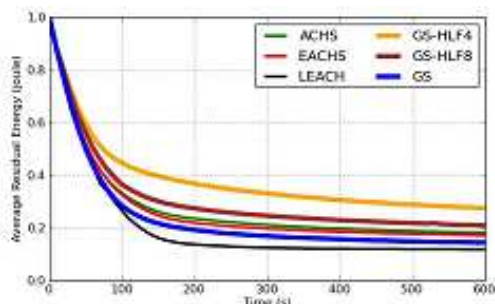


Figure 7. The average residual energy of 100 nodes for uniform deployment.

The results of throughput per sink and PDR on Figure. 4 projected that the proposed gives the highest values about 8 kbps per sink and around 75 per cent of PDR respectively. Additionally, the lowest values of the average absolute SD of load per sink and NRL which is shown on Figure. 5 is demonstrated that the proposed can distribute cluster loads more equally and the proposed can satisfyingly be performed together with AODV routing even more.

The results of the traditional Grid Sectoring in term of latency and energy consumption on Figure. 6 gives the similar values to LEACH's results but gives little poorer than ACHS' and EACHS' results. However, the results of the improved Grid Sectoring (GS-HLF8 and GS-HLF4) can be clearly illustrated that the proposed can reduce both latency and energy consumption as well.

The final results of the first experiment on Figure. 7 with uniform deployment is the network lifetime, the traditional Grid Sectoring outperforms than LEACH but ACHS and EACHS gives better results. By contrast, the results of GS-HLF8 and GS-HLF4 are shown that the average residual energy of all nodes gives better than the others.

2) *Non-Uniform Deployment*: There are 500 nodes scattered over the network area, the nodes would be fallen into four regions as seen in Figure. 8. The study divided each regions into sectors depending on node density. In short, there will be number of sectors or CHs about 5 per cent within each regions. The study evaluated the average absolute SD of MNs per cluster and the average hop count.

~60 nodes 3 CHs	~100 nodes 5 CHs
~120 nodes 6 CHs	~220 nodes 11 CHs

Figure 8. The density of nodes in each region deployed over the network area for non-uniform random deployment.

The results from the chart on Figure. 9 is illustrated that the proposed gives the lowest values for both average absolute standard deviation and average hop count. In particular, the average absolute SD of the proposed gives the results less than the others double values.

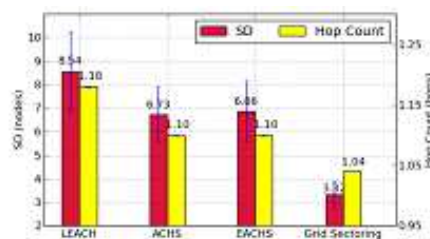


Figure 9. The average absolute standard deviation of the number of MNs per cluster of 500 nodes for non-uniform deployment.

## V. CONCLUSIONS

From this studying, The results can be convinced that the proposed algorithm, Grid Sectoring and the improved Grid Sectoring (GS-HLF) are the simple, flexible for any topologies (uniform or non-uniform deployment), load balanced, energy-efficient concept for the cluster head election to handle with large-scale WSNs. Furthermore, the results of the proposed show that the proposed not only be able to equally distribute the load balanced better than the others but further save much more energy as well as can achieve advance in the network lifetime.

## ACKNOWLEDGMENT

The research reported here was supported by a research grant from the Telecommunications Research and Industrial Development Institute (TRIDI).

## REFERENCES

- [1] W. Heinzelman, A. Chandrakasa, and H. Balakrishnan, "Energy Efficient Communication Protocols for Wireless Microsensor Networks," Proceedings of Hawaiian International Conference on Systems Science, January 2000.
- [2] C. Nam, H. Jeang, and D. Shin, "The Adaptive Cluster Head Selection in Wireless Sensor Networks," IEEE International Workshop on Semantic Computing and Application, pp. 147-149, July 2008.
- [3] C. Nam, Y. Ku, J. Yoon, and D. Shin, "Cluster Head Selection for Equal Cluster Size in Wireless Sensor Networks," in Proceedings of IEEE. New Trends in Information and Service Science, 2009. NISS, pp. 618-623, July 2009.
- [4] Y. Zuang, J. Pan, and G. Wu, "Energy-Optimal Grid-Based Clustering in Wireless Microsensor Networks," Distributed Computing Systems Workshops: ICDCS Workshop '09, 29th IEEE International Conference, June 2009.
- [5] W. Liu, Y. Sun, J. Yu, and B. Cao, "A Grid-based Distributed Multi-hop Routing Protocol for Wireless Sensor Networks," Embedded and Ubiquitous Computing 2008: EUC '08, IEEE/IFIP international conference, pp. 330-336, December 2008.
- [6] L. Li, X. Jiang, S. Zhong, and L. Hu, "Energy Balancing Clustering Algorithm for Wireless Sensor Network," Networks Security, Wireless Communications and Trusted Computing: NSWCTC '09, pp.61-64, April 2009.
- [7] M. Zhang, C. Gong, Y. Feng, and Y. Lu, "An Novel Energy-Efficient Dynamic Target Tracking Protocol for Wireless Sensor Networks," Networks Security, Wireless Communications and Trusted Computing: NSWCTC '09, pp.649-652, April 2009.
- [8] Paul Bourke, "Calculation Area and th Centroid of A Polygon," <http://paulbourke.net/geometry/polyarea/>

## ประวัติผู้เขียน

ชื่อ สกุล	นายอนิรุท ทองกลั่น	
รหัสประจำตัวนักศึกษา	5210120107	
วุฒิการศึกษา		
วุฒิ	ชื่อสถาบัน	ปีที่สำเร็จการศึกษา
วิศวกรรมศาสตรบัณฑิต (วิศวกรรมคอมพิวเตอร์)	มหาวิทยาลัยสงขลานครินทร์	2552

### ทุนการศึกษา (ที่ได้รับระหว่างการศึกษา)

1. ทุน NTC Scholarship สำนักงานคณะกรรมการกิจการโทรคมนาคมแห่งชาติ (สททช.)
2. ทุนอุดหนุนการดำเนินงานวิจัย คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์

### การตีพิมพ์เผยแพร่ผลงาน

1. Anirooth Thongklin and W. Suntiarnorntut, "Grid Sectoring: A Cluster Head Election for Load Balancing over Wireless Sensor Networks," In Proceedings of ICSAP 2011 the 3rd International Conference on Signal Acquisition and Processing, IEEE, Singapore, February 2011, pp. 86 - 91.
2. Anirooth Thongklin and W. Suntiarnorntut, "A Load Balanced Cluster Head Election for Uniform and Non-uniform Deployment over Wireless Sensor Networks," In Proceedings of ICCCD 2011 the International Conference on Computer and Communication Devices, IEEE, Bali Island, Indonesia, April 2011, pp. 99 - 103.
3. Anirooth Thongklin and W. Suntiarnorntut, "Load Balanced and Energy Efficient Cluster Head Election in Wireless Sensor Networks," In Proceedings of ECTI-CON 2011 the 8th Electrical Engineering/Electronics, Computer, Telecommunication and Information Technology Conference, IEEE, Khon-Kaen, Thailand, May 2011.