

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวโดยสังเขปถึงทฤษฎีและงานวิจัยที่เกี่ยวข้องกับโปรโตคอลค้นหาเส้นทางที่เหมาะสมสำหรับการสื่อสารระหว่างยานพาหนะ โดยในหัวข้อที่ 2.1 จะกล่าวถึงโปรโตคอลค้นหาเส้นทาง (routing protocols) สำหรับเครือข่ายเคลื่อนที่เฉพาะกิจ (Mobile Ad hoc Network: MANET) ถัดไปในหัวข้อที่ 2.2 จะกล่าวถึงโปรโตคอลค้นหาเส้นทางสำหรับเครือข่ายยานพาหนะ (Vehicular Ad hoc Network: VANET) ถัดไปในหัวข้อที่ 2.3 จะกล่าวถึงการประยุกต์ใช้โปรโตคอลค้นหาเส้นทางสำหรับเครือข่ายยานพาหนะในชีวิตประจำวัน ถัดไปในหัวข้อที่ 2.4 จะกล่าวถึงตัวแบบหรือแบบจำลองทางคณิตศาสตร์ที่งานวิจัยชิ้นนี้ใช้อ้างอิงในการตัดสินใจเลือกเส้นทางที่เหมาะสมสำหรับจัดส่งข้อมูลไปยังโหนดปลายทางภายในเครือข่าย และท้ายที่สุดในหัวข้อที่ 2.5 จะกล่าวถึงตัวแบบการเคลื่อนที่

2.1 โปรโตคอลค้นหาเส้นทางสำหรับเครือข่ายเคลื่อนที่เฉพาะกิจ (Mobile Ad hoc Network Routing Protocols)

เครือข่ายเคลื่อนที่เฉพาะกิจ (Mobile Ad hoc Network: MANET) เป็นเครือข่ายของอุปกรณ์คำนวณแบบเคลื่อนที่ได้ ที่เชื่อมต่อเพื่อรับส่งข้อมูลระหว่างกันโดยไม่อาศัยสถานีฐานไม่มีจุดศูนย์กลางในการให้บริการเชื่อมต่อไม่มีการกำหนดเส้นทางที่ใช้เชื่อมต่อกันในเครือข่าย ทุก ๆ โหนดสามารถเคลื่อนที่และเชื่อมต่อกับโหนดใด ๆ ที่อยู่ในพื้นที่ครอบคลุม (coverage area) ของตนเองได้อย่างอิสระ ส่งผลให้ทอพอโลยี (topology) หรือ รูปแบบการต่อเชื่อมของโหนดภายในเครือข่ายสามารถเปลี่ยนแปลงได้ตลอดเวลา อัตราความเร็วในการเปลี่ยนแปลงรูปแบบการต่อเชื่อมขึ้นกับความเร็วและทิศทางในการเคลื่อนที่ของโหนดต่าง ๆ ภายในเครือข่าย

โปรโตคอลค้นหาเส้นทางที่ถูกพัฒนาสำหรับเครือข่ายเคลื่อนที่เฉพาะกิจแบ่งออกได้เป็น 2 กลุ่มใหญ่ (Royer & Chai-Keong, 1999) คือ กลุ่มของโปรโตคอลที่ทำงานแบบขับเคลื่อนด้วยตาราง (table-driven) (Charles E. Perkins & Bhagwat, 1994) (Ching-Chuan & Gerla, 1997) (Murthy & Garcia-Luna-Aceves, 1996) และกลุ่มของโปรโตคอลที่ทำงานเมื่อถูกทวงถาม (on-demand) (C. E. Perkins & Royer, 1999) (Johnson & Maltz, 1996) (Park & Corson, 1997) (Chai-Keong, 1996) (Dube, Rais, Kuang-Yeh, & Tripathi, 1997)

2.1.1 กลุ่มของโปรโตคอลที่ทำงานแบบขับเคลื่อนด้วยตาราง (Table-Driven Routing Protocols)

กลุ่มของโปรโตคอลที่ทำงานแบบขับเคลื่อนด้วยตารางมีลักษณะร่วมที่สำคัญคือ เป็นโปรโตคอลที่ทำงานแบบโปรแอกทีฟ (proactive) กล่าวคือ โหนดแต่ละโหนดในเครือข่ายจัดเตรียมตารางซึ่งบรรจุข้อมูลเส้นทางที่เหมาะสมที่สุดสำหรับการจัดส่งข้อมูลไปยังโหนดปลายทางใด ๆ ภายในเครือข่ายไว้ล่วงหน้า และมีการปรับเปลี่ยนข้อมูลในตารางให้ทันต่อเหตุการณ์อยู่เสมอตามช่วงเวลาที่กำหนด (period) ข้อดีของโปรโตคอลในกลุ่มนี้คือ เนื่องจากมีการเตรียมข้อมูลเส้นทางไว้ล่วงหน้าแล้ว เวลาที่ต้องเสียไปสำหรับประมวลผลที่แต่ละโหนดจึงมีค่อนข้างน้อย ซึ่งก็จะส่งผลต่อเนื่องให้ค่าการหน่วงตลอดเส้นทาง (end-to-end delay) มีค่าน้อยตามไปด้วย อย่างไรก็ตาม โปรโตคอลในกลุ่มนี้มีข้อเสียที่สำคัญหลายประการ ได้แก่ การสิ้นเปลืองพื้นที่สำหรับจัดเก็บตารางข้อมูลเส้นทางที่ทุก ๆ โหนดในเครือข่าย โดยขนาดของตารางจะแปรผันตรงกับจำนวนโหนดในเครือข่าย นอกจากนี้ การอัปเดตข้อมูลเส้นทางในตารางตามช่วงเวลาที่กำหนดก็จัดเป็นโอเวอร์เฮด (overhead) หรือค่าใช้จ่ายสำหรับโปรโตคอลในกลุ่มนี้อีกด้วย

ในหัวข้อนี้จะกล่าวถึงตัวอย่างของโปรโตคอลที่ทำงานแบบขับเคลื่อนด้วยตาราง ได้แก่ โปรโตคอลค้นหาเส้นทาง Destination-Sequenced Distance-Vector (DSDV) (Charles E. Perkins & Bhagwat, 1994) โปรโตคอลค้นหาเส้นทาง Clusterhead Gateway Switch Routing (CGSR) (Ching-Chuan & Gerla, 1997) และโปรโตคอลค้นหาเส้นทาง Wireless Routing Protocol (WRP) (Murthy & Garcia-Luna-Aceves, 1996)

2.1.1.1 โปรโตคอลค้นหาเส้นทาง Destination-Sequenced Distance-Vector (DSDV)

โปรโตคอลค้นหาเส้นทาง Destination-Sequenced Distance-Vector (DSDV) (Charles E. Perkins & Bhagwat, 1994) มีพื้นฐานจากอัลกอริทึมสำหรับค้นหาเส้นทางแบบกระจายของ Bellman-Ford¹ (Arslan, Chen, & Di Benedetto, 2006) ที่รู้จักกันอย่างแพร่หลาย โดยมีการเพิ่มเติมส่วนปรับปรุงเพื่อป้องกันไม่ให้มีลูปในเส้นทางสำหรับจัดส่งข้อมูลไปยังโหนดปลายทางภายในเครือข่าย

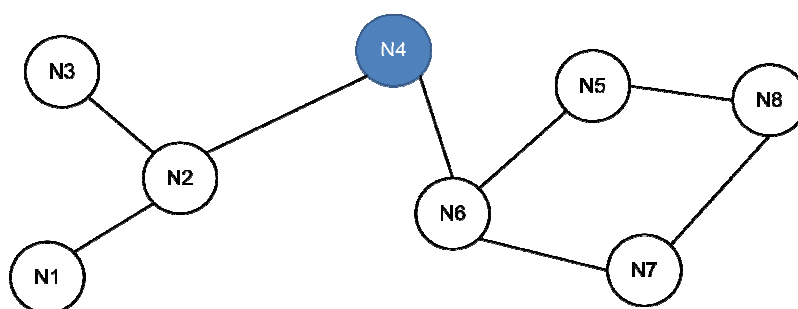
¹ Distributed Bellman-Ford (DBF) algorithm

โปรโตคอลค้นหาเส้นทาง DSDV กำหนดให้แต่ละโหนดภายในเครือข่ายจะต้องจัดสร้างตารางส่งต่อ (forwarding table) ซึ่งภายในบรรจุข้อมูลโหนดถัดไป (next hop) สำหรับส่งต่อแพ็กเก็ตไปยังโหนดปลายทางใด ๆ ภายในเครือข่าย และข้อมูลประกอบอื่น ๆ ตารางที่ 2.1 แสดงตัวอย่างตารางส่งต่อที่จัดสร้างโดยโหนด N4 ภายในเครือข่ายดังแสดงในภาพที่ 2.1

ตารางที่ 2.1
ตารางส่งต่อสร้างโดยโหนด N4

Destination	Next Hop	Metric	Sequence number
N1	N2	2	S406_N1
N2	N2	1	S128_N2
N3	N2	2	S564_N3
N4	N4	0	S710_N4
N5	N6	2	S392_N5
N6	N6	1	S076_N6
N7	N6	2	S128_N7
N8	N6	3	S050_N8

ภาพที่ 2.1
เครือข่ายตัวอย่าง DSDV



เมื่อถึงเวลาที่กำหนดหรือเมื่อมีการเปลี่ยนแปลงข้อมูลภายในตารางส่งต่อ แต่ละโหนดภายในเครือข่ายจะทำการแพร่สัญญาณ หรือบรอดแคสต์ (broadcast) ตารางส่งต่อไปยังโหนดเพื่อนบ้าน การบรอดแคสต์ตารางส่งต่อแบ่งออกเป็น 2 แบบด้วยกันคือ ส่งต่อแบบทั้งตาราง (full dump) การส่งตารางส่งต่อแบบนี้ทำเมื่อถึงเวลาที่กำหนดไว้ และแบบที่สองส่งเฉพาะส่วนที่มีการเปลี่ยนแปลงเท่านั้น ตารางที่ 2.2 แสดงตัวอย่างของตารางส่งต่อที่โหนด N4 จะทำการบรอดแคสต์ไปยังโหนดเพื่อนบ้าน

ตารางที่ 2.2

ตารางส่งต่อที่ N4 จะทำการบรอดแคสต์ออกไปยังโหนดเพื่อนบ้าน

Destination	Metric	Sequence number
N1	2	S406_N1
N2	1	S128_N2
N3	2	S564_N3
N4	0	S710_N4
N5	2	S392_N5
N6	1	S076_N6
N7	2	S128_N7
N8	3	S050_N8

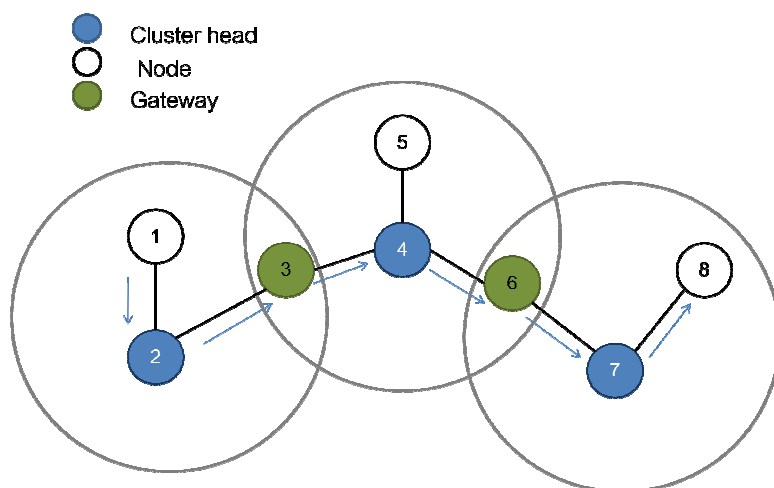
โปรโตคอล DSDV มีข้อดีดังต่อไปนี้ ไม่เกิดลูปในการค้นหาเส้นทาง และเวลาที่ต้องเสียไปสำหรับการประมวลผลที่แต่ละโหนดใช้เวลาน้อยเนื่องจากได้มีการเตรียมข้อมูลเส้นทางไว้ล่วงหน้าแล้ว แต่ยังมีข้อเสีย คือ สิ้นเปลืองพื้นที่สำหรับจัดเก็บตารางข้อมูลเส้นทางที่ทุก ๆ โหนดในเครือข่าย โดยขนาดของตารางจะแปรผันตรงกับจำนวนโหนดในเครือข่าย ไม่เหมาะกับเครือข่ายที่มีการเคลื่อนที่อยู่ตลอดเวลาเพราะต้องทำการปรับปรุงตารางส่งต่ออยู่บ่อยครั้ง

2.1.1.2 โปรโตคอลค้นหาเส้นทาง Clusterhead Gateway Switch Routing (CGSR)

โปรโตคอลค้นหาเส้นทาง Clusterhead Gateway Switch Routing (CGSR) (Ching-Chuan & Gerla, 1997) มีการจัดเรียงโหนดต่างๆ ในเครือข่ายออกเป็นลำดับชั้นภาพที่ 2.2 แสดงเครือข่ายตัวอย่าง CGSR ซึ่งแบ่งออกเป็น 3 ส่วน ดังนี้

- คลัสเตอร์เฮด (cluster head) ทำหน้าที่ในการเลือกเกตเวย์ให้กับโหนดที่เป็นสมาชิกในกลุ่มของตัวเอง
- โหนด (node) เป็นสมาชิกที่อยู่ภายในพื้นที่ครอบคลุมการส่งสัญญาณของคลัสเตอร์เฮด (cluster head)
- เกตเวย์โหนด (gateway node) ทำหน้าที่เป็นโหนดที่ส่งต่อข้อมูลไปให้ cluster head อีกกลุ่มหนึ่ง

ภาพที่ 2.2
เครือข่ายตัวอย่าง CGSR



ที่มา : “A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks,”
โดย Royer, E. M., & Chai-Keong, T. (1999). *Personal Communications, IEEE*,
6(2), 46-55.

โปรโตคอลค้นหาเส้นทาง CGSR กำหนดให้ทุกโหนดภายในเครือข่ายทำการเลือกคลัสเตอร์เฮดของตนเองจำนวน 1 โหนด ยกตัวอย่างเช่น จากภาพที่ 2.2 โหนด 1 เลือกโหนด 2 เป็นคลัสเตอร์เฮด โหนด 8 เลือกโหนด 7 เป็นคลัสเตอร์เฮด เป็นต้น นอกจากนี้ แต่ละโหนดภายในเครือข่ายจะต้องจัดสร้างตารางข้อมูลอีกจำนวน 2 ตาราง ได้แก่ ตารางสมาชิกคลัสเตอร์ (cluster member table) ซึ่งจัดเก็บข้อมูลคลัสเตอร์เฮดของโหนดปลายทางแต่ละโหนดในเครือข่ายไว้ (ตารางที่ 2.3 แสดงตารางสมาชิกคลัสเตอร์ของโหนด 1 ที่บรรจุข้อมูลคลัสเตอร์เฮดของโหนดปลายทางหมายเลข 8 ไว้) และตารางส่งต่อซึ่งจัดเก็บข้อมูลโหนดถัดไป (next hop) ที่ต้องส่งต่อแพ็คเก็ตข้อมูลไปสำหรับแต่ละคลัสเตอร์ปลายทางเอาไว้ (ตารางที่ 2.4 แสดงตารางส่งต่อข้อมูลของโหนด 1)

เมื่อโหนดใด ๆ ในเครือข่ายต้องการจัดส่งแพ็คเก็ตข้อมูลให้ส่งต่อแพ็คเก็ตไปให้กับคลัสเตอร์เฮดของกลุ่มตน และให้คลัสเตอร์เฮดเป็นคนส่งต่อข้อมูลไปยังเกตเวย์ที่เหมาะสมต่อไป ตารางที่ 2.4 แสดงข้อมูลว่าเมื่อโหนด 1 ต้องการส่งข้อมูลไปยังโหนด 8 ซึ่งมีคลัสเตอร์เฮดคือ โหนด 7 จะต้องส่งต่อข้อมูลไปยังโหนด 2 ซึ่งก็คือ คลัสเตอร์เฮดของโหนด 1 นั่นเอง

ตารางที่ 2.3

ตารางสมาชิกคลัสเตอร์ของโหนด 1

Destination	Destination clusterhead	Sequence number
8	7	20

ตารางที่ 2.4

ตารางส่งต่อข้อมูลของโหนด 1

Destination Cluster	Next hop	Metric	Sequence number
7	2	7	15

เมื่อถึงช่วงเวลาที่กำหนดโหนดแต่ละโหนดภายในเครือข่ายจะทำการบรอดแคสต์ (broadcast) ตารางสมาชิกคลัสเตอร์ให้กับโหนดเพื่อนบ้านภายในเครือข่าย และหลังจากนั้นทำการปรับปรุงตารางส่งต่อเพื่อให้ได้คลัสเตอร์เฮดที่ใกล้โหนดปลายทางมากที่สุด

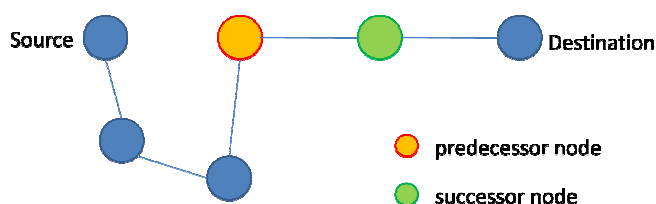
ข้อดีของโปรโตคอล CGSR คือ ลดพื้นที่สำหรับการจัดเก็บตารางส่งต่อข้อมูลในแต่ละโหนดลง (1 ระเบียบต่อ 1 คลัสเตอร์เฮด) ในส่วนข้อเสียของโปรโตคอล CGSR ได้แก่ คลัสเตอร์เฮดและเกตเวย์โหนดอาจทำให้เกิดภาวะคอขวดขึ้นได้เมื่อโหนดมีความต้องการส่งข้อมูลไปให้โหนดปลายทางพร้อม ๆ กัน และเมื่อโหนดมีการเคลื่อนที่อยู่ตลอดเวลาทำให้ต้องเลือกคลัสเตอร์เฮดอยู่บ่อย ๆ ดังนั้นจึงเป็นสาเหตุทำให้เกิดโอเวอร์เฮดที่สูงตามมา

2.1.1.3 โปรโตคอลค้นหาเส้นทาง Wireless Routing Protocol (WRP)

โปรโตคอลค้นหาเส้นทาง Wireless Routing Protocol (WRP) (Murthy & Garcia-Luna-Aceves, 1996) เป็นโปรโตคอลที่ทำงานบนพื้นฐานอัลกอริทึมเวกเตอร์ระยะทาง (distance-vector) ซึ่งกำหนดให้โหนดในเครือข่ายทำการสร้างตารางข้อมูลต่าง ๆ ไว้ 4 ตารางด้วยกัน คือ ตารางระยะทาง (distance table) เก็บข้อมูลระยะทางจากโหนดเพื่อนบ้านของโหนดต้นทางไปยังโหนดปลายทางที่ต้องการ ตารางเส้นทาง (routing table) เก็บข้อมูลระยะทางจากโหนดต้นทาง (source node) ไปยังโหนดปลายทาง (destination node) โหนดบัพกอน (predecessor node) และโหนดบัพหลัง (successor node) (ภาพที่ 2.3) ตารางค่าใช้จ่ายของลิงค์ (link-cost table) ประกอบไปด้วย ค่าใช้จ่ายของลิงค์ที่เชื่อมโยงไปยังโหนดเพื่อนบ้าน และตารางเมสเสจรีทราสมิสชันลิสต์ (message retransmission list table: MRL) เก็บข้อมูลโหนดเพื่อนบ้านซึ่งยังไม่ได้ตอบกลับมาภายในเวลาที่กำหนด

ภาพที่ 2.3

แสดงโหนดในเครือข่ายที่ทำหน้าที่เป็นโหนดบัพกอน และโหนดบัพหลัง



เมื่อถึงเวลาที่กำหนด หรือเมื่อค่าใช้จ่ายของลิงค์มีการเปลี่ยนแปลง ทุกโหนดจะทำการแลกเปลี่ยนตารางเส้นทาง (routing table) กับโหนดเพื่อนบ้าน หลังจากนั้นทำการปรับปรุงข้อมูลในตารางเส้นทาง (routing table) และตารางระยะทาง (distance table) หากข้อมูลที่ได้รับมาใหม่แสดงเส้นทางที่ดีกว่า

ข้อดีของโปรโตคอล WRP คือ ไม่เกิดลูปในเส้นทางที่ทำการจัดส่งข้อมูล และเมื่อเส้นทางเชื่อมต่อขาดสามารถเชื่อมต่อเส้นทางใหม่ได้รวดเร็ว ข้อเสียของโปรโตคอลนี้ก็คือ สิ้นเปลืองพื้นที่สำหรับจัดเก็บตารางข้อมูลจำนวน 4 ตารางที่ทุก ๆ โหนดในเครือข่าย และสิ้นเปลืองแบนด์วิดท์ที่ใช้ในการแลกเปลี่ยนตารางข้อมูล

2.1.2 กลุ่มของโปรโตคอลที่ทำงานเมื่อถูกทวงถาม (On-Demand Routing Protocols)

กลุ่มของโปรโตคอลที่ทำงานเมื่อถูกทวงถามมีลักษณะร่วมที่สำคัญคือ จะทำการค้นหาเส้นทางไปยังโหนดปลายทางใด ๆ เมื่อโหนดต้นทางต้องการจัดส่งข้อมูลไปยังโหนดปลายทางเท่านั้น ไม่มีการจัดเตรียมตารางเส้นทางที่เหมาะสมที่สุดสำหรับการจัดส่งข้อมูลไปยังโหนดปลายทางใด ๆ ภายในเครือข่ายไว้ก่อนล่วงหน้า ข้อดีของโปรโตคอลในกลุ่มนี้ คือ โอเวอร์เฮด (overhead) หรือค่าใช้จ่ายในการค้นหาเส้นทางลดลงเมื่อเปรียบเทียบกับกลุ่มโปรโตคอลที่ทำงานแบบขับเคลื่อนด้วยตาราง อย่างไรก็ตาม โปรโตคอลในกลุ่มนี้มีข้อเสีย คือ เสียเวลาในการค้นหาเส้นทางถ้าโหนดปลายทางไม่เคยถูกทวงถามมาก่อน

ในหัวข้อนี้จะกล่าวถึงตัวอย่างของโปรโตคอลที่ทำงานแบบเมื่อถูกทวงถาม ได้แก่ โปรโตคอลค้นหาเส้นทาง Ad hoc On-demand Distance Vector Routing (AODV) (C. E. Perkins & Royer, 1999) โปรโตคอลค้นหาเส้นทาง Dynamic Source Routing (DSR) (Johnson & Maltz, 1996) โปรโตคอลค้นหาเส้นทาง Temporally-Ordered Routing algorithm (TORA) (Park & Corson, 1997) โปรโตคอลค้นหาเส้นทาง Associativity-Based Routing (ABR) (Chai-Keong, 1996) และโปรโตคอลค้นหาเส้นทาง Signal Stability-based Adaptive Routing (SSR) (Dube, et al., 1997)

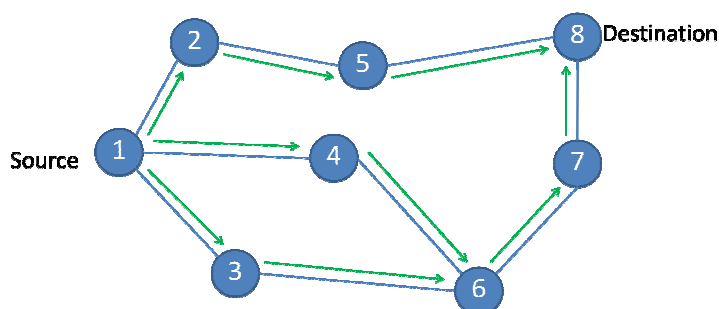
2.1.2.1 โปรโตคอลค้นหาเส้นทาง Ad hoc On-demand Distance Vector Routing (AODV)

โปรโตคอลค้นหาเส้นทาง Ad hoc On-demand Distance Vector Routing (AODV) (C. E. Perkins & Royer, 1999) เป็นโปรโตคอลที่ได้มีการเพิ่มเติมส่วนปรับปรุงมาจากโปรโตคอล DSDV โดยจะทำงานก็ต่อเมื่อโหนดต้นทางมีข้อมูลที่จะจัดส่งไปให้โหนดปลายทางที่ต้องการเท่านั้น

เมื่อโหนดต้นทางมีข้อมูลที่ต้องการจัดส่งไปให้กับโหนดปลายทาง โปรโตคอล AODV กำหนดให้โหนดดังกล่าวทำการบรอดแคสต์แพ็คเก็ตร้องขอเส้นทาง (route request packet: RREQ) ให้กับโหนดเพื่อนบ้าน โหนดเพื่อนบ้านก็จะทำการบรอดแคสต์แพ็คเก็ตร้องขอเส้นทางไปยังโหนดเพื่อนบ้านของตนเองต่อไปจนกระทั่งแพ็คเก็ตร้องขอเส้นทางเดินทางไปถึงโหนดปลายทาง หรือโหนดที่ทราบเส้นทางที่จะไปยังโหนดปลายทาง ดังภาพที่ 2.4

ภาพที่ 2.4

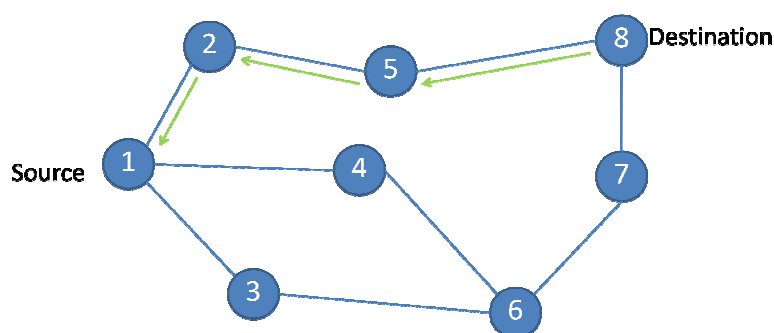
แสดงเส้นทางในการส่งแพ็คเก็ตร้องขอเส้นทางจากโหนดต้นทาง 1 ไปยังโหนดปลายทาง 8



หลังจากนั้นโหนดปลายทางหรือโหนดที่ทราบเส้นทางไปยังโหนดปลายทางทำการส่งแพ็คเก็ตตอบกลับ (route reply packet) ไปให้โหนดต้นทางโดยใช้เส้นทางเดียวกับที่แพ็คเก็ตร้องขอใช้ในการเดินทางมาดังภาพที่ 2.5 และระหว่างทางที่แพ็คเก็ตตอบกลับเดินทางผ่านโหนดต่าง ๆ ให้โหนดดังกล่าวทำการจัดเก็บข้อมูลเกี่ยวกับเส้นทางที่ค้นพบลงในตารางเส้นทาง (route table) ส่วนของการจัดเก็บเส้นทางตามโปรโตคอล AODV ให้โหนดต้นทางจัดเก็บเส้นทางเพียง 1 เส้นทางสำหรับเดินทางไปยังโหนดปลายทางแต่ละโหนด

ภาพที่ 2.5

แสดงเส้นทางที่แพ็คเกิดตอบกลับ (route reply packet) ใช้ในการเดินทางจากโหนด 8 ไปยังโหนดต้นทาง 1



ข้อดีของโปรโตคอล AODV คือ ไม่เกิดลูปในเส้นทางที่จะจัดส่งข้อมูลไปให้โหนดปลายทางที่ต้องการ และใช้แบนด์วิดท์ได้อย่างมีประสิทธิภาพ ข้อเสียของโปรโตคอล AODV คือ ไม่มีการทำแคชเก็บเส้นทางอื่นที่สามารถไปถึงโหนดปลายทางได้ทำให้ต้องประมวลผลใหม่ทุกครั้งเมื่อเส้นทางเดิมขาด

2.1.2.2 โปรโตคอลค้นหาเส้นทาง Dynamic Source Routing (DSR)

โปรโตคอลค้นหาเส้นทาง Dynamic Source Routing (DSR) (Johnson & Maltz, 1996) ประกอบไปด้วยการทำงานหลัก 2 ส่วนคือ ส่วนการค้นหาเส้นทาง (route discovery) ใช้ในการค้นหาเส้นทางเมื่อโหนดต้นทางไม่ทราบเส้นทางที่จะไปยังโหนดปลายทาง และการบำรุงรักษาเส้นทาง (route maintenance) ใช้เมื่อโหนดต้นทางพบว่าเส้นทางที่ใช้จัดส่งข้อมูลเกิดการเปลี่ยนแปลง

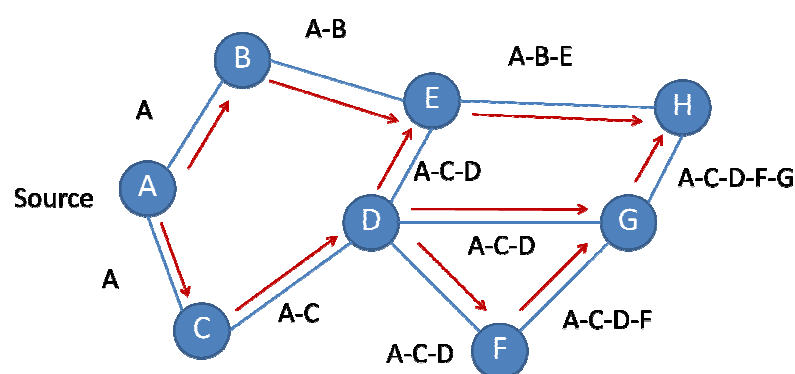
เมื่อโหนดต้นทางต้องการจัดส่งข้อมูลไปให้กับโหนดปลายทางที่ต้องการ โปรโตคอล DSR กำหนดให้โหนดดังกล่าวบรรจุแคสท์แพ็คเก็ตค้นหาเส้นทาง (route request packet) โหนดที่ได้รับแพ็คเก็ตค้นหาเส้นทางให้ทำการเพิ่มเลขที่อยู่ (address) ของตัวเองลงในแพ็คเก็ตก่อนจะส่งต่อ จนแพ็คเก็ตค้นหาเส้นทางไปถึงโหนดปลายทาง (ภาพที่ 2.6) หลังจากที่ได้รับแพ็คเก็ตปลายทางได้รับแพ็คเก็ตร้องขอแล้ว ให้โหนดปลายทางส่งแพ็คเก็ตตอบกลับ (route reply packet) ไปยัง

โหนดต้นทาง (ภาพที่ 2.7) และให้โหนดต้นทางทำการจัดเก็บเส้นทางที่ได้ลงในแคชเพื่อใช้ในการจัดส่งข้อมูล

โปรโตคอล DSR ต่างจากโปรโตคอล AODV กล่าวคือ โหนดต้นทางมีเส้นทางที่ไปยังโหนดปลายทางที่ต้องการได้หลายเส้นทาง เมื่อเส้นทางหนึ่งเส้นทางใดขาดก็สามารถเลือกใช้เส้นทางอื่นได้ทันที

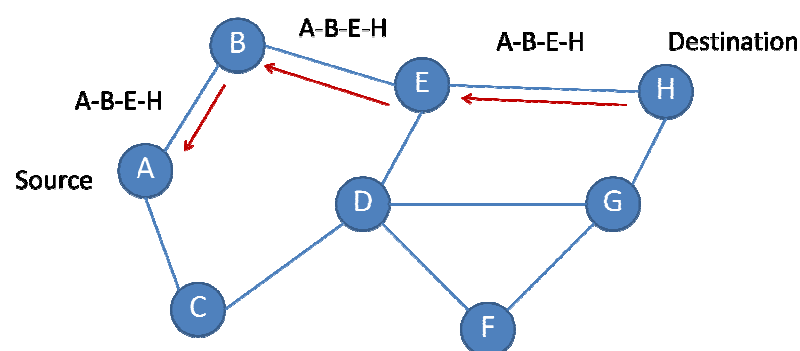
ภาพที่ 2.6

แสดงเส้นทางที่แพ็คเก็ตค้นหาเส้นทางเดินทางจากโหนดต้นทาง A ไปยังโหนดปลายทาง H



ภาพที่ 2.7

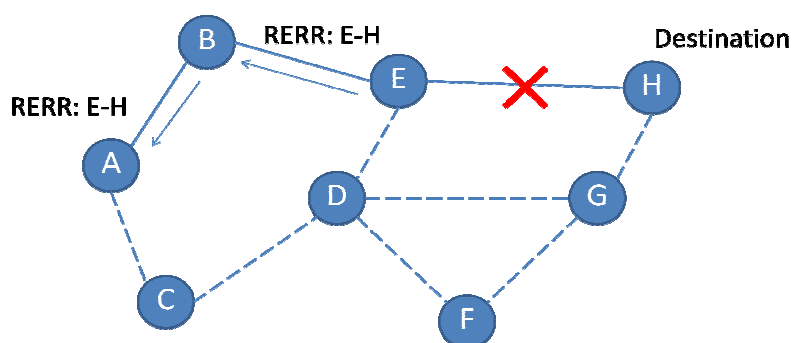
แสดงเส้นทางที่แพ็คเก็ตตอบกลับเดินทางจากโหนดปลายทาง H ไปยังโหนดต้นทาง A



เมื่อเส้นทางที่ใช้ในการจัดส่งข้อมูลขาด โหนดจะเข้าสู่กระบวนการบำรุงรักษาเส้นทาง (route maintenance) โดยให้โหนดที่ทราบเส้นทางระหว่างโหนดที่ใช้ในการจัดส่งข้อมูลทำการส่งแพ็คเกจแจ้งข้อมูลผิดพลาดของเส้นทาง (route error) เพื่อแจ้งให้โหนดต้นทางทราบและทำการลบโหนดดังกล่าวออกจากแคชและใช้เส้นทางอื่นที่มีในแคชแทนเส้นทางเก่า

ภาพที่ 2.8

โหนด E ส่ง route error ไปให้โหนด A หลังจากได้ถึงกระหวาโหนด E-H ขาด



ข้อดีของโปรโตคอลค้นหาเส้นทาง DSR คือ มีแคชเอาไว้เก็บข้อมูลเส้นทางไปยังโหนดปลายทางที่ต้องการหลายเส้นทาง เมื่อเส้นทางหนึ่งขาดก็สามารถใช้เส้นทางอื่นที่มีอยู่ในแคชได้ทันทีไม่ต้องเสียเวลาในการประมวลผลใหม่ และลดการใช้แบนด์วิดท์ ข้อเสีย คือ เมื่อจำนวนโหนดในเครือข่ายมีจำนวนมากขึ้นอาจทำให้เกิดพาร์ทิชัน (partition) ในกรณีที่โหนดใด ๆ เคลื่อนที่ออกไปทำให้ไม่สามารถจัดส่งข้อมูลไปให้กับโหนดที่เคยเชื่อมต่อได้เพราะไม่มีเส้นทางอื่นที่สามารถเข้าถึงได้

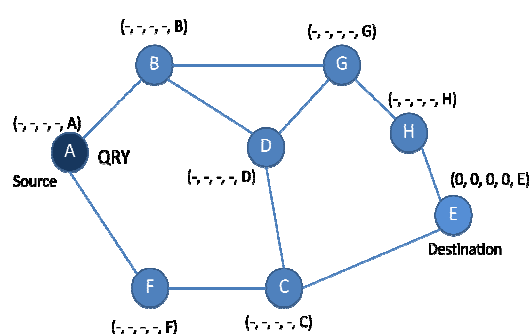
2.1.2.3 โปรโตคอลค้นหาเส้นทาง Temporally-Ordered Routing Algorithm (TORA)

โปรโตคอลค้นหาเส้นทาง Temporally-Ordered Routing algorithm (TORA) (Park & Corson, 1997) เริ่มต้นการทำงานเมื่อโหนดต้นทางมีแพ็คเกจที่ต้องการจะส่งไปยังโหนดปลายทางให้ฟลัด (flood) แพ็คเกจทดสอบถาม (query packet: QRY) ให้ทั่วทั้งเครือข่ายจนกระทั่งโหนดปลายทางหรือโหนดที่ทราบเส้นทางไปยังโหนดปลายทางได้รับแพ็คเกจทดสอบถาม จากนั้นให้บรอดแคสต์อัปเดตแพ็คเกจ (update packet: UPD) กลับไปให้โหนดต้นทางและกำหนดค่า

ความสูง (height) ของโหนดปลายทางเท่ากับ 0 ส่วนโหนดอื่น ๆ ในเครือข่ายที่ไม่ใช่โหนดต้นทาง เมื่อได้รับอัปเดตแพ็คเก็ตนี้ให้ทำการเพิ่มค่าความสูงทีละ 1 เมื่อโหนดต้นทางได้รับอัปเดตแพ็คเก็ตแล้วจะทำการเลือกเส้นทางที่สั้นที่สุด (ในที่นี้จะหมายถึงเส้นทางที่มีค่าความสูงต่ำที่สุด) ภาพที่ 2.9 - ภาพที่ 2.14 แสดงขั้นตอนในกระบวนการสร้างเส้นทางสำหรับจัดส่งแพ็คเก็ตข้อมูล

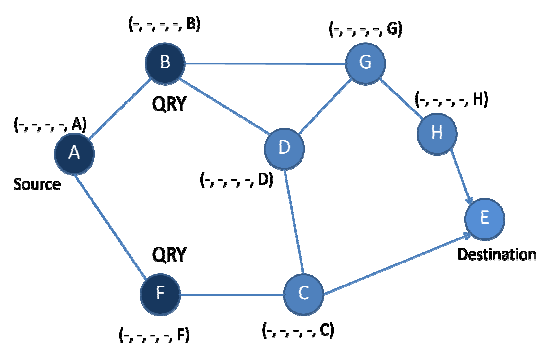
ภาพที่ 2.9

เครือข่ายตัวอย่าง เมื่อโหนดต้นทาง A ทำการส่ง QRY packet ออกไปในเครือข่าย



ภาพที่ 2.10

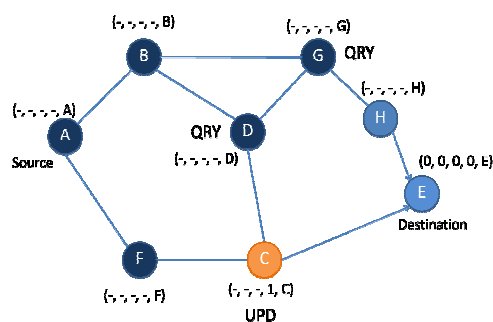
โหนด B และ F ได้รับ QRY packet และทำการส่งต่อ QRY packet



ภาพที่ 2.11

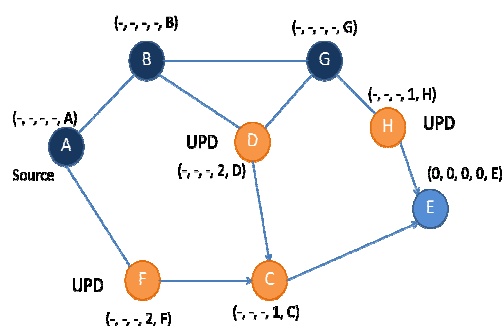
โหนด G และ D ทำการส่งต่อ QRY packet ส่วนโหนด C สร้างและส่ง UPD packet กลับไปให้

โหนด A



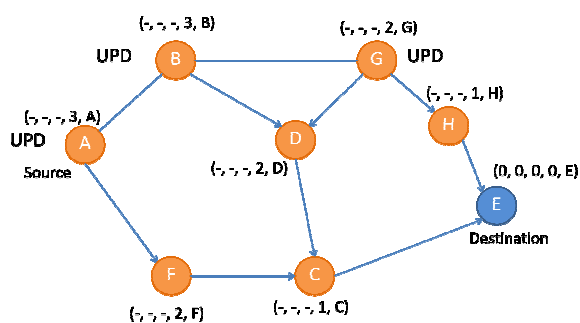
ภาพที่ 2.12

โหนด F และ D ส่งต่อ UPD packet ส่วนโหนด H สร้างและส่ง UPD packet กลับไปให้โหนด A



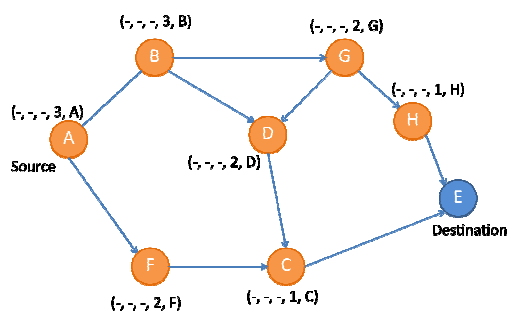
ภาพที่ 2.13

โหนด A ได้รับ UPD packet และโหนด B และ G ส่งต่อ UPD packet



ภาพที่ 2.14

กระบวนการสร้างเส้นทางในการจัดส่งแพ็คเก็ตข้อมูลจากโหนด A – E เสริมจลิน



ข้อดีของโปรโตคอลค้นหาเส้นทาง TORA มีหลายประการดังนี้ สร้างเส้นทางได้รวดเร็วและไม่เกิดลูป ตรวจสอบลิงค์ที่ขาดและลบเส้นทางที่ไม่ถูกต้องได้ภายในเวลาที่ยอมรับได้ มีการสำรองเส้นทางในการจัดส่งข้อมูลไปยังโหนดปลายทางที่ต้องการในกรณีที่ลิงค์ขาดสามารถใช้ได้ทันทีไม่ต้องทำการประมวลผลใหม่ ข้อเสีย คือ เกิดการหน่วง (delay) ระหว่างการประมวลผลเพื่อค้นหาเส้นทางใหม่

2.1.2.4 โปรโตคอลค้นหาเส้นทาง Associativity-Based Routing (ABR)

โปรโตคอลค้นหาเส้นทาง Associativity-Based Routing (ABR) (Chai-Keong, 1996) ทำการค้นหาเส้นทางโดยใช้ค่าดีกรีเสถียรภาพสนธิของโหนดภายในพื้นที่เป็นเกณฑ์ (degree of association stability) โดยแต่ละโหนดจะส่งสัญญาณ (beacon) เพื่อบอกสถานะแก่โหนดเพื่อนบ้าน และเมื่อโหนดเพื่อนบ้านได้รับสัญญาณก็จะทำการบันทึกค่าลงในตารางสนธิการ (associativity table) โหนดที่มีค่าดีกรีเสถียรภาพสนธิสูง แสดงว่าโหนดนั้นมีการเคลื่อนที่ต่ำในทางตรงกันข้ามโหนดที่มีการเคลื่อนที่สูงจะมีค่าดีกรีเสถียรภาพสนธิต่ำ ค่าที่อยู่ในตารางนี้จะถูกจัดเรียงใหม่เมื่อโหนดเพื่อนบ้านหรือตัวโหนดเองมีการเคลื่อนที่ กระบวนการทำงานของโปรโตคอลค้นหาเส้นทาง ABR แบ่งออกเป็น 3 ส่วนด้วยกันคือ การค้นหาเส้นทาง (route discovery) การสร้างเส้นทางใหม่ (route reconstruction: RRC) และการลบเส้นทาง (route deletion)

เมื่อโหนดต้นทางมีความต้องการจัดส่งข้อมูลให้โหนดปลายทางที่ต้องการให้เริ่มกระบวนการค้นหาเส้นทาง (route discovery) โดยbroadcastข้อความร้องขอเส้นทาง (broadcast query message: BQ) เมื่อโหนดที่ไม่ใช่โหนดปลายทางได้รับ BQ จะใส่ค่าเสถียรภาพสนธิระหว่างโหนด

เพื่อนบ้านลงไปในข้อความและบรอดแคสต์ข้อความร้องขอเส้นทางต่อไป เมื่อโหนดปลายทางได้รับ BQ แล้วให้ส่งแพ็คเกจตอบกลับให้กับโหนดต้นทางโดยเลือกเส้นทางที่ดีที่สุดในการส่ง

เมื่อเส้นทางที่ใช้ในการจัดส่งข้อมูลไปยังโหนดปลายทางเกิดความเสียหาย ให้ใช้กระบวนการสร้างเส้นทางใหม่ (route reconstruction: RRC)

หลังจากที่โหนดต้นทางไม่มีความต้องการในการจัดส่งข้อมูลไปให้กับโหนดปลายทางนั้นแล้ว ให้ทำการบรอดแคสต์แพ็คเกจในการลบเส้นทางจัดส่งข้อมูลนั้นทิ้งไป ซึ่งอยู่ในขั้นตอนการลบเส้นทาง (route delete)

ข้อดีของโปรโตคอลค้นหาเส้นทาง ABR คือ โหนดปลายทางจะไม่ได้รับแพ็คเกจข้อมูลซ้ำซ้อน (duplicate packet) เนื่องจากมีการบำรุงรักษาเส้นทางแค่เส้นทางเดียว ข้อเสีย คือ การส่งสัญญาณระบุตำแหน่งบ่อยทำให้สิ้นเปลืองพลังงาน จัดเป็นโอเวอร์เฮดของโปรโตคอล

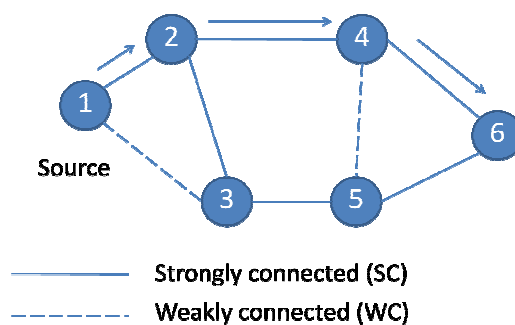
2.1.2.5 โปรโตคอลค้นหาเส้นทาง Signal Stability Routing (SSR)

โปรโตคอลค้นหาเส้นทาง Signal Stability-based Adaptive Routing (SSR) (Dube, et al., 1997) เป็นโปรโตคอลที่เลือกเส้นทางโดยดูจากความแรงของสัญญาณระหว่างโหนดการทำงานของโปรโตคอลนี้ประกอบไปด้วยโปรโตคอลย่อย 2 โปรโตคอลด้วยกันคือ Forwarding Protocol (FP) และ Dynamic Routing Protocol (DRP)

ขั้นตอนการทำงานของโปรโตคอลนี้เริ่มจากโหนดต้นทางบรอดแคสต์ข้อความค้นหาเส้นทาง (Route-search message) เพื่อหาเส้นทางไปยังโหนดปลายทางที่ต้องการ โหนดที่ได้รับข้อความค้นหาเส้นทางจากลิงค์ที่มีสัญญาณอ่อน (Weakly Connection: WC) ให้รีบัพแพ็คเกจทิ้ง ในทางกลับกันถ้าโหนดได้รับข้อความค้นหาเส้นทางจากลิงค์ที่มีสัญญาณแรง (Strongly Connection: SC) ให้บรอดแคสต์ข้อความค้นหาเส้นทางต่อไปให้กับโหนดเพื่อนบ้านจนไปถึงโหนดปลายทาง

ภาพที่ 2.15

แสดงเส้นทางที่ route-search message ถูกส่งต่อ

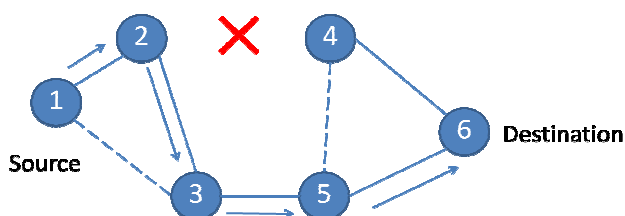


เมื่อโหนดปลายทางได้รับข้อความค้นหาเส้นทาง แล้วให้ส่งแพ็คเกจตอบกลับ (route reply packet) ไปให้โหนดต้นทาง และเมื่อโหนดต้นทางได้รับแพ็คเกจตอบกลับแล้วให้ทำการบันทึกข้อมูลเส้นทางเอาไว้ในตารางส่งต่อ และใช้เส้นทางนั้นในการจัดส่งแพ็คเกจส่งต่อไปให้กับโหนดปลายทาง

การปรับปรุงเส้นทางจะเกิดขึ้นก็ต่อเมื่อทราบว่าลิงค์ระหว่างโหนดเพื่อนบ้านขาดให้ส่งข้อความแสดงข้อผิดพลาด (error message) บอกโหนดต้นทางเพื่อให้โหนดต้นทางทำการค้นหาเส้นทางใหม่และส่งข้อความลบเส้นทาง (erase message) ลบเส้นทางที่ลิงค์ขาดออกไป (ภาพที่ 2.16)

ภาพที่ 2.16

แสดงเส้นทางใหม่ที่ใช้ในการจัดส่งข้อมูลให้กับโหนดปลายทางเมื่อพบว่าเส้นทางระหว่างโหนด 2 และ 4 ขาด



ข้อดีของโปรโตคอลค้นหาเส้นทาง SSR คือ ทำการเลือกเฉพาะเส้นทางส่งต่อที่มีสัญญาณแรงจึงทำให้ลดภาระในการบำรุงรักษาเส้นทาง ข้อเสียคือ ถ้าลิงค์ที่ใช้ในการส่งต่อขาดต้องเริ่มทำการสร้างเส้นทางส่งต่อใหม่ทั้งหมด

2.2 โปรโตคอลค้นหาเส้นทางสำหรับเครือข่ายยานพาหนะ (Vehicular Ad hoc NETWORK Routing Protocols)

เครือข่ายยานพาหนะ (Vehicular Ad hoc NETWORK: VANET) เป็นอีกรูปแบบหนึ่งของเครือข่ายเคลื่อนที่เฉพาะกิจโดยมีรถยนต์เป็นโหนดภายในเครือข่าย รถยนต์ภายในเครือข่ายยานพาหนะสามารถเคลื่อนที่และสื่อสารกันได้อย่างอิสระเหมือนโหนดในเครือข่ายเคลื่อนที่เฉพาะกิจ เนื่องจากมีการติดตั้งอุปกรณ์สื่อสารแบบไร้สายตามเทคโนโลยี IEEE 802.11 ไว้ภายใน โหนดภายในเครือข่ายยานพาหนะ คือรถยนต์สามารถเคลื่อนที่ได้ด้วยความเร็วค่อนข้างสูง ส่งผลให้รูปแบบการเชื่อมต่อของโหนดภายในเครือข่ายมีการเปลี่ยนแปลงรูปแบบการต่อเชื่อมอย่างรวดเร็ว

โปรโตคอลค้นหาเส้นทางภายในเครือข่ายยานพาหนะที่มีประสิทธิภาพ จะทำการค้นหาเส้นทางแบบฐานตำแหน่ง (position-based routing protocols) กล่าวคือ อาศัยตำแหน่งข้อมูลปัจจุบันของโหนดต้นทาง (source node) ข้อมูลตำแหน่งของโหนดเพื่อนบ้าน (neighboring node) และข้อมูลตำแหน่งของโหนดปลายทาง (destination node) ในการตัดสินใจเลือกโหนดส่งต่อ (forwarding node) ที่เหมาะสมที่สุดจำนวนหนึ่งโหนดเพื่อให้แพ็คเก็ตข้อมูลเดินทางไปยังโหนดปลายทางได้อย่างรวดเร็ว

โปรโตคอลค้นหาเส้นทางที่ถูกพัฒนาสำหรับเครือข่ายยานพาหนะ (Vehicular Ad hoc NETWORK routing protocol) จัดแบ่งออกได้เป็น 3 กลุ่มใหญ่ (Chennikara-Varghese, Chen, Altintas, & Cai, 2006) คือ กลุ่มของโปรโตคอลที่ใช้กลยุทธ์เชิงละโมภในการส่งต่อแพ็คเก็ต (Greedy Packet Forwarding scheme) (Karp & Kung, 2000) กลุ่มของโปรโตคอลที่ไม่ใช้เครื่องบอกตำแหน่ง (non-beacon routing scheme) (Füßler, Widmer, Käsemann, Mauve, & Hartenstein, 2003) และกลุ่มของโปรโตคอลที่ใช้ข้อมูลทางภูมิศาสตร์มาวางทับบนเครือข่าย (Overlay Routing) (Lochert, et al., 2003) (Jing, Lu, & Rothermel, 2003)

2.2.1 กลุ่มของโปรโตคอลที่ใช้กลยุทธ์เชิงละโมภในการส่งต่อแพ็คเก็ต (Greedy Packet Forwarding scheme)

กลุ่มของโปรโตคอลที่ใช้กลยุทธ์เชิงละโมภในการส่งต่อแพ็คเก็ต (Greedy Packet Forwarding scheme) ยกตัวอย่างเช่น Greedy Perimeter Stateless Routing (GPSR) (Karp & Kung, 2000) มีลักษณะร่วมที่สำคัญ คือ เลือกโหนดส่งต่อเป็นโหนดเพื่อนบ้านที่อยู่ใกล้กับโหนดปลายทางมากที่สุดจำนวนหนึ่งโหนด ข้อดี คือ เหมาะกับเครือข่ายที่มีการเปลี่ยนแปลงสูง ข้อเสียของโปรโตคอลในกลุ่มนี้ คือ วิธีการส่งต่อแบบนี้ใช้ไม่ได้ผลเมื่อตัวโหนดที่ทำการประมวลผลอยู่เป็นโหนดที่อยู่ใกล้กับโหนดปลายทางมากที่สุดเมื่อเทียบกับโหนดเพื่อนบ้านทั้งหมด

ในหัวข้อนี้จะกล่าวถึงตัวอย่างของโปรโตคอลที่ใช้กลยุทธ์เชิงละโมภในการส่งต่อแพ็คเก็ต ได้แก่ Greedy Perimeter Stateless Routing (GPSR) (Karp & Kung, 2000)

2.2.1.1 โปรโตคอลค้นหาเส้นทาง Greedy Perimeter Stateless Routing (GPSR)

โปรโตคอลค้นหาเส้นทาง Greedy Perimeter Stateless Routing (GPSR) (Karp & Kung, 2000) เป็นโปรโตคอลที่ทำการค้นหาและส่งต่อแพ็คเก็ตข้อมูลให้กับโหนดเพื่อนบ้านที่มีตำแหน่งใกล้กับโหนดปลายทางมากที่สุดจนกระทั่งแพ็คเก็ตข้อมูลไปถึงโหนดปลายทางที่ต้องการ การส่งต่อแพ็คเก็ตลักษณะนี้เรียกว่า วิธีการเชิงละโมภ หรือวิธีการเชิงกิริติ (greedy mode) แต่เมื่อแพ็คเก็ตข้อมูลไม่สามารถส่งต่อไปยังโหนดเพื่อนบ้านที่อยู่ใกล้กับโหนดปลายทางได้อีกต่อไปจะเข้าสู่กระบวนการกู้คืนโดยใช้วิธีการที่เรียกว่าเพอริมิเตอร์ (perimeter mode) กล่าวคือ จะทำการส่งต่อแพ็คเก็ตข้อมูลไปให้กับโหนดเพื่อนบ้านในทิศทางทวนเข็มนาฬิกาจนกระทั่งแพ็คเก็ตข้อมูลไปถึงโหนดปลายทางที่ต้องการ

ข้อดีของโปรโตคอล GPSR คือ สามารถหาเส้นทางส่งต่อข้อมูลใหม่ได้เร็วเมื่อโหนดมีการเคลื่อนที่ และข้อมูลที่โหนดต้องใช้ในการตัดสินใจเลือกโหนดส่งต่อแพ็คเก็ตมีเพียงตำแหน่งของโหนดเพื่อนบ้านซึ่งมีระยะทางแค่เพียงหนึ่งฮอป (hop) เท่านั้น ข้อเสียที่สำคัญ ได้แก่ วิธีการส่งต่อแบบนี้ใช้ไม่ได้ผลเมื่อตัวโหนดที่ทำการประมวลผลอยู่เป็นโหนดที่อยู่ใกล้กับโหนดปลายทางมากที่สุดเมื่อเทียบกับโหนดเพื่อนบ้านทั้งหมด นอกจากนี้ โปรโตคอล GPSR จำเป็นต้องใช้ข้อมูลตำแหน่งปัจจุบันจากส่วนบริการตำแหน่ง (location service) ในการตัดสินใจเลือกโหนดส่งต่อที่เหมาะสม ซึ่งบริการดังกล่าวยังไม่ครอบคลุมในทุกพื้นที่ และมีค่าใช้จ่ายที่ค่อนข้างสูง

2.2.2 กลุ่มของโปรโตคอลที่ไม่ใช้เครื่องบอกตำแหน่ง (non-beacon routing scheme)

กลุ่มของโปรโตคอลที่ไม่ใช้เครื่องบอกตำแหน่ง (non-beacon routing scheme) ยกตัวอย่างเช่น Contention-Based Forwarding (CBF) (Füßler, et al., 2003) มีลักษณะร่วมที่สำคัญ คือ เป็นโปรโตคอลที่ไม่มีการส่งสารบอกตำแหน่ง (beacon message) ระหว่างโหนดภายในเครือข่าย ข้อดีของโปรโตคอลในกลุ่มนี้ คือ ลดการใช้แบนด์วิดท์ ข้อเสีย คือ ข้อมูลไปไม่ถึงโหนดปลายทางที่ต้องการเมื่อเกิดทอพอโลยีโฮลด์ (topology hole)

ในหัวข้อนี้ จะกล่าวถึงตัวอย่างของโปรโตคอลที่ไม่ใช้เครื่องบอกตำแหน่ง ได้แก่ Contention-Based Forwarding (CBF) (Füßler, et al., 2003)

2.2.2.1 โปรโตคอลค้นหาเส้นทาง Contention-Based Forwarding (CBF)

โปรโตคอลค้นหาเส้นทาง Contention-Based Forwarding (CBF) (Füßler, et al., 2003) เป็นโปรโตคอลมีรูปแบบการส่งต่อเป็นแบบเชิงละโมบที่ไม่ใช้สารบอกตำแหน่งในการประมวลผลเพื่อตัดสินใจเลือกโหนดถัดไป ลักษณะการทำงานของ CBF มี 3 ขั้นตอน คือ การเลือกโหนดส่งต่อแพ็คเก็ต (forwarding node) ขั้นตอนในช่วงการแข่งขัน (contention period) และขั้นตอนยับยั้ง (suppress)

เมื่อโหนดต้นทางมีความต้องการส่งแพ็คเก็ตข้อมูลไปให้โหนดปลายทางใด ๆ ให้บรอดแคสต์แพ็คเก็ตข้อมูลนั้นไปให้กับโหนดเพื่อนบ้าน หลังจากนั้นโหนดเพื่อนบ้านจะทำการแข่งขันกัน ในการที่จะมีสิทธิเป็นคนส่งแพ็คเก็ต วิธีการแข่งขันก็คือ เมื่อโหนดเพื่อนบ้านได้รับแพ็คเก็ตแล้วให้ทำการคำนวณระยะห่างระหว่างตนเองกับโหนดปลายทาง จากนั้นกำหนดเวลาคอยก่อนที่จะจัดส่งแพ็คเก็ตข้อมูลต่อไปยังโหนดเพื่อนบ้านของตนเอง ซึ่งเวลาดังกล่าวจะเป็นสัดส่วนตรงกับระยะทางระหว่างโหนดตนเองกับโหนดปลายทาง โหนดใดที่คอยจนครบตามเวลาก่อน จะเป็นผู้ที่มีสิทธิในการส่งต่อแพ็คเก็ต และโหนดที่ชนะการแข่งขันจะไปยับยั้ง (suppress) การส่งต่อแพ็คเก็ตของโหนดอื่น และจะกลายเป็นโหนดส่งต่อที่มีสิทธิส่งต่อแพ็คเก็ตข้อมูลไปให้กับโหนดปลายทาง

ข้อดีของโปรโตคอล CBF คือ ลดการใช้แบนด์วิดท์ในการกระจายสารบอกตำแหน่ง และไม่สิ้นเปลืองเนื้อที่ในการจัดเก็บข้อมูลของโหนดเพื่อนบ้าน ข้อเสีย คือ โหนดปลายทางอาจจะไม่ได้รับแพ็คเก็ตส่งต่ออันเนื่องมาจากเครือข่ายเกิดทอพอโลยีโฮลด์ (topology hole)

2.2.3 กลุ่มของโปรโตคอลที่ใช้ข้อมูลทางภูมิศาสตร์มาวางทับบนเครือข่าย (Overlay Routing)

กลุ่มของโปรโตคอลที่ใช้ข้อมูลทางภูมิศาสตร์มาวางทับบนเครือข่าย หรือโอเวอร์เลย์ (overlay routing) ยกตัวอย่างเช่น Geographic Source Routing (GSR) (Lochert, et al., 2003) และ Spatially-Aware Routing (SAR) (Jing, et al., 2003) มีลักษณะร่วมที่สำคัญคือ เป็นโปรโตคอลที่นำเอาข้อมูลตำแหน่งของโหนด โครงสร้างของถนน และสภาพแวดล้อมโดยรอบมาใช้ในกระบวนการค้นหาเส้นทาง โดยแผนที่ถนนจะถูกมองให้อยู่ในรูปของกราฟซึ่งประกอบด้วยจุด (vertices) สำหรับใช้แทนทางแยกและเส้นขอบ (edges) สำหรับใช้แทนส่วนของถนน นอกจากการใช้แผนที่ถนนจริงซ้อนทับบนเครือข่ายแล้ว โปรโตคอลบางตัวในกลุ่มนี้ยังมีความแตกต่างจากโปรโตคอลในกลุ่มอื่น ๆ กล่าวคือ แทนที่จะทำการคัดเลือกโหนดสำหรับส่งต่อโดยตรง โปรโตคอลจะทำการคัดเลือกทางแยกที่เหมาะสมสำหรับเป็นจุดส่งต่อเพื่อเกิดข้อมูลแทนที่จะทำการคัดเลือกโหนดส่งต่อโดยตรง โหนดใด ๆ บริเวณทางแยกที่ได้รับการคัดเลือกก็จะทำหน้าที่เป็นโหนดส่งต่อเพื่อเกิดข้อมูลโดยอัตโนมัติ ข้อดีของโปรโตคอลในกลุ่มนี้ คือ เป็นโปรโตคอลที่เหมาะสมกับสภาพแวดล้อมในเมือง หรือมีรูปแบบการเชื่อมต่อของโหนดภายในเครือข่ายที่ไม่แน่นอนเนื่องจากโหนดมีการเคลื่อนที่อยู่ตลอดเวลา ข้อเสียที่สำคัญ ได้แก่ แพ็คเก็ตข้อมูลอาจถูกส่งต่อไปยังทางแยกซึ่งไม่มีโหนดใด ๆ อยู่บริเวณนั้น ส่งผลให้โหนดปลายทางไม่ได้รับแพ็คเก็ตข้อมูล นอกจากนี้ โปรโตคอลในกลุ่มนี้ ยังจำเป็นต้องอาศัยข้อมูลตำแหน่งปัจจุบันของโหนดปลายทางในการคำนวณเส้นทางสำหรับส่งต่อแพ็คเก็ตข้อมูลด้วย

ในหัวข้อนี้จะกล่าวถึงตัวอย่างของโปรโตคอลที่ใช้ข้อมูลทางภูมิศาสตร์มาวางทับบนเครือข่าย ได้แก่ Geographic Source Routing (GSR) (Lochert, et al., 2003) และ Spatially-Aware Routing (SAR) (Jing, et al., 2003)

2.2.3.1 โปรโตคอลค้นหาเส้นทาง Geographic Source Routing (GSR)

โปรโตคอลค้นหาเส้นทาง Geographic Source Routing (GSR) (Lochert, et al., 2003) เริ่มต้นที่โหนดต้นทางทำการค้นหาเส้นทางที่สั้นที่สุดด้วย Dijkstra's algorithm ซึ่งเส้นทางที่คำนวณได้นั้นจะเป็นลำดับของทางแยกที่แพ็คเก็ตถูกส่งต่อไปจนถึงโหนดปลายทางที่ต้องการ เมื่อโหนดต้นทางคำนวณเส้นทางได้แล้วก็ให้ทำการส่งแพ็คเก็ตข้อมูลตามเส้นทางที่คำนวณได้จากข้างต้นจนกระทั่งแพ็คเก็ตข้อมูลไปถึงโหนดปลายทางที่ต้องการ

ข้อดีของโปรโตคอล GSR คือ เป็นโปรโตคอลซึ่งเหมาะกับการใช้งานในเมืองที่มีสิ่งกีดขวางหรืออุปสรรค เช่น อาคารบ้านเรือนและทางแยก ส่วนข้อเสียของโปรโตคอล GSR มีดังนี้ การไล่เส้นทางของแพ็คเก็ตกำกับไว้ในแพ็คเก็ตเฮดเดอร์นั้นเป็นการเพิ่มค่าใช้จ่าย หรือโอเวอร์เฮด (overhead) ถ้าหากระยะทางจากโหนดต้นทางกับโหนดปลายทางอยู่ใกล้กันมากขนาดของแพ็คเก็ตก็ต้องมีขนาดใหญ่ตามไปด้วย นอกจากนี้ การเชื่อมต่อระหว่างทางแยกอาจจะไม่มีอยู่จริงก็ได้ เนื่องจากไม่มีโหนดที่เชื่อมต่อถึงกันได้

2.2.3.2 โปรโตคอลค้นหาเส้นทาง Spatially-Aware Routing (SAR)

โปรโตคอลค้นหาเส้นทาง Spatially-Aware Routing (SAR) (Jing, et al., 2003) เริ่มต้นเมื่อโหนดต้นทางมีแพ็คเก็ตข้อมูลที่ต้องการส่งไปให้โหนดปลายทางให้ทำการค้นหาเส้นทางที่สั้นที่สุดไปยังโหนดปลายทางโดยใช้ Dijkstra's algorithm เส้นทางนี้ประกอบไปด้วยลำดับของสถานที่ที่น่าสนใจ (significant place) ระหว่างทางที่เชื่อมต่อกัน หลังจากนั้นให้กำกับเส้นทางไว้ในแพ็คเก็ตข้อมูลที่ต้องการจะส่งทุกแพ็คเก็ต จากนั้นเมื่อโหนดส่งต่อได้รับแพ็คเก็ตข้อมูลจะเลือกโหนดเพื่อนบ้านที่อยู่ใกล้กับโหนดปลายทางมากที่สุดเป็นโหนดส่งต่อโดยกำหนดตำแหน่งของโหนดเพื่อนบ้านที่อยู่บริเวณใกล้เคียงลงในกราฟและเลือกส่งต่อแพ็คเก็ตข้อมูลไปให้กับโหนดเพื่อนบ้านที่อยู่ใกล้กับโหนดปลายทางมากที่สุดจนกระทั่งแพ็คเก็ตข้อมูลเดินทางไปถึงโหนดปลายทางที่ต้องการ

ข้อดีของโปรโตคอล SAR คือ เหมาะกับเครือข่ายที่มีการเคลื่อนที่อยู่ตลอดเวลา เนื่องจากโปรโตคอล SAR จะทำการค้นหาโหนดส่งต่อจุดต่อจุด ข้อเสีย คือ อาจมีโอกาที่โหนดปลายทางจะไม่ได้รับแพ็คเก็ตข้อมูลที่โหนดต้นทางต้องการส่งไปให้เมื่อเครือข่ายไม่มีการเชื่อมต่อกันระหว่างโหนด (partition) และโปรโตคอลต้องการบริการข้อมูลตำแหน่งปัจจุบันของโหนดปลายทางในการคัดเลือกโหนดสำหรับส่งต่อแพ็คเก็ตข้อมูลด้วย

2.3 การประยุกต์ใช้โปรโตคอลค้นหาเส้นทางสำหรับเครือข่ายยานพาหนะในชีวิตประจำวัน

การประยุกต์ใช้งานที่นำเอาเทคโนโลยีการสื่อสารแบบไร้สาย ระบบจีพีเอส (GPS) ส่วนให้บริการตำแหน่ง (location service) และโปรโตคอลค้นหาเส้นทางสำหรับเครือข่ายยานพาหนะมาใช้ในการค้นหารถแท็กซี่ที่อยู่ใกล้กับผู้ใช้โดยสารมากที่สุด ยกตัวอย่างเช่น ในประเทศสิงคโปร์มี

การนำเอาระบบระบุตำแหน่งโดยอัตโนมัติและระบบการจัดส่ง (GPS-based Automatic Vehicle Location and Dispatch System: AVLDS) (Ziqi, 2003) มาใช้ในการค้นหารถแท็กซี่ที่อยู่ใกล้กับผู้โดยสารมากที่สุดเพื่อให้การบริการที่รวดเร็ว เป็นต้น ลักษณะการใช้งานคือ ผู้โดยสารทำการร้องขอการใช้บริการแท็กซี่ผ่านการสื่อสารไร้สายไปยังพนักงานขับรถแท็กซี่ที่อยู่ภายในรัศมี 10 กิโลเมตร หากพนักงานขับรถคนใดยอมรับการร้องขอให้ทำการยืนยันโดยการตอบรับไปยังศูนย์ควบคุมโดยกดปุ่มที่ติดตั้งไว้ในรถแท็กซี่ ถ้าหากศูนย์ควบคุมไม่ได้ยินการตอบรับใด ๆ ภายในเวลา 10 วินาทีระบบจะทำการค้นหารถแท็กซี่ที่อยู่ใกล้บริเวณนั้นอีกครั้ง เมื่อได้รับการตอบรับการร้องขอแล้วจะมีการส่งข้อมูลของหมายเลขรถแท็กซี่และเวลาโดยประมาณที่คาดว่าจะรถแท็กซี่จะไปถึงตำแหน่งที่ผู้โดยสารอยู่ให้ผู้โดยสารทราบ

นอกจากนี้ เทคโนโลยีดังกล่าวข้างต้นอาจจะนำไปประยุกต์ใช้ในการค้นหารถพยาบาลที่อยู่ใกล้กับผู้ป่วยฉุกเฉินมากที่สุด ซึ่งผู้ป่วยฉุกเฉินอาจจะเป็นผู้สูงอายุที่ไม่ได้อาศัยอยู่กับลูกหลานได้อีกด้วย

2.4 ตัวแบบหรือแบบจำลองเชิงคณิตศาสตร์ (Mathematical Models)

มีตัวแบบหรือแบบจำลองทางคณิตศาสตร์จำนวนมากที่สามารถนำมาใช้ประโยชน์ในการจำลองปัญหาเกี่ยวกับการตัดสินใจเลือกตัวเลือกที่เหมาะสมที่สุด การเลือกใช้ตัวแบบหรือแบบจำลองต้องพิจารณาให้ตรงกับลักษณะ องค์ประกอบ และเงื่อนไขบังคับ (constraints) ของปัญหา เพื่อให้ได้ตัวแบบซึ่งสามารถจำลองสถานการณ์ของปัญหาได้อย่างถูกต้องที่สุด ยกตัวอย่างเช่น ตัวแบบ Set Covering (Set Covering Model) (Daskin, 1995, p. 92) เป็นแบบจำลองเชิงเส้นตรงสำหรับจำลองปัญหาที่ต้องการเลือกกลุ่มตัวแทนที่มีขนาดเล็กที่สุด หรือ กลุ่มตัวแทนที่เสียค่าใช้จ่ายในการลงทุนน้อยที่สุด แต่ต้องสามารถตอบสนองความต้องการของกลุ่มเป้าหมาย หรือ ต้องสามารถครอบคลุมกลุ่มเป้าหมายให้ได้ทั้งหมด ภาพที่ 2.17 แสดงการใช้ตัวแบบ Set Covering เพื่อจำลองปัญหาการเลือกพื้นที่เพื่อทำการติดตั้งไวร์เลสแอ็คเซสพอยต์ (Wireless Access Point: WAP) กระจายสัญญาณให้กับโหนดหรือคอมพิวเตอร์ที่อยู่ในละแวกใกล้เคียง ซึ่งมีวัตถุประสงค์เพื่อให้มีการใช้เงินลงทุนในการติดตั้งน้อยที่สุด แต่ต้องสามารถรองรับความต้องการเชื่อมต่อจากโหนดหรือคอมพิวเตอร์ได้ทุกเครื่อง จากภาพที่ 2.17 สมการที่ (2.17 a) แสดงฟังก์ชันวัตถุประสงค์ (objective function) ของปัญหา นั่นคือ ต้องการใช้เงินลงทุนสำหรับติดตั้งไวร์เลสแอ็คเซสพอยต์ให้น้อยที่สุด ในขณะที่สมการที่ (2.17 b) เป็นเงื่อนไขบังคับ (constraint) ที่

กำหนดว่าโหนดหรือคอมพิวเตอร์ทุกเครื่องจะต้องได้รับการเชื่อมต่อจากไวร์เลสแอ็คเซสพอยต์อย่างน้อยหนึ่งจุด และท้ายที่สุดสมการที่ (2.17 c) เป็นเงื่อนไขบังคับปริพันธ์ (integrality constraint) ซึ่งกำหนดให้ค่าของตัวแปรตัดสินใจ X_j ต้องเป็นเลขจำนวนเต็ม 0 หรือ 1 เท่านั้น

ภาพที่ 2.17

การใช้ตัวแบบ Set Covering เพื่อจำลองปัญหา

ข้อมูลนำเข้า (inputs)

$$a_{ij} = \begin{cases} 1 & \text{เมื่อไวร์เลสแอ็คเซสพอยต์ที่ } j \text{ สามารถให้บริการการเชื่อมต่อกับโหนด } i \text{ ได้} \\ 0 & \text{เมื่อไวร์เลสแอ็คเซสพอยต์ที่ } j \text{ ไม่สามารถให้บริการการเชื่อมต่อกับโหนด } i \text{ ได้} \end{cases}$$

f_j = ค่าใช้จ่ายในการติดตั้งไวร์เลสแอ็คเซสพอยต์ที่ตำแหน่ง j

ตัวแปรตัดสินใจ (decision variables)

$$X_j = \begin{cases} 1 & \text{เมื่อเลือกติดตั้งไวร์เลสแอ็คเซสพอยต์ที่ตำแหน่ง } j \\ 0 & \text{เมื่อไม่เลือกติดตั้งไวร์เลสแอ็คเซสพอยต์ที่ตำแหน่ง } j \end{cases}$$

$$\text{MINIMIZE} \quad \sum_j f_j X_j \quad (2.17 \text{ a})$$

$$\text{SUBJECT TO:} \quad \sum_j a_{ij} X_j \geq 1 \quad \forall i \quad (2.17 \text{ b})$$

$$X_j = 0, 1 \quad \forall j \quad (2.17 \text{ c})$$

ตัวแบบ Maximum Covering Location (Daskin, 1995, p. 110) เป็นแบบจำลองเชิงเส้นตรงอีกตัวแบบหนึ่ง ซึ่งสามารถจำลองสถานการณ์ของปัญหาที่มีลักษณะ องค์ประกอบ และ เงื่อนไขบังคับ ที่แตกต่างกันออกไปจากกลุ่มปัญหาที่สามารถจำลองได้ด้วยตัวแบบ Set Covering โดยลักษณะเฉพาะของปัญหาที่สามารถจำลองได้โดยใช้ตัวแบบ Maximum Covering Location มีดังนี้

- มีวัตถุประสงค์ในการรองรับหรือตอบสนองความต้องการของกลุ่มเป้าหมายให้ได้มากที่สุด
- มีการจำกัดจำนวนหรือขนาดของกลุ่มตัวแทน
- กลุ่มเป้าหมายมีความต้องการหรือข้อเรียกร้องที่แตกต่างกันได้

ภาพที่ 2.18 แสดงการใช้ตัวแบบ Maximum Covering Location เพื่อจำลองปัญหาการติดตั้งไวร์เลสแอ็กเซสพอยต์ (Wireless Access Point: WAP) เพื่อกระจายสัญญาณให้กับโหนดหรือคอมพิวเตอร์ที่อยู่ในละแวกใกล้เคียง ซึ่งมีวัตถุประสงค์เพื่อให้สามารถรองรับความต้องการเชื่อมต่อของโหนดหรือคอมพิวเตอร์ให้ได้มากที่สุด แต่จะต้องไม่ติดตั้งไวร์เลสแอ็กเซสพอยต์เกินกว่าจำนวนที่กำหนด จากภาพที่ 2.18 สมการที่ (2.18 a) แสดงฟังก์ชันวัตถุประสงค์ของปัญหา นั่นคือ ต้องการครอบคลุมความต้องการเชื่อมต่อของโหนดหรือคอมพิวเตอร์ให้ได้มากที่สุด ในขณะที่ สมการที่ (2.18 b) เป็นเงื่อนไขบังคับ (constraint) สำหรับควบคุมความถูกต้องว่า หากเลือกที่จะรองรับความต้องการเชื่อมต่อของโหนดหรือคอมพิวเตอร์ i แล้ว ($Z_i = 1$) จะต้องเลือกติดตั้งไวร์เลสแอ็กเซสพอยต์ที่โหนด j ใด ๆ อย่างน้อยหนึ่งโหนด ซึ่งสามารถกระจายสัญญาณ (ตามปริมาณที่ต้องการ) ให้กับโหนด i ได้ ต่อจากนั้น สมการที่ (2.18 c) เป็นเงื่อนไขบังคับที่ระบุว่าไม่สามารถติดตั้งไวร์เลสแอ็กเซสพอยต์ได้มากกว่าจำนวนที่กำหนด (P) และท้ายที่สุด สมการที่ (2.18 d) และ (2.18 e) เป็นเงื่อนไขบังคับปริพันธ์ (integrality constraint) ซึ่งกำหนดให้ค่าของตัวแปรตัดสินใจ X_j และ Z_i ต้องเป็นเลขจำนวนเต็ม 0 หรือ 1 เท่านั้น

ภาพที่ 2.18

การใช้ตัวแบบ Maximum Covering Location เพื่อจำลองปัญหา

ข้อมูลนำเข้า (inputs)

 h_i = ความต้องการเชื่อมต่อกับไวร์เลสแอ็คเซสพอยต์ของโหนด i P = จำนวนไวร์เลสแอ็คเซสพอยต์ทั้งหมดที่สามารถติดตั้งได้

$$a_{ij} = \begin{cases} 1 & \text{เมื่อไวร์เลสแอ็คเซสพอยต์ที่โหนด } j \text{ สามารถกระจายสัญญาณให้กับคอมพิวเตอร์ที่โหนด } i \text{ ได้} \\ 0 & \text{เมื่อไวร์เลสแอ็คเซสพอยต์ที่โหนด } j \text{ ไม่สามารถกระจายสัญญาณให้กับคอมพิวเตอร์ที่โหนด } i \text{ ได้} \end{cases}$$

ตัวแปรตัดสินใจ (decision variables)

$$Z_i = \begin{cases} 1 & \text{เมื่อเลือกที่จะรองรับความต้องการที่จะเชื่อมต่อของโหนด } i \\ 0 & \text{เมื่อเลือกที่จะไม่รองรับความต้องการที่จะเชื่อมต่อของโหนด } i \end{cases}$$

$$X_j = \begin{cases} 1 & \text{เมื่อเลือกติดตั้งไวร์เลสแอ็คเซสพอยต์ที่ตำแหน่งโหนด } j \\ 0 & \text{เมื่อเลือกไม่ติดตั้งไวร์เลสแอ็คเซสพอยต์ที่ตำแหน่งโหนด } j \end{cases}$$

$$\text{MAXIMIZE} \quad \sum_i h_i Z_i \quad (2.18 \text{ a})$$

$$\text{SUBJECT TO:} \quad Z_i \leq \sum_j a_{ij} X_j \quad \forall i \quad (2.18 \text{ b})$$

$$\sum_j X_j \leq P \quad (2.18 \text{ c})$$

$$X_j = 0, 1 \quad \forall j \quad (2.18 \text{ d})$$

$$Z_i = 0, 1 \quad \forall i \quad (2.18 \text{ e})$$

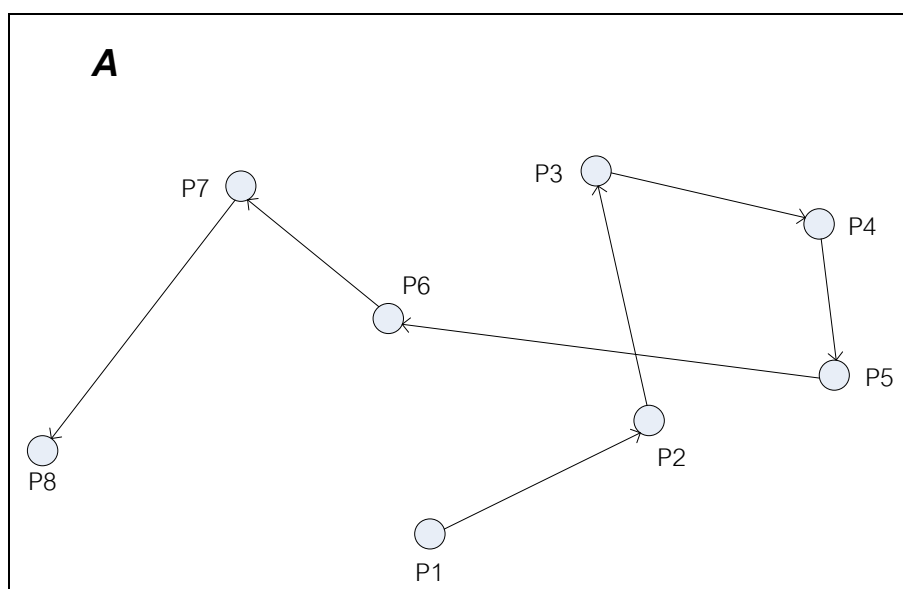
2.5 ตัวแบบการเคลื่อนที่ (Mobility Model)

2.5.1 ตัวแบบการเคลื่อนที่แบบแรนดอมเวย์พอยต์ (Random Waypoint Mobility Model: RW)

ตัวแบบการเคลื่อนที่แบบแรนดอมเวย์พอยต์ (RW, 2011) เป็นโมเดลที่ประกอบไปด้วยโหนดที่เคลื่อนที่ไปยังตำแหน่งปลายทางใดก็ได้โดยอิสระด้วยการสุ่มตำแหน่งปลายทางและความเร็ว หลังจากนั้นเมื่อโหนดเคลื่อนไปถึงตำแหน่งปลายทางแล้วจะหยุดรอเป็นระยะเวลาหนึ่งและทำการสุ่มเลือกตำแหน่งใหม่ไปเรื่อย ๆ ดังแสดงในภาพที่ 2.19 แสดงลักษณะการเคลื่อนที่ของโหนดตามตัวแบบการเคลื่อนที่แบบแรนดอมเวย์พอยต์ การเคลื่อนที่แบบแรนดอมเวย์พอยต์ (RW) นี้นิยมนำไปใช้ในการวัดสมรรถนะโปรโตคอลค้นหาเส้นทางสำหรับเครือข่ายเคลื่อนที่เฉพาะกิจ (MANET) เนื่องจากเป็นโมเดลที่ง่ายและถูกใช้งานอย่างแพร่หลาย

ภาพที่ 2.19

แสดงลักษณะการเคลื่อนที่ของโหนดตามตัวแบบการเคลื่อนที่แบบแรนดอมเวย์พอยต์



2.5.2 ตัวแบบการเคลื่อนที่แบบแมนฮัตตันกริด (Manhattan Grid Mobility Model: MG)

ตัวแบบการเคลื่อนที่แบบแมนฮัตตันกริด (MG, 2011) เป็นโมเดลอีกหนึ่งในหลายโมเดลที่ถูกนำไปใช้ในการวัดสมรรถนะโปรโตคอลค้นหาเส้นทางสำหรับเครือข่ายยานพาหนะ (VANET) ตัวแบบการเคลื่อนที่แบบแมนฮัตตันกริดเป็นโมเดลการเคลื่อนที่ของโหนดซึ่งเคลื่อนที่อยู่บนถนนที่เป็นระเบียบลักษณะเป็นกริดดังแสดงตามภาพที่ 2.20 การเคลื่อนที่ของโหนดในตัวแบบการเคลื่อนที่แบบแมนฮัตตันกริด (MG) สามารถเคลื่อนที่ได้ 2 ทิศทางด้วยกันคือ แนวตั้งและแนวนอนตามเส้นทางบนถนน เมื่อโหนดเคลื่อนที่มาถึงทางแยกโหนดจะสุ่มเลือกทิศทางที่จะมุ่งหน้าไป เช่น เลี้ยวซ้าย เลี้ยวขวา หรือเคลื่อนที่ตรงไป โดยใช้ความน่าจะเป็นเข้ามาช่วยในการเลือกทิศทางของโหนด

ภาพที่ 2.20

แสดงตัวอย่างถนนในตัวแบบการเคลื่อนที่แบบแมนฮัตตันกริด

