5907101012 มุขยา วงศ์ดาราพานิช

AODV Protocol

ประวัติย่อ

AODV ถูกพัฒนาขึ้นเมื่อช่วงปี 2003 โดย Nokia Research Center, University of California และ University of Cincinati

เนื้อหา

AODV ย่อมาจาก Ad Hoc On-Demand Distance Vector ซึ่งเป็น routing protocol designed มาสำหรับ wireless และ mobile ad hoc network

อีกทั้ง AODV ยังเป็น routing protocol ที่ใช้อยู่ใน ZigBee อีกด้วย ซึ่งเป็นระบบแบบ low poer low datarate wireless ad hoc network พูดอีกนัยหนึ่งคือใช้พลังงานต่ำ และเป็นwireless บริเวณกว้าง

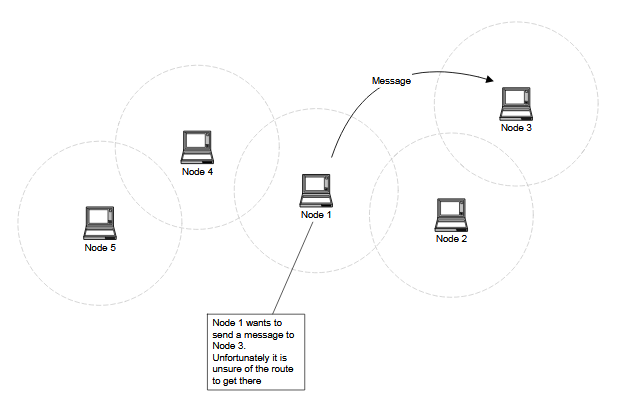
โดย protocol AODV นั้นจะทำการ route เส้นทางไปที่ปลายทางโดยขึ้นอยู่กับ demand รวมไปถึง support ทั้งการทำ unicast and multicast routing

AODV คือ routing protocol ที่เป็น ad hoc mobile network ซึ่งเหมาะกับการใช้เมื่อมี node จำนวนมาก เพราะ algorithm ของ AODV นั้นสร้าง route หรือเส้นทางระหว่าง node เท่านั้น และจะทำต่อเมื่อมีการ request การทำ routes จาก source node ซึ่งทำให้ network มีความ flexillity สำหรับให้ node เข้ามาเป็น route หรือการนำ node ออกจาก route ก็ทำได้ง่ายเพราะการทำ route หรือเส้นทาง route นั้นจะยังคง active อยู่ ตราบใดที่ data หรือ packet นั้นยังเดินทางไปตามเส้นทาง route จาก source เพื่อไปที่ destination node เมื่อ source ทำการหยุดส่ง packet เมื่อนั้น path ในการ route ก็จะ time out และหยุด

ตัวอย่าง รูปแบบการทำ routing ใน AODV โดยอ้างอิงจากรูปแบบการส่ง routing messages ระหว่าง node mobile computers หรือ node เพื่อส่งผ่าน messages ไปยัง node ปลายทางผ่าน neighbors ไปยัง nodes ปลายทางเพราะไม่สามารถสื่อสารได้โดยตรงด้วยตัวเอง

โดย AODV จะทำการ discover routes ในระหว่างที่ทำการส่ง message เพื่อตามหาช่องทางที่สั้นที่สุดในการส่งข้อมูลไปยังปลายทาง

รวมไปถึงยังสามารถยืนยันได้ว่าจะไม่มีการทำให้ข้อมูลที่ส่งนั้นกลายเป็น loop (การส่งข้อมูลวนกลับมาที่ source) ด้วย

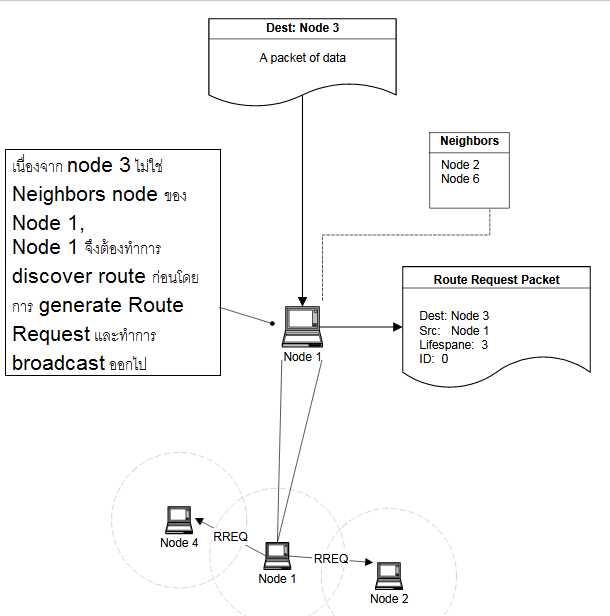


*รูปข้างต้นแสดงการทำ routing ของ node 1 โดย node 1 ต้องการจะส่งข้อมูลไปยัง node 3*

*เส้นประโดยรอบคือระยะการสื่อสารโดยประมาณของ node แต่ละ node ซึ่งจะเห็นได้ว่า ระยะของ node 1 นั้นไม่สามารถไปถึง node 3 ที่ต้องการสื่อสารได้จึงต้องส่งข้อมูลผ่าน neighbors*

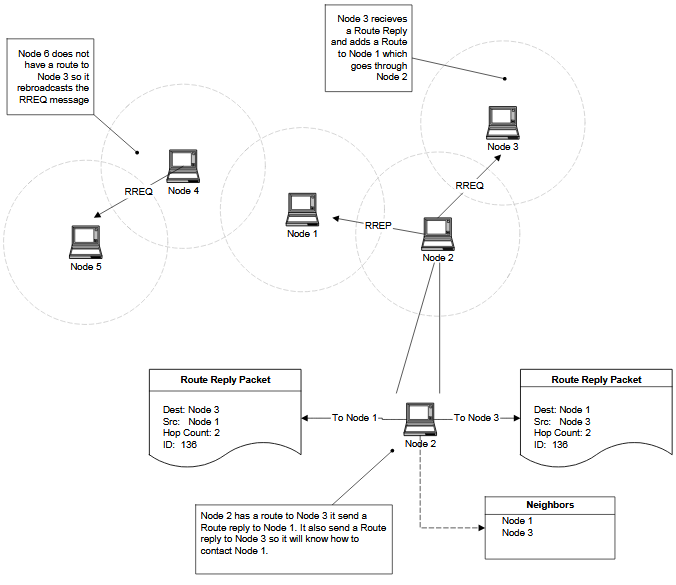
Nodes ที่สามารถคุยหรือสื่อสารกันได้จะถูกนับว่าเป็น Neighbors ซึ่งกันและกัน ซึ่ง Node จะทำการสื่อสารกันด้วยการรอฟัง Hello packet จาก Node ข้างเคียงที่ทำการ broadcast ออกมา

เมื่อ Node ใด Node หนึ่งต้องการส่งข้อความไปยัง Node อื่นที่ไม่ใช่ Neighbors ตัว Source Node (ต้นทาง) จะทำการ broadcast Route Request (RREQ) message

โดย Route Request Packet นี้ จะมีส่วนที่เป็น key bits ซึ่งมีข้อมูลจำพวก source, destination, lifespan of message และ sequence number (unique ID)

เมื่อ Neighbors ของ Node 1 ได้รับ RREQ message, Neighbors จะเลือกทำอย่างใดอย่างหนึ่ง

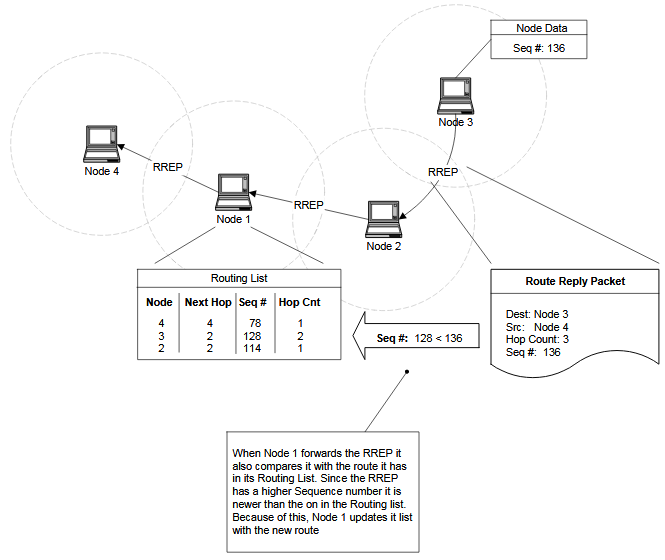
1. ถ้า Neighbors รู้ route ที่นำไปสู่ destination หรือเป็นตัว destination เอง Neighbors Node จะทำการส่ง Route Reply (RREP) message กลับมาที่ Node 1
2. ทำการ Rebroadcast RREQ ไปที่ Neighbors ของตัวเองอีกที และจะทำแบบนั้นไปเรื่อยๆจนกว่า lifespan ของ message จะหมดลง

ถ้า Node 1 ไม่ได้รับ reply ในช่วงระยะเวลาหนึ่ง Node 1 จะทำการ rebroadcast RREQ message อีกครั้งเว้นแต่ว่า ครั้งนี้จะเพิ่มจำนวน lifespan เข้าไปอีก และเปลี่ยนตัว ID Number (Sequence Number) เนื่องจาก Node ทุกตัวจะใช้ ID, Sequence Number ในการเช็คว่า ไม่ให้ตัวเองทำการ rebroadcast ข้อมูลเดิมซ้ำ

Sequence Numbers

Sequence Numbers จะมาในรูปแบบของ time stamps เพื่อให้ node สามารถเปรียบเทียบความ “ fresh “ ของข้อมูลกับ node อื่นๆ

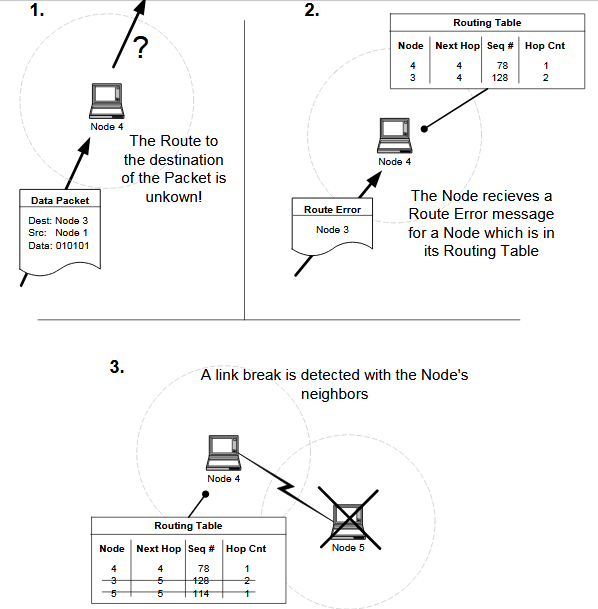
ทุกครั้งที่มีการส่ง message แบบไหนออกไปก็ตาม มันจะทำการเพิ่มตัว Sequence Numbers ของตัวเองขึ้นไปทุกครั้ง และ Node แต่ละ Node ก็จะบันทึก Sequence Numbers ของ Node อื่นๆที่เป็น Neighbors ด้วย

Sequence Number ยิ่งสูง ก็แปลว่าเป็น routing table ที่ fresh ยิ่งขึ้น

*จากรูปจะเห็นว่าNode1 กำลังทำการ forward RREP ไปที่ Node 4 และตัว Node 1 ก็เห็นว่า route ใน RREP มี Sequence number ที่สูงกว่า route ที่มีอยู่ใน routing list ของตัวเอง Node 1 จึงทำการแทนค่า routeเดิมที่มีอยู่ ด้วย route จาก RREP*

Error Messages

Route Error Message (RERR) เป็น message ที่จะทำให้ AODV สามารถจัดการการทำ routes ได้ เมื่อมีการเคลื่อนย้าย rotate node ไปในจุดอื่น

ยังไงก็ตาม เมื่อ Node ได้รับ RERR มันจะทำการมองไปที่ routing table แล้วทำการ remove routes ที่มี bad node ทิ้งทั้งหมด

รูปอธิบายเกี่ยวกับ scenario ของการส่ง RERR ซึ่งแบ่งออกเป็นสามกรณีดังอธิบายด้านล่าง

โดยการที่ Node ใดๆนั้น จะทำการ broadcast RERR message นั้นจะมีด้วยกัน 3 กรณี

1. Node ได้รับ Data packet ที่สมควรจะได้รับการ forward แต่ดันไม่มี route ไปที่ destination โดยสามารถแบ่งได้เป็นสองกรณี คือ ตัว Node เองไม่มี routes หรืออีกกรณีคือ Node อื่นคิดว่าเส้นทาง route ที่เหมาะสมทำหรับการ forward ไปที่ destination คือผ่าน node นี้
2. Node ได้รับ RERR message อย่างน้อย 1 ครั้งซึ่งเป็น Route error ที่มีอยู่แล้วใน routing table ของตัวเอง ซึ่งจะทำให้ node ปัจจุบันนั้นทำการ forward RERR ไปที่ node อื่นๆ
3. Node ตรวจพบว่าไม่สามารถสื่อสารกับ Neighbors ของตัวเองได้ เมื่อเกิดเหตุการณ์แบบนี้ขึ้น ตัว Node จะทำการมองไปที่ route table ของตัวเอง เพื่อดู route ที่เป็น next hop แล้วค่อยทำการ mark ให้เป็น invalid node จากนั้นค่อยส่ง RERR message ไปยัง Neighbors ถึงเรื่อง invalid node, routes

สรุป เอกลักษณ์ของ AODV

• สร้าง routes ต่อเมื่อมีการขอ request

• ใช้ Sequence Numbers ในการ track ความแม่นยำของข้อมูล

• ทำการจดจำ next hop route แทนที่จะจดจำทั้ง route ของทุก node

• ใช้ HELLO message ในการ track Neighbors