Analyzing RPL DODAG construction

Using 'Cooja' — a simulator for wireless sensors networks — to analyze process of DODAG formation according to RPL protocol.

Your task is to simulate a sensor network with
• one server
• four clients out which two should be out of server's range (but reachable by other clients).

Then in Wireshark go through the list of captured RPL messages in sequence and explain specifically according to your network topology how DODAG construction takes place. Also identify the root of the DODAG tree.
Hints:
• All nodes get a link-local IP address in the form of fe80::201:1:1:1 (node #1), fe80::202:2:2:2 (node #2), and so on.
• ff02::1a is a special IPv6 multicast address.
• In DIO messages, keep an eye on Rank and DODAGID fields.
Deliverables
• Your simulation file *.csc (In Cooja File > Save simulation as)
• Your packet capture *.pcap
• A document containing your analysis and explanation.

SOLUTION

- **1) Setup**

  The green node (labelled 1) is server node and the remaining yellow nodes (labelled 2,3,4 & 5) are client nodes.

L200921 (Aisha Muhammad Nawaz)
Introduction to Internet of Things
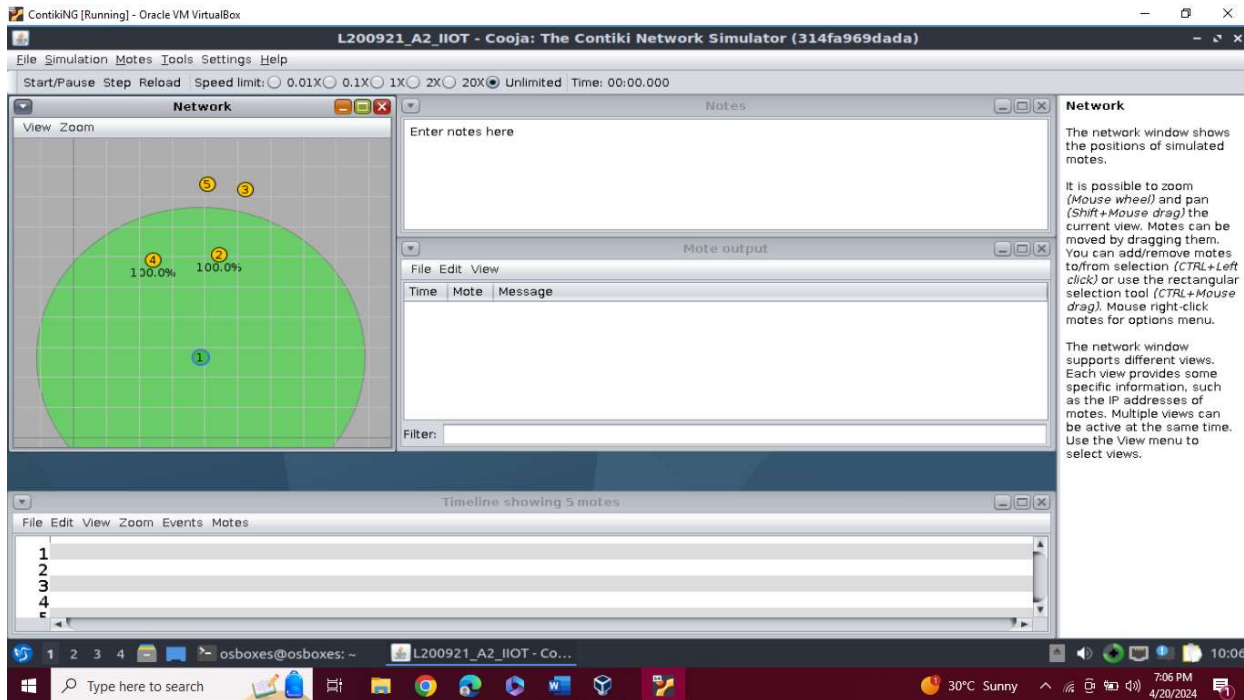**Assignment # 2 BSCS 8A Spring 2024**

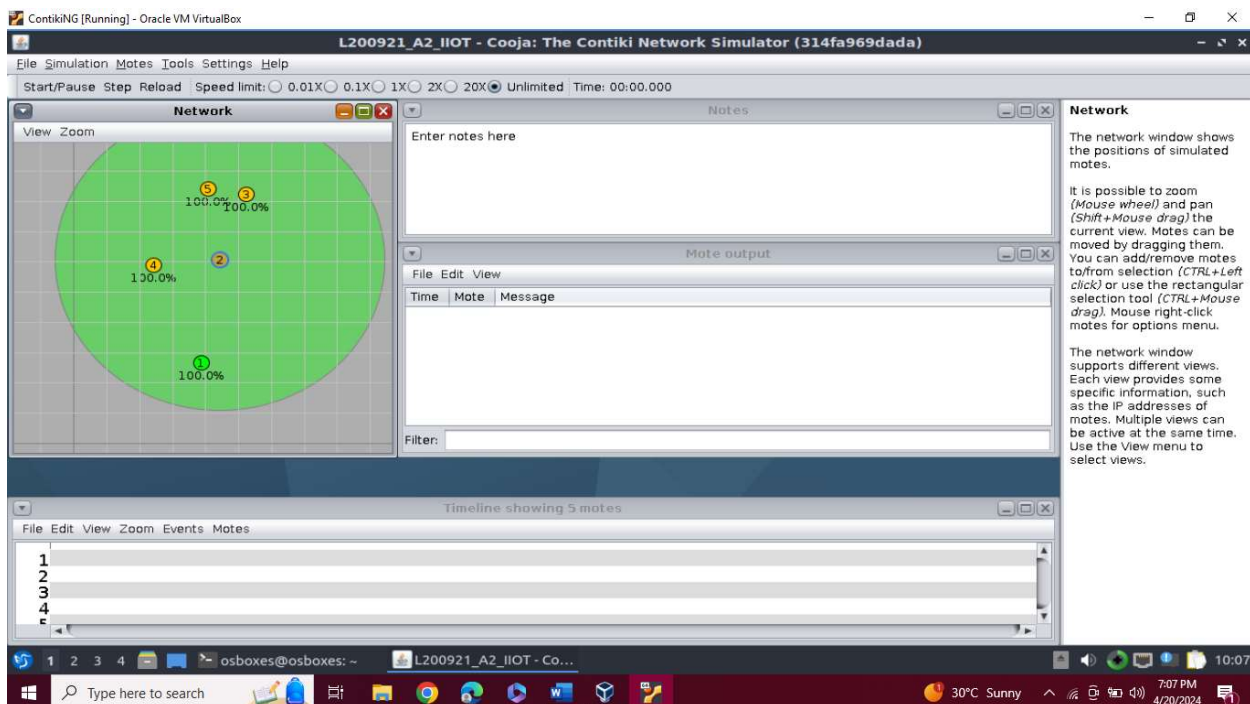**Fig 1: As visible above, the server cannot reach two of the four client nodes**

Fig 2: As visible above, the client 2 node can reach all other client nodes
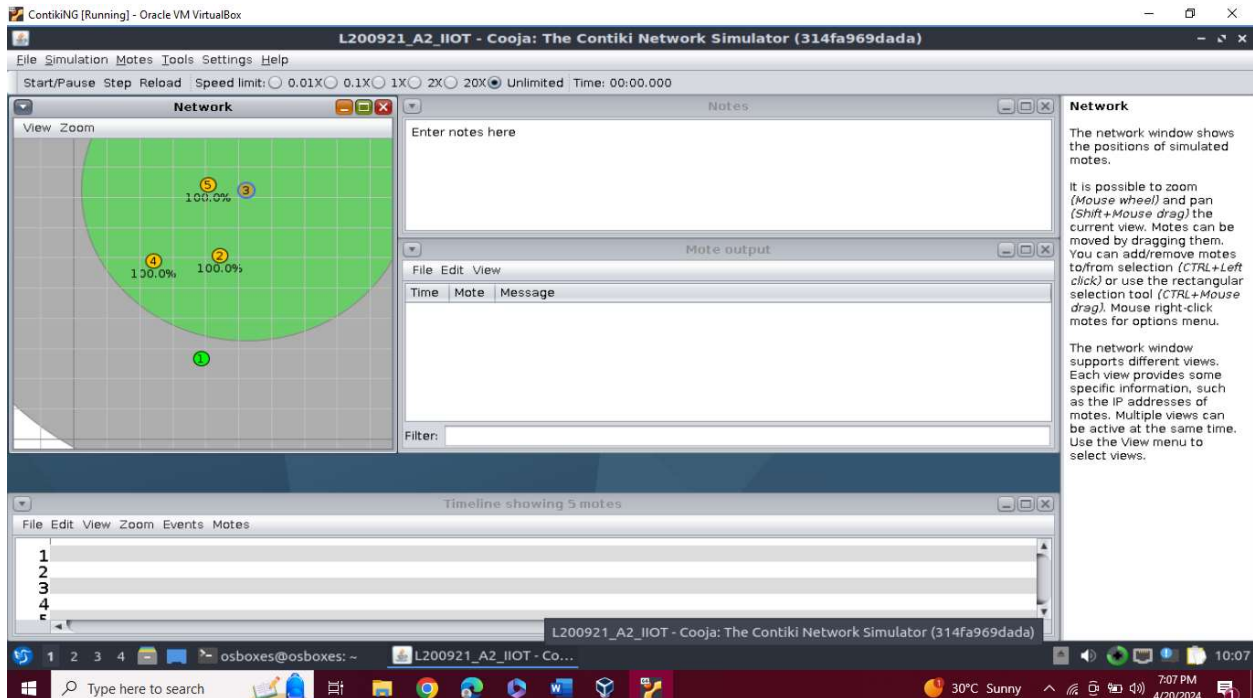
Fig 3: As visible above, the client 3 node can reach all other client nodes
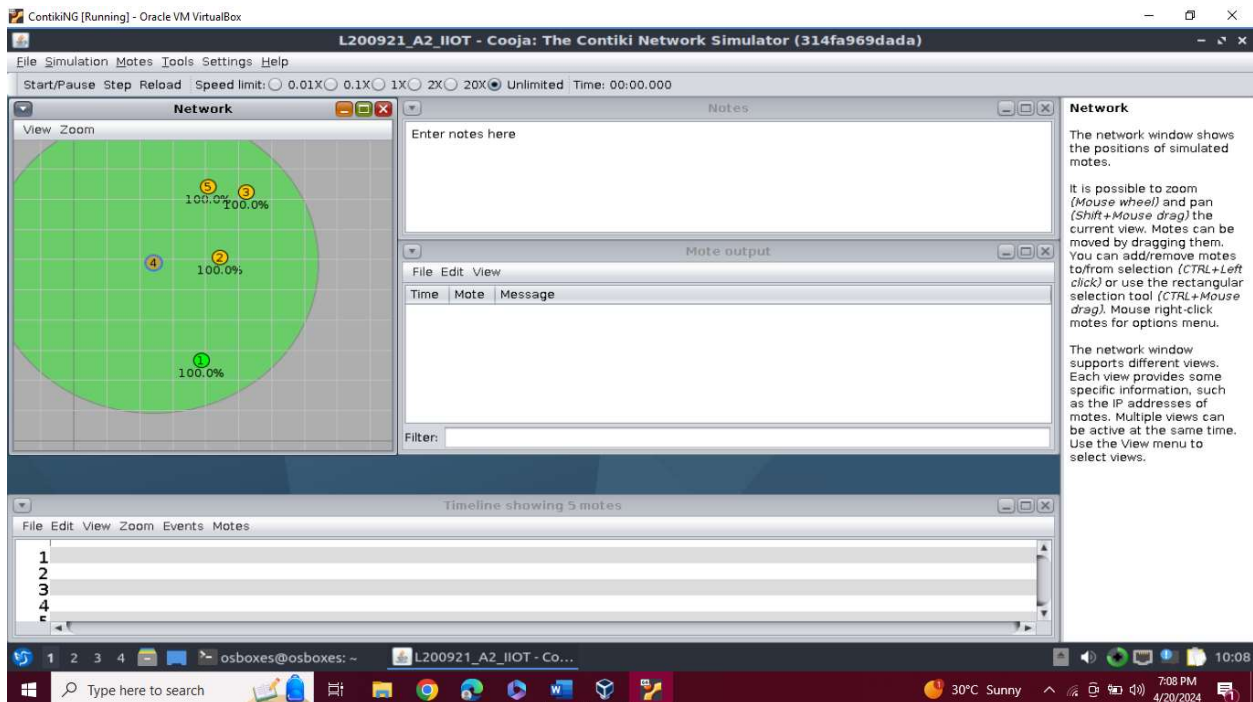
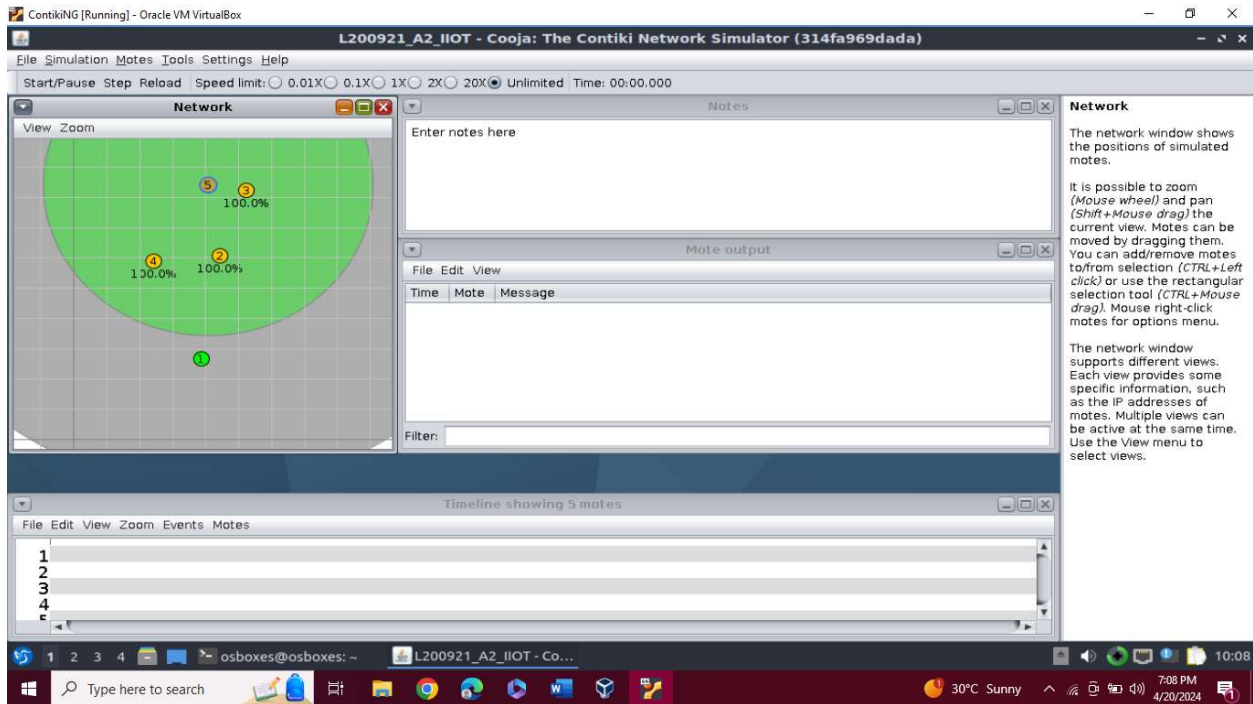**Fig 4: As visible above, the client 4 node can reach all other client nodes**

Fig 5: As visible above, the client 5 node can reach all other client nodes

- **2) Starting Simulation**
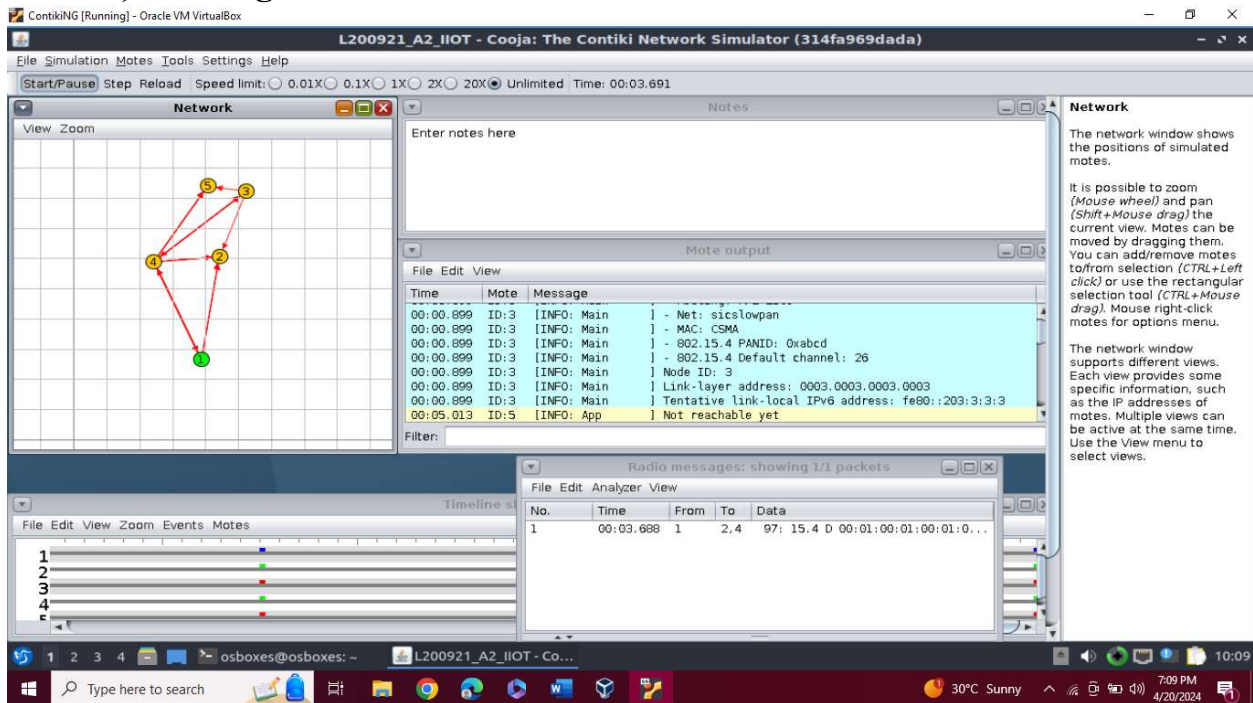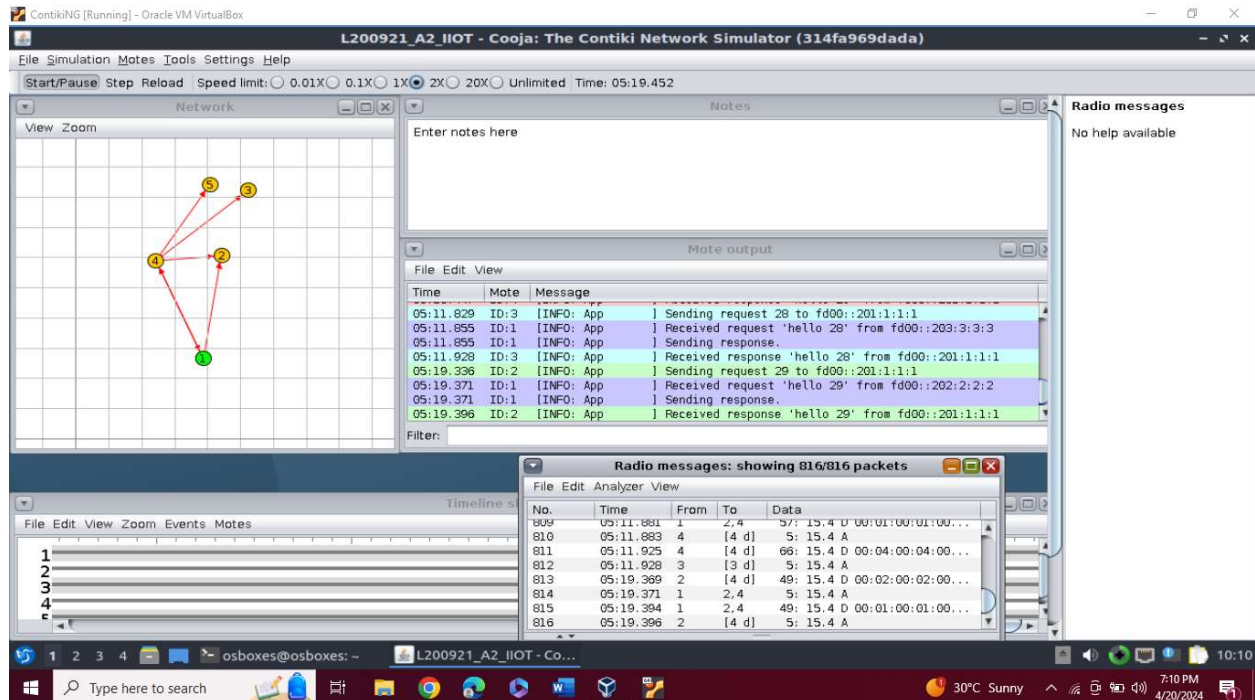

Fig 6: Simulation running at 20X
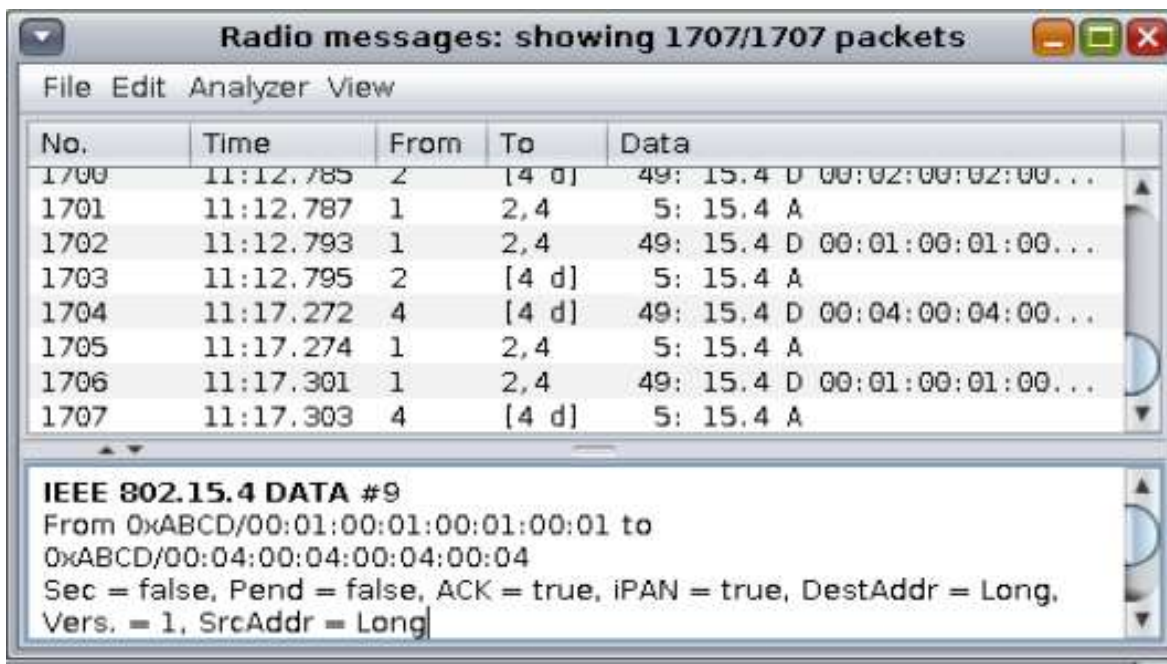
Fig 7: Simulation Running at 1X

Fig 8: Radio Messages during Simulation

- ## 3) ANALYSING RPL MESSAGES IN WIRESHARK

No.	Time	Source	Destination	Protocol	Length	Info
1 0.000000		fe80::201:1:1:1	ff02::1a	ICMPv6	97	RPL Control (DODAG Information Object)
2 3.063000		fe80::204:4:4:4	fe80::201:1:1:1	ICMPv6	102	RPL Control (DODAG Information Object)
4 3.164000		fe80::202:2:2:2	fe80::201:1:1:1	ICMPv6	102	RPL Control (DODAG Information Object)
6 3.986000		fe80::204:4:4:4	fe80::201:1:1:1	ICMPv6	102	RPL Control (DODAG Information Object)
8 4.647000		fe80::204:4:4:4	fe80::201:1:1:1	ICMPv6	102	RPL Control (DODAG Information Object)
10 4.800000		fe80::204:4:4:4	fe80::201:1:1:1	ICMPv6	102	RPL Control (DODAG Information Object)
12 5.774000		fe80::202:2:2:2	fe80::201:1:1:1	ICMPv6	102	RPL Control (DODAG Information Object)

Fig 9: DODAG Information Objects

Wireshark · Packet 1 · radiolog-1713622173325.pcap

```
─Rank: 128
> Flags: 0x08, Mode of Operation (MOP): Non-Storing Mode of Operation
─Destination Advertisement Trigger Sequence Number (DTSN): 240
─Flags: 0x00
─Reserved: 00
─DODAGID: fd00::201:1:1:1
> ICMPv6 RPL Option (DODAG configuration)
> ICMPv6 RPL Option (Prefix Information fd00::/64)
```
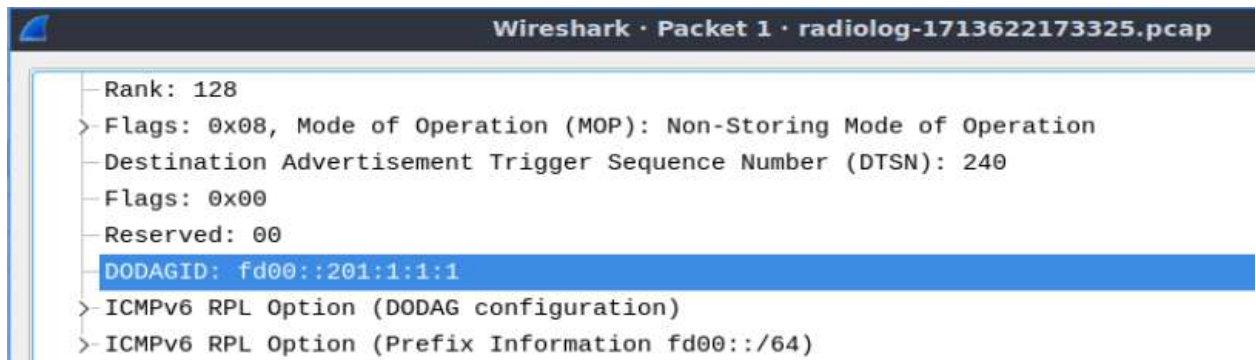
Fig 9.1 Observing Rank and DODAGID in Packet 1

As visible in fig 9, DODAG Information Objects are sent initially in the following order:

- o   node 1 to multicast address (with Rank 128, DODAGID: fd00::201:1:1:1)
- o   node 4 to node 1 (with Rank 65535, DODAGID: fd00::201:1:1:1)
- o   node 2 to node 1 (with Rank 65535, DODAGID: fd00::201:1:1:1)

Basically what is happening is that node 1 multicasts DODAG Information Objects after which nodes 4 and 2 that are actually in the range of server node 1, send DODAG Information Objects to Node 1 in order to inform the node 1 that I am part of a DODAG do you want to join? The DODAGID is fd00::201:1:1:1. Initially only root should send it and since here it was sent first by node 1 so node 1 is clearly the ROOT node.
These packets have to be constantly transmitted/broadcasted with no regular interval. Hence the duplicates are visible in Fig 9. Moreover, these are designed to be adaptive meaning, if network stable, frequency will be less else if too much disruptions going on like links breaking etc, the network is not considered stable and frequency of DIO sent is increased. Here the frequency is less showing that network is quite stable.

| 14 7.309000 | ::204:4:4:4 | ::201:1:1:1 | ICMPv6 | 85 RPL Control (Destination Advertisement Object) |
| 16 7.318000 | ::201:1:1:1 | ::204:4:4:4 | ICMPv6 | 43 RPL Control (Destination Advertisement Object Acknowled |

Fig 10: DAO and DAO ack between node 4 and node 1

After the exchange of DIOs, node 4 sends the Destination Advertisement Object to node 1 to basically ask "Can I join you as a child?". Node 1 replies with Destination Advertisement Object Acknowledgement confirming that node 1 is now parent of node 4. As visible in fig 10.

21 11.031000	fe80::204:4:4:4	ff02::1a	ICMPv6	97 RPL Control (DODAG Information Object)
22 11.186000	fe80::202:2:2:2	fe80::201:1:1:1	ICMPv6	102 RPL Control (DODAG Information Object)
24 11.846000	fe80::205:5:5:5	fe80::204:4:4:4	ICMPv6	102 RPL Control (DODAG Information Object)
26 11.874000	fe80::205:5:5:5	fe80::204:4:4:4	ICMPv6	102 RPL Control (DODAG Information Object)
28 14.159000	fe80::203:3:3:3	fe80::204:4:4:4	ICMPv6	102 RPL Control (DODAG Information Object)
30 14.336000	fe80::203:3:3:3	fe80::204:4:4:4	ICMPv6	102 RPL Control (DODAG Information Object)

Fig 11: DODAG Information Objects

Wireshark · Packet 21 · radiolog-1713622173332

```
Version: 240
Rank: 328
Flags: 0x08, Mode of Operation (MOP): Non-Storing Mode of Operation
Destination Advertisement Trigger Sequence Number (DTSN): 240
Flags: 0x00
Reserved: 00
DODAGID: fd00::201:1:1:1
ICMPv6 RPL Option (DODAG configuration)
```
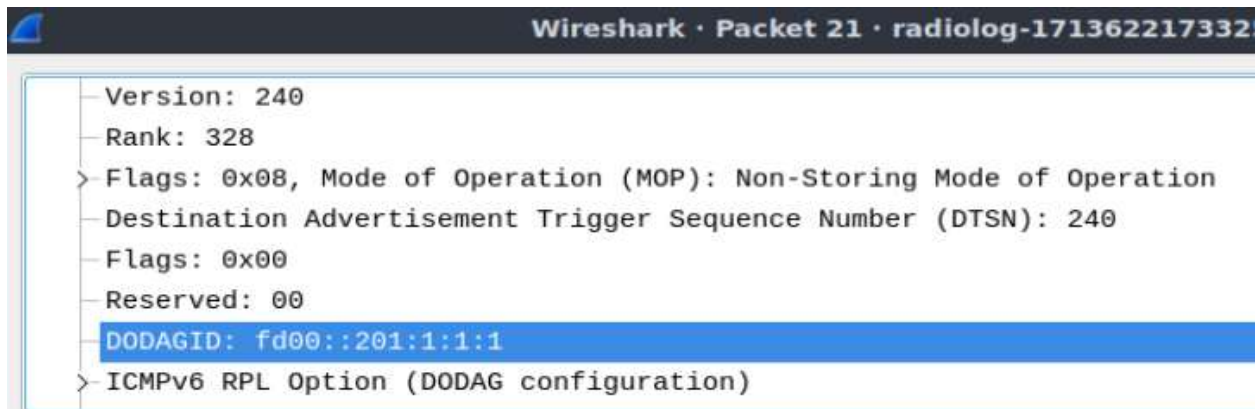
Fig 11.1 Observing Rank and DODAGID in Packet 21

After node 4 becomes child of root node (node 1), it multicasts a DODAG Information Object with rank 328 and DODAGID: fd00::201:1:1:1 as visible in fig 11 and fig 11.1. After this, a couple more of the DODAG Information Objects are observed to be sent by node not reachable before by node 1 (root node). These are sent by node 5 and Node 3 to node 4 with rank 65535 and DODAGID: fd00::201:1:1:1.

| 32 14.953000 | ::202:2:2:2 | ::201:1:1:1 | ICMPv6 | 85 RPL Control (Destination Advertisement Object) |
| 34 14.973000 | ::201:1:1:1 | ::202:2:2:2 | ICMPv6 | 43 RPL Control (Destination Advertisement Object Acknowle |

Fig 12: DAO and DAO ack between node 2 and node 1

After the exchange of another round of DIOs, node 2 sends the Destination Advertisement Object to node 1 to basically ask "Can I join you as a child?". Node 1 replies with Destination Advertisement Object Acknowledgement confirming that node 1 is now parent of node 2. As visible in fig 12.

| 45 18.484000 | fe80::202:2:2:2 | ff02::1a | ICMPv6 | 97 RPL Control (DODAG Information Object) |

Fig 13: DODAG Information Object Multicast by Node 2


```
RPLInstanceID: 0
Version: 240
Rank: 329
Flags: 0x08, Mode of Operation (MOP): Non-Storing Mode of Operatio
Destination Advertisement Trigger Sequence Number (DTSN): 240
Flags: 0x00
Reserved: 00
DODAGID: fd00::201:1:1:1
```
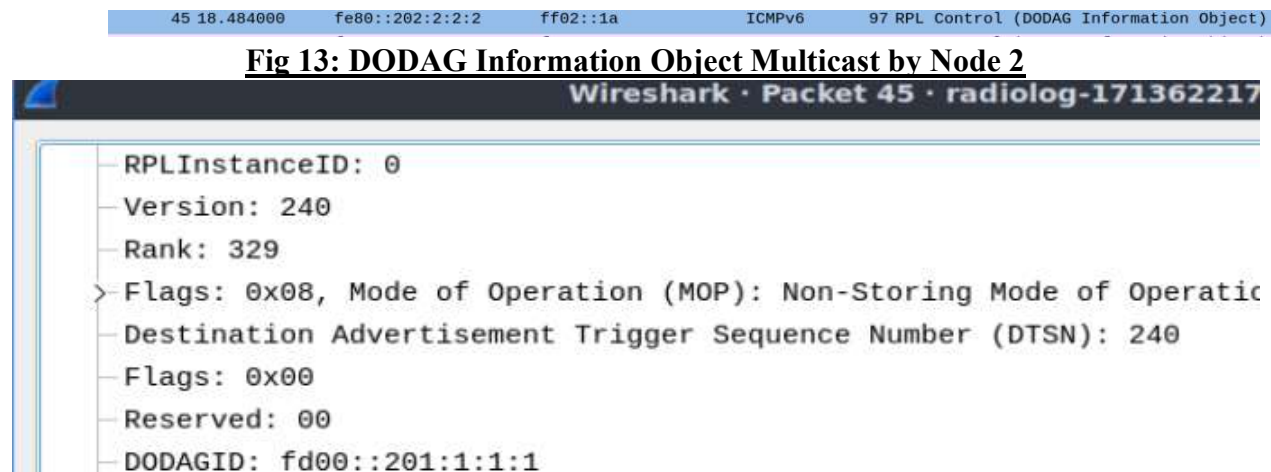
Fig 13.1 Observing Rank and DODAGID in Packet 21

After node 2 becomes child of root node (node 1), it multicasts a DODAG Information Object with rank 329 and DODAGID: fd00::201:1:1:1 as visible in fig 13 and 13.1.

| 52 22.374000 | ::205:5:5:5 | ::201:1:1:1 | ICMPv6 | 94 RPL Control (Destination Advertisement Object) |
| 54 22.410000 | ::201:1:1:1 | ::205:5:5:5 | ICMPv6 | 51 RPL Control (Destination Advertisement Object Acknowle |

Fig 14: DAO and DAO ack between node 5 and node 1

node 5 sends the Destination Advertisement Object to node 1 to basically ask "Can I join you as a child?". Node 1 replies with Destination Advertisement Object Acknowledgement confirming that node 1 is now parent of node 5 as well. As visible in fig 14.

66 23.662000	fe80::201:1:1:1	ff02::1a	ICMPv6	97 RPL Control (DODAG Information Object)
67 23.991000	::203:3:3:3	::201:1:1:1	ICMPv6	93 RPL Control (Destination Advertisement Object)
69 24.015000	::203:3:3:3	::201:1:1:1	ICMPv6	94 RPL Control (Destination Advertisement Object)
71 24.059000	::201:1:1:1	::203:3:3:3	ICMPv6	51 RPL Control (Destination Advertisement Object Acknowle
73 24.099000	::201:1:1:1	::203:3:3:3	ICMPv6	60 RPL Control (Destination Advertisement Object Acknowle

Fig 15: DIO, DAO and DAO Ack

The root node, node 1 is then observed to once again multicast DODAG Information Object with with Rank 128, DODAGID: fd00::201:1:1:1. After which node 3, which was not previously reachable by node 1, sends a Destination Advertisement Object to node 1 to basically ask "Can I join you as a child?". Node 1 replies with Destination Advertisement Object Acknowledgement confirming that node 1 is now parent of node 3 as well. As visible in fig 15. After this all nodes (2,3,4 & 5) have now become children of root node (node 1) and can continue to send DODAG Information objects to each other as they are now all part of the network.

This is how the DODAG construction takes place by first sending DODAG Information Objects by each neighboring nodes then DAO, and then finally DAO Ack by parent. In this simulation, I was not able to find any DIS Type packets indicating there wasn't any node that had to itself request for information and all announcements were made to every node.