
Comparing MAC Layer Implementations using Contiki-OS

Shantanoo Desai
prepared for:
Prof. Dr. Anna Förster

Sustainable Communication Networks
University of Bremen

November 20, 2015

Outline

Parameters under Observation

Required Firmware

Steps for Configuration

- Simulation for ContikiMAC

- Results for ContikiMAC Simulation

- Simulation for CXMAC

- Results for CX-MAC Simulation

Result

Contents of this section

Parameters under Observation

Parameters Under Observation

- Packet Reception Rate (PRR):Percent of received packets in an interval
 - related to throughput
 - if PRR is low, Packets dropped High
 - if PRR is low, overall throughput is low
- Average Duty-Cycle of sensor node
 - pertaining to Power Consumption of nodes
 - Lower the duty cycle less power consumed and longer lifetime of node
 - energy savings high when node is sleeping

Contents of this section

Required Firmware

Required Firmware

We use the example mentioned in the following folder:

examples/ipv6/rpl-udp

The folder contains:

- **udp-server.c**: A UDP server with RPL¹ abilities with IPv6 addressing
- **udp-client.c**: A UDP client with RPL abilities with IPv6 addressing
- **rpl-udp-powertrace.csc**: Cooja simulator file for powertracing feature
- **rpl-udp.csc**: Cooja simulator file to understand RPL based routing of UDP packets in network

¹Routing Protocol over Low-power Lossy Networks

Contents of this section

Steps for Configuration

- Simulation for ContikiMAC

- Results for ContikiMAC Simulation

- Simulation for CXMAC

- Results for CX-MAC Simulation

Steps

- For easy access of MAC and RDC (Radio Duty Cycling) layer, we make use of a **project-conf.h** file
- use the following (in the *ipv6/rpl-udp* folder):

```
$ gedit project-conf.h
```

Add the following configuration

- Channel Check Rate: 8 Hz
- MAC Layer: CSMA
- RDC Layer: ContikiMAC

```
#define NETSTACK_CONF_CHANNEL_CHECK_RATE 8
```

```
#define NETSTACK_CONF_MAC csma_driver
```

```
#define NETSTACK_CONF_RDC contikimac_driver
```


Steps

Do not forget to add the **project-conf.h** to the Makefile by adding this:

```
CFLAGS += -DPROJECT_CONF_H=\"project-conf.h\"
```

Previously mentioned configuration is **default** even if there is no **project-conf.h** mentioned in your work.

We use the available **rpl-udp-powertrace.csc** file for simulation.

- To make the Simulator run do the following the available .csc file:

```
make TARGET=cooja rpl-udp-powertrace.csc
```

Or you can create your own simulation for e.g. One UDP Sink and 10 UDP clients

Cooja Simulator view

The screenshot displays the Cooja Simulator interface for a network simulation. The main window is titled "Data collection network using IPv6 and RPL - Cooja: The Contiki Network Simulator".

Simulation control: Includes a "Run Speed limit" section with "Start", "Pause", "Step", and "Reload" buttons. Below these, it shows "Time: 00:00.000" and "Speed: ---".

Network: A central window showing a network topology with 30 nodes, each represented by a small circle with a number inside. The nodes are scattered across the network area.

Simulation script editor *active*: A window on the right containing a script for packet handling. The script includes the following code:

```
19 senderID = parseInt(msgArray[0]);
20 packetsReceived[senderID]++;
21
22 log.log("SenderID " + senderID + " PRR " + packetsRe
23 totalReceived = totalSent = 0;
24 for(i = serverID + 1; i <= nodeCount; i++) {
25     totalReceived += packetsReceived[i];
26     totalSent += packetsSent[i];
27 }
28 totalPPR = totalReceived / totalSent;
29 log.log("Total PRR " + totalPPR + " recv " + totalRe
30 } else if(msgArray.length == 6) {
31     // Sent packet
32     packetsSent[id]++;
33 }
34 }
35 }
36 }
```

Simulation for ContikiMAC

Before starting the simulation:

- click on **View** in Network window and choose **Radio Traffic** and **Mote Type** options
- on the Toolbar, click on **Mote Radio Duty Cycle** and adjust the window to view all the motes
- Start the simulation and keep it on for 5 minutes (NOTE: don't rely on the time in the Simulator, use your Watch)
- Check on **Mote Output** window and observe the starting commands (scroll all the way to the top to see what the nodes are configured to)
it should be the **same as your project-conf.h file**
- Observe the values collected in the Radio duty cycle and save it for comparison (use Screenshot tool in the Applications)

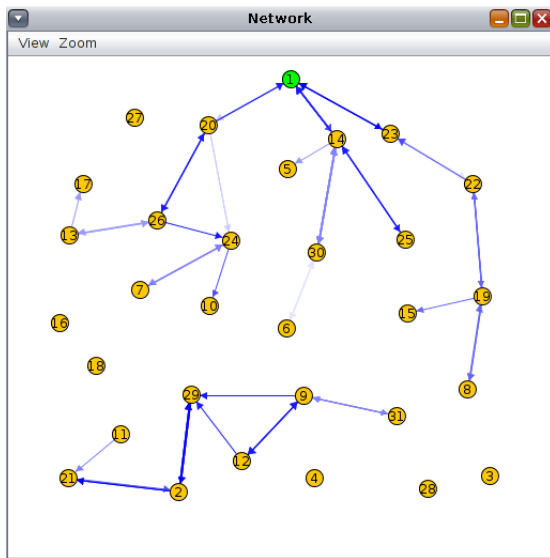
Mote Output for ContikiMAC Simulation

```
File Edit View
Mote output
Time Mote Message
00:00.784 ID:10 Rime started with address 0.18.116.10.0.10.10.10
00:00.792 ID:10 MAC 00:12:74:0a:00:0a:0a:0a Contiki 2.7 started. Node id is set to 10.
00:00.800 ID:10 CSMA ContikiMAC, channel check rate 8 Hz, radio channel 26
00:00.810 ID:10 Tentative link-local IPv6 address fe80:0000:0000:0000:0212:740a:000a:0a0a
00:00.812 ID:10 Starting 'UDP client process'
00:00.815 ID:10 UDP client process started
00:00.819 ID:10 Client IPv6 addresses: aaaa::212:740a:a:a0a
00:00.822 ID:10 fe80::212:740a:a:a0a
00:00.827 ID:10 Created a connection with the server :: local/remote port 8765/5678
00:00.866 ID:25 Rime started with address 0.18.116.25.0.25.25.25
Filter:
```

observe that all the motes are configured to:

- CSMA
- ContikiMAC
- Channel Check Rate 8 Hz
- Radio Channel number is 26

Network Topology for Simulation



Results for ContikiMAC

PowerTracker: 31 motes

Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	94.41%	1.21%	7.10%
Sky 2	5.20%	2.65%	0.27%
Sky 3	3.95%	1.61%	0.56%
Sky 4	3.48%	1.56%	0.22%
Sky 5	7.35%	1.92%	1.45%
Sky 6	9.36%	3.41%	1.39%
Sky 7	8.34%	2.47%	1.43%
Sky 8	6.34%	2.83%	0.64%
Sky 9	7.65%	3.13%	0.87%
Sky 10	4.63%	1.71%	0.51%
Sky 11	3.94%	1.87%	0.30%
Sky 12	4.16%	1.57%	0.51%
Sky 13	4.17%	1.98%	0.23%
Sky 14	10.94%	2.43%	2.32%
Sky 15	7.36%	2.84%	0.99%
Sky 16	4.34%	2.36%	0.12%
Sky 17	4.14%	1.81%	0.49%
Sky 18	5.33%	2.23%	0.54%
Sky 19	4.04%	1.79%	0.33%
Sky 20	6.96%	2.20%	1.11%
Sky 21	5.41%	2.35%	0.66%
Sky 22	5.88%	2.55%	0.47%
Sky 23	5.12%	1.45%	0.85%
Sky 24	8.52%	2.56%	1.60%
Sky 25	4.01%	1.74%	0.28%
Sky 26	6.88%	2.56%	0.86%
Sky 27	3.41%	1.51%	0.19%
Sky 28	3.81%	1.61%	0.27%
Sky 29	6.06%	2.25%	0.80%
Sky 30	13.12%	3.91%	2.31%
Sky 31	3.56%	1.59%	0.27%
AVERAGE	8.77%	2.18%	0.97%

Print to console/Copy to clipboard Reset

changing to CX-MAC

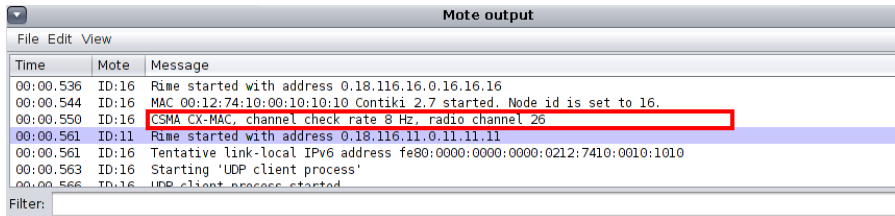
- before changing the RDC layer, close the current simulation and Cooja.
- do the following changes (via terminal):

```
#define NETSTACK_CONF_RDC  cxmac_driver
```

and compile the program once again

```
make TARGET=cooja rpl-udp-powertrace.csc
```
- follow the same step as previously mentioned and run the simulation

Mote Output for CXMAC



Mote output

File Edit View

Time	Mote	Message
00:00.536	ID:16	Rime started with address 0.18.116.16.0.16.16.16
00:00.544	ID:16	MAC 00:12:74:10:00:10:10:10:10 Contiki 2.7 started. Node id is set to 16.
00:00.550	ID:16	CSMA CX-MAC, channel check rate 8 Hz, radio channel 26
00:00.561	ID:11	Rime started with address 0.18.116.11.0.11.11.11
00:00.561	ID:16	Tentative link-local IPv6 address fe80:0000:0000:0000:0212:7410:0010:1010
00:00.563	ID:16	Starting 'UDP client process'
00:00.566	ID:16	UDP client process started

Filter:

Observe that all the motes are configured as:

- CSMA
- CX-MAC
- Channel Check Rate 8 Hz
- Radio Channel number is 26

To View the Nodes IEEE 802.15.4 Radio interface

- in the Network window, right-click on any Mote
- in the drop down menu go to **More Tools for Sky ..**
- click on **Mote Interface Viewer**
- in the window click on drop down menu on the top right corner
- select **IEEE 802.15.4 Radio**

This window will show you what the selected mote is doing (Listening or Idle) and signal strength

In the end compare the **Average values** from the Radio Duty Cycle window for both CX-MAC and ContikiMAC.

Simulation for CX-MAC

PowerTracker: 31 motes

Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	96.94%	0.58%	8.61%
Sky 2	13.47%	1.34%	1.13%
Sky 3	10.17%	0.94%	0.48%
Sky 4	12.97%	1.42%	1.31%
Sky 5	19.06%	3.20%	1.59%
Sky 6	19.97%	2.43%	2.35%
Sky 7	22.50%	1.68%	3.35%
Sky 8	14.38%	1.38%	1.41%
Sky 9	16.03%	2.05%	1.64%
Sky 10	20.63%	2.20%	2.56%
Sky 11	17.86%	2.22%	1.65%
Sky 12	15.96%	1.60%	1.55%
Sky 13	19.01%	2.67%	1.65%
Sky 14	20.14%	3.01%	1.90%
Sky 15	19.91%	2.40%	2.22%
Sky 16	14.53%	1.96%	0.90%
Sky 17	16.22%	2.27%	1.34%
Sky 18	20.73%	2.25%	1.90%
Sky 19	11.75%	1.34%	0.88%
Sky 20	16.41%	1.67%	1.83%
Sky 21	12.90%	1.79%	0.87%
Sky 22	22.45%	2.81%	1.10%
Sky 23	16.70%	1.43%	2.08%
Sky 24	23.58%	2.10%	2.89%
Sky 25	20.55%	2.06%	2.41%
Sky 26	22.93%	2.80%	2.94%
Sky 27	20.22%	2.56%	1.16%
Sky 28	10.90%	1.56%	0.59%
Sky 29	19.68%	1.54%	2.58%
Sky 30	20.65%	2.27%	2.71%
Sky 31	13.34%	1.56%	1.52%
AVERAGE	20.08%	1.97%	1.97%

Print to console/Copy to clipboard Reset

Contents of this section

Result

Changes and Result

- change the data transmission interval to 1 or any value in seconds in **udp-client.c** in the following section:

```
#ifndef  
#define PERIOD 1  
#endif
```

this will affect the PRR values and duty cycle values accordingly

- test a self created MAC protocol versus an already defined MAC layer (CXMAC or ContikiMAC)

RESULT: ContikiMAC is has better performance than the CX-MAC in terms of PRR and is more energy efficient for nodes (9 percent v/s 20 percent respectively).

The SINK node remains on for almost close to 100 percent since it keeps receiving packets from Clients.