

# Energy Saving Use-Cases Operation

## I. Energy saving under cell utilization xApp:

**This method optimizes energy consumption by monitoring PRB usage and shutting down underutilized cells when conditions are met.**

### Execution Steps:

#### 1. Run FlexRIC

```
ola@PF3A12C6:~$ ./RIC_TaaP/flexric/build/examples/ric/nearRT-RIC
[UTIL]: Setting the config -c file to /usr/local/etc/flexric/flexric.conf
[UTIL]: Setting path -p for the shared libraries to /usr/local/lib/flexric/
[NEAR-RIC]: nearRT-RIC IP Address = 127.0.0.1, PORT = 36421
[NEAR-RIC]: Initializing
[NEAR-RIC]: Loading SM ID = 146 with def = TC_STATS_V0
[NEAR-RIC]: Loading SM ID = 145 with def = SLICE_STATS_V0
[NEAR-RIC]: Loading SM ID = 148 with def = GTP_STATS_V0
[NEAR-RIC]: Loading SM ID = 2 with def = ORAN-E2SM-KPM
[NEAR-RIC]: Loading SM ID = 3 with def = ORAN-E2SM-RC
[NEAR-RIC]: Loading SM ID = 143 with def = RLC_STATS_V0
[NEAR-RIC]: Loading SM ID = 142 with def = MAC_STATS_V0
[NEAR-RIC]: Loading SM ID = 144 with def = PDCP_STATS_V0
[iApp]: Initializing ...
[iApp]: nearRT-RIC IP Address = 127.0.0.1, PORT = 36422
[NEAR-RIC]: Initializing Task Manager with 2 threads
```

**2. Select the energy-saving under cell utilization scenario** in RIC-TaaP Studio or manually configure the parameters (any configuration can be used to observe xApp operation).

Hide simulation setup

Show AI policy manager

Connect to FlexRIC: (If true, FlexRIC need to be running)
☒

hoSnrDifference (dB):
3

IndicationPeriodicity (s):
0.1

simTime (s):
1000

e2TermIp:127.0.0.1

KPM\_E2functionID:2
RC\_E2functionID:3
N\_MmWaveEnbNodes:4
N\_Ues:15

CenterFrequency (Hz):3.5e9

Bandwidth (Hz):20e6

IntersideDistanceUEs (m):1000

IntersideDistanceCells (m):500

Scenario:
scratch/Energy\_saving\_with\_cell\_utilization\_scenario.cc

Start new

Stop

Reset

Scenario flags:☒

Status: off

Warning: If there are no Cells/UEs on the grid after max 30s, please check if ns3 is running properly typing in terminal "ps -a" or "cat ns-3-mmwave-ora-ns3\_run.log"

Cell energy states management:

ES status: off

Turn ES xApp on

Turn ES xApp off

Observe simulation grid

Observe KPIs

Observe ES

4,000

3,000

2,000

1,000

0

0

1000

2000

3000

4000

5000

6000

3. Start the simulator and verify that the scenario has been initialized.



4. Launch the xApp by executing the following command:

```
./flexric/build/examples/xApp/c/ctrl/energy_saving_with_CU
```

```
[xApp]: E42 RIC SUBSCRIPTION REQUEST tx RAN_FUNC_ID 2 RIC_REQ_ID 1
[xApp]: SUBSCRIPTION RESPONSE rx
[xApp]: Successfully subscribed to RAN_FUNC_ID 2
[xApp]: E42 RIC SUBSCRIPTION REQUEST tx RAN_FUNC_ID 2 RIC_REQ_ID 2
[xApp]: SUBSCRIPTION RESPONSE rx
[xApp]: Successfully subscribed to RAN_FUNC_ID 2
[xApp]: E42 RIC SUBSCRIPTION REQUEST tx RAN_FUNC_ID 2 RIC_REQ_ID 3
[xApp]: SUBSCRIPTION RESPONSE rx
[xApp]: Successfully subscribed to RAN_FUNC_ID 2
[xApp]: E42 RIC SUBSCRIPTION REQUEST tx RAN_FUNC_ID 2 RIC_REQ_ID 4
[xApp]: SUBSCRIPTION RESPONSE rx
[xApp]: Successfully subscribed to RAN_FUNC_ID 2
[xApp]: E42 RIC SUBSCRIPTION REQUEST tx RAN_FUNC_ID 2 RIC_REQ_ID 5
[xApp]: SUBSCRIPTION RESPONSE rx
[xApp]: Successfully subscribed to RAN_FUNC_ID 2
Waiting for KPM measurements...
```

5. The energy-saving xApp monitors **PRB usage** for each mmWave node within a defined sample window (MAX\_PRB\_HISTORY in mmwave-enb-net-device.h).

To monitor logs, use: tail -f ns3.log

```
+1.319428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Current PRB Value: 68.2746 History Size: 3/5
+1.319428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Not enough points yet, returning -1
+1.319428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Current PRB Value: 19.9443 History Size: 3/5
+1.319428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Not enough points yet, returning -1
+1.319428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Current PRB Value: 85.6215 History Size: 3/5
+1.319428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Not enough points yet, returning -1
+1.319428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Current PRB Value: 36.9202 History Size: 3/5
+1.319428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Not enough points yet, returning -1
```

6. If any cell meets the PRB threshold (default **40%**), the simulator starts reporting the **SINR map** for that cell. In this scenario, **Cell 2** meets the condition with **34.71% PRB usage**.

```
+1.619428571s 1 MmWaveEnbNetDevice:BuildAndSendReportMessage(): [DEBUG] Send NR DU
+1.619428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Current PRB Value: 0 History Size: 5/5
+1.619428571s 1 MmWaveEnbNetDevice:CalculatePrbAverage(): [DEBUG] Returning PRB Average: 34.731
+1.619428571s 1 MmWaveEnbNetDevice:CheckReportingFlag(): [DEBUG] Current PRB Average: 34.731 Threshold: 40 Should Report: 1
```

7. The xApp analyzes the SINR map based on a specific number of RIC\_INDICATION events.

```

=== Target Cell Selection for UE 15 ===
Current Cell: 2
No suitable target cell found (SINR > -10 dB required)
Serving Cell 2 - UE 13: 19.00 dB
Updated neighbor cell 3 for UE 13: SINR -1.50 (sample 2/5)
Updated neighbor cell 5 for UE 13: SINR -3.50 (sample 2/5)
Serving Cell 2 - UE 15: 2.00 dB
Updated neighbor cell 5 for UE 15: SINR -1.00 (sample 2/5)
Updated neighbor cell 4 for UE 15: SINR -4.00 (sample 2/5)

```

8. If a target cell meets the **MIN\_SINR** condition, the xApp initiates **handover commands**.

```

[xApp]: data.toTargetCell= 3 ..
[xApp]: Send Handover Control message to move IMSI 13 from cellId 2 to target cellId 3
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 0
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 1
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 2
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 3
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 4

```

```

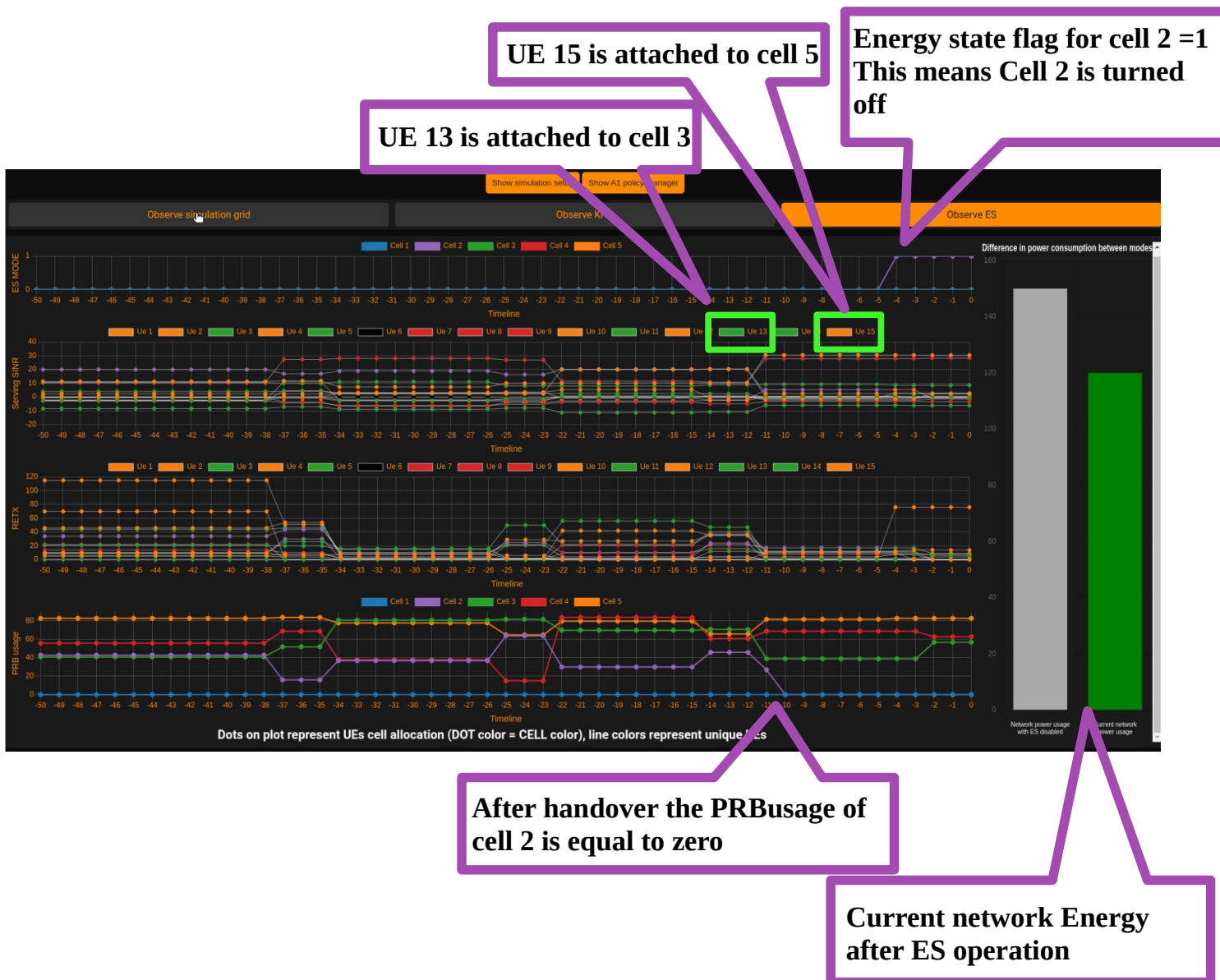
Current Cell: 2
Evaluating Cell 5: SINR -0.60 dB
Found better cell: 5 (SINR: -0.60 dB)
Selected Target Cell: 5 (SINR: -0.60 dB)
[xApp]: data.toTargetCell= 5 ..
[xApp]: Send Handover Control message to move IMSI 15 from cellId 2 to target cellId 5
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 0
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 1
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 2
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 3
[xApp]: CONTROL-REQUEST tx
[xApp]: CONTROL ACK rx
[xApp]: Successfully received CONTROL-ACK
[xApp]: Handover request sent successfully to node 4

```

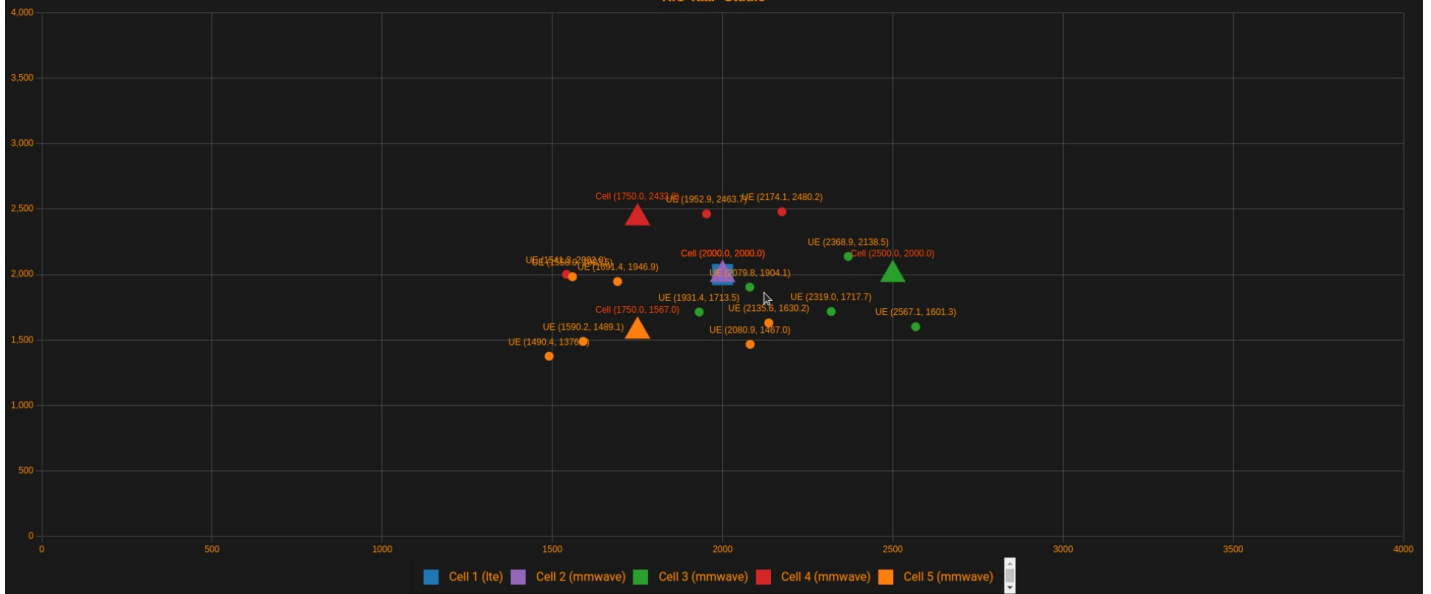
## 9 - Observe the handover operation in the GUI

UE 15 is moved to Cell 5.

- UE 13 is moved to Cell 3.
- Cell 2 PRB usage drops to zero.
- Energy state flag for Cell 2 = 1, indicating that Cell 2 has been powered down.
- Current network power is reduced after energy-saving operations.



RIC TaaP Studio





## II- Energy saving with load balancing:

This approach focuses on optimizing network performance while minimizing energy consumption through intelligent resource allocation and dynamic load redistribution.

Let us demonstrate how to simulate this scenario using our RIC TaaP tester:

1. Ensure that FlexRIC is running in the background.
2. Set the simulation parameters as follows and press **Start new** :

The screenshot shows the RIC TaaP tester interface with several parameters set. Annotations highlight specific values:

- IntersideDistanceUEs = 700** (Red box)
- N\_UEs = 10** (Yellow box)
- IntersideDistanceCells = 500** (Green box)

The interface includes a top section for simulation setup with buttons like 'Hide simulation setup', 'Show AI policy manager', and 'Start new'. Below this is a 'Cell energy states management' section with buttons for 'Observe simulation grid', 'Observe KPIs', and 'Observe ES'. The bottom section displays two graphs: 'ES MODE' and 'Serving SINR', both plotted against 'Timeline'.

After the nodes and UEs are running and appear in the simulator, we will observe the following:

**Cell 2 is connected to 4 UEs**



UE's 3,8,9 and 10 are attached to cell 2

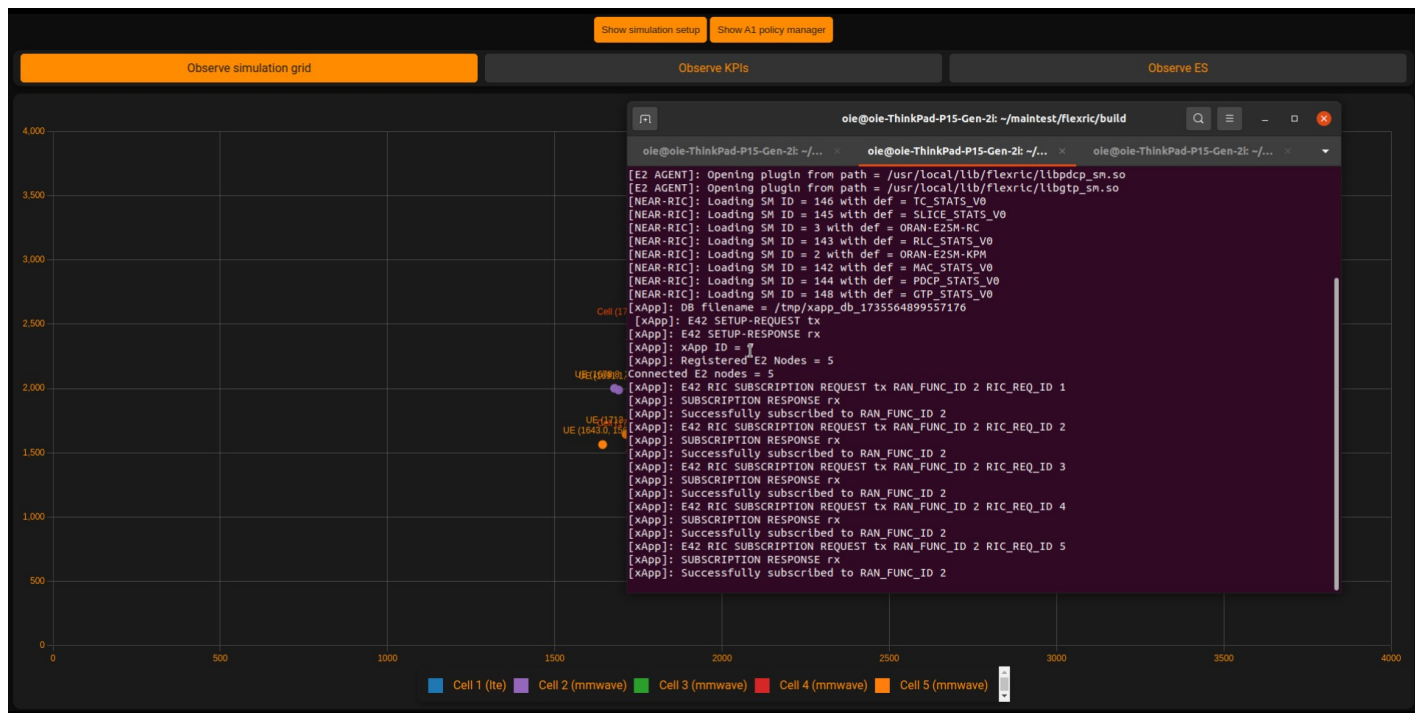


Cell 2 has the biggest PRB usage

3. Run the xapp\_energy\_saving\_with\_LB by executing the following command in the terminal:



./flexric/build/example/xapp/c/ctrl/xapp\_energy\_saving\_with\_LB



## What is the action of the xApp?

The xApp sends a RIC control message to Cell 2, instructing it to move the UEs attached to it to the nearest cells and then power it down. Let us demonstrate this action in the following :

**UE 10 is attached to cell 5**

**UE 9 is attached to cell 3**

**UE 8 is attached to cell 4**

**UE 3 is attached to cell 5**

**Energy state flag for cell 2 =1**  
**This means Cell 2 is turned off**



**After handover the PRUsage of cell 2 is equal to zero**

### Current network power after Energy saving operation

