**3GPP TSG-RAN WG3 Meeting #114bis-e *R3-221446***

**17-26 Jan 2022, E-meeting**

**Title:** TP to 37.817 on AI/ML based load balancing

**Source:** ZTE

**Agenda item:** 18.4.2

**Document Type:** pCR

# 1. Introduction

This TP tries to reflect agreement on the solution of AI/ML-based load balancing from CB # AIRAN3\_LB.

# 5. Reference

1. R3-221435, SoD\_CB # AIRAN3\_LB

# Annex – TP for TR 37.817

//////////////////////////////////////////////////////////////irrelevant operations skipped/////////////////////////////////////////////////////////////////////

## 5.2 Load Balancing

### 5.2.1 Use case description

The rapid traffic growth and multiple frequency bands utilized in a commercial network make it challenging to steer the traffic in a balanced distribution. To address the problem, load balancing had been proposed. The objective of load balancing is to distribute load evenly among cells and among areas of cells, or to transfer part of the traffic from congested cells or from congested areas of cells, or to offload users from one cell, cell area, carrier or RAT to improve network performance. This can be done by means of optimization of handover parameters and handover actions. The automation of such optimisation can provide high quality user experience, while simultaneously improving the system capacity and also to minimize human intervention in the network management and optimization tasks.

However, the optimization of the load balancing is not an easy task as follows:

* Currently the load balancing decisions relying on the current/past-state cell load status are insufficient. The traffic load and resource status of the network changes rapidly, especially in the scenarios with high-mobility and large number of connections, which may lead to ping-pong handover between different cells, cell overload and degradation of user service quality.
* It is difficult to guarantee the overall network and service performance when performing load balancing. For the load balancing, the UEs in the congested cell may be offloaded to the target cell, by means of handover procedure or adapting handover configuration. For example, if the UEs with time-varying traffic load are offloaded to the target cell, the target cell may be overloaded with new-arrival heavy traffic. It is difficult to determine whether the service performance after the offloading action meets the desired targets.

To deal with the above issues, solutions based on AI/ML model could be introduced to improve the load balancing performance. Based on collection of various measurements and feedbacks from UEs and network nodes, historical data, etc. AI/ML model-based solutions and predicted load could improve load balancing performance, in order to provide higher quality user experience and to improve the system capacity.

### 5.2.2 Solutions and standard impacts

*Editor Note: Capture the solutions for the use case, including potential standard impacts on existing Nodes, functions, and interfaces*

#### 5.2.2.1 Locations for AI/ML Model Training and AI/ML Model Inference

The following solutions can be considered for supporting AI/ML-based load balancing:

* AI/ML Model Training is located in the OAM and AI/ML Model Inference is located in the gNB.
* AI/ML Model Training and AI/ML Model Inference are both located in the gNB.

In case of CU-DU split architecture, the following solutions are possible:

* AI/ML Model Training is located in the OAM and AI/ML Model Inference is located in the gNB-CU.
* AI/ML Model Training and Model Inference are both located in the gNB-CU.

Note: gNB is also allowed to continue model training based on AI/ML model trained in the OAM.

Other possible locations of the AI/ML Model Inference are FFS.

#### 5.2.2.2 AI/ML Model Training in OAM and AI/ML Model Inference in a NG-RAN node

A high-level signalling flow for the AI/ML use case related to Load Balancing with Model Training in OAM and Model Inference in NG-RAN is shown in Figure 5.2.2-1 below.



Figure 5.2.2-1 Model Training at OAM, Model Inference at NG-RAN

Step 0: NG-RAN node 2 is assumed to have an AI/ML model optionally, which can provide NG-RAN node 1 with useful input information, such as predicted resource status, etc.

Step 1: The NG-RAN node 1 requests the UE to provide measurements and/or location information (e.g., RRM measurements, MDT measurements, velocity, position).

Step 2: The UE collects and reports to NG-RAN node 1 requested measurements and/or location information (e.g., UE measurements related to RSRP, RSRQ, SINR of serving cell and neighbouring cells, velocity, position).

Step 3: NG-RAN node 1 further sends UE measurement reports together with other input data for Model Training to OAM. NG-RAN node 2 also sends input data for Model Training to OAM.

Step 4: AI/ML Model Training is located at OAM. The required measurements are leveraged to train the AI/ML models for load balancing.

Step 5: OAM deploys/updates AI/ML model into the NG-RAN node(s). The NG-RAN node is allowed to continue model training based on the received AI/ML model from OAM.

Note: This step is out of RAN3 Rel-17 scope.

Step 6: The UE collects and reports to NG-RAN node 1 requested measurements or location information.

Step 7: The NG-RAN node 1 receives from the neighbouring NG-RAN node 2 the input information for load balancing model inference.

Step 8: NG-RAN node 1 performs Mobility Load Balancing predictions (e.g. for cells of NG-RAN node 1).

Step 9. The NG-RAN 1 sends the model performance feedback to OAM if applicable.

Note: This step is out of RAN3 scope.

Step 10: NG-RAN nod 1 executes Mobility Load Balancing actions and UEs are moved from NG-RAN node 1 to NG-RAN node 2.

Step 11: NG-RAN node 1 and NG-RAN node 2 send feedback information to OAM.

#### 5.2.2.3 AI/ML Model Training and AI/ML Model Inference in a NG-RAN node

A high-level signalling flow for the AI/ML use case related to Load Balancing with Model Training and Model Inference in a NG-RAN node is shown in Figure 5.2.2-2 below.



Figure 5.2.2-2: Model Training and Model Inference in a NG-RAN node

Step 0: NG-RAN node 2 is assumed to have an AI/ML model optionally, which can provide NG-RAN node 1 with useful input information, such as predicted resource status, etc.

Step 1: The NG-RAN node 1 requests UE to provide measurements and/or location information(e.g., RRM measurements, MDT measurements, velocity, position).

Step 2: The UE collects and reports to NG-RAN node 1 the requested measurements and/or location information (e.g., UE measurements related to RSRP, RSRQ, SINR of serving cell and neighbouring cells, velocity, position).

Step 3: The NG-RAN node 1 requests the neighbouring NG-RAN node 2 the input information for load balancing model training.

Step 4: The NG-RAN node 1 receives from the neighbouring NG-RAN node 2 the input information for load balancing model training.

Step 5: An AI/ML Model Training is located at NG-RAN node 1. The required measurements are leveraged to train the AI/ML model.

Steps 6: NG-RAN node 1 receives UE measurements and/or location information.

Step 7: NG-RAN node 1 can receive from the neighbouring NG-RAN node 2 the input information for load balancing model inference.

Step 8: NG-RAN node 1 performs Mobility Load Balancing predictions (e.g., for cells of NG-RAN node 1).

Step 9: NG-RAN node 1 takes Mobility Load Balancing decision and UEs are moved from NG-RAN node 1 to NG-RAN node 2.Step 10: NG-RAN node 2 sends feedback information to NG-RAN node 1 (e.g. resource status updates after load balancing, etc).

5.2.2.4 Input of AI/ML-based Load Balancing

To predict the optimized load balancing decisions, NG-RAN may need following information as input data for AI/ML-based load balancing:

From the local node:

* Own resource status information (e.g. per cell, per SSB Area): e.g., this can be calculated using predictions of some or all of the resource information specified in current XnAP
* Predicted own resource status information: e.g., this can be calculated using predictions of some or all of the resource information specified in current XnAP
* UE trajectory prediction

From the UE:

* UE location information (e.g., coordinates, serving cell ID, moving velocity) interpreted by gNB implementation when available
* UE Radio Measurements, e.g., RSRP, RSRQ, SINR
* UE Mobility History Information

From neighbour NG-RAN Nodes:

* Neighbour resource status information (e.g. per cell, per SSB Area): it may include, e.g., some or all of the resource information in current Xn: Resource Status Update procedure
* Predicted neighbour resource status information: this can be calculated using, e.g., predictions of some or all of the resource information specified in current XnAP
* UE performance measurement at traffic offloaded neighbour cell

Editor’s Note: FFS other input information required for AI/ML-based load balancing.

5.2.2.5 Output of AI/ML-based Load Balancing

AI/ML-based load balancing model can generate following information as output:

* Selection of target cell for mobility load balancing
* Predicted own resource status information: this can be calculated using, e.g., predictions of some or all of the resource information specified in current XnAP
* Predicted resource status information signalled from neighbor NG-RAN node(s): this can be calculated using, e.g., predictions of some or all of the resource information specified in current XnAP
* Validity time for the Model Inference output predictions. FFS whether validity time is applied to all outputs produced by the Model Inference function.
* The predicted UE(s) selected to be handed over to target NG-RAN node (will be used by RAN node internally)

Editor’s Note: FFS other output information expected from AI/ML-based load balancing.

5.2.2.6 Feedback of AI/ML-based Load Balancing

To optimize the performance of AI/ML-based load balancing model, following feedback can be considered to be collected from NG-RAN nodes:

* UE performance information from target NG-RAN (for those UEs handed over from the source NG-RAN node)
* Resource status information updates from target NG-RAN
* System KPIs (e.g., throughput, delay, RLF of current and neighbours)

Editor’s Note: FFS other feedback expected from AI/ML-based load balancing

5.2.2.7 Standard impacts

To improve the load balancing decisions at a gNB (gNB-CU), a gNB can request load predictions from a neighbouring node. Details of the procedure are FFS.

If existing UE measurements are needed by a gNB for AI/ML-based load balancing, RAN3 shall reuse the existing framework (including MDT and RRM measurements). FFS on whether new UE measurements are needed.

To increase the awareness of the traffic dynamics and enable more improved traffic steering decisions it is possible to complement load measurements currently exposed over RAN interfaces with information related to predicted load from neighbouring RAN nodes as well as UE measurements and information.

* An NG-RAN node can also predict its own load. This can be achieved by considering the own load and load information received from neighbour RAN nodes. Load predictions can be signalled between RAN nodes.
* An NG-RAN node can also derive load prediction using UE measurements and information, for example MDT and RRM measurements, or UE location information (e.g. velocity, position). For the aspects concerning the configuration and the reporting of UE measurements and information the impacted protocol is RRC. RAN2 needs to be consulted for details during the normative phase.

Signalling of information used to derive Model Inference outputs may be achieved over the Xn interface by reusing existing or new procedures. The details are to be discussed during normative work.

**Potential interface impacts:**

* (FFS) MDT/RRM enhancement in order to collect consecutive UE information.
* New or enhanced existing signaling procedure to request/retrieve predicted resource status information from neighbouring nodes via Xn interface.
* New or enhanced existing signaling procedure to request/retrieve predicted load balancing strategy information from neighbouring nodes via Xn interface.
* New or enhanced existing procedure to request/retrieve feedback information via Xn interface.

-----------------------------------End of Changes-----------------------------------