3GPP TSG-RAN WG4 Meeting #116 R4-2511887

Bengaluru, India, August 25th – 29th, 2025

Agenda Item: 7.18.1

Source: *CAICT, NTU, Ericsson, Qualcomm, APPLE, Huawei, Hisilicon, OPPO, CATT, CMCC, NTT DOCOMO, INC., Vivo, Nokia, Xiaomi, Mediatek, Rohde & Schwarz, Samsung, Intel, ZTE Corporation, Sanechips,* *Korea Testing Laboratory*

Title: Proposed update for TR 38.843 with RAN4 part

Document for: Discussion

# 1 Introduction

In this contribution, we provide update for TR 38.843 with RAN4 part with the agreements till RAN4#116.

# 2 Text Proposal

## ------------------------------------------- Change --------------------------------------------------------3.1 Terms

For the purposes of the present document, the terms given in TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

…….

**AI/ML model delivery:** A generic term referring to delivery of an AI/ML model from one entity to another entity in any manner. Note: An entity could mean a network node/function (e.g., gNB, LMF, etc.), UE, proprietary server, etc.

**AI/ML model Inference:**  A process of using a trained AI/ML model to produce a set of outputs based on a set of inputs.

**AI/ML model testing:** A subprocess of training, to evaluate the performance of a final AI/ML model using a dataset different from one used for model training and validation. Differently from AI/ML model validation, testing does not assume subsequent tuning of the model.

NOTE: the term is not applicable for performance/conformance testing.

**……….**

**AI/ML model transfer:** Delivery of an AI/ML model over the air interface in a manner that is not transparent to 3GPP signalling, either parameters of a model structure known at the receiving end or a new model with parameters. Delivery may contain a full model or a partial model.

**AI/ML model validation:** A subprocess of training, to evaluate the quality of an AI/ML model using a dataset different from one used for model training, that helps selecting model parameters that generalize beyond the dataset used for model training.

NOTE: the term is not applicable for performance/conformance testing.

**Data collection:** A process of collecting data by the network nodes, management entity, or UE for the purpose of AI/ML model training, data analytics and inference.

……..

**Functionality activation:** A process of enabling an applicable functionality to perform inference.

**Functionality deactivation**: A process of network deactivating an active functionality.

……

## 7.4 Interoperability and testability aspects

### 7.4.1 Introduction

In this section, the study of requirements and testing frameworks to validate AI/ML based performance enhancements and ensuring that UE and gNB with AI/ML meet or exceed the existing minimum requirements, if applicable, are documented.

The need and implications for AI/ML processing capabilities definition is considered.

### 7.4.2 Common framework

#### 7.4.2.1 General

The general requirements and testing frameworks for AI/ML based performance enhancements mainly focus on

* how to define requirements and tests for inference
* evaluate feasibility and necessity of requirements/tests for LCM
* requirements for data collection (in particular for training) could/need be defined

Requirements/tests for training will not be studied unless training procedures are defined. The design of test should ensure performance is guaranteed and avoid that a UE can pass the test but perform poorly in the field.

The testing goal is to verify whether the minimum performance of AI/ML functionality/feature can be achieved. LCM would also be tested.

#### 7.4.2.2 Principles on the definition of requirements

For the definition of AI/ML requirements, the following cases related to legacy performance should be considered

* For the cases with the existing legacy performance
  + Take the legacy performance as baseline for existing use cases/procedures/functionalities /measurements that are to be enhanced by AI/ML based methods
    - Further study may be needed on what is baseline performance in conditions different to the requirement condition but within the expected range of operation.
  + New or enhanced performance requirements/tests could be considered for existing use cases/procedures/functionalities/measurements that are to be enhanced by AI/ML based methods
* For the cases without the existing legacy performance
  + New performance requirements/tests could be considered for the use cases/procedures/functionalities/measurements that are carried out or are to be enhanced by AI/ML based methods

The following procedure can be considered for defining core requirements

* Performance monitoring procedure, including performance evaluation and decision-making procedure for AI/ML functionalities/models
* Functionality/Model management procedure, including functionality/model selection/activation/deactivation, and functionality/model switching/fallback/transfer/delivery/update
* Latency/interruption requirement for above procedures

The following LCM related requirements can be considered:

* Model/Functionality select/switch/activate/deactivate/fallback
* Model/Functionality monitoring
* On whether requirements for data collection (in particular for training) could/need be defined:
  + Data collection requirements would only be defined if data collection procedure is defined in 3GPP specifications.
* On requirements for model transfer/update:
  + Requirements would only be defined if model transfer/update would be defined in 3GPP specifications.

Legacy RRM requirements (non-AI/ML) are applicable to the corresponding legacy RRM procedures even during the AI/ML operation mode and RAN4 to assess what RRM requirements are needed for each case. The legacy framework for RRC/MAC-CE/DCI based core requirements (e.g., define delay requirements based on multiple delay components) can be used as the baseline for LCM procedures if the LCM related requirements are agreed to be introduced. If new procedures which legacy framework is not applicable to are introduced, additional core requirement framework can be discussed.

LCM related tests should consider how the framework can address the possibility of updates/activation/deactivation /switching to the functionalities/models after the deployment of the devices in the field.

Both static and non-static scenarios/configurations could be needed for AI testing.

* Static: channel model and SNR settings are fixed and do not change over the test, specific channel realizations may be dynamic
* Non-static: Non-static scenarios/configuration can be further considered in application to use cases.

How to use them, including whether to use static scenarios/configurations as baseline need to be discussed and decided case by case. Static scenarios are assumed by default. Non-static scenarios are introduced on a use-case specific basis if needed for testing.

#### 7.4.2.3 Reference block diagrams for testing

Reference block diagrams provide test modules/functionalities of TE/DUT and testing framework for different use cases. Both reference block diagrams for 1-sided model and 2-sided model are studied.

##### 7.4.2.3.1 Reference block diagram for 1-sided model

Figure 7.4.2.3.1-1 provides the reference block diagram for 1-sided model. LCM in the figure includes functionality and/or model ID based LCM. The link between TE and DUT are physical and not logical. The logical link will depend on the functionality being tested. The scope of the figure includes both performance and potentially LCM tests. Offline training is assumed and some blocks may not be used in some of the tests. LCM may not be tested depending on the purpose of the test.

LCM

Verification

AI/ML functions

Test configuration/controller

Signal generator

inference

LCM

DUT

TE

Figure 7.2.3.1-1 Reference block diagram for 1-sided model

##### 7.4.2.3.2 Reference block diagram for 2-sided model

Figure 7.4.2.3.1-2 provides the reference block diagram for 2-sided model. LCM in the figure includes functionality and/or model ID based LCM. The link between TE and DUT are physical and not logical. The logical link will depend on the functionality being tested. The scope of the figure includes both performance and potentially LCM tests. Offline training is assumed and some blocks may not be used in some of the tests. LCM may not be tested depending on the purpose of the test.

LCM

Verification

AI/ML functions

Test configuration/controller

Signal generator

inference

LCM

DUT

TE

inference

Figure 7.2.3.1-2 Reference block diagram for 2-sided model

#### 7.4.2.4 Test encoder/decoder for 2-sided model

(NOTE: At the current stage the framework in this session applies to CSI compression case.)

In order to determine the test encoder/decoder, the following issues are considered:

* Common assumptions for proposals of the test decoder / encoder (and the paired encoder/ decoder) for tester
* The need for and potential definition and derivation procedure of intermediate KPI for decoder evaluation and selection
* Data collection/generation for decoder evaluation, and the common assumptions/environment needed for data collection/generation
* How to minimize the impact of possible variations/differences in the test decoder/ test encoder design/implementation on UE/ gNB performance verification
* The impact of test decoder/ encoder for testing complexity to UE/gNB performance verification, and the advantage/disadvantage analysis of high/low complexity decoders.

The test decoder/encoder design should take into account complexity limitations based on e.g., feasibility of TE implementation and complexity levels considered feasible by network vendors/UE vendors for decoder/encoder deployment.

The choice of test decoder/encoder should aim as much as possible to avoid limiting the implementation choices, including e.g. complexity, back-bone model etc, of UE/gNB encoders/decoders operating in the field (this principle may not be fully achievable in practice).

Specification on the test may include some high-level parameters for the test decoder/encoder (e.g. parameters related to processing complexity, model structure, etc).

Following the above principles, the considered options of test decoder are listed below

* Option 1: DUT provides the decoder
* Option 2: Infra vendor provides the decoder
* Option 3: Full decoder specification in standard
* Option 4: TE vendor provides the decoder

Option 3 target is that a single decoder defined in the specifications for at least a single test for any DUTs.

For option 4, the following aspects should be considered

* TE vendor should be able to develop the decoder based on the specifications
* Test repeatability should be ensured (variation among TE vendor implementations should be bound)
* Other vendors should also be able to develop such a decoder and which can deliver similar performance
* Interoperability should be ensured based on the parameters that need to be specified
  + Parameters that need to be specified are FFS
* Candidate parameters/conditions that may be considered for defining test decoder include
  + Training data set for TE decoder training
  + Model structure (Activation function is included in the model structure)
  + Performance parameters for the TE decoder (e.g. cosine similarity, loss function, etc)
  + Maximum FLOPs allowed for the test decoder
  + Maximum number/size of model parameters
  + Compression ratio of decoder (output size/input size)
  + Quantization level
  + Other parameters are not precluded and to be further discussed.
  + Note: Feasibility of definition of parameters needs further investigated.

Option 4 target is that a single decoder implemented by each TE vendor will be enough for at least a single test for any DUTs. TE vendor should be able to implement the test decoder for Option 4 without any involvement from another party. If this is found infeasible, another option in which TE vendors need to collaborate with DUT/infra vendors to implement the decoder could be considered.

Further clarifications and analysis of the four options of test decoder are included in table 7.4.2.3-1. It is assumed that for Option 4 the TE vendors can implement the decoder just based on the specifications (no other party involved). The table would need to be revised if collaboration between TE vendor and DUT/infra vendor is needed.

**Table 7.4.2.4-1 Comparison of the four options of test decoder**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Option 1** | **Option 2** | **Option 3** | **Option 4** |
| **Clarification of options** | | | | |
| Source of the test decoder | DUT vendor | Decoder vendor (infra vendor in case of testing UEs) | RAN4 specifications | TE vendor, decoder developed based on RAN4 specifications |
| Source of decoder training data | Up to DUT vendor (no need to be specified) | Up to decoder implementer (infra vendor)  FFS whether coordination with encoder vendor is required | Not needed, decoder fully specified (used as part of the RAN4 procedure to specify the decoder) | FFS  Could be specified depending on how Option 4 will be defined |
| DUT vendor knowledge of the test decoder | Full knowledge | No or partial or enough or full knowledge based on alignment with infra vendors or specifications | Full knowledge based on the specifications | Partial knowledge – based on the RAN4 specification |
| Supported training collaboration type between DUT and decoder provider (source of training data should be consistent with the collaboration type) |  |  |  |  |
| Test decoder performance verification procedure at TE | Need to ensure that decoder performance is not degraded (as intended by the decoder provider) on the TE | - Need to ensure that decoder performance is not degraded (as intended by the decoder provider) on the TE  - Need to ensure that decoder performance is good enough to enable a DUT that meets the minimum requirements to pass the test | Not needed as long as the standardized model implementation can be similar enough between TE vendors | Not needed as long as the model implementation can be similar enough between TE vendors |
| Feasibility of test decoder verification procedure | FFS | FFS | FFS | FFS |
| **Pros/Cons analysis** | | | | |
| Reflection on the real deployment (likelihood that test decoder would be used in actual field deployments ) |  |  |  |  |
| TE requirements to deploy the decoder (e.g. training, complexity, interoperability) | Higher than Option 3/4 in terms of that maybe more than one decoder are implemented by TE  Lower than Option 3/4 in terms of that no training at TE is required | Higher than Option 3/4 in terms of that maybe more than one decoder are implemented by TE  Lower than Option 3/4 in terms of that no training at TE is required | Lower complexity than Option 1/2 in terms of that only one decoder is implemented by TE  Lower than Option 4 in terms of that no training at TE is required | Lower complexity than Option 1/2 in terms of that only one decoder is implemented by TE  Higher than Option 3 in terms of that training at TE is required  Note: How to ensure compatibility/interoperability between TE and DUT needs further study. |
| Specification Effort (defining test decoder and requirements) | Low | Low | Highest  RAN4 needs to standardize the entire decoder | High  RAN4 needs study and decide on what to standardize |
| Confidentiality/ IP issues in the testing procedure(after specs are published) |  |  | No | No |
| Applicability to different scenarios/conditions/ configurations |  |  |  |  |
| Complexity of testing for the ecosystem | Testing the encoder at DUT  Higher than Option 3/4  Need for interaction between TE vendor | Testing the encoder at DUT  Higher than Option 3/4  Testing complexity higher also than option 1. | Testing the encoder at DUT  Low – no need for interaction between TE vendors and other parties | Testing the encoder at DUT  Low – no need for interaction between TE vendors and other parties |
| Complexity of verifying/testing the test decoder | Higher than option 3/4  FFS compared to option 2 | Higher than Option 3/4  FFS compared to Option 1 | Low | Low |
| Complexity of deploying for the ecosystem |  |  |  |  |
| Friendly to STOA(state of the art) model test / Forward compatibility when new AI models are invented |  |  |  |  |
| Relationship with reference decoder/encoder(used by RAN4 to define the performance requirements) for defining requirement |  |  |  |  |
| Whether model transfer/delivery is needed during the test procedure |  |  |  |  |

##### 7.4.2.4.1 Terms for feasibility study

The following terms are defined for the feasibility study of test encoder/decoder for 2-sided model

**Reference decoder:** A decoder used in RAN4 discussions at least for simulation alignment/requirement derivation and/or verification of the decoder implemented by the TE.

**Reference encoder:** An encoder used in RAN4 discussions at least for simulation alignment/requirement derivation, test decoder derivation and/or test decoder verification.

**Own encoder/decoder**: A decoder/encoder trained by the individual companies (in contrast of the frozen encoder/decoder and test decoder).

##### 7.4.2.4.2 Evaluation assumptions for option 3 and 4 feasibility study

Table 7.4.2.4-2 provides the agreed parameters for verifying the feasibility of aligning model among companies and additional parameters of the encoder/decoder and feedback are given in Table 7.4.2.4-3.

**Table 7.4.2.4-2 System-level simulation parameters for** **verifying the feasibility of aligning model**

|  |  |  |
| --- | --- | --- |
| Parameter | | Value |
| Duplex, Waveform | | FDD, OFDM |
| Multiple access | | OFDMA |
| Scenario | | Dense Urban (Macro only) |
| Frequency Range | | FR1 only, [2GHz, 4GHz] |
| Inter-BS distance | | 200m |
| Channel model | | According to TR 38.901 |
| Antenna setup and port layouts at gNB | | Companies need to report which option(s) are used between  - 32 ports: (8,8,2,1,1,2,8), (dH,dV) = (0.5, 0.8)λ |
| Antenna setup and port layouts at UE | | 4RX: (1,2,2,1,1,1,2), (dH,dV) = (0.5, 0.5)λ for (rank 1-4) |
| BS Tx power | | 44dBm for 20MHz |
| BS antenna height | | 25m |
| UE antenna height & gain | | Follow TR36.873 |
| UE receiver noise figure | | 9dB |
| Numerology | Slot/non-slot | 14 OFDM symbol slot |
| SCS | Baseline: 15kHz for 2GHz;  Optional: 30kHz for 4GHz |
| Simulation bandwidth | | Baseline: 10 MHz for 15kHz  Optional: 20 MHz for 30kHz |
| Frame structure | | Slot Format 0 (all downlink) for all slots |
| MIMO scheme | | SU-MIMO |
| MIMO layers | | Baseline: 1  Optional: 2 |
| CSI feedback | | Feedback assumption at least for baseline scheme  - CSI feedback periodicity (full CSI feedback): 5 ms (baseline)  - Scheduling delay (from CSI feedback to time to apply in scheduling): 4 ms |
| Traffic load (Resource utilization) | | Baseline: 50%  Optional: 20/70% |
| UE distribution | | CSI compression: 80% indoor (3 km/h), 20% outdoor (30 km/h) |
| UE receiver | | MMSE-IRC as the baseline receiver |
| Feedback assumption | | Realistic |
| Channel estimation | | Realistic or ideal channel estimation |

**Table 7.4.2.4-3 additional parameters of the encoder/decoder and feedback**

|  |  |  |
| --- | --- | --- |
| **Assumptions** | | **Value** |
| **CSI generation part** | **AI/ML model backbone** | MLP, CNN, Transformer |
| **Pre-processing** | SVD to get channel eigenvectors |
| **CSI reconstruction part** | **AI/ML model backbone** | [MLP, CNN, Transformer] |
| **Common description** | **Input type** | Eigenvectors of channel matrix |
| **Output type** | Eigenvectors of channel matrix |
| **Quantization /dequantization method** | Scalar quantization |
| **Dataset description** |
| **Ratio between testing and training dataset** | 10% of the training dataset as baseline |
| **Ground-truth CSI quantization method (including scalar/codebook based quantization, and the parameters)** | Floating point (float 32 for real data and imaginary data) |
| **Other parameters** | **Laten/reporting size** | 64bits |

***Logistical issues:***

For AI/ML model formats:

* During the study, ONNX format was used for sharing.
* Some Companies also shared PyTorch models
* The models in different formats from one company were the same
* Use ONNX version v1.16 or later
* ONNX version and opset version number were included in the file

For dataset format

* NumPy was used for dataset sharing
* se pickled data(compression mechanism) was not used
* npy – single array was used in each file

Dataset/model input file format:

* N (samples) X 2 (IQ) X nSB (number of subbands) X nPorts (number of CSI-RS ports) X nLayers (number of layers)
  + - * + In other words, dataset contains the following info:

1st dimension: Number of samples

2nd dimension: Real and imaginary

3rd dimension: Number of sub-bands

4th dimension: Number of antenna ports

5th dimension: Number of layers – present only if >1 layer

Note: Each element of the dataset will be a float32 real number

The dataset included eigenvectors calculated based on average covariance channel matrix for the given MIMO layer for each sub-band based on pre-processing assumptions in WF R4-2414447

Single file for fixed scalar quantization (only dataset),

Dataset files could be split into multiple files to enable easier upload.

* Use the 2 digits for split files starting from 00, increment for each additional file

Split files and then archive each file.

For file naming scheme

* Folders for AI/ML data sharing and current WI/use case were created under “RAN4 folder”
* Subfolder created for each meeting

File naming scheme (ML model file and dataset file)

* a unique identifier for the company (4 characters – list to be maintained by RAN4 secretary)
* meeting number
* differentiate model and dataset by identifier (2 characters ml and ds)
* additional identifier with 2 characters
  + - * + ec for encoder
        + dc for decoder
        + ei for encoder input
        + do for decoder output
        + lt for latent (decoder input or encoder output)
        + others to be added as needed
* dataset could be split in multiple files – 2 digits
* files were compressed to zips and uploaded

Files to be shared, were decided based on option under study

*  Dataset containing encoder input per subband
*  Encoder and/or decoder model

##### 7.4.2.4.3 Feasibility study of Option 3

The basic steps for the feasibility study of Option 3 are as follows:

* Step 1: Based on the test decoder and reference encoder AI/ML model structure, individual companies jointly train the reference encoder and test decoder based on a dataset. Check on performance alignment -> see simulation results from contributing companies
  + The dataset used for jointly training could be a company specific training dataset or a mixed training dataset
  + repeat simulations until good alignment is achieved
  + move to next step after alignment
* Step 2: Share models (encoder or decoder or both)/datasets (training/testing/inference)
* Step 3: Select one or more decoder for further analysis (called the frozen decoder in the following text)
* Step 4: Each company brings results for training of “own encoder” with selected decoder(s)
  + Check/discuss performance alignment
  + use “own” data or data shared by other companies or mixed test dataset
* Step 5: Conclude on overall feasibility of Option 3
  + consider the conditions under which Option 3 is feasible if found feasible

It should be noted that parameters agreed are just for the feasibility study of testing options and if/when RAN4 discusses requirement definition, RAN4 will define a new test decoder which may or may not reuse any of the parameters agreed in the feasibility study.

Test decoder derivation procedure:

companies bring encoder + decoder set based on agreed parameters. RAN4 chooses one of the decoders and interested companies further check if an encoder can be trained with this decoder to obtain similar performance/complexity (or other evaluation criteria)

For decoder(s) selection in step 3, Choose the decoders with Low, medium, high SGCS among the decoders submitted by companies. Continue the feasibility study on the two tracks below:

Track 1(Decoder trained over company own dataset): Select 3 decoders, which are trained in step-1 with companies’ own training dataset. Companies train own encoders (with own Eigenvector dataset) and check performance against decoders (with at least own test dataset)

* The 3 decoders were selected based on low, medium, high mean SGCS out of available decoders
* At least an encoder with the agreed model structure should be considered when making own encoder. Optionally companies can try own encoder with their preferred structure. Companies should report the structure if different.

Track 2(Decoder trained over mixed dataset): Create a mixed dataset, companies train decoders based on the mixed dataset in Step-1. One or more decoder selected and companies develop encoders and check encoder performance with own or the mixed dataset checked against decoder(s).

* At least an encoder with the agreed model structure should be considered when making own encoder. Optionally companies can try own encoder with their preferred structure. Companies should report the structure if different.

For track 1, decoders with 2 bit quantization, and with no quantization are provided

* No quantization aware training when model is unquantized
* Quantization aware training when model is quantized
* For the model with quantization:
  + Agree on a scaling and quantization codebook
  + Include Sigmoid in model file in the encoder, and inverse Sigmoid in the decoder
  + Codebook (1/8, 3/8, 5/8, 7/8) in model file
  + Do not including mapping to 2 bits. Quantizer function of converting to 2 bits should not be in the model file
* For the model without quantization
  + For no quantization models, do not include sigmoid at the model output and no inverse Sigmoid in the decoder

For the input / output dimensionality of track 1,

* define Encoder Input and Decoder Output as data dimension (n, 2, 13, 32) with data type float32 where n is the dynamic batch size, and we have 2 for I/Q, 13 sub-bands, 32 Tx ports.
* define Encoder Output and Decoder Input as data dimension (n, 32) with data type float
* No additional inputs

Take mixed dataset comprising Mediatek, Ericsson, Vivo, Oppo, Nokia.

CSI\_Feasibility Table1 in attached Spreadsheets provide simulation results for the feasibility study of option 3.

Preliminary observations for Dataset

* Datasets consist of separate training data and separate test data.
* All companies have produced datasets based on the same simulation assumptions. However, differences can be observed between performance with different companies’ datasets.
* The “mixed dataset” provides much more consistent results than training or testing using different individual company datasets
* The exercise demonstrates that in order to agree on test decoder(s) and or reference encoder(s), some effort is needed to align dataset(s).

The focus in this study has been on convergence between SGCS reported for different company UE encoders when operating with test decoders, in order to understand whether it is feasible to specify a test decoder (or, for option 4, information used to create a test decoder) that can be used for consistent performance testing across multiple UE vendor implementations. Absolute SGCS has not been considered, since the assumed model structure is somewhat arbitrary and the purpose of the study has been to consider testing.

**Observations for option 3 track 1**

* It is possible for UE vendors to train an encoder based on another company’s frozen test decoder that is functional
* The performance of the encoder is very dependent on the dataset used for training the encoder and the dataset used for training the frozen decoder, and the dataset
  + If the encoder is trained using a dataset of the UE vendor, the UE vendor dataset should align to the dataset used to develop the test decoder.
* The variation in the performance depending on the training datasets and test datasets is as much as 30-50% in SGCS. Further details are available in the results spreadsheet.
* For option 3, track 1 to be useful, more careful attention would be needed to the alignment of datasets
* Only one encoder structure was considered for option 3 track 1. Other lower complexity encoder structures were not examined.

**Observations for option 3 track 2**

* There is reasonable SGCS convergence when companies jointly train an encoder-decoder pair using the mixed dataset and test using the mixed test dataset
  + Around +-2% variation in SGCS; see results spreadsheet for more details
* It is possible for UE vendors to train an encoder based on a frozen test decoder that is functional
* When training “own” encoder with the frozen decoder, with all training (frozen decoder and encoder) and testing using the mixed dataset, there is good convergence in average SGCS results
  + Around - 2% to +1.4% variation in SGCS from the average
* When training “own” encoder with the frozen decoder, with training of the frozen decoder using the mixed dataset, own encoder using own dataset and testing using the mixed dataset, there is convergence in average SGCS results from results presented so far
  + Around -+-4% variation in SGCS from the average (+-1.5% with outlier performance excluded)
* When trained “own” encoder with the frozen decoder, with training of the frozen decoder using the mixed dataset, own encoder using own dataset and testing using the own dataset, there is much less convergence in average SGCS results from results presented so far
  + Around -40% to 13% variation in SGCS from the average (+-8% to 5% with outlier performance excluded)
* The results considering “own” datasets demonstrate that there is divergence in the datasets, and attention must be paid to alignment of the training datasets.
  + Note: Alignment of datasets does not refer to aligning UE algorithms such as channel estimation, CSI estimation etc.

**Observations from lower complexity encoder evaluations:**

* It is important to differentiate in this evaluation between the impacts of different encoder backbones and the impacts of different complexity levels
* The following observations are valid with the mixed dataset; there can be dataset dependency for other datasets
* 1-1 joint training:
  + For 1-1 joint training, some, although not all, companies observed that the low complexity MLP encoder (Encoder 4 as defined in [R4-2508084]) had a reduced performance compared to the other encoders.
* Training based on the selected decoder
  + When trained using the Samsung Frozen decoder, many, although not all, companies observed that the MLP encoder had a loss compared to the other encoders
  + One company saw SGCS loss for all of the low complexity encoders (Transformer, CNN and MLP) when trained with the Samsung frozen encoder.
  + Based on the majority company results, it might be concluded that the lower complexity transformer and CNN encoders are compatible with the Samsung frozen decoder (which was trained using the agreed “high complexity” encoder), but the MLP may not be.
* Training based on N-1 decoder (as defined in [R4-2508084])
  + It is proposed to name training based on the N-1 decoder as “N-to-1 Multi-encoder structure aware training”
  + With N-to-1 Multi-encoder structure aware training, most companies observed that the performance of the MLP low complexity encoder could improve relative to the other encoders (depending on company, possibly still a slight loss)
  + One company observed that if lower complexity CNN encoders are used with N-1 joint training then the SGCS becomes lower for the low complexity CNN encoders.
  + One company observed that the choice of anchor and method used to train the N-to-1 Multi-encoder structure aware training may impact the performance and capability with different encoders.
* Summary conclusion:
  + If a high complexity test decoder is used, it seems to be compatible with different encoder backbones (at least CNN and transformer)
  + Whether a low complexity decoder can be compatible with different encoder structures was not studied
  + The benefits of N-to-1 Multi-encoder structure aware training are not fully clear, however it may improve the performance of the lowest complexity MLP encode

***Potential specification impact***

For option 3, for each test, a test decoder needs to be captured in the specs, a reference encoder may be needed to derive the test decoder and/or requirements and/or to validate the test decoder implementation in the TE. the same decoder may be used in multiple tests or each test could have a difference decoder. In option 3, the reference decoder is the test decoder. The training dataset used to train the test decoder in 3GPP should be captured.

The need to define a reference encoder was discussed for the test decoder for option 3 and it is agreed that

* To verify the performance of CSI compression test decoder for calibration of test equipment, agree to define the reference encoder including encoder structure and parameters in RAN4
  + For option 3 the reference encoder should have been joint trained with the test decoder.

whether to specify it in TS or TR, and capture the conclusion in the TR conclusion part could be decided during a normative phase

* + The reference encoder is the encoder that is jointly trained with the decoder
  + At a normative phase, 3GPP would decide whether RAN4 or RAN5 will specify it
* The encoder that corresponds to the standardized test decoder is stored within 3GPP
  + Whether in a TR, or in any TS, whether in RAN4 or RAN5, or whether simply kept in a 3GPP database (with no reference from TR or TS) would be decided in a normative phase

##### 7.4.2.4.4 Feasibility study of Option 4

The feasibility of two option 4 sub-options was studied:

* Option 4a (Dataset based): specify at least dataset based on which the test decoder can be implemented by TE vendors
  + Potentially other things may need to be specified (e.g. model structure for decoder or a reference encoder)
* Option 4b (Encoder based): encoder is documented in the specifications, used to derive a decoder to be implemented by TE vendors
  + The training dataset (channel information) may or may not also need to be specified

The basic step for the feasibility study of Option 4a was as follows

* Step 1-3: Reuse results of Option 3
* Step 4: Select one or more Eigenvector dataset(s) for further analysis based option 3
  + Selection criteria: select the dataset(s) generated from the selected encoder and decoder pair(s) from Option 3 track 1
  + Or mixed dataset from track 2
  + If multiple datasets are selected (as was the case of Option 3 track 1), the subsequent steps of the feasibility procedure will be applied on each one of the selected datasets separately
* Step 5: Label selected dataset with encoder input/output using encoder corresponding to the selected test decoder from option 3.
* Step 6: Companies bring results for training of “own encoder(s) and decoder(s)” with selected dataset(s)
  + Performance alignment to be checked/discussed
* Step 7: Conclude on overall feasibility of Option 4a
  + Dataset for feasibility evaluation to be discussed; may be common test dataset or own test dataset]

The basic step for the feasibility study of Option 4b was as follows:

* Step 1-3: Reuse results of Option 3
* Step 4: Select one or more encoder(s) for further analysis based on option 3
  + Selection criteria: select the encoder(s) from the selected encoder and decoder pair(s) of Option 3 track 1
  + Or encoder(s) from the selected encoder and decoder pair(s) of Option 3 track 2
* Step 5: Company brings results for training of “own decoder(s)” with selected encoder(s)
  + Aligned and own datasets were both considered during the evaluation
* Step 6: Company trains “own encoder(s)” with “own decoder(s)” from step 5
* Aligned and own datasets were both considered during the evaluation Step 7: Conclude on overall feasibility of Option 4b
  + Dataset for feasibility evaluation to be discussed; may be common test dataset or own test dataset]

Encoder(s) in step 5 at least have the agreed structure. Companies were allowed to bring analysis/results with other encoders (using different structures)

CSI\_Feasibility Table2 in attached spreadsheets provides simulation results for the feasibility study of option 4.

**Observations for option 4**

* For option 4b: It is feasible to train a functional “own” decoder based on the frozen encoder, and then a functional “own” encoder based on the “own” decoder when the mixed dataset is used
* For option 4a: For the mixed dataset, a similar observation to option 4b can be expected (in this case, the labelled dataset would be the same latent space as the selected encoder (created using joint training) acting on the mixed dataset, and so option 4a converges to option 4b)
* If the frozen encoder is created using the mixed dataset, TE/UE vendors create decoders using the mixed dataset, UE vendors create an encoder using the mixed dataset and testing uses the mixed test dataset then good convergence is seen in SGCS
  + Around +- 1.5% from average SGCS
  + UE encoders were shown to be interoperable with all companies own decoders
  + This represents the situation where there is a well-defined dataset, and UE vendors use the same dataset to train their encoder models
* Some companies observed that, if
  + the frozen encoder is created using the mixed dataset,
  + TE/UE vendors create decoders using the mixed dataset,
  + UE vendors create an encoder using their own dataset and
  + testing uses the **mixed** test dataset

then good convergence is seen in SGCS, but there is more variation than when mixed dataset is used for training the encoder

* + Around +- 3% from average SGCS
  + UE encoders were shown to be interoperable with all companies own decoders
  + This represents the situation where there is a well-defined dataset that is used for all 3GPP responsibilities (Creating the frozen decoder, providing information for the TE vendor to train the decoder, providing test data), but the UE vendor uses their own dataset to create the encoder
* Some companies observed that, if
  + the frozen encoder is created using the mixed dataset,
  + TE/UE vendors create decoders using the mixed dataset,
  + UE vendors create an encoder using their own dataset and
  + testing uses the **own** test dataset

then convergence is seen in SGCS, but there is more variation than when mixed dataset is used for training the encoder. The variation is similar between using the mixed dataset and own dataset for testing.

* + Around +- 1.6 to +4.7% from average SGCS
  + UE encoders were shown to be interoperable with all companies own decoders
  + This represents the situation where there is a well-defined dataset that is used for all 3GPP responsibilities (Creating the frozen decoder, providing information for the TE vendor to train the decoder, providing test data), but the UE vendor uses their own simulation to create the encoder
* Two lower complexity encoders were considered and tested based on the frozen decoder.
  + A transformer with 3.63 MFlops
  + A CNN encoder with 4.14 MFlops
* When the lower complexity encoders were examined using the mixed dataset for all training and the mixed test set for testing, SGCS convergence (- 1 to 3.5%) and interoperability with all companies’ “own” decoders were observed for the considered structures.

***Potential specification impact***

For option 4a, the following aspects should be considered for specification:

* Dataset is specified for option 4a.
* Assume reference decoder is captured.
* Reference encoder should be captured
* The study only considered the case of a common assumption on model structure for the “own” test decoder. This corresponds to standardized model structure.
  + Structure refers to backbone, numbers of layers, type of layers and all description of the model, but not the parameters.
  + Interoperability in the case that the structure of the TE decoder is not specified has not been investigated

For option 4b, the following aspects should be considered for specification:

* Assume reference decoder is captured.
* Specify dataset to train frozen encoder and decoder
  + TE vendor shall use the dataset to train the test decoder
  + This is the same dataset as used to train the frozen encoder
* The study only considered the case of a common assumption on model structure for the “own” test decoder. This corresponds to standardized model structure.
  + Structure refers to backbone, numbers of layers, type of layers and all description of the model, but not the parameters.
  + Interoperability in the case that the structure of the TE decoder is not specified has not been investigated

##### 7.4.2.4.5 Recommendations for WI

Encoder / Decoder selection criteria: Achievable performance, complexity and robustness in different conditions need to be taken into account in selecting the model.

Performance requirements: Agree a minimum performance requirement level per set of side conditions during Rel-20 WI.

Options comparison: Option 4 can work as long as the test decoder structure is fully specified and the same dataset is used by all TE vendors and 3GPP. Some validation of the TE decoder may be needed**.**

#### 7.4.2.5 Data collection/generation for testing

Different generating methods of test dataset can be used for different tests. The following candidate methods are to be considered:

* Dataset based on TR 38.901, e.g. UMa channel, UMi channel, CDL channel, “legacy approach”, etc.
  + “Legacy approach” refers legacy test in which a channel model is used
* Field dataset (data collected directly from field measurements)
* TE generates dataset for test based on assumptions/parameters defined by RAN4 (e.g. by defining some rules/function to generate data)
* Other methods are not precluded

Synthetic channels are used as baseline, and check whether it can be used for the individual use case. An analysis for each use case to determine the reliability of using synthetic channels for test data in evaluating models trained on real data may be conducted and the field data can be considered for the analysis.

#### 7.4.2.6 Data collection/generation for training

Some conditions and/or accuracy requirements for the training dataset or training data generation could only be introduced if the training procedure is defined in 3GPP specifications.

#### 7.4.2.7 Generalization/scalability aspects

The necessity and feasibility of defining requirements or test to verify the generalization of AI/ML is studied.

The goals of generalization test are to verify whether the minimum level of performance of AI/ML functionality/model can be achieved/maintain under the identified scenarios and/or configurations, while the performance won’t be significantly degraded in other scenarios and/or configurations. The following aspects should be considered for generalization/scalability related testing:

* details about the scenarios and/or configurations for test and the corresponding AI/ML models/functionality
* what the minimum level performance for each identified scenario and/or configuration is
* what the significant degradation for other scenarios and/or configurations is

It should also be considered that generalization and/or scalability related requirements for different scenarios/ configurations can be implicitly handled in the test case definition.

The high level and general guidelines for the generalization test are provided as follow. The exact decisions are to be made case by case.

* define requirements for each AI/ML functionality
* Define one test per UE capability as a minimum
  + The possibility more than one test per UE capability could be considered depending on capability definitions]
* Define a minimum set of test configurations (including NW sided conditions if any) as mandatory for testing of AI/ML-enabled Feature

As for the handling of generalization tests, the following option is considered as baseline:

Signaling based LCM procedures and performance monitoring are considered in dedicated test cases and are excluded in tests verifying generalization. RAN4 may define multiple tests with different conditions. In each of the test, TE configures the same specified UE configuration, and therefore the same specified UE configuration is tested under different conditions to verify its generalizability. (environment differs in each test but not changing dynamically during the test)

* + Specified UE configuration includes functionality and/or model ID if defined

There shall be consistency between applicable conditions signalling and testing. No test will be defined that implies a change of network condition or associated ID during the test. Whether a mixed dataset can be created for testing generalization, and whether such a mixed dataset would be a static or non-static scenario should be considered.

Potential areas to consider for generalization testing includes:

* gNB array parameters
  + port layouts, array size, antenna virtualization
* propagation conditions
* Deployment scenarios
  + Carrier frequencies
  + Speeds
  + Indoor/outdoor
  + Bandwidth
* SNR

#### 7.4.2.8 AI/ML processing capability

The practical processing capability and implementation complexity for device under test should be assumed when specifying RAN4 requirements.

* The UE capability may be needed to handle different complexity for one side and two-side models.
* The complexity of UE should also be studied when making assumption on gNB side model, and vice versa.

#### 7.4.2.9 Post deployment handling

When operating in the field, two aspects of UE operation may impact performance:

* Update or fine tuning or addition/removal of models, if applicable, resulting in a change of functionality which may impact functionality performance
* Data drift / mismatch between the conditions encountered by the UE in the field and the training data, which may impact functionality performance.

For dealing with drift / mismatch between the conditions encountered by the UE in the field and the training data for the model, which may impact functionality performance, monitoring is needed.

* + Performance monitoring will be designed in other groups
  + RAN4 may consider the need and feasibility of requirements and tests to ensure consistency and accuracy of monitoring metrics or other monitoring related data sent from the UE, and set requirements as feasible/needed.
  + Monitoring can be used for managing fallback, change in functionality, model update/model switching/model transfer, if applicable

For dealing with potential changes in the performance of functionalities, two options may be available. The options are not mutually exclusive:

* Option 1 (Post-deployment pre-activation functionality/configuration update testing): Conduct the validation of a change in AI model/functionality before its deployment/activation in already deployed UEs
  + Validation takes into account the UE hardware in which the model is to be deployed/activated.
  + Other aspects not precluded.
* Option 2 (Post-deployment post-activation functionality testing based on performance monitoring): Using performance monitoring and LCM procedures
  + Performance monitoring will be designed in other groups
  + RAN4 may consider the need and feasibility of requirements and tests to ensure consistency and accuracy of monitoring metrics or other monitoring related data sent from the UE, and set requirements as feasible/needed.

Option2 is prioritized. Reliable and testable monitoring procedures can be established for all cases.

### 7.4.3 CSI feedback enhancement

Both time domain CSI prediction and spatial-frequency domain CSI compression are considered.

PMI reporting framework (follow PMI vs. random PMI test, use of γ as criteria, etc.) is taken as starting point for CSI related tests. Other metrics/framework is not precluded.

For metrics for CSI requirements/tests, the following test metrics are identified:

* Option 1: Throughput/relative throughput
* Option 2: SGCS, NMSE
* Option 3: CSI prediction accuracy

Option 1 should be used as baseline. For option 3, further discuss is needed on the feasibility to define the CSI prediction accuracy in WI. For metrics for CSI monitoring, further discussion is needed in WI.

### 7.4.4 Beam management

Both spatial-domain DL beam prediction and temporal DL beam prediction are considered.

For metrics for beam management requirements/tests, the following test metrics are identified and could be considered

* Option 1: RSRP accuracy
* Option 2: Beam prediction accuracy
  + Top-1 (%) : the percentage of “the Top-1 strongest beam is Top-1 predicted beam”
  + Top-K/1 (%) : the percentage of “the Top-1 strongest beam is one of the Top-K predicted beams”
  + Top-1/K (%) : the percentage of “the Top-1 predicted beam is one of the Top-K strongest beams”
* Option 3: The successful rate for the correct prediction which is considered as maximum RSRP among top-K predicted beams is larger than the RSRP of the strongest beam – x dB,
  + Related measurement accuracy can be considered to determine x
* Option 4: combinations of above options

The overhead/latency reduction should be considered for the requirements as the side condition.

### 7.4.5 Positioning accuracy enhancements

Both direct AI/ML positioning and AI/ML assisted positioning are considered.

For metrics for positioning requirements/tests, the candidate options include

* Option 1: positioning accuracy: Ground truth vs. reported
  + only option available for direct positioning
* Option 2: CIR/PDP, channel estimation accuracy
* Option 3: ToA, RSTD and RSRP, and RSRPP
* Option 4: others (e.g., intermediate KPIs, LoS/NLoS)/combinations of the above

The feasibility and testability of different options should be further justified in WI.