Science Fund
of the Republic of Serbia

Grant Agreement No.: 7750824

Call: IDEJE

Hybrid Integrated Satellite and Terrestrial Access Network

D4.2: Simulation and verification of the ITCU module

Work package	WP 4
Subactivity	T4.1, T4.2
Due date	
Submission date	
Deliverable lead	
Version	1.0
Authors	Ivan Vajs, Srđan Brkić
Reviewers	Predrag Ivaniš, Dejan Drajić, Goran Đorđević

Document Revision History

Version	Date	Description of change	List of contributor(s)
V0.1	03.01.2024.	Document created, Sections 3 and 4 added	Ivan Vajs
V0.2	0.4.01.2024.	Section 2 added	Srđan Brkić
V0.3	05.01.2024.	1st version of D4.2	Ivan Vajs, Srđan Brkić
V0.4	05.1.2024.	2nd version of D4.2	Dejan Drajić
V1.0	05.01.2024.	Final deliverable completed	Ivan Vajs, Srđan Brkić

COPYRIGHT NOTICE

ACKNOWLEDGMENT

This deliverable has been written in the context of hi-STAR project who has received funding from the Science Fund of the Republic of Serbia, Programme IDEJE under grant agreement n° 7750284.

EXECUTIVE SUMMARY

The hi-STAR project addresses one of the most critical challenges for the next generation wireless networks, which is integration of non-terrestrial networks with terrestrial 5G network. The general objective of the WP4 is to develop a traffic control unit (TCU) that will benefit from multiple RANs (Radio Access Networks) and increase the reliability of users' communication. In order to develop the TCU and verify its performance, it is necessary to create a simulation environment and propose the handover procedure that will improve the user experience.

This deliverable is a result of the work done in the context of WP4 Subtasks T4.1 (Simulation environment creation) and T4.2 (Design of traffic control module placed in HUT). We briefly review the simulation environment that includes LEO (Low Earth Orbit) satellite-to-ground communication links and a user-centric handover execution method. Following this, a description of the expert system based on various machine learning (ML) models including (neural networks) is given, alongside the dataset containing the simulations for its evaluation. A comparison between various ML models is given, and a review of system performance in terms of two types of adaptive coding and modulation scenarios.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

ML	**Machine learning**
LR	**Linear regression**
ITCU	**Intelligent Traffic Control Unit**
KNN	**K nearest neighbors**
SVM	**Support vector machine**
LEO	**Low-Earth-Orbit**
NN	**Neural network**
WP	**Work Package**
MAE	**Mean absolute error**
NMAE	**Normalized mean absolute error**
MSE	**Mean squared error**
MODCOD	**Modulation and Coding**
SNORE	**Signal-to-Noise-Interference Ratio estimator**
NMSE	**Normalized mean squared error**
RAN	**Radio access network**
DNN	**Deep neural network**
AI	**Artificial intelligence**

SECTION 1 - INTRODUCTION

The increased need for communication, followed by the necessity of infrastructure development can be considered one of the main challenges in the communication industry [1]. Handling multiple RANs and developing handover execution and load balancing strategies presents a crucial step towards this direction.

The System that can successfully optimize communication through multiple RANs and minimize the usage of resources requires reliable information regarding the channel through which the communication is to be performed. In terms of LEO satellites, this would imply an accurate estimation of the channel's SNR, so that the optimal modulation can be used. This would ensure not only higher spectral efficiency, but also a lower transmission error rate, and could contribute to using a lower number of handovers to obtain the necessary communication requirements. With the development of AI, more specifically ML algorithms and complex neural networks, this problem has been observed from the data science perspective, providing models that can aid communication performance in different scenarios [2]–[4].

The Deliverable D4.2 summarizes the continued work carried out within WP4 subtasks T.4.1 and T4.2. In the above subtasks we have investigated the possibility to perform user-centric handover in heterogeneous radio environment, where the end user could establish connectivity through 5G base station, or multiple LEO satellites. Deliverable 4.2 focuses on various ML models used for SNR prediction, in order to obtain a higher spectral efficiency and a lower transmission error rate. The ML models are evaluated using a dataset of simulations with various conditions and 2 RANs each, and with two different MODCOD options.

This deliverable is structured as follows: In Section 2 a review of the system model is given. Section 3 defines the expert system that is used for the decision making, including various ML algorithms, the evaluation protocol and the dataset that is used for evaluation. Section 4 presents the results of the evaluation accompanied by a discussion regarding the obtained metrics. Section 5 concludes the document.

SECTION 2 – SYSTEM MODEL

2.1. ITCU OPERATION SUMMERY

In this section we briefly explain the system model of the ITCU (*Intelligent Traffic Control Unit*), which is from most part inherited from the Deliverable D4.1. In Fig. 1 we depict typical physical layer of RAN and the position of ITCU in it. One of the basic operation pre-required for the handover execution is measurement of the channel state information, i.e., measurements of instantaneous SNR. In our simulation environment we use data-added with pilots variant of so called SNORE (Signal-to-Noise-Interference Ratio estimator) algorithm, developed in NASA Jet Propulsion Laboratory. The ITCU collects channel state information from each available RAN, transformed them into decision making policy and sends control information via a backchannel to an access point of each RAN.

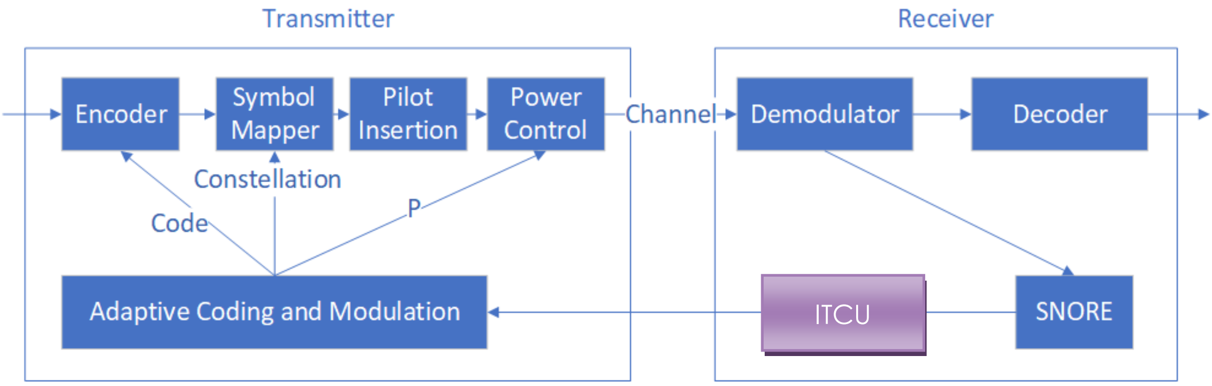

Figure 1. Block diagram of physical layer RAN.

Typical RAN supports a set of MODCODs spectral efficiencies M={M_1,M_2,…,M_K}, each associated with monotonously increasing function $T: M \rightarrow \mathbb{R}$: where $T(M_i)$ represents minimal SNR value needed to operate in the MODCOD M_i, also called ***the threshold***. If we denote the instantaneous SNR in the channel by γ then the *optimal MODCOD* is the greatest element in a set $S_\gamma \subseteq M, S_\gamma = \{M_i | T(M_i) \leq \gamma\}$, i.e., supremum of S_γ denoted as $\sup(S_\gamma)$. In a special case when $S_\gamma = \emptyset$, the lowest spectral efficiency M_1 will be used for transmission.

Let us enumerate all of total N RANs that end-user can use to establish communication by a set I={1,2,…N}. At a time interval (t_{j-1},t_j] the end-user measures instantaneous SNRs for all the currently available RANs, i.e., RANs from a set $I_j \in I$. The principle of forming the I_j set will be discussed later. The measured values form a set $\Gamma_j = \{\hat{\gamma}_{j,i} | i \in I_j\}$, where $\hat{\gamma}_{j,i}$ represents measured SNR in [dB] of the i-th RAN, during time interval (t_{j-1},t_j]. Then, a prediction function $f: \mathbb{R}^3 \rightarrow M$ is executed, which for all the available RANs produces the MODCODs that will potentially be used until the next SNR measurements are completed, i.e., in the time interval

$(t_j, t_{j+1}]$. The goal of the function $f()$ is to make predictions regarding the optimal MODCOD, based on the current and previous SNR measurements. Let $S \subseteq M$ be a set of MODCOD efficiencies form in the following way

$$P = \{M_m | g(\hat{\gamma}_{j,i}, \hat{\gamma}_{j-1,i}, \hat{\gamma}_{j-2,i}) \geq T(M_m)\}$$

where $\gamma_{j,i} \in \Gamma_j$, while $g: \mathbb{R}^3 \to \mathbb{R}^+$ produces **prediction of channel state information (i.e., SNR value)**. Thus we have

$$f(\hat{\gamma}_{j,i}, \hat{\gamma}_{j-1,i}, \hat{\gamma}_{j-2,i}) = \sup(P)$$

The end-user simply chooses the RAN that will produce the MODCOD with the highest spectral efficiency, i.e.,

$$f\left(\hat{\gamma}_{j,n_j}, \hat{\gamma}_{j-1,n_j}, \hat{\gamma}_{j-2,n_j}\right) \geq f(\hat{\gamma}_{j,i}, \hat{\gamma}_{j-1,i}, \hat{\gamma}_{j-2,i}), \qquad \forall i \in I_j.$$

In case that there are multiple RANs that achieve maximal efficiently, and the previously used RAN is among then, the end-user does not perform a handover. Otherwise, the least used RAN in the current time window in selected, in order to reduce the possibility that selected RAN leaves a set of available RANs. If $n_j \neq n_{j-1}$ the handover procedure is triggered and the information of the preferred RAN is send to the network core via reverse link.

Note that in Deliverable 4.1. we optimized the threshold margin, however our further findings revealed that the prediction of SNR could be directly made, by state-of-the-art machine learning models, as discussed in the subsequent section.

Formally, we recapitulate the steps of the handover procedure from D.4.1 as follows:

- The end user makes statistics of RANs usage in a predefined time window. All RANs, which utilization (in the time window) is below some threshold, compose a set of *available RANs*;
- The end user also collects pilot symbols from all the available RANs and periodically measures SNR values;
- Based on the current and previous SNR measurements end user makes predictions of future SNR values for each available RAN, by using DNN;
- Based on the SNR predictions, MODCOD for potential subsequent communication is obtained for each RAN;
- If there exists available RAN with the predicted MODCOD that offers higher spectral efficiency than correctly used RAN, handover execution is triggered by the end user.

2.2. RECAPITULATION OF THE CHANNEL MODEL

In this deliverable we are only interested to investigate what effects instantaneous SNR, denoted by $\gamma(t)$ has on handover procedure, defined as following

$$\gamma(t) = \frac{P_T}{\sigma^2} \frac{|h(t)|^2}{d^\beta},$$

where P_T is transmitted power, σ^2 is noise variance, d is distance between source and destination, β is path-loss factor and $|h(t)|^2$ is time-varying channel power gain. The probability density function of the channel gain *h(t)* depends on the propagation environment, which could be different in satellite-to-ground and terrestrial communications. In this deliverable we assumed that channel gain between RAN access point and end-user is composed of two time-vary components: i) scattering and ii) line-of-sight (LOS) component, i.e., [9-11]

$$h(t) = a(t)e^{j\alpha(t)} + z(t)e^{j\alpha_0},$$

where $a(t)$ represents instantaneous scattering amplitude which is Rayleigh distributed, $\alpha(t)$ is uniformly distributed random phase, $z(t)$ is amplitude of the LOS component which is Nakagami-m distributed, and α_0 is deterministic phase. The probability density function of channel gain amplitude $r(t)=|h(t)|$ can be represented as

$$f_R(r) = \left(\frac{2b_0 m}{2b_0 m + \Omega}\right)^m \frac{r}{b_0} e^{-\frac{r^2}{2b_0}} {}_1F_1\left(m, 1, \frac{\Omega r^2}{2b_0(2b_0 m + \Omega)}\right), \qquad r \geq 0,$$

where $2b_0$ denotes the average power of scattering component, Ω is average power of LOS component, m represents parameter of Nakagami-m distribution and ${}_1F_1()$ is confluent hypergeometrical function. By using three degrees of freedom (b_0, Ω and m) the introduced channel model can accurately describe different propagation environments, as well as different LEO satellite elevation angles. Usually, quality of the channel can be described by the level of shadowing included in the model and we distinguish low, average and heavy shadowing, with parameters given in Table 1.

Propagation scenario	b_0	Ω	m
Infrequent low shadowing	0.158	1.29	19.4
Average shadowing	0.126	0.835	10.1
Frequent heavy shadowing	0.063	0.000897	0.739

Table 1. Channel parameters for different scenarios.

SECTION 3 – OVERVIEW OF THE IMPLEMENTED EXPERT SYSTEM AND ML MODELS

This section presents an overview of implemented expert system, ML models used for the channel state prediction task, the evaluation dataset, evaluation protocol, and evaluation metrics.

3.1. EXPERT SYSTEM

The implemented expert system is structured around the channel states of RANs, more specifically simulated SNR values, and performs the prediction for the optimal MODCOD and RAN that should be used for data transfer in the following time interval. The system inputs are the SNR values from RAN channels in a sequence form, based on several observed timepoints. Based on these inputs, the system predicts the SNR values for the RAN channels for the next timepoint and makes a decision on which MODCOD and which RAN should be used.

The expert system is focused on the maximum amount of data that can be correctly transmitted, choosing the RAN with the higher SNR value, and the MODCOD with the highest transmission rate that can be successfully transmitted.

The system used several different ML algorithms for SNR prediction, including linear regression (LR), support vector machine (SVM), k nearest neighbors (KNN) and a fully connected neural network (NN). Each of the system evaluation protocols was also done using two different MODCOD scenarios. In the first scenario end-used implements small portion of different MODCODs (the same assumption is used in Deliverable 4.1) which are the most used operation points in DVB-S2X protocol. The second scenario includes all of the operation points, used when communicate with short frames of the DVB-S2X protocol. The spectral efficiencies (M_i) and the SNR thresholds ($T(M_i)$, minimal SNR value needed to operate with the efficiency M_i) for the MODCODs, for both scenarios, are listed in Table 2.

The metrics that were used for evaluation were the mean square error (MSE), mean absolute error (MAE), normalized mean square error (NMSE) and normalized mean absolute error (NMAE). Each metric was calculated between the predicted and true value of the respective SNR on the test set. The evaluation of the system was done using leave-one-out cross-validation, where data from a single simulation is used for testing, and the data from other simulations is used for training the ML algorithm. This is repeated 12 times, i.e. the number of simulations that were performed to create the dataset. The final system evaluation includes the performance of each expert system on the test set, measured by the average spectral efficiency obtained on the test set, as well as the transmission error rate. Expert system evaluation also includes the number of handover events that occurred, and the number of handovers is compared between various versions of the system.

Table 2 : MODCODs used for system evaluation.

MODCOD 1		
Modulation	M_i [b/s/Hz]	$T(M_i)$ [dB]
8PSK	1.22	3.95
16PSK	1.63	6.1
16APSK	1.86	7.1
16APSK	2.02	7.75
16APSK	2.48	9.8
MODCOD 2		
Modulation	M_i[b/s/Hz]	$T(M_i)$ [dB]
BPSK-S 1/5	0.1	-9.9
BPSK-S 11/45	0.12	-8.3
BPSK 1/5	0.2	-6.1
BPSK 4/15	0.27	-4.9
BPSK 1/3	0.33	-3.72
QPSK 11/45	0.49	-2.5
QPSK 4/15	0.53	-2.24
QPSK 14/45	0.62	-1.46
QPSK 7/15	0.93	0.6
QPSK 8/15	1.07	1.45
QPSK 32/45	1.42	3.66
8PSK 8/15	1.60	4.71
8PSK 26/45	1.73	5.52
16APSK 7/15	1.87	5.99
16APSK 8/15	2.13	6.93
16APSK 26/45	2.31	7.66
16APSK 3/5	2.40	8.1
16APSK 32/45	2.84	9.81
32APSK 2/3	3.33	11.41
32APSK 32/45	3.56	12.18

3.2. THE EVALUATION DATASET

The dataset used for the evaluation of the developed system contains 12 distinct simulations with different parameters. Each simulation contains between 2030 and 2045 timepoints and each timepoint contains data from two RANs, and each in each timepoint each RAN is described by a true SNR value and a measured SNR value. These time series are obtained by Monte Carlo simulation, described in Deliverables D2.1 and D4.1. True SNR values are used solely for the evaluation process, while the measured values are used for ML inputs and decision making, to present a realistic implementation.

Half of the data was generated with a low Doppler frequency shift (10Hz) and the other half with a high Doppler frequency shift (100Hz), so that two different channel characteristics can be considered in the evaluation. In each half of the data, a range of expected SNR values were used as well, from 7 dB to 12 dB to access system performance with both higher and lower available SNR values. A visualization of the simulations can be seen in Fig. 2, including a small segment to show the temporal characteristics, and a violin plot [5] to show the distribution of the SNR values for the entire simulation. The y axis of all subplots in Fig. 2 is shared, so that the SNR values can be compared.

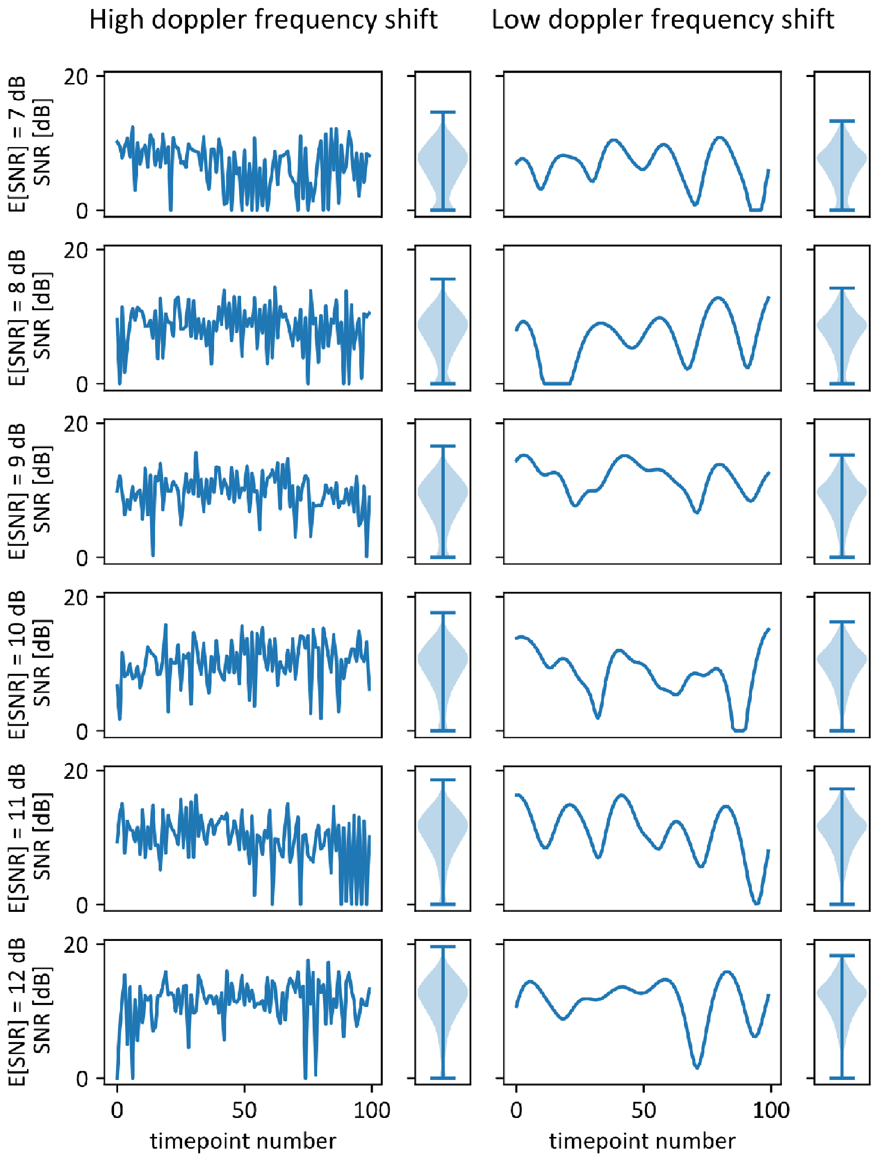

Figure 2. Dataset visualization with included violin plots.

3.3. ML MODELS

The ML models were used for the prediction of future SNR values to make the MODCOD and satellite selection process more accurate. For the evaluation scenario, the predictions were based on 10 consequent timepoints, to provide enough inputs to the algorithms to make a reliable prediction, yet limit the possibilities for overfitting the data.

For each simulation SNR prediction was performed for each RAN separately. A window of 10 samples from measured SNR was used with a stride of 1, and for each 10 consequent measured SNR samples, the label was the measured SNR value from the subsequent timepoint. The input-output pairs from the simulations used for training were used to train the models, and the data from the test simulation was used for evaluation. The ML models that were used are LR, SVM, KNN, and NN.

The training process of the ML algorithms represents the adaptation of the ML model parameters to optimize a certain function. This function is called the loss function, and in supervised learning, which is the type of learning used in the described research, this function quantifies the disagreement between the model predictions and the true values the model is trying to predict. The learning process essentially represents the adaptation of parameters so that the loss function has the lowest possible value on the data from the training set. This main principle is the same for all stated ML algorithms, however, the difference between the models is the nature of their parameters and how they function.

LR represents a simple algorithm that is linear in nature, and essentially has a coefficient for each input value and a bias factor [6]. These parameters are optimized in order to obtain the lowest MSE between the predictions and labels of the train set. KNN is also a simple algorithm that does not have parameters in a conventional sense [7]. This algorithm creates a prediction on the test set based on the “closest” neighboring values that can be found in the train set. SVM is a type of algorithm that has the optimization complexity as linear regression but uses kernels to transform the inputs into a higher dimensional plane [8]. This essentially enables SVM to adapt its parameter to a more dispersed group of input features and potentially achieve better results.

The NN is a type of algorithm that mimics the functionalities of the human brain. It is based on matrix multiplication and nonlinearities introduced through activation functions that are applied after matrix multiplication. All stated algorithms have hyperparameters that are out of scope for this phase of the research, but NN architecture and hyperparameter tuning pose a much more complicated task than the other mentioned ML algorithms. NN algorithms can also model much more complex dependencies between inputs and outputs when compared to other algorithms but have multiple caveats that can lead to suboptimal performance.

For the testing performed on the dataset, the problem of NN architecture and ML hyperparameter tuning can be considered out of scope. Therefore, all default settings for the algorithms are kept as are by default in the scikit learn library [9], which was used for all ML model implementations. The exception is the NN, where the number of neurons in the hidden layer was lowered from 200 to 50 considering the amount of available data, and the maximum number of epochs was increased from 200 to 500 so that the optimization could converge.

SECTION 4 – RESULTS AND DISCUSSION

4.1. SNR PREDICTION

The evaluation metrics for the SNR prediction problem using the described ML algorithms as well as the outdated information approach as a baseline is presented in Table 3.

Table 3. SNR prediction metrics averaged over all test sets and averaged for both satellites.

	MAE [dB]	NMAE	MSE [dB^2]	NMSE
Outdated information	2.336	0.675	14.165	1.164
LR	1.960	0.572	7.294	0.612
KNN	1.312	0.329	3.048	0.255
SVM	1.375	0.398	5.361	0.445
NN	1.385	0.402	4.547	0.377

The outdated information which simply considers the latest SNR value of the channel to be the same as the one that follows it, clearly shows the worst performance. This is expected as there is no mechanism to compensate for any faster changes in the channel that occur when there is a high doppler frequency shift.

In comparison, the ML algorithms show lower error values, but LR has a distinctly bad performance. The simplicity of the LR algorithm, and the two different characteristics of the channels available in the dataset (slow changing and quick changing based on the frequency shift) allow it to only make minor improvements when compared to the outdated information. On the other hand, The KNN, SVM and NN have a similar performance, with the best being KNN in all cases.

The KNN algorithm is the simplest after the LR, however it owes its best performance to its non-linear estimation that is based directly on observing the distance from the values contained in the training set. The obtained MAE value of 1.312 dB is almost half the initial value that is obtained using the outdated information. It is also important to note that the outdated information principle can give great results when the channel has slow changes, which is the case in half of the used dataset.

The goal of the simulations was to cover the two extremes that can occur in practice, and in a realistic scenario, the outdated information approach would most likely have a worse performance as compared to the one shown in Table 2. In contrast, the other ML algorithms (perhaps aside from LR) are not expected to have any difficulties, since they show great results even if they are trained and evaluated on extreme scenarios. In general, the expansion of the

dataset and the usage of data that has a continuous span of characteristics is expected only to improve the performance of the implemented ML algorithms.
The results of the algorithm performance for each fold of the cross-validation, i.e. each simulation file separately is shown in Fig. 3, 4, 5 and 6. Each figure shows the obtained metric for each satellite separately, for each observed algorithm, each frequency shift condition and for each expected SNR value. Fig. 3 shows the MAE metric, Fig. 4 the NMAE, Fig. 5 the MSE and Fig. 6 the NMSE.

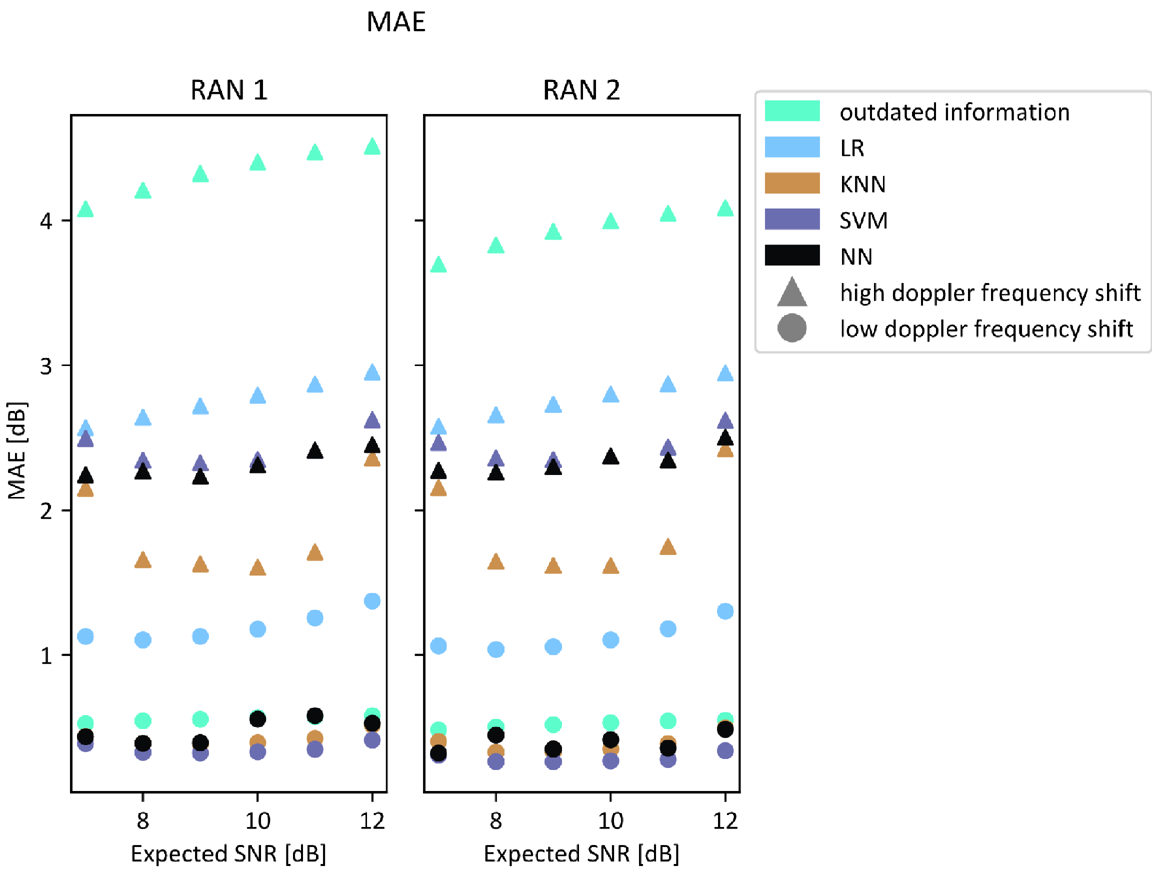

Figure 3. The MAE for each individual algorithm, satellite and simulation condition.

The results shown in Fig. 3 present several important findings. Firstly, the performance of all algorithms is worse when evaluated on the high Doppler frequency shift data, in comparison to the lower Doppler frequency. This is expected, as it is much easier to predict the forthcoming SNR value if the changes in the channel are slower. This is so prominent that the best performing algorithm for the higher Doppler frequency always has a bigger error than the worst performing algorithm for the lower Doppler frequency, regardless of the expected SNR.
Another important finding is that the outdated information approach has a much higher error for the higher Doppler frequency than it has for the lower Doppler frequency, when compared to the other algorithms. The LR, which has the worst performance out of all the ML models, even has a higher error for the low Doppler frequency that the outdated information approach. For

the low Doppler frequency, the difference between the ML algorithms is also a lot more subtle. This could indicate that for channels which have slower changes, the outdated information approach can successfully be used, and the implementation and introduction of ML algorithms could be redundant. In contrast, channels that have quicker changes are not at all suitable for this approach, and substantial benefits can be obtained even from LR.

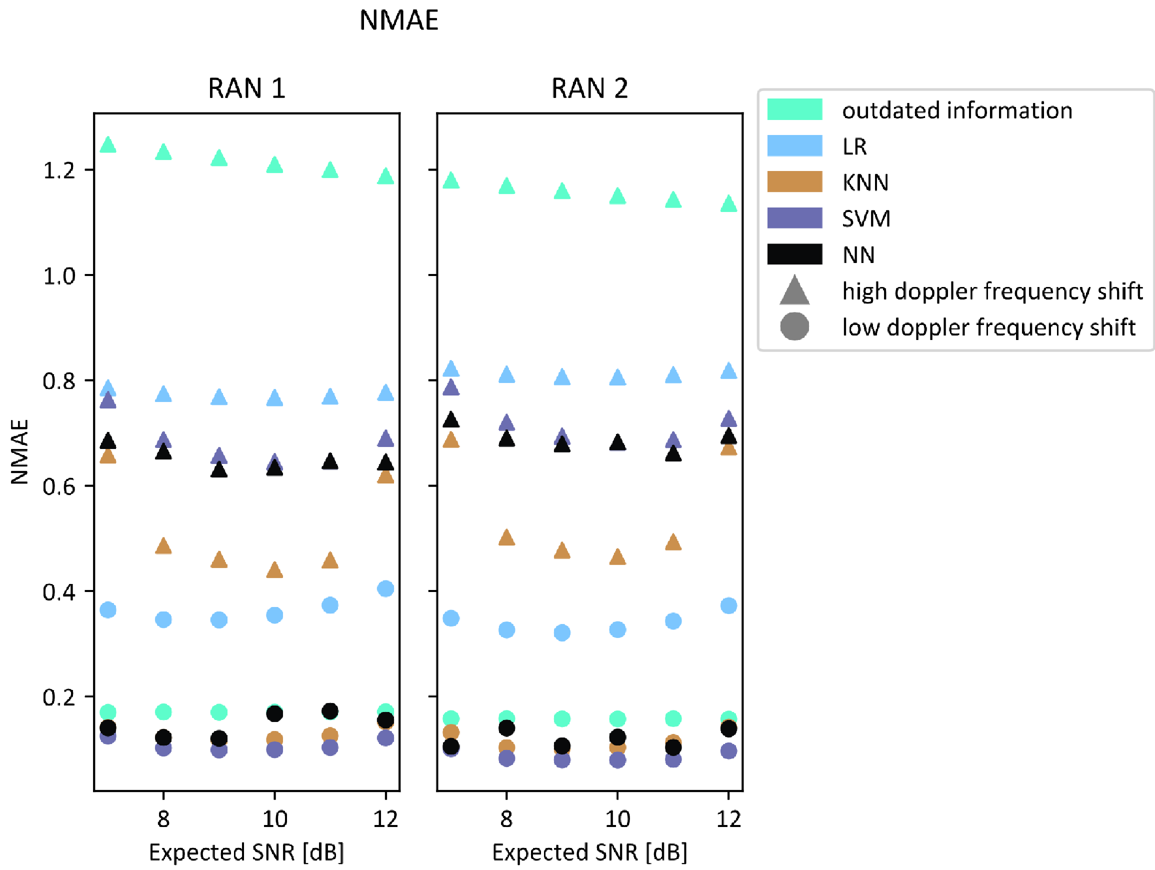

Figure 4. The NMAE for each individual algorithm, satellite and simulation condition.

For the NMAE, the results closely resemble the ones presented for MAE. Since NMAE is normalized using the standard deviation of the signal, the NMAE error that is over 1 essentially means that the estimation power of the algorithm is worse than just predicting the mean value of the signal for each and every sample. This is the case for all evaluations of the outdated information approach and stands in line with the importance of using ML algorithms for high Doppler frequency shift channels.

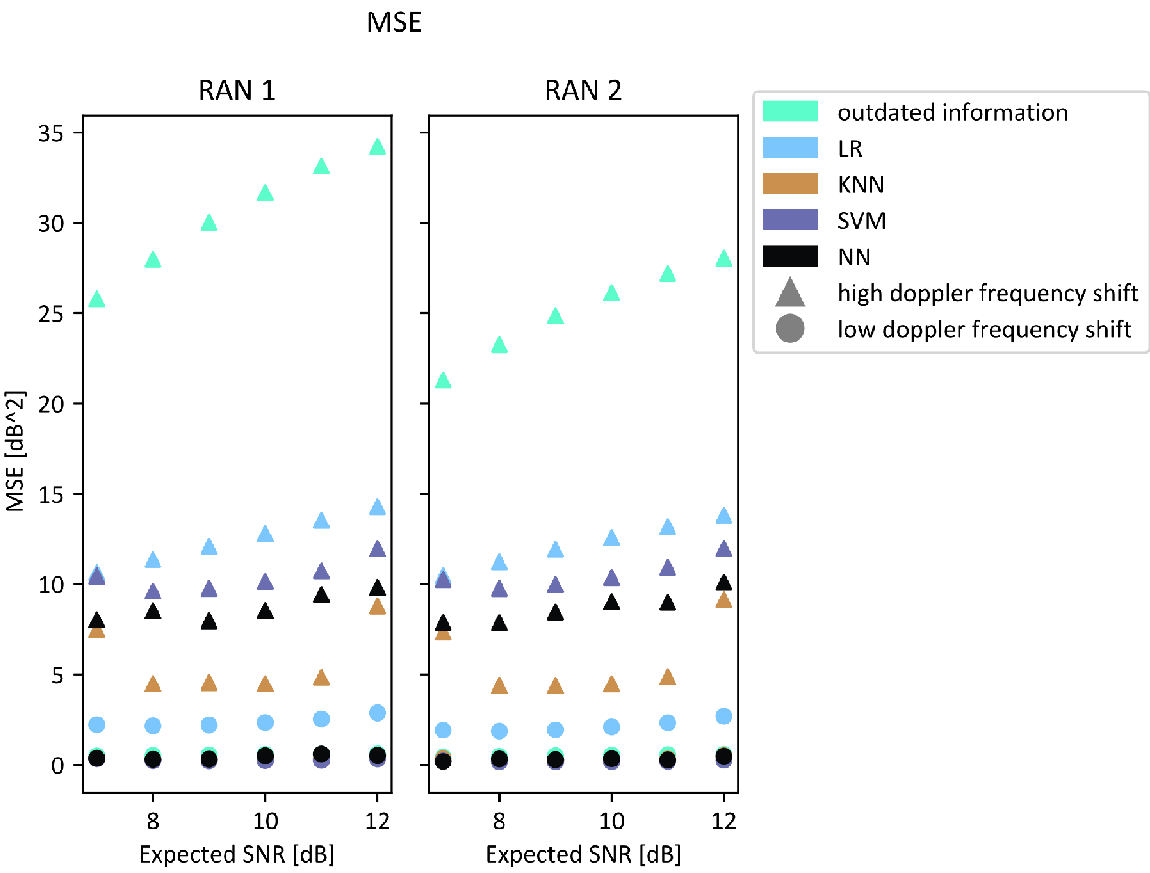

Figure 5. The MSE for each individual algorithm, satellite and simulation condition.

The MSE metric provides similar conclusions as the MAE and NMAE. The comparison between algorithm performance stands and there is no real difference in the conclusions that can be draw, aside from perhaps a smaller visual difference between the algorithms for the low Doppler frequency shift data. The MSE and MAE metric are similar in nature, and the potential difference that could be observed might be due to outlier errors, which are higher for MSE than for MAE. In the observed data, however, the distribution of the data is such that there are no significant outliers which is why the conclusions drawn are the same.

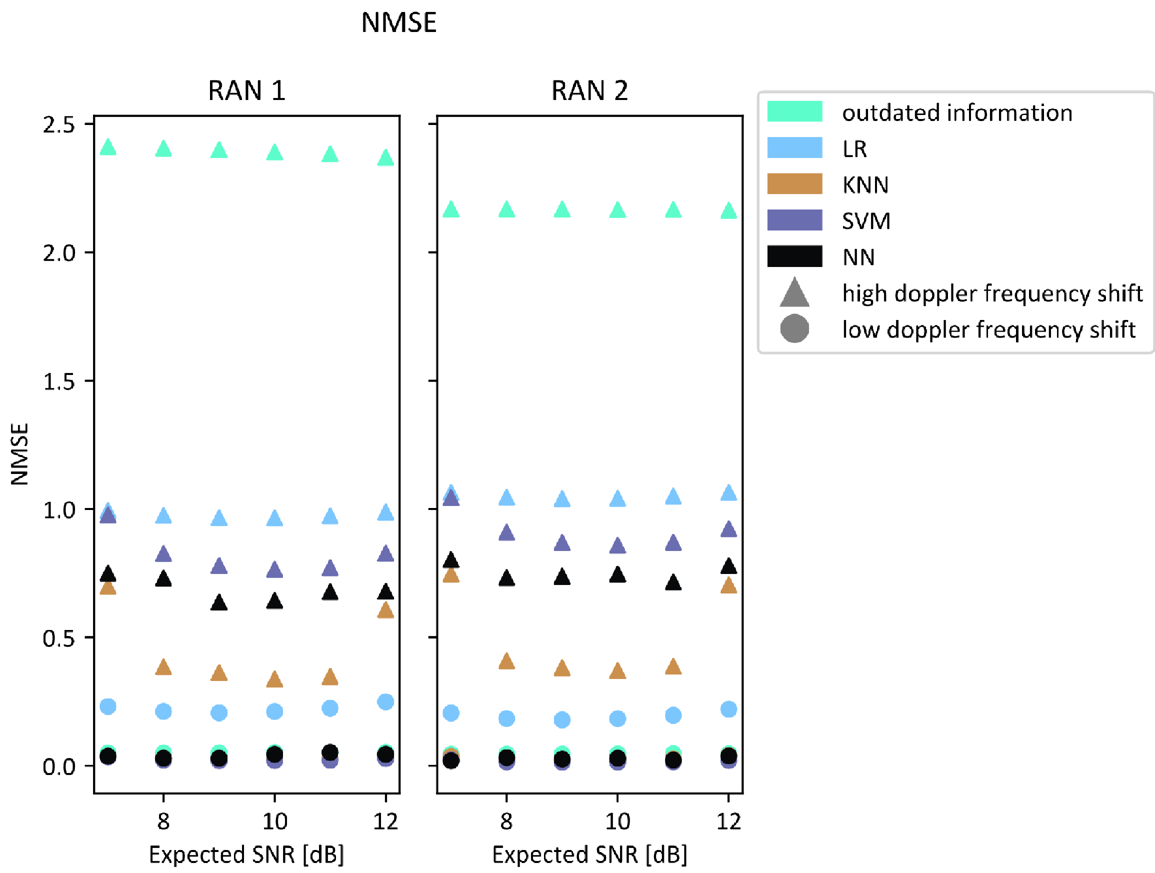

Figure 6. The NMSE for each individual algorithm, satellite and simulation condition.

The NMSE, as well as all other metrics, gives similar results. One result which is important to note is that the SVM and LR which had a NMAE value lower than 1 for the quick changing channel data, now have a NMSE value close to 1. This means that perhaps if normalized by the variance of the signal, and observing the square error, these algorithms do not give the best performance. However, the KNN and NN still show great results.

An interesting occurrence that is consistent throughout the presented results is that the KNN algorithm has a bigger error for SNRs of 7 and 12, when compared to other SNR values, i.e. a worse performance in the extreme cases. This is because of the nature of the KNN algorithm, which performs worse if it has to perform estimates outside of the data values that are available in the train set. Still, the results in these extreme cases are still superior to the ones of other algorithms.

4.2. EXPERT SYSTEM PERFORMANCE

The evaluation of the expert system was performed by observing the average spectral efficiencies and the error rate, averaged over all test sets during cross-validation. The results are presented for all used algorithms, and both observed MODCOD scenarios in Table 4.

Table 4. Average spectral efficiencies and error rate averaged for all test sets.

	MODCOD 1		MODCOD 2	
	Average M_i [b/s/Hz]	Error rate [%]	Average M_i[b/s/Hz]	Error rate [%]
Outdated information	1.587	28.6	1.762	36.8
LR	1.638	27.3	1.879	33.0
KNN	1.900	15.8	2.233	21.6
SVM	1.814	19.8	2.137	25.1
NN	1.826	18.5	2.083	25.7

The obtained spectral efficiencies and error rates compare the ML algorithms in almost the same way as the observed SNR regression errors. The only difference that can be observed is that for MODCOD 2 the SVM outperforms the NN, which was not the case for the SNR regression evaluation. This, although not major, especially since the best performing algorithm is still the KNN, opens up an important question relating to the usage of ML algorithms in the expert system. SNR prediction plays a crucial role in the system pipeline, but the final metrics that are relevant for system performance are the spectral efficiency and error rate. Although an algorithm could be better at predicting SNR values, it could do so in the cases (SNR range) that is not crucial for decision making, and that would not change the MODCOD selected for data transfer. In this sense, it is important to evaluate all parts of the system individually, but also the system as a whole, to gain both a micro and a macro evaluation perspective.
When comparing the results between the two MODCOD scenarios, it can be seen that the MODCOD 2 offers a higher spectral efficiency but also a higher error rate in comparison to MODCOD 1, for each of the proposed algorithms. This is due to a higher number of available modulations in the MODCOD 2 list, allowing for a higher spectral efficiency for a given SNR, but also being more prone to errors.

Observing the implemented system, since the decision system always chooses the satellite with the higher SNR, the number of handovers does not depend on the used MODCOD. The average number of handovers per simulation, averaged over all cross-validation fold is shown in Table 5.

Table 5. Average number of handovers on the test set for all observed algorithms.

Number of handovers	
Outdated information	610
LR	464
KNN	589
SVM	515
NN	565

The number of handovers gives an interesting conclusion, once again indicating the importance of observing the system as a whole. The outdated information averages at the highest number of handovers, and even in terms of handover optimization would not be the greatest solution. However, the KNN algorithm which offered the lowest SNR prediction errors as well as the lowest transmission errors and spectral efficiencies shows the second highest number of handovers. If this number is not crucial for system performance, the KNN could of course be used, but if it is, then perhaps a different ML algorithm could be more efficient. The algorithms that have higher spectral efficiencies and lower error and SNR prediction rates, also have a higher number of handovers (starting with KNN, followed by NN, SVM and LR). This would indicate that there should perhaps be a tradeoff between the spectral efficiency and the number of handovers. It could also be of interest to develop a system that would not always choose a channel with a higher SNR, but rather balance the need for a higher spectral efficiency with the number of handovers. Out of all the algorithms, it is important to note, that NN can have complex loss functions, and can be trained with various constraints, so perhaps the entire system can be designed with several (or one complex) neural network that can learn to make decisions taking into consideration handover and other resource limitations.

SECTION 5 – CONCLUSIONS

This Deliverable has presented a review of the simulation protocol and a handover procedure that is based on the SNR measurements. An expert system has been proposed that incorporates an ML model to predict future SNR values, and based on the prediction the system decides which RAN should be used for the connection. The system is evaluated for various ML algorithms including NNs, based on 12 different simulations with varying expected SNR values and Doppler frequency shifts. The obtained results have been observed in terms of pure SNR prediction metrics, as well as average spectral efficiencies, transmission error rates and number of handovers for two different MODCOD scenarios. The obtained results show that a clear improvement can be obtained using ML models, and direct future work towards optimizing the expert system directly on the parameters of interest, i.e., spectral efficiency, error rate and handover number.

REFERENCES

[1] G. Pan, J. Ye, J. An, and M. Alouini, “Latency Versus Reliability in LEO Mega-Constellations: Terrestrial, Aerial, or Space Relay?,” *IEEE Trans. Mob. Comput.*, vol. 22, no. 09, pp. 5330–5345, Sep. 2023.

[2] Y. Zhang, Y. Wu, A. Liu, X. Xia, T. Pan, and X. Liu, “Deep Learning-Based Channel Prediction for LEO Satellite Massive MIMO Communication System,” *IEEE Wirel. Commun. Lett.*, vol. 10, no. 8, pp. 1835–1839, 2021.

[3] T. Mortlock and Z. M. Kassas, “Assessing Machine Learning for LEO Satellite Orbit Determination in Simultaneous Tracking and Navigation,” in *2021 IEEE Aerospace Conference (50100)*, 2021, pp. 1–8.

[4] Y. Zhang, Z. Wang, Y. Huang, W. Wei, G. F. Pedersen, and M. Shen, “A Digital Signal Recovery Technique Using DNNs for LEO Satellite Communication Systems,” *IEEE Trans. Ind. Electron.*, vol. 68, no. 7, pp. 6141–6151, 2021.

[5] J. L. Hintze and R. D. Nelson, “Violin Plots: A Box Plot-Density Trace Synergism,” *Am. Stat.*, vol. 52, no. 2, pp. 181–184, May 1998.

[6] S. Weisberg, *Applied linear regression*, vol. 528. John Wiley & Sons, 2005.

[7] Z. Zhang, “Introduction to machine learning: k-nearest neighbors.,” *Ann. Transl. Med.*, vol. 4, no. 11, p. 218, Jun. 2016.

[8] V. Kecman, “Support Vector Machines – An Introduction BT - Support Vector Machines: Theory and Applications,” L. Wang, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 1–47.

[9] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in {P}ython,” *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.