User-Level Handover Decision Making Based on Machine Learning Approaches

João P. S. H. Lima, Alvaro A. M. de Medeiros, Eduardo P. de Aguiar, Vicente A. de Sousa Jr. and Tarciana C. B. Guerra

***Abstract*—This Letter covers a broad comparison of methods for classification and regression applications for a user-level handover decision making in scenarios with adverse propagation conditions involving buildings, coverage holes, and shadowing effects. The simulation campaigns are based on network simulator *ns-3*. The comparison encompasses classical machine learning approaches, such as KNN, SVM, and neural networks, but also state-of-the-art fuzzy logic systems and latter boosting machines. The results indicate that SVM and MLP are the most suitable for the classification of the best handover target, although fuzzy system SOFL can perform similarly with lower processing time. Additionally, for the download time estimation, LightGBM provides the smallest error with short processing time, even in hard propagation scenarios.**

***Index Terms*—Machine learning, Fuzzy, Handover, ns-3.**

I. Introduction

Two key concepts adopted by next-generation networks are cell densification and operation at high frequencies. Although larger bandwidth is available, enabling higher data rates, the propagation on higher frequencies limits the cell coverage area.

A fundamental cellular procedure directly affected by this scenario is the handover (HO), which is the transfer of a communication session (e.g., a call, a video stream, a file download) from one cell to another without loss or interruption of service. Since the User Equipment (UE) must switch between physical channels during such procedure, it requires very rapid decisions from the cellular network in order to guarantee the Quality of Experience (QoE). The number of handovers is expected to increase notably, specially considering propagation-intensive scenarios (e.g. high-frequency urban cells, outdoor-to-indoor coverage) whose severe propagation situations cause areas with meaningful signal degradation, creating non-deterministic coverage holes.

Three characteristics are required from evolved handover procedures in order to provide solid work in upcoming mobile communication systems: **seamless** (no interruption); **spectral-efficiency aware** (controlled signaling load); and **smart** (decision-making leveraged by machine learning and the vast amount of information available in the network).

The current HO schemes in 3GPP networks (4G and 5G) are set upon some characteristic events, as depicted in Table I [1]. UEs are supposed to provide frequent measurement reports to the base station, known as enhanced Node-B (eNB) in 4G standard, containing received signal metrics, such as Reference Signal Received Power (RSRP) and Reference Signal Received Quality (RSRQ). Accordingly, the HO procedures occur with simple power level comparisons based on events of those measurements, being denominated as deterministic HOs. Despite its simple implementation, this configuration may lead to numerous inefficient, unnecessary or ping-pong HOs, flooding network channels with counterproductive signaling load, and degrading spectral efficiency.

TABLE I
Characteristic events for 3GPP Handover [1].

Event	Description
A1	Primary cell (PC) signal power becomes better than a threshold
A2	PC signal power becomes worse than a threshold
A3	Secondary cell (SC) signal power becomes better than PC by an offset
A4	SC signal power becomes better than a threshold

Machine learning (ML) applications to develop smarter handovers are numerous. The authors in [2] implement a Bayesian regression method for HO improvements in high-speed trains in South Korea, whereas authors in [3] applies *K-Nearest Neighbours* (KNN) for possible real-time HO decisions in vehicular networks. In [4], ML approaches are used to reduce latency and classify the best cell available for HO. In [5] and [6] fuzzy logic and reinforcement learning is used to for optimize traditional HO parameters, such as time-to-trigger and HO margin. The work from [7] compares different computational intelligence models for HO parameter tuning while solution in [8] employs *LightGBM* to predict mobile network traffic. A lane-changing algorithm for autonomous vehicles is developed in [9] based on *Extreme Gradient Boosting*. The authors of [10] promote a rich survey on HO management and 4G and 5G tendencies, while [11] offers a vast survey on autonomous HO management in the heterogeneous network context. In [12], the development of neural networks in HO mechanism were implemented at different levels, and an extensive database was produced. These works have demonstrated the capacity of different configurations of neural networks-handover integration to outperform classical 3GPP HO methods.

As a plenty of methodologies are developed for smart HO, this work aims to bring a broad comparison of methods in user-level scenarios of mobility in 3GPP networks, based in data set from [12]. In this Letter, the classification addresses the determination of the best HO target, whereas the regression estimates the time and percentage of download that a mobile user performs while moving and requesting a HO. Coverage holes and shadowing effects are modeled into simulation scenarios to emulate an urban environment. The coverage

João Lima is with CPQD (e-mail: jsales@cpqd.com.br). Alvaro Medeiros and Eduardo Aguiar are with Federal University of Juiz de Fora, Brazil (e-mails: {alvaro, eduardo.aguiar}@engenharia.ufjf.br). Alvaro Medeiros is also with Munster Technological University, Cork, Ireland. Vicente Sousa and Tarciana Guerra are with Federal University of Rio Grande do Norte, Brazil (e-mails: {tarciana.guerra.051, vicente.sousa}@ufrn.edu.br). This study was financed in part by FUNTTEL/Finep and the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. The proof of concept simulations provided by this Letter was supported by High Performance Computing Center (NPAD/UFRN).

Digital Object Identifier: 10.14209/jcis.2022.11

hole is a complete lack of coverage in a particular region, representing a connection interruption, e.g., due to a mmWave severe propagation condition. Thus, this Letter extends the solution and results of [12] with the following contributions:

- Enhancing the comparison of ML classification approaches, including most recent fuzzy logic systems;
- Estimating download duration and percentage of completion, providing new inputs for HO decision, using classical and state-of-the-art approaches, such as latter boosting machines (not increasing data acquisition complexity to feed learning algorithms);
- Evaluating of fuzzy-based HO methods on urban scenarios with the presence of buildings, shadowing effects, and coverage holes (evaluation also includes network-related Key Performance Indicators (KPIs) and algorithms' processing time).

II. System Modelling and Evaluated Scenarios

The environment setup relies on [13] using version 3.22 of Network Simulator *ns-3*, as presented in Fig. 1. It counts with 3 eNBs, 3 UEs, and an obstacle near eNB 2.

The UE 1 starts simultaneously to download a file and move straight with a random angle from $-60°$ to $30°$, with a constant speed of 60 km/h. Quickly, it escapes from the coverage area of eNB 1 and enters the coverage area of eNBs 2 and 3, requesting handover. For each scenario, about 1200 runs are analyzed. The levels of RSRP and RSRQ are captured every 200 ms, feeding the input database to evaluate ML algorithms. The download process uses the well-known TCP protocol with the file size of 15 MB. Finally, the simulation is carried for 100 s.

In this Letter, two scenarios are explored. The first one is based on the Okumura-Hata propagation model, pondering only path loss as large scale attenuation, chosen for being a widely used deterministic model for characterizing urban and suburban areas. This scenario may represent a situation with averaged RSRQ and RSRP, in which instantaneous values are filtered (e.g., moving average filter), flattening shadowing and small-scale fading effects. The second scenario sums the random shadowing effect to the Okumura-Hata model, indicating measurements that are more resembling to the fluctuations of RSRP and RSRQ. For both scenarios, the coverage hole is modeled by the presence of a building whose dimensions are extensive enough to emulate a region of connection interruption, with a very high path loss [13].

More details about scenario modeling, including simulation parameters and the SINR Radio Environment Maps (REMs) of eNB 2 for both scenarios can be found in [12].

III. The Proposed Evaluation

The authors of [12] develop studies to explore the possibilities of how the machine learning models can be integrated with the handover decision making process coordinated by the eNBs, but they did not include most recent gradient boosting machines nor fuzzy systems. Moreover, in this work a statistical analysis is also developed to corroborate the initial results presented. Nonetheless, this work also implements new metric predictions, which are regression machine learning tasks. The prediction of a download time and percentage of completion can be of high value for network architects in order to design optimal handover algorithms.

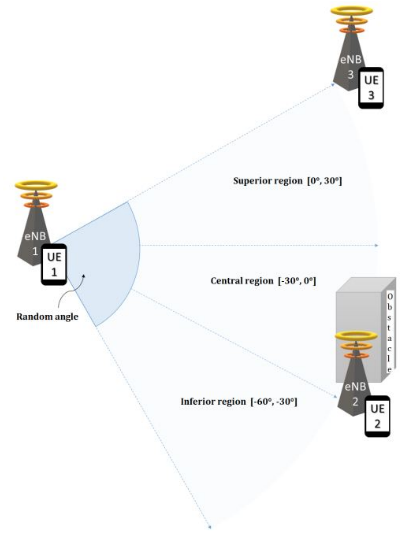

Fig. 1. The simulation environment.

A handover triggering can be considered unnecessary if a solid estimation indicates that the ongoing download will be completed. On the other hand, a handover shall be anticipated if the estimation indicates that it will not be completed.

Thus, similar to [12], this Letter searches for the best eNB in terms of download completion and duration (the classification problem), including state-of-the-art fuzzy systems. The authors of [14] and [15] have indicated that fuzzy strategies are capable of providing equivalent (or even better) performances while consuming less computational resources compared to traditional Artificial Intelligence tools. They had never been applied to the HO problem targeted by this Letter. We also propose regression techniques to estimate the percentage of completed download and the download duration.

The methods applied for the HO decision (classification problem) are Autonomous Learning Multimodel System (ALMMo) [15] (which has no tuning parameters, since it extracts all the features and adjustments from data); Self-Organizing Fuzzy Logic Classifier (SOFL) [16] (using *Mahalanobis* distance and *Granularity Level* of 2.9); Type-2 Fuzzy Logic Classifier (T2FLS) [17], [18] (being the learning parameter $\alpha = 0.01$, tolerance $\epsilon = 10^-8$, $\beta_1 = 0.9$ and $\beta_2 = 0.999$); Support Vector Machine Classifier (SVM) [7], [19] (with linear kernel and penalty parameter of 10 and 100 for Scenarios 1 and 2, respectively); and Multilayer Perceptron Classifier (MLP) [20], [21] (with 6 neurons in hidden layers and solver *lbfgs*).

Regarding the regression problem (estimation of the completed download percentage and the download duration), six methods are compared: Multilayer Perceptron Regressor (MLP) [20], [21] (with 22 and 4 neurons in hidden layers, *tanh* and *logistic* activation functions and *lbfgs* solver for Scenarios 1 and 2, respectively); KNN [3], [20] with 4 and 6 neighbors considered for each Scenario; Random Forest (RF) [7], [22] with 94 and 106 trees in the forest, in each case; Gradient Boosting Machine (GBM) [23], with 84 and 120 trees in their ensemble; Extreme Gradient Boosting (XGBoost) [9], with 174 and 120 estimators each; and Light Gradient Boosting Machine (LightGBM) [8], which used 148 and 139 estimators in the ensemble for each Scenario.

There are some considerations on how the ML models could be implemented in a real network. First, it would be

TABLE II
INITIAL RESULTS FOR SCENARIO 1.

Method	Accuracy (%)	Std. Dev.	Time (s)
MLP	99.72	0.10	5.25
SVM	99.74	0.08	0.54
SOFL	99.11	0.22	12.51
T2FLS	98.94	0.68	5845.78
ALMMo	99.61	0.11	257.47

TABLE III
INITIAL RESULTS FOR SCENARIO 2.

Method	Accuracy (%)	Std. Dev.	Time (s)
MLP	86.22	0.96	15.94
SVM	86.34	0.33	21.26
SOFL	85.32	0.71	14.60
T2FLS	72.40	0.98	7672.62
ALMMo	68.06	1.16	357.74

necessary a setup step, in which the network would act without the models' action because it is necessary to collect/store data and to train models. In our analysis, the RSRP and RSRQ measurements occur in this initial phase, along with the information about download completions and their required times. They are models' inputs so that they can be trained. This also clarifies how important is the processing time, since the models must be updated as fast as possible, with no harm to the network. After this initial phase, the models would be available to act on the HO decision making at the eNBs. Furthermore, if network alters e.g., an introduction of a new eNB, it would be necessary to retrain the models to account such changes [12].

IV. RESULTS AND DISCUSSION

The selected methods were tested with Python 3.7 and Matlab scripts on an i7-6700HQ processor computer (2.6 GHz). To provide statistical robustness, the *k-Fold* technique [20] was implemented, with $k = 5$. Additionally, the test was carried 33 times for each method employed. In order to facilitate the reproducibility of the proposal discussed in this Letter, all the codes for training, testing, and the resulting parameters of all algorithms, besides our simulation campaign numerical data are available in [24].

A. Handover Decision (Classification Problem)

For the classification of the best eNB for HO, three metrics were used for comparison: the average prediction accuracy (the percentage of correct predictions by the classifier), its standard deviation and processing time. The initial results are presented in Tables II and III, for Scenarios 1 and 2, respectively.

We verify the statistical validity of the data obtained from the proposed solution by using the two-sample t-test [25], whose t parameter is given by

$$t = \frac{\bar{G}_1 - \bar{G}_2}{\sqrt{\frac{{s_{G_1}}^2}{l} + \frac{{s_{G_2}}^2}{h}}}, \tag{1}$$

where $\bar{G}_1$ and $\bar{G}_2$ are the means, s_{G_1} and s_{G_2} the standard deviation and h and l are the size of samples G_1 and G_2, respectively. In addition to the evaluation of t, it is also important to infer the hypothesis $H_0 : \bar{G}_1 = \bar{G}_2$ and $H_1 = \bar{G}_1 \neq \bar{G}_2$, where the null hypothesis H_0 indicates that both G_1 and G_2 methods have obtained the same accuracy, while H_1 is the alternative hypothesis which indicates that the accuracy levels are distinct. Given a significance level α_t, the p-value, which is calculated from t-test, represents the lowest possible value to reject H_0 [25]. Values lower than α_t indicates the rejection of H_0 in $(1 - \alpha_t) \times 100\%$ of the cases (i.e., if p-value $< \alpha_t$, the alternative hypothesis H_1 is valid). Here, we consider $\alpha_t = 0.05$.

TABLE IV
T-TEST RESULTS FOR SCENARIO 1.

G_1	G_2	p-value	Low. b.	Upp. b.	H
SVM	ALM	3.56E-07	0.0009	0.0018	1
SVM	**MLP**	**0.425**	**-2.70E-4**	**6.34E-4**	**0**
SVM	SOFL	4.66E-19	0.0055	0.0071	1
SVM	T2FL	1.17E-07	0.0056	0.0105	1
MLP	ALM	2.13E-05	0.0007	0.0017	1
MLP	SOFL	5.12E-19	0.0053	0.007	1
MLP	T2FL	1.83E-07	0.0054	0.0103	1
SOFL	ALM	9.99E-16	-0.0058	-0.0041	1
SOFL	**T2FL**	**0.1803**	**-0.0008**	**0.0042**	**0**
ALM	T2FL	3.39E-06	0.0042	0.0091	1

TABLE V
T-TEST RESULTS FOR SCENARIO 2.

G_1	G_2	p-value	Low. b.	Upp. b.	H
SVM	ALM	1.26E-44	0.1786	0.1871	1
SVM	**MLP**	**0.5019**	**-0.0024**	**0.0048**	**0**
SVM	SOFL	1.66E-9	0.0075	0.013	1
SVM	T2FL	2.47E-44	0.1358	0.1431	1
MLP	ALM	1.86E-60	0.1764	0.1869	1
MLP	SOFL	5.76E-5	0.0049	0.0132	1
MLP	T2FL	7.31E-57	0.1334	0.143	1
SOFL	ALM	6.14E-55	0.1679	0.1773	1
SOFL	T2FL	1.78E-54	0.125	0.1334	1
ALM	T2FL	2.47E-24	-0.0487	-0.0381	1

In this stage, the t-test compares the performance of the adopted strategies for obtaining the greatest accuracy on classifying the best HO target for UE 1. Table IV and V present the evaluations for Scenario 1 and 2, respectively. In both Tables, the p-value is presented, as well as the confidence interval on the difference of the population means, and the hypothesis inferred (0 for H_0 and 1 for H_1).

Based on Tables II and IV, the results indicate that all methods perform optimally when there are no shadowing effects to disturb predictions. However, *SVM* and *MLP* have the best scores and reduced processing time. The p-value of the t-test is greater than α_t only when comparing *SVM* to *MLP* and *SOFL* to *T2FL*, which indicates the validity of the null hypothesis ($H = 0$). Therefore, the t-test demonstrates there is no statistical difference between these algorithms in these cases and it confirms that the best models for this classification task are *SVM* and *MLP*. Moreover, the fuzzy logic-based *ALMMo* also demonstrates excellent accuracy, but it fails to deliver it quickly. We credit the longer time required for T2FLS and ALMMo mainly due to the training process. ALMMo extracts features autonomously, without further parameters and form its structure empirically from the observed data. The T2FLS on the other hand requires larger pre-processing calculations that could affect the training phase.

Furthermore, looking at the results for Scenario 2 on Tables III and V, in which the shadowing effects are present, some algorithms are still achieving reasonable precision, especially *SVM* and *MLP* classifiers, although the accuracy falls considerably (around 13%). Again, they outperform

TABLE VI
REGRESSION RESULTS FOR SCENARIO 1.

Method	MAE	Std. Dev.	Time (s)
MLP	0.27163	0.00800	21.85
KNN	0.12791	0.00490	1.24
RF	0.11987	0.00524	49.34
GBM	0.12452	0.00727	32.98
XGBoost	0.12343	0.00714	25.67
LightGBM	0.11553	0.00317	14.81

TABLE VII
REGRESSION RESULTS FOR SCENARIO 2.

Method	MAE	Std. Dev.	Time (s)
MLP	6.32415	0.07108	8.98
KNN	6.00046	0.07525	1.68
RF	5.99726	0.05373	53.05
GBM	6.19593	0.07841	34.15
XGBoost	5.88485	0.10150	17.53
LightGBM	5.08845	0.04832	13.76

the others on the comparison, and they do not present relevant differences to each other, confirmed by the t-test. However, in this case, it is important to accentuate that fuzzy rule-based *SOFL* classifier was capable of reaching competitive accuracy while requiring the shortest processing time, which is meaningful to the fuzzy logic context.

B. Download Time Estimation (Regression Problem)

For regression, each Scenario demanded a different regression variable. For the first one, the prediction was made for the download duration, since the majority of the downloads are able to be completed within simulation time. However, for the second one, the prediction was made for the percentage of completed download, seeing that it is a more challenging scenario and the majority of downloads could not be completed at the end of 100 s of simulation. Therefore, the analyzed metrics are the mean absolute error (MAE) between the actual download time and the predicted value, the standard deviation, and processing time. The results are presented in Tables VI and VII below, for Scenarios 1 and 2, respectively.

We notice on Table VI that all methods presented a relatively low MAE, being *LightGBM* the most accurate with MAE = 0.11553; and *MLP* being the least accurate with MAE = 0.27163. In Table VII, due to the presence of shadowing, the values of mean absolute error are greater, as expected. Again, the most precise was *LightGBM* and the least precise was *MLP*.

Regarding execution time, *KNN* obtained the best result, possibly explained by the database not being so numerous and the hyperparameter K being considerably small, reducing the computational cost. The second lowest time was presented by *LightGBM*, which has processing speed as an advantage. Differently, *Random Forest* was the slowest, probably due to the number of trees created during training.

The two-sample t-test [25] is applied once again, now seeking statistical differences between regression methods G_1 and G_2. Hence, Tables VIII and IX are obtained for the first and second Scenarios, respectively.

Analyzing Tables VIII and IX, the hypothesis that *XGBoost* and *GBM* are equivalents is rejected in Scenario 1. The same applies to *RF* and *KNN* in Scenario 2. Therefore, it is clear that *LightGBM* is the one that best fits into the database obtained from this simulation campaigns, also presenting a diminished execution time, due to the fact that it has the smallest value for MAE and the t-test confirms there is no other model with equivalent performance. It is worth to mention that *KNN* offers an acceptable performance while requiring an extremely low execution time, which suggests its use for similar applications with real-time regressions.

TABLE VIII
T-TEST RESULTS FOR REGRESSION IN SCENARIO 1.

G_1	G_2	p-value	Low. b.	Upp. b.	H
XGB	Light	6.85E-7	0.0052	0.0106	1
XGB	**GBM**	**0.5785**	**-0.0047**	**0.0027**	**0**
XGB	MLP	8.11E-40	-0.1530	-0.1406	1
XGB	RF	0.0493	-0.0014	0.0063	1
XGB	KNN	0.0049	-0.0076	-0.0014	1
Light	GBM	1.88E-7	-0.0118	-0.0061	1
Light	MLP	2.16E-34	0.1605	0.1489	1
Light	RF	9.43E-6	0.0068	-0.0029	1
Light	KNN	5.89E-17	-0.0143	-0.0104	1
GBM	MLP	2.16E-40	-0.1520	-0.1395	1
GBM	RF	0.0128	0.0009	0.0073	1
GBM	KNN	0.0347	-0.0067	-0.0002	1
MLP	RF	1.30E-36	0.1439	0.1558	1
MLP	KNN	8.43E-36	0.1364	0.1486	1
RF	KNN	4.92E-8	0.0051	0.0100	1

TABLE IX
T-TEST RESULTS FOR REGRESSION IN SCENARIO 2.

G_1	G_2	p-value	Low. b.	Upp. b.	H
XGB	Light	1.99E-37	0.7565	0.8363	1
XGB	GBM	3.00E-20	-0.3563	-0.2658	1
XGB	MLP	1.13E-27	-0.4830	-0.3956	1
XGB	RF	1.19E-6	-0.1576	-0.0731	1
XGB	KNN	2.71E-6	-0.1602	-0.0710	1
Light	GBM	1.95E-53	-1.1401	-1.0748	1
Light	MLP	3.94E-60	-1.2661	-1.2053	1
Light	RF	3.51E-57	-0.9399	-0.8836	1
Light	KNN	1.42E-50	-0.9437	-0.8803	1
GBM	MLP	3.50E-9	-0.1656	-0.0908	1
GBM	RF	3.95E-16	0.1602	0.2314	1
GBM	KNN	5.43E-15	0.1571	0.2338	1
MLP	RF	3.89E-28	0.2904	0.3576	1
MLP	KNN	2.75E-26	0.2871	0.3602	1
RF	**KNN**	**0.9867**	**-0.0344**	**-0.0350**	**0**

V. CONCLUSIONS

Since 3GPP HO management relies basically on power level comparisons, several inefficiencies arise during such procedures. In this context, we presented a user-level simulation-based performance analysis of algorithms for classification and regression applications in 3GPP networks. The classification aims to predict the best HO target, whereas the regression estimates the download time and its completed percentage. Classical computational intelligence approaches, such as *KNN*, *MLP*, *SVM* and also recent fuzzy logic systems and latter gradient boosting machines were implemented.

The results indicate a valuable performance even in adverse propagation conditions while requiring short processing time. For classification, *SVM* and *MLP* have the best performance, although the fuzzy system *SOFL* has similar accuracy with lower processing time. Regarding the regression applications, *LightGBM* is certainly the one that best adapts to these work conditions and presents minor mean absolute error and processing time. However, it is worth to mention that *KNN* offers extremely low time values.

REFERENCES

[1] *3GPP TS 36.133*. 3GPP, 2009, Evolved Universal Terrestrial Radio Access (E-UTRAN); Requirements for Support of Radio Resource Management.

[2] J.-h. Bang, S. Oh, K. Kang, and Y.-J. Cho, "A bayesian regression based lte-r handover decision algorithm for high-speed railway systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 10 160–10 173, 2019, DOI: https://doi.org/10.1109/TVT.2019.2935165.

[3] L. Yan, H. Ding, L. Zhang, J. Liu, X. Fang, Y. Fang, M. Xiao, and X. Huang, "Machine Learning-Based Handovers for Sub-6 GHz and mmWave Integrated Vehicular Networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 10, pp. 4873–4885, 2019, DOI: https://doi.org/10.1109/TWC.2019.2930193.

[4] R. Zhohov, A. Palaios, H. Rydén, R. Moosavi, and J. Berglund, "Reducing latency: Improving handover procedure using machine learning," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*. IEEE, 2021, pp. 1–5, DOI: https://doi.org/10.1109/VTC2021-Spring51267.2021.9448875.

[5] M. Saeed, H. Kamal, and M. El-Ghoneimy, "Novel type-2 fuzzy logic technique for handover problems in a heterogeneous network," *Engineering Optimization*, vol. 50, no. 9, pp. 1533–1543, 2018, DOI: https://doi.org/10.1080/0305215X.2017.1402012.

[6] A. Abdelmohsen, M. Abdelwahab, M. Adel, M. S. Darweesh, and H. Mostafa, "LTE Handover Parameters Optimization Using Q-Learning Technique," in *2018 IEEE 61st International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2018, pp. 194–197, DOI: https://doi.org/10.1109/MWSCAS.2018.8623826.

[7] T. Elmahdy and A. F. Bendary, "Novel technique in 4G Handover parameter tuning and prediction using statistical trend analysis and supervised machine learning," in *2021 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE, 2021, pp. 1–5, DOI: https://doi.org/10.1109/ISNCC52172.2021.9615773.

[8] H. Xia, X. Wei, Y. Gao, and H. Lv, "Traffic Prediction Based on Ensemble Machine Learning Strategies with Bagging and LightGBM," in *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2019, pp. 1–6, DOI: https://doi.org/10.1109/ICCW.2019.8757058.

[9] X. Gu, Y. Han, and J. Yu, "A Novel Lane-Changing Decision Model for Autonomous Vehicles Based on Deep Autoencoder Network and XGBoost," *IEEE Access*, vol. 8, pp. 9846–9863, 2020, DOI: https://doi.org/10.1109/ACCESS.2020.2964294.

[10] M. Tayyab, X. Gelabert, and R. Jäntti, "A Survey on Handover Management: From LTE to NR," *IEEE Access*, vol. 7, pp. 118 907–118 930, 2019, DOI: https://doi.org/10.1109/ACCESS.2019.2937405.

[11] A. Stamou, N. Dimitriou, K. Kontovasilis, and S. Papavassiliou, "Autonomic Handover Management for Heterogeneous Networks in a Future Internet Context: A Survey," *IEEE Communications Surveys Tutorials*, vol. 21, no. 4, pp. 3274–3297, 2019, DOI: https://doi.org/10.1109/COMST.2019.2916188.

[12] T. Cabral de Brito Guerra, Y. Dantas, and V. Sousa Jr, "A Machine Learning Approach for Handover in LTE Networks with Signal Obstructions," *Journal of Communication and Information Systems*, vol. 35, no. 1, pp. 271–289, Nov. 2020, DOI: https://doi.org/10.14209/jcis.2020.28.

[13] Z. Ali, N. Baldo, J. Mangues-Bafalluy, and L. Giupponi, "Simulating LTE Mobility Management in Presence of Coverage Holes with ns-3," in *Proceedings of the 8th International Conference on Simulation Tools and Techniques*, ser. SIMUTools '15. Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2015, p. 279–283, DOI: https://doi.org/10.4108/eai.24-8-2015.2260998.

[14] P. P. Angelov and X. Gu, "Empirical fuzzy sets," *International Journal of Intelligent Systems*, vol. 33, no. 2, pp. 362–395, 2018, DOI: https://doi.org/10.1002/int.21935.

[15] P. P. Angelov, X. Gu, and J. C. Príncipe, "Autonomous learning multimodel systems from data streams," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 4, pp. 2213–2224, 2018, DOI: https://doi.org/10.1109/TFUZZ.2017.2769039.

[16] X. Gu and P. P. Angelov, "Self-organising fuzzy logic classifier," *Information Sciences*, vol. 447, pp. 36–51, 2018, DOI: https://doi.org/10.1016/j.ins.2018.03.004.

[17] C. PH, R. MG, A. RP, V. MM, T. R, and de Aguiar EP, "An enhanced aircraft engine gas path diagnostic method based on upper and lower singleton type-2 fuzzy logic system," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 41, pp. 1–14, 2019, DOI: https://doi.org/10.1007/s40430-019-1567-4.

[18] E. P. de Aguiar, R. P. Amaral, M. M. Vellasco, and M. V. Ribeiro, "An enhanced singleton type-2 fuzzy logic system for fault classification in a railroad switch machine," *Electric Power Systems Research*, vol. 158, pp. 195–206, 2018, DOI: https://doi.org/10.1016/j.epsr.2017.12.018.

[19] S. Aghabozorgi, A. Bayati, K.-K. Nguyen, C. Despins, and M. Cheriet, "Toward Predictive Handover Mechanism in Software-Defined Enterprise Wi-Fi Networks," in *2019 IEEE Sustainability through ICT Summit (StICT)*, 2019, pp. 1–6, DOI: https://doi.org/10.1109/STICT.2019.8789369.

[20] S. S. Haykin, *Neural networks and learning machines*, 3rd ed. Pearson Education, 2009.

[21] R. A. Campos, R. P. F. Amaral, N. Soares, L. G. da Fonseca, M. L. L. Júnior, and E. P. de Aguiar, "A new model to distinguish welds performed by short-circuit GMAW based on FRESH algorithm and MLP ANN," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 41, 03 2019, DOI: https://doi.org/10.1007/s40430-019-1607-0.

[22] L. Zhang, Q. Ni, M. Zhai, J. Moreno, and C. Briso, "An Ensemble Learning Scheme for Indoor-Outdoor Classification Based on KPIs of LTE Network," *IEEE Access*, vol. 7, pp. 63 057–63 065, 2019, DOI: https://doi.org/10.1109/ACCESS.2019.2914451.

[23] N. Nesa, T. Ghosh, and I. Banerjee, "Outlier detection in sensed data using statistical learning models for IoT," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, 2018, pp. 1–6, DOI: https://doi.org/10.1109/WCNC.2018.8376988.

[24] J. P. S. H. Lima, "Notebooks GitHub Repository." [Online]. Available: https://github.com/jpshlima/systems_notebooks

[25] P. G. Moore, "The two-Sample t-test based on range," *Biometrika*, vol. 44, no. 3-4, pp. 482–489, 12 1957, DOI: https://doi.org/10.1093/biomet/44.3-4.482.