O-RAN Intelligent Application for Cellular Mobility Management

1st Baud Haryo Prananto
School of Electrical Engineering and Informatics
Bandung Institute of Technology
Bandung, Indonesia
baud.prananto@students.itb.ac.id

2nd Iskandar
School of Electrical Engineering and Informatics
Bandung Institute of Technology
Bandung, Indonesia
iskandar@stei.itb.ac.id

3rd Adit Kurniawan
School of Electrical Engineering and Informatics
Bandung Institute of Technology
Bandung, Indonesia
adit@stei.itb.ac.id

2022 International Conference on ICT for Smart Society (ICISS) | 978-1-6654-7135-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICISS55894.2022.9915221

***Abstract*—Machine Learning (ML) is gaining a more important role in cellular networks. Mobility management can be improved using ML due to its complexity and criticality to network performance. Managing mobility while maintaining the network connection is very important because the user's movement may disrupt their data transmission. In some cases, the traditional mobility management algorithm is not reliable enough. Here is where ML may provide a more intelligent mobility management algorithm. ML implementation in the cellular network requires a major modification in the network element software logic and architecture. This may cause difficulties in real-world implementation. Open Radio Access Network (O-RAN) consortium provides a modular solution to implement ML algorithms by adding an optional Radio Intelligent Controller (RIC) to host the intelligent application without major modification to the existing network elements. Using this RIC, a lot of intelligent use cases can be implemented modularly in the network. In this paper, we prove that the ML algorithm can be used to improve mobility management in some particular cases. As the experiment results, we demonstrated that machine learning has superior performance compared to the traditional handover algorithm.**

Keywords—Cellular, Mobility, Machine Learning, O-RAN,LTE, 5G

I. Introduction

The service area of a cellular mobile telecommunication network is covered by multiple interconnecting service coverages instead of just one single service coverage. These coverages are called cells, hence the name. This is the main characteristic of a cellular network since its first generation in the 1980s (AMPS, NMT), followed by the second generation (GSM), and ultimately the currently emerging fifth generation (5G/NR). As a mobile telecommunication system, these systems also enable the user's movement since they are connected wirelessly to the network through cells. The movement of the user requires the user to jump from one cell to another cell to be still connected to the network, this activity is called handover. Handover can be defined as the process that prevents ongoing communication from getting interrupted as the mobile equipment changes its attachment point such as cells [1]. During handover, some disruptions may occur in the active communication due to packet losses and delays. These disruptions may result in a significant loss of performance, especially in Ultra-Reliable Low-Latency Communications (URLLC) applications such as mobile automation.

The fifth-generation cellular system uses a higher frequency spectrum and this results in smaller cell coverage [2]. The higher network capacity requirement also mandates the cells to be smaller and denser. Smaller cells mean more frequent handovers in high mobility users, and more handovers may result in more traffic disruption if not managed properly. The disturbance caused by handover may be calculated in the terms of Mobility Interruption Time (MIT).

3GPP defines MIT as the shortest time duration supported by the system during which a user terminal cannot exchange user plane packets with any base station during transitions [3]. MIT consists mainly of Handover Interruption Time (HIT), interruption time in a successful handover, and Handover Failure Time (HOF), interruption time in a failed handover [4]. Based on the measurements in the LTE network, HIT is reported at around 50 milliseconds, while HOF ranges from several hundred milliseconds to a few seconds [5]. From this fact, it can be concluded that failed handovers contributed more to MIT compared to successful ones. Therefore, MIT can be more significantly reduced by reducing the probability of handover failure by avoiding unnecessary handovers and making sure handovers are to the correct target cells.

A lot of research has already been done to improve MIT by reducing handover failure probability. Some of the most prominent methods are fast measurements [6], dual connectivity [7], conditional handover [8]–[10], and predictive handover [9], [11]–[15]. This paper utilizes the predictive handover method, which is predicting the target cell in advance using various techniques on top of the legacy measurement-based algorithm.

In legacy or traditional handover algorithm, the target cell is determined by the air interface measurement (typically RSRP, signal strength and/or RSRQ, signal quality) of the current cell and neighbor cells reported by the user equipment (UE). The serving base station will then determine the target cell based on those measurements. In a predictive handover algorithm, the target cell may be predicted based on the other methods such as user behavior or network condition and may employ

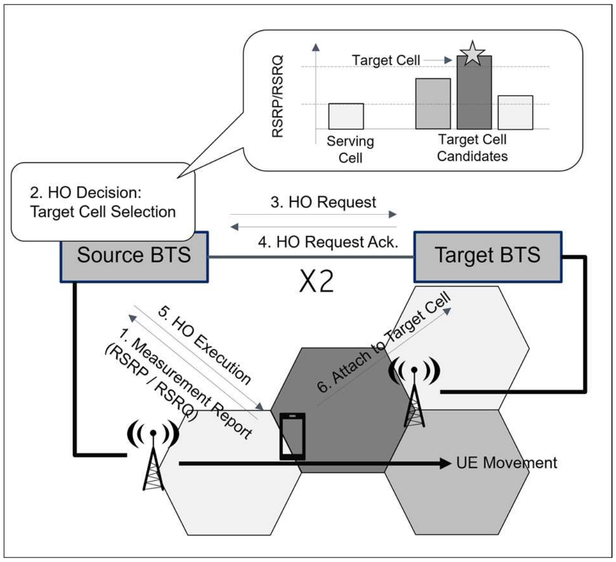

Fig. 1. Traditional measurement-based handover algorithm.

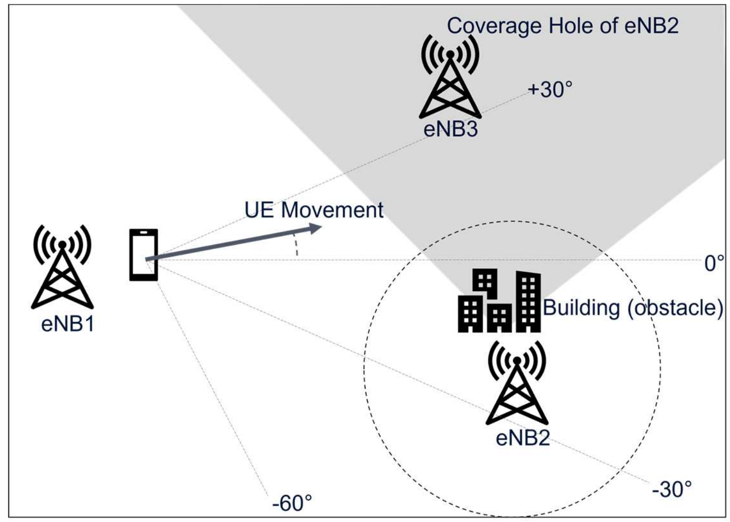

Fig. 2. Simulation design, a network with 3 eNBs and a coverage hole.

machine learning techniques.

Machine learning (ML) is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed [16]. It studies the computer algorithms that improve automatically through experience [17]. This research uses predictive handover with an ML algorithm to determine the best target cell.

II. Machine Learning for Mobility Management

Many types of research apply ML/AI for improving handover [18] usually predictive handover method. There are several approaches for ML: supervised learning, unsupervised learning, and reinforcement learning [19]. Supervised learning is the ML task of learning a function that maps an input to an output based on example input-output pairs [20]. Unsupervised learning looks for previously undetected patterns in a data set with no pre-existing labels and with a minimum of human supervision [21]. Reinforcement learning is concerned with how software agents ought to take actions in an environment to maximize the notion of cumulative reward [22].

Supervised learning is widely used for handover improvement. Neural networks method is one of the most popular technique used in several researches [9], [23]–[27]. Some studies uses support vector machine [28] and K-nearest neighbor [29], [30]. Unsupervised learning techniques are also used by some studies, for example, K-means [14], [31] and long short-term memory [12]. Reinforced learning is used by some researchers that usually employ Q-learning algorithms [11], [32].

III. Neural Network Simulation Experiment

In this experiment, we proved that ML-based handover performs better than traditional measurement-based handover. In the traditional handover, the target cell is determined by the serving base station based on the measurement report of serving and neighbor cells sent by the UE (in the form of RSRP and/or RSRQ). The target cell is simply the strongest cell among all, or any cell with a certain threshold better than the serving cell (Fig.1).

ML can process the measurement report more intelligently. In our simulation, based on previous research [23], we determined the target cell using a neural network based on the measurement report sent by UE. In the ideal condition, the traditional handover algorithm works well and is reliable. However, in a non-ideal condition such as the presence of a coverage hole obstructed by a building, it faces some performance degradation [33].

We simulated handover cases in an environment containing coverage holes. As the simulation result, we showed that ML-based handover performs better than traditional handover. This is because the ML-based algorithm can "learn" the behavior of the previous handover cases and may calculate the presence of a coverage hole. In the simulation, we create an environment containing 3 eNBs using the Lena NS3 simulator [34] (Fig.2). We put a building near eNB2 to create a coverage hole. The UE will move, while downloading a file, on every simulation run with a random trajectory angle. Due to this movement, the UE will perform a handover to either eNB2 or eNB3 using the traditional handover algorithm. We also monitor the download status for every case. After 1000 simulation runs, some downloads were successfully performed using the traditional handover algorithm (86%) and the rest (14%) failed (Fig.3).

We took the RSRP and RSRQ measurement reports for every simulation run and put them as the input of our neural network simulation using MATLAB (Fig.4). The result of the neural network is a recommendation, on whether it is good to perform handover to eNB2/eNB3 or not. The result is if the UE follows the neural network recommendation, not the traditional handover decision, more successful download cases happen (Fig.3). However, this neural network success rate depends on

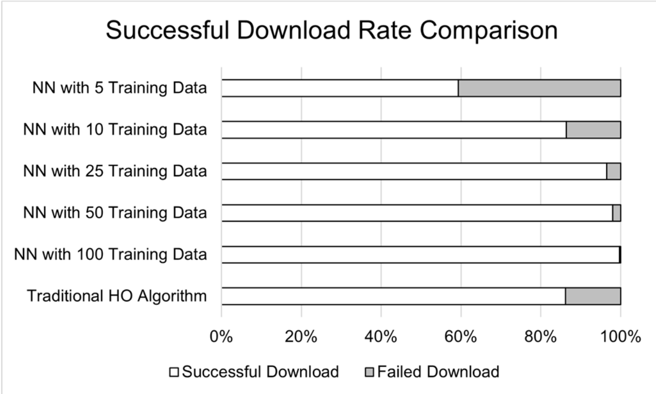

Fig. 3. Comparison of traditional algorithm and MATLAB NN simulations.

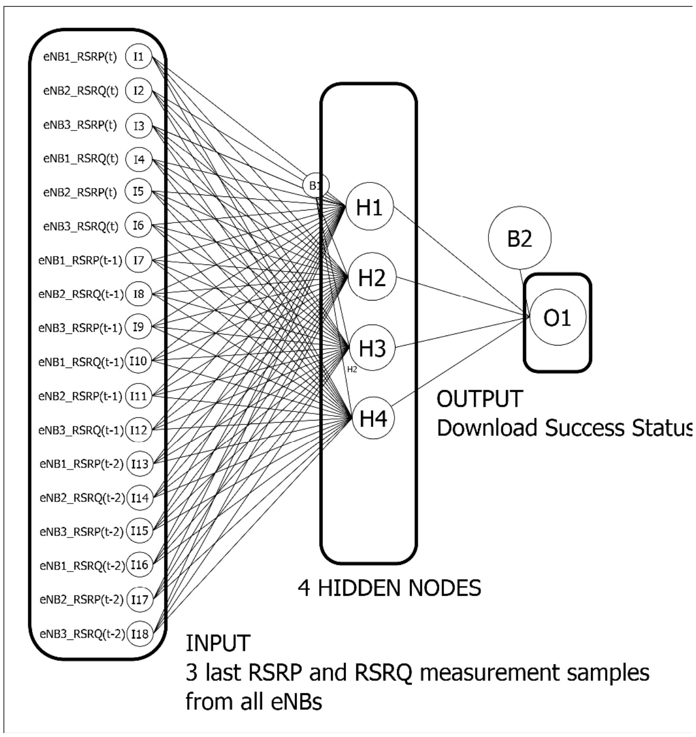

Fig. 4. Neural network design for MATLAB simulation.

the size of the training data. The ML-based handover will be better if we have more than 10 training data. The training data was taken earlier from the same simulation environment.

IV. O-RAN Radio Intelligent Controller

ML is originally not part of the cellular system Radio Access Network (RAN) standard architecture, all algorithm logic is explicitly programmed in the network elements such as base stations. Since a lot of research presents the benefit of ML and network intelligence in a cellular network, the network vendors and service providers already propose some workaround on top of the standard architecture to implement them. The earliest candidate of ML implementation can be seen in the Self Organizing Network (SON) standard released in LTE 3GPP Release-8. 3GPP has recently started a new Release-17 study on the applications of AI/ML to RAN [35].

A newly formed O-RAN Alliance is committed to evolving radio access networks making them more open and smarter than previous generations [36]. This consortium standardizes a new network function called Radio Intelligent Controller (RIC) to run the network intelligent functions of the RAN (Fig.5). This RIC is divided into Non-Real Time (Non-RT RIC) and Near Real-Time (Near-RT RIC). As the name implies, the two variants serve different intelligent functions based on the required response of the logic.

Non-RT RIC is installed as an Open Networking Automation Platform (ONAP). Usually, it is located in the centralized Management and Orchestration (MANO) or Network Management System (NMS), the management subsystem that a service provider already has in the legacy standard. This Non-RT RIC hosts an intelligent function that doesn't require an immediate response or long-term functions and policies.

Near-RT RIC is a new network function that is installed in an Edge Cloud platform. The fast-response nature of the function requires it to be installed near the RAN, while a centralized cloud environment cannot provide the required latency requirement. The Near-RT RIC can communicate with RAN using a newly standardized E2 interface. Using this interface, it can perform immediate control to RAN such as mobility management control.

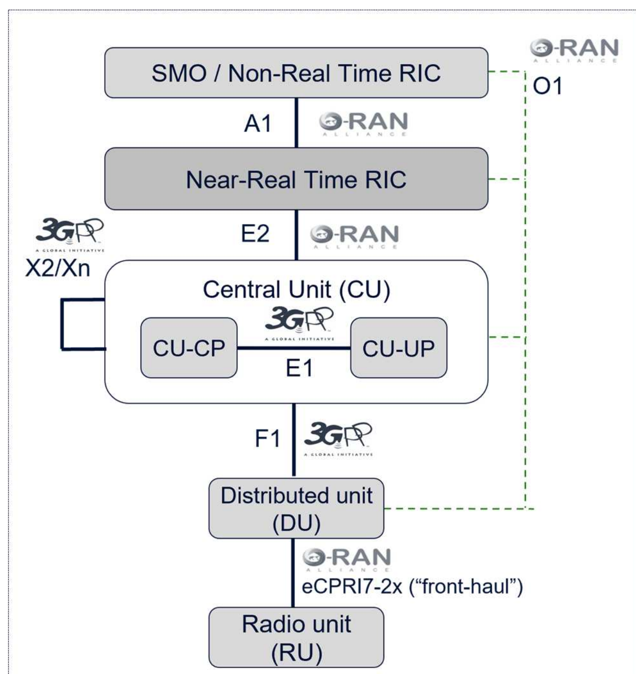

Fig. 5. O-RAN Architecture.

The ML algorithm implementations are usually done by improving the internal algorithm in the base stations. Therefore to implement them, there must be a major modification in the base station software. Our research aims to minimize this major modification by outsourcing the ML algorithm to an external network element. We focus on improving handover performance using predictive handover by employing an ML algorithm. As a mobility management function, this algorithm requires near real-time response and therefore is hosted in Near-RT RIC.

V. Near-RT RIC Implementation

With the introduction of RIC in the O-RAN standard, there is a possibility to implement ML to RAN without heavily modifying the existing base stations. The ML algorithm can also be implemented modularly without disturbing the existing base station software.

O-RAN is a relatively novel standard and currently introduces several use cases for open and intelligent RAN [37]. However, the exact implementation of the use case is given to specific vendors. For example, Nokia prioritizes Traffic Steering and Network Anomaly Detection use case for its RIC solution [38].

Our research implements ML-based applications in Near-RT RIC. We created a Near-RT RIC using software provided by O-RAN Software Community (O-RAN SC) [39]. The Near-RT RIC is a very versatile open-source software that can be installed in various environments. In our implementation, we installed the Near-RT RIC on a Linux Ubuntu virtual machine (Fig.6).

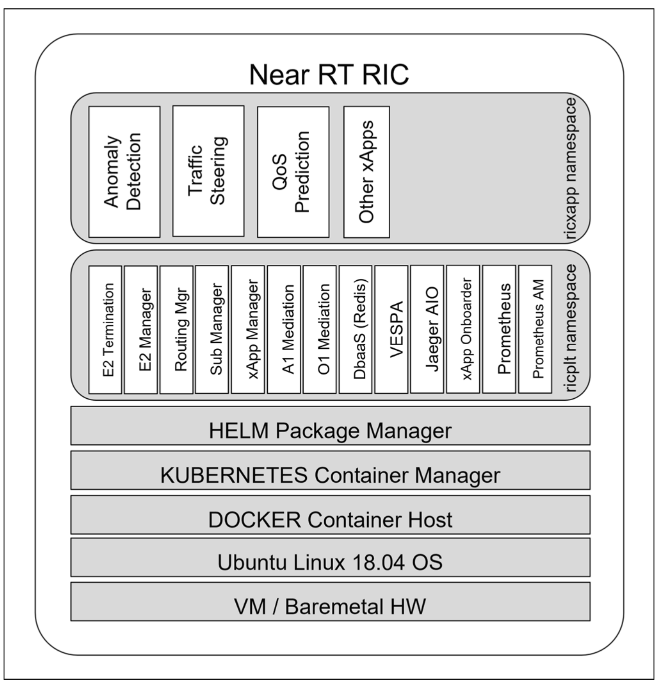

Fig. 6. Near-RT RIC Architecture.

Near-RT RIC utilizes containerization technologies that employ Docker as the container host, Kubernetes as the container manager, and Helm as the package manager. We installed Anomaly Detection use case xApp on top of the Near-RT RIC platform [40]. The programs of this use case are provided by

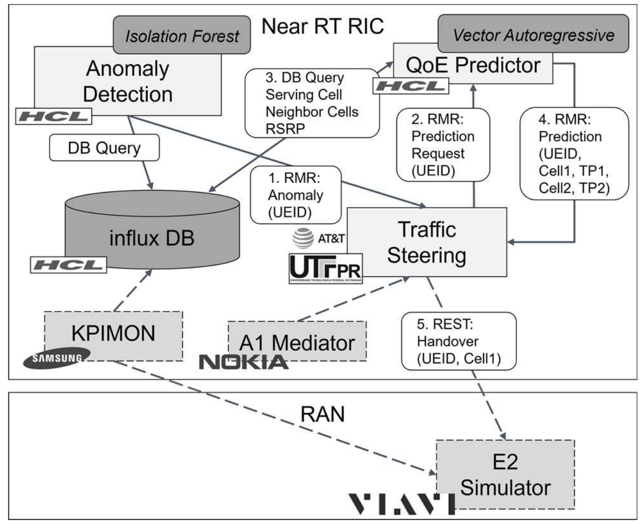

Fig. 7. Anomaly Detection Use Case.

several companies, as members of the O-RAN SC, and are available as open-source software (Fig.7).

This Anomaly Detection use case consists of three xApps: Anomaly Detection, Traffic Steering, and QoE Predictor. The goal of this use case is to detect some anomalous condition in the UE (such as degradation of radio condition) and perform an action to RAN. The flow starts with the Anomaly Detection xApp reviews the UE data. In this preliminary stage of our research, the data was taken from a database. When an anomaly is detected, the Anomaly Detection xApp will send a notification to Traffic Steering xApp using the RMR protocol, an internal Near-RT RIC protocol. Traffic Steering xApp will send a request to QoE Predictor xApp that will provide the prediction of the target cell to get the best quality of service. After that, the Traffic Steering xApp will send commands to the RAN, such as a handover command using the REST API interface.

Ideally, the use case should involve some other xApps which is not present in this demonstration. KPIMON shall provide the UE data, but in this demonstration, the UE data is already provided in the database. The E2 Simulator acts as the RAN but this application is replaced by a dummy destination.

In the original provided software, the algorithm and training data of QoE Predictor xApp didn't consider the UE movement. It considers only the static improvement of cells throughput (i.e., which cells are getting better, regardless of UE movement). This will result in the same prediction for all UE cases.

QoE Predictor xApp predicts the cell throughput using Vector Autoregressive (VAR), a statistical method to forecast time-series data. It learns from past throughput data as training data to predict the future throughput.

As a contribution to the software by this research, we add RSRP measurement from UE as additional consideration for QoE Predictor xApp to predict the cells' throughput. This will put the UE movement into consideration for the throughput prediction. We also test our modification with

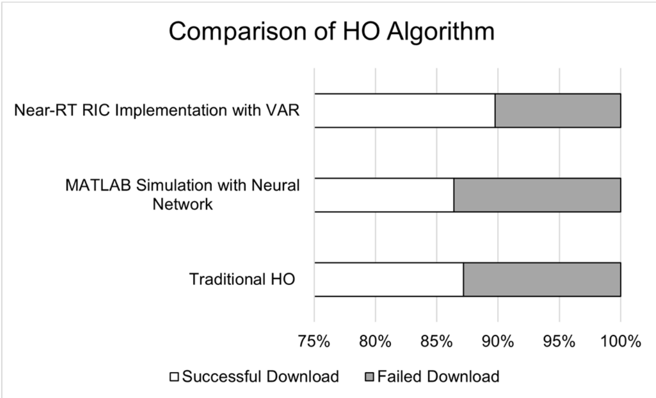

Fig. 8. Comparison of HO Algorithms.

the same simulation data run in NS3 as described in the previous experiment. Finally, we can compare the result of our experiments in Fig.8.

The VAR algorithm in Near-RT RIC provides better target cell prediction for the handover process compared to the traditional handover algorithm. To compare with our previous MATLAB neural network simulation, we took the result from 10 training data because it is the amount of training data used by VAR. The QoE Predictor xApp predicts the cell throughput based on the previous 10 throughput data using VAR.

It was studied that more than 10 training data in a neural network can perform much better. Therefore, in our future work, we plan to implement a neural network in Near-RT RIC to replace VAR.

VI. Conclusion and Future Works

In this paper, we showed that the ML algorithm gives better performance in mobility management, in this case, handover, compared to the traditional algorithm. This benefit is more significantly shown in the presence of a coverage hole where the traditional handover algorithm cannot perform in ideal conditions.

We have shown the improvement using two experiments: MATLAB simulation using neural network and Near-RT RIC implementation using VAR. Both experiments showed improvement in the handover performance with a better successful handover rate in the network environment with a coverage hole.

The neural network method provides better performance along with more training data. Therefore for the implementation of this research, we plan to replace the original VAR algorithm in Near-RT RIC with a neural network and better training data.

References

[1] M. S. Obaidat and P. Nicopolitidis, *Smart Cities and Homes: Key Enabling Technologies*, 2016.

[2] 3GPP, "NR ; User Equipment (UE) radio transmission and reception; part 2: Range 2 standalone-Rel. 16," 2020.

[3] 3GPP TR 38.913, "Study on Scenarios and Requirements for Next Generation Access Technologies (Release 16)," 2020.

[4] H. S. Park, Y. Lee, T. J. Kim, B. C. Kim, and J. Y. Lee, "Handover Mechanism in NR for Ultra-Reliable Low-Latency Communications," *IEEE Network*, vol. 32, no. 2, pp. 41–47, 2018.

[5] 3GPP TR 36.881, "Evolved Universal Terrestrial Radio Access (E-UTRA) Study on latency reduction techniques for LTE (Release 14)," 2016.

[6] A. Chincholi, M. Menon, and L. Hsu, "Measurement Gap Enhancements for BL/CE UEs," 2019.

[7] M. Polese, M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, "Improved Handover Through Dual Connectivity in 5G mmWave Mobile Networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 9, pp. 2069–2084, 2017.

[8] H. Martikainen, I. Viering, A. Lobinger, and T. Jokela, "On the Basics of Conditional Handover for 5G Mobility," *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC*, vol. 2018-Septe, no. 2, 2018.

[9] C. Lee, H. Cho, S. Song, and J.-m. Chung, "Prediction-Based Conditional Handover for 5G mm-Wave Networks: A Deep-Learning Approach," *IEEE Vehicular Technology Magazine*, vol. 15, no. March, pp. 54–62, 2020.

[10] I. L. Da Silva, C. Eklöf, J. Muller, and R. Zhohov, "This is the key to mobility robustness in 5G networks." [Online]. Available: https://www.ericsson.com/en/blog/2020/5/the-key-to-mobility-robustness-5g-networks

[11] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, "Reinforcement learning based predictive handover for pedestrian-aware mmWave networks," *INFOCOM 2018 - IEEE Conference on Computer Communications Workshops*, pp. 692–697, 2018.

[12] C. Wang, L. Ma, R. Li, T. S. Durrani, and H. Zhang, "Exploring Trajectory Prediction Through Machine Learning Methods," *IEEE Access*, vol. 7, pp. 101 441–101 452, 2019.

[13] R. V. Akhpashev and V. G. Drozdova, "Spatial interpolation of LTE measurements for minimization of drive tests," *International Conference of Young Specialists on Micro/Nanotechnologies and Electron Devices, EDM*, vol. 2018-July, pp. 136–138, 2018.

[14] A. Suresh Kumar, S. Vanmathi, B. Praveen Sanjay, S. Ramya Bharathi, and M. Sakthi Meena, "Handover forecasting in 5G using machine learning," *International Journal of Engineering and Technology(UAE)*, vol. 7, no. 2, pp. 76–79, 2018.

[15] A. Masri, T. Veijalainen, H. Martikainen, J. Ali-tolppa, S. Mwanje, M. Kajo, and N. B. Labs, "Machine Learning Based Predictive Handover."

[16] Expert.AI, "What is Machine Learning? A definition - Expert System — Expert.ai," 2020. [Online]. Available: https://www.expert.ai/blog/machine-learning-definition/

[17] T. Mitchell, *Machine Learning*. McGraw Hill, 1997.

[18] P. V. Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A Survey of Machine Learning Techniques Applied to Self-Organizing Cellular Networks," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 4, pp. 2392–2431, 2017.

[19] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[20] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Prentice Hall, 2010.

[21] G. Hinton and T. Sejnowski, *Unsupervised Learning: Foundations of Neural Computation*. MIT Press, 1999.

[22] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement Learning: A Survey," *Journal of Artificial Intelligence Research*, vol. 4, no. 9, pp. 237–285, may 1996. [Online]. Available: http://arxiv.org/abs/cs/9605103 https://www.jair.org/index.php/jair/article/view/10166

[23] Z. Ali, N. Baldo, J. Mangues-Bafalluy, and L. Giupponi, "Machine learning based handover management for improved QoE in LTE," *Proceedings of the NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, no. 5, pp. 794–798, 2016.

[24] N. Sinclair, D. Harle, I. A. Glover, J. Irvine, and R. C. Atkinson, "An advanced SOM algorithm applied to handover management within lte," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 1883–1894, 2013.

[25] N. M. Alotaibi and S. S. Alwakeel, "A neural network based handover management strategy for heterogeneous networks," *Proceedings - 2015 IEEE 14th International Conference on Machine Learning and Applications, ICMLA 2015*, pp. 1210–1214, 2016.

[26] M. A. F. Rihani, M. Mroue, J. C. Prevotct, F. Nouvel, and Y. Mohanna, “A Neural Network Based Handover for Multi-RAT Heterogeneous Networks with Learning Agent,” *Proceedings of the 13th International Symposium on Reconfigurable Communication-Centric Systems-on-Chip, ReCoSoC 2018*, 2018.

[27] B. Shubyn, N. Lutsiv, O. Syrotynskyi, and R. Kolodii, “Deep Learning based Adaptive Handover Optimization for Ultra-Dense 5G Mobile Networks,” *Proceedings - 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET 2020*, pp. 869–872, 2020.

[28] X. Chen, F. Meriaux, and S. Valentin, “Predicting a user’s next cell with supervised learning based on channel states,” *IEEE Workshop on Signal Processing Advances in Wireless Communications, SPAWC*, pp. 36–40, 2013.

[29] 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.

[30] T. C. d. B. Guerra, “Machine Learning Based Handover Management for LTE Networks with Coverage Holes,” Ph.D. dissertation, Universidade Federal do Rio Grande do Norte, 2018.

[31] L. L. Vy, L. P. Tung, and B. S. P. Lin, “Big data and machine learning driven handover management and forecasting,” *2017 IEEE Conference on Standards for Communications and Networking, CSCN 2017*, no. September 2019, pp. 214–219, 2017.

[32] V. Yajnanarayana, H. Rydén, and L. Hévizi, “5G Handover using Reinforcement Learning,” pp. 1–6, 2019. [Online]. Available: http://arxiv.org/abs/1904.02572

[33] Z. Ali, N. Baldo, J. Mangues-Bafalluy, and L. Giupponi, “Simulating LTE mobility management in presence of coverage holes with ns-3,” *SIMUTOOLS 2015 - 8th EAI International Conference on Simulation Tools and Techniques*, 2015.

[34] NS3, “Design Documentation — Model Library.” [Online]. Available: https://www.nsnam.org/docs/models/html/lte-design.html#handover

[35] G. Masini, Y. Gao, and S. Sirotkin, “Artificial Intelligence and Machine Learning in NG-RAN: New Study in RAN3,” 2021. [Online]. Available: https://www.3gpp.org/news-events/2201-ai_ml_r3

[36] O-RAN, “O-RAN : Towards an Open and Smart RAN. White Paper. October 2018,” no. October, 2018. [Online]. Available: https://www.o-ran.org/s/O-RAN-WP-FInal-181017.pdf

[37] A. Akman, “O-RAN Working Group 1 Use Cases Detailed Specification,” 2020.

[38] Nokia, “Nokia launches first commercial Service Enablement Platform to drive Open RAN innovation.” [Online]. Available: https://www.nokia.com/about-us/news/releases/2021/03/11/nokia-launches-first-commercial-service-enablement-platform-to-drive-open-ran-innovation/

[39] F. Cefalu, “O-RAN SC Wiki: Getting Started,” 2020. [Online]. Available: https://wiki.o-ran-sc.org/display/GS/Getting+Started

[40] D. Karnwal, “Anomaly Detection Use Case,” 2021. [Online]. Available: https://wiki.o-ran-sc.org/display/RICP/Anomaly+Detection+Use+Case