Mini Project Report

on

Different Strategies for Optimal Network Selection

Submitted by

Kanishk Bajpai (20bec020)

Kavya Garg (20bcs069)

Under the guidance of

Dr. Mukesh Kumar Mishra

Assistant Professor

Department of Electronics and Communication Engineering

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY DHARWAD

Contents

1 Introduction

The project focuses on utilizing opportunistic networks (ON) and software-defined networking (SDN) to enhance the capacity and performance of wireless networks, particularly in the context of increasing data traffic from mobile devices. The Macro Base Station (MBS) in the ON configuration collects a significant amount of data traffic, making it necessary to offload some users to nearby small cells to ensure satisfactory service. To facilitate this offloading process, the MBS offers optimal financial incentives to access points (APs) and selects users strategically to achieve maximum offloading. The proposed approach combines SDN and the Stackelberg game model to optimize the selection of users and achieve efficient offloading, prioritizing users with the least service quality.

A Comparative Study of Network Selection Algorithms for Wireless Communication Systems using GPS Technology proposes an network selection algorithm. The algorithm provided is a simulation and visualization of a network selection process. The purpose of the project is to provide a tool for network planners and decision-makers to compare and select the most appropriate wireless network based on several attributes, such as technology, bandwidth, energy efficiency, delay jitter, and price. The algorithm generates a list of networks and calculates a score for each network based on the attributes provided. The networks are then sorted based on their scores, and the sorted list is presented to the user. In addition to network selection, the algorithm also includes functionality to calculate distances between coordinates on Earth using the Haversine formula and to visualize the network and base station coordinates on a map using the Folium library. These features could be useful in planning and optimizing the coverage area of the selected network. The project aims to provide a tool for decision-makers to select the most appropriate wireless network based on various attributes, while also providing functionality to aid in the planning and optimization of the network's coverage area.

2 Related Works

In [1], Several Software Defined Networking (SDN) based architectures have been proposed for wireless networks. Among them, the proposal of Reconfigurable Base Station (RBS) pro- grammable with different Radio Access Technologies (RATs) dynamically by an SDN con- troller has the potential to improve mobility.This paper explains us briefly about the use of SDN for Network Selection Strategy using Reconfigurable Base Station. In[2], The authors have developed an evaluation platform to compare the performance of our proposal with the standard 3GPP architecture. Results demonstrate significant gains in the network performance of 5G-Flow RAN over the existing 3GPP 5G network. In [2] presents a novel SDN assisted architecture for futuristic wireless networks which augments network capacity on need basis using unsupervised Machine Learning (ML) to create ON cells with appropriate RAT. Subsequently, we define utilities for the Wireless Network Infrastructure (WNI) and the User Equipment (UE) to evaluate the benefit of creation of ON cells. A game theoretic model is developed to understand the strategies of the two players, i.e., WNI and UE, while using the ON cell resources. As the consumption of mobile data among the users is hiking, it is not possible to handle the entire traffic by the base stations. Hence, it is required to offload the abundant traffic to the small cells. Though offloading pertains to a solution, many challenges are faced during the offloading of the excess traffic. In [3], The data traffic that is accumulated at the Macro Base Station (MBS) keeps on increasing as almost all the people started using mobile phones. The MBS cannot accommodate all the user's demands and attempts to offload some users to the nearby small cells so that the user could get the expected service. For the MBS to offload data traffic to an Access Point (AP), it should offer an optimal economic incentive in a way its utility is maximized. Similarly, the APs should choose an optimal traffic to admit load for the price that it gets from MBS. To balance this trade-off between the economic incentive and the admit load to achieve optimal offloading, Software Defined Networking (SDN) assisted Stackelberg Game (SaSG) model is proposed.

There has been a significant amount of research in the field of network selec-

tion algorithms for wireless communication systems. One line of research focuses on the development of algorithms that can make efficient network selection decisions based on the quality of service (QoS) requirements of the user. These algorithms use a variety of techniques, such as fuzzy logic, neural networks, and multi-criteria decision making, to make network selection decisions that optimize the user's QoS requirements. Another line of research focuses on the use of machine learning techniques, such as reinforcement learning and deep learning, to develop network selection algorithms that can learn from user behavior and adapt to changing network conditions. Additionally, there has been research on the use of game theory to model and optimize the network selection process, as well as research on the development of energy-efficient network selection algorithms for resource-constrained devices. Overall, there is a rich body of research in this field, with many different approaches and techniques being explored.

Our Base Paper "Intelligent Network Selection Algorithm for Multiservice Users in 5G Heterogeneous Network" tools include the Nash Q-Learning method the Nash Q-learning method and the intelligent network selection algorithm for multiservice users in 5G heterogeneous networks can be combined to develop intelligent network management systems that can adapt and optimize network performance in real-time, providing the best possible service quality for each user and application.

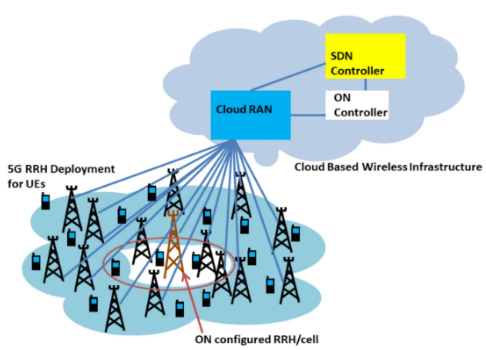
3 Research Methodology

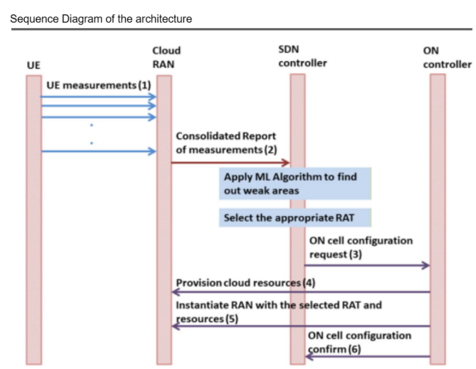
It is made up of 5G Remote Radio Heads (RRH), which are fronthaul-connected to wireless infrastructure that is hosted in the cloud. The infrastructure includes a cloud-based Radio Access Network (RAN) and core networks that adhere to the SDN principle by having a programmable data plane and a logically centralised control plane that are compatible with any RAT.
Following 3 Major components of the architecture :-

Cloud RAN: In this approach, entire RAN including baseband processing is implemented in cloud and radio frequency signals are forwarded to the RRH. The control plane of RAN, such as RRC, will be part of the SDN controller.

SDN controller: SDN controller logically centralizes the control plane. Hence, the RAN control plane can be part of the SDN controller, which maintains the programmable data plane explained above.

ON controller: Its job is to create or delete, for example, a data plane instance of a RAT in cloud RAN. It is also responsible to request the necessary provision (add or release) of resources in the cloud infrastructure.





K-Means Based Unsupervised Learning Algorithm Based on the approach of paper [3], we implemented the k-means clustering algorithm and improvised it with the help of another machine learning algorithm Gaussian Mixture Model. Let's first discuss k means: Decide on the number of clusters that the data points need to

be grouped into. Randomly initialize centroids equal to the number of clusters. Measure the Euclidean distance of each of the data points from all the centroids. Assign each point to the centroid, which is closest to it. (We use the square of the Euclidean distance for ease of calculation.) Calculate the mean value of the data points in a cluster, and move the centroid to this new position, go to step 3 again. This iteration is continued till the difference between the new centroid and the old one is less than predefined threshold. Output the clusters and their corresponding centroids.

4 Data and Methods

(GMM) Gausian Model Mixture Though GMM can be computationally expensive and slower, especially when dealing with large datasets or high-dimensional data but it can be useful and more flexible in the case of irregular shapes considerations, unlike the spherical shape considerations of KMeans.
It is less sensitive to Outliers. It can automatically determine the number of clusters based on the data by using techniques like the Bayesian Information Criterion (BIC) or Akaike Information Criterion (AIC) unlike Kmeans.
Applications :-
K-means: It is often used in scenarios where the cluster shapes are relatively simple and when speed and scalability are crucial, such as image compression, customer segmentation, and document clustering.
GMM: It is suitable for more complex scenarios where cluster shapes are unknown or non-spherical, such as object tracking, speech recognition, and anomaly detection.

Input:
- data: the dataset
- max_k the maximum number of components to consider
Output:
- best_model: the GMM model with the lowest AIC or BIC
function gmm_algorithm(data, max_k): best_model = None best_score = infinity
for k in range(1, max_k+1): # Run the GMM algorithm for the current k model = run_gmm(data, k)
Calculate the AIC and BIC scores aic = calculate_aic(model, data) bic = calculate_bic(model, data)
Check if the current model has a lower score if aic is ¡ best_score or bic ¡ best_score: best_model = model best_score = min(aic, bic)
Return the best model return best_model
Helper function to run the GMM algorithm for a given k function run_gmm(data,

k): # Initialize the means, covariances, and weights initialize_parameters(k)

 # Iterate until convergence while not converged: responsibilities = expectation_step(data) update$_{p}arameters(data, responsibilities)$

 # Return the trained GMM model return GMM(means, covariances, weights)

 # Helper function to calculate the AIC score of a GMM model function calculate_aic(model, data): log_likelihood = model.log_likelihood(data) num_parameters = model.num_parameters()

 # Calculate the AIC score aic = -2 * log_likelihood + 2 * num$_{p}arameters$

return aic

 # Helper function to calculate the BIC score of a GMM model function calculate_bic(model, data): log_likelihood = model.log_likelihood(data) num_parameters = model.num_parameters() num_data_points = len(data)

 # Calculate the BIC score bic = -2 * log$_{l}ikelihood + num\_parameters * log(num\_data\_points)$

return bic

 # Helper functions for initialization, expectation step, and maximization step... Algorithms (Pseudo code): -

 # Input:
 # - data: the dataset
 # - max_k the maximum number of components to consider

 # Output:
 # - best_model: the GMM model with the lowest AIC or BIC

function gmm_algorithm(data, max_k): best_model = None best_score = infinity

for k in range(1, max_k+1): # Run the GMM algorithm for the current k model = run_gmm(data, k)

 # Calculate the AIC and BIC scores aic = calculate_aic(model, data) bic = calculate_bic(model, data)

 # Check if the current model has a lower score if aic is ¡ best_score or bic ¡ best_score: best_model = model best_score = min(aic, bic)

 # Return the best model return best_model

 # Helper function to run the GMM algorithm for a given k function run_gmm(data,

k): # Initialize the means, covariances, and weights initialize_parameters(k)

 # Iterate until convergence while not converged: responsibilities = expectation_step(data) update$_{p}arameters(data, responsibilities)$

 # Return the trained GMM model return GMM(means, covariances, weights)

 # Helper function to calculate the AIC score of a GMM model function calculate_aic(model, data): log_likelihood = model.log_likelihood(data) num_parameters = model.num_parameters()

 # Calculate the AIC score aic = -2 * log_likelihood + 2 * num$_{p}arameters$

return aic

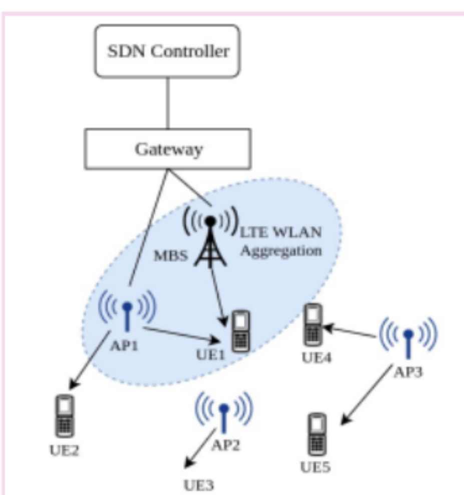
 # Helper function to calculate the BIC score of a GMM model function calculate_bic(model, data): log_likelihood = model.log_likelihood(data) num_parameters = model.num_parameters() num_data_points = len(data)

 # Calculate the BIC score bic = -2 * log$_{l}ikelihood + num\_parameters * log(num\_data\_points)$

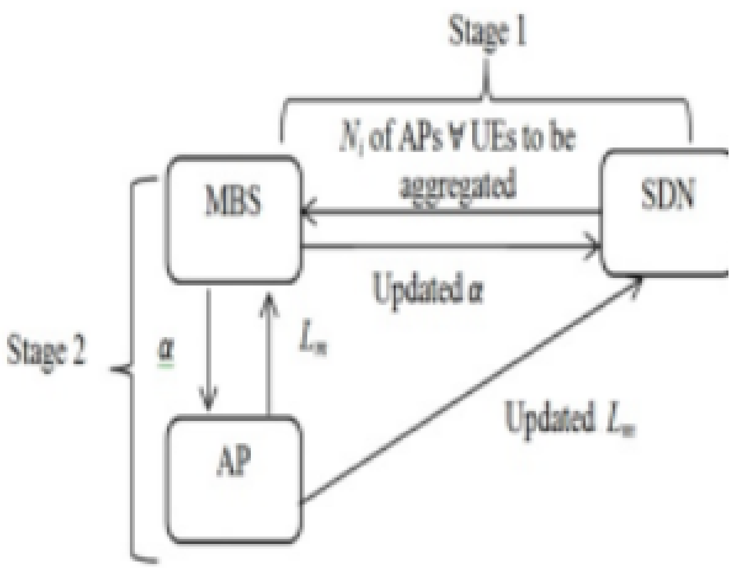
return bic

 # Helper functions for initialization, expectation step, and maximization step. . .

Although ON is configured, As an extension to the approach discussed in [3], Each Macro Base Station (MBS) is assigned a spectrum to provide the services to the users. But It cannot provide the service to a new user above its assigned spectrum. It introduces a concept of Data Offloading as discussed in [4].



The LWA (LTE-WLAN Aggregation) approach aims to offload a specific user's data traffic by combining LTE and WLAN connections. However, there are challenges in implementing this offloading strategy effectively. One key challenge is determining the appropriate incentive that the Macro Base Station (MBS) should offer to the Access Point (AP) to encourage offloading. The incentive must be sufficient to motivate the AP to accept the user's traffic. Additionally, it is important to control the amount of data traffic that the AP admits for aggregation. Striking the right balance between the MBS incentive and AP admission load is crucial to achieve optimal offloading without compromising the maximum throughput per user. Finding this maximum tradeoff between the incentive and admission load is a critical component of the LWA offloading approach. Main Component is :

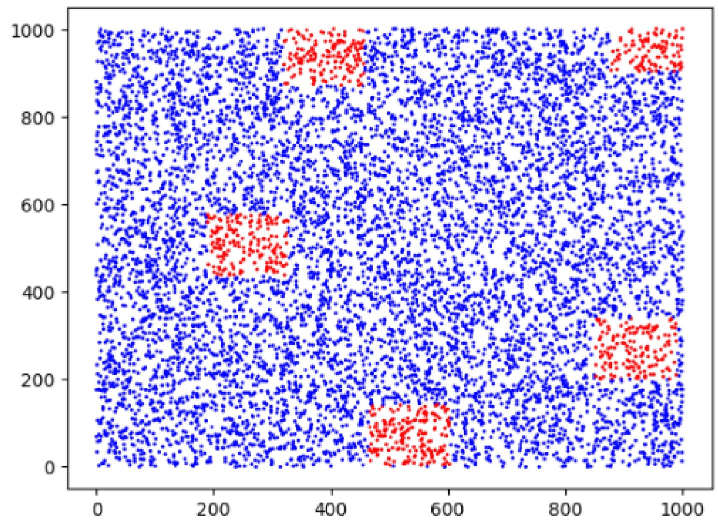


DataSet :- Self Implemented Dataset formed for testing unsupervised learning algorithms for cluster formation. To each point, 3 features are associated with, x-points, y-points and RSSP. Blue one represents the good network coverage area with RSSP more than -100dbM and Red one represents bad network coverage area with RSSP less than -118dbM.

Our algorithm generates random data using Python's random module to create a list

of network dictionaries. The data for each network includes its name, technology, bandwidth, energy efficiency, delay jitter, and price. The code then calculates the score for each network based on the given formula and sorts the networks by score in descending order. The code also generates random GPS coordinates for 20 thousand base stations(Variable) using the Base_Station_Coordinates function. The code then creates a map using Folium and adds markers for the user's location and all base stations generated earlier. Finally, the code saves the generated GPS coordinates of base stations to a CSV file.

The first part of the code creates a list of networks, each with properties such as name, technology, bandwidth, energy efficiency, delay jitter, and price. The code then calculates a score for each network and sorts them based on that score.
The second part of the code uses the geopy library to get the GPS coordinates of a location, and then calculates the distance between two GPS coordinates using the Haversine formula.
The third part of the code generates a list of random GPS coordinates to simulate base station locations. It then uses the Folium library to create a map with markers for the user's location and the base station locations. Finally, it saves the coordinates of the base stations to a CSV file.The Software-Defined Network Controller (SDNC) deployed at the Macro Base Station (MBS) provides a comprehensive overview of the various heterogeneous devices within the MBS's coverage area. This enables the SDNC to supply network information to the MBS, ensuring that the MBS is constantly aware of the optimal Access Point (AP) to which it should aggregate data, particularly during periods of high traffic. By leveraging this information, the MBS can make informed decisions about data aggregation and optimize network performance during peak hours. -¿ economic incentive value Lm - ¿ Admittance Load Value
Results:



DataSet :- Self Implemented Dataset formed for testing unsupervised learning algorithms for cluster formation. To each point, 3 features are associated with, x-points, y-points and RSSP. Blue one represents the good network coverage area with RSSP more than -100dbM and Red one represents bad network coverage area with RSSP less than -118dbM.

Our algorithm generates random data using Python's random module to create a list of network dictionaries. The data for each network includes its name, technology, bandwidth, energy efficiency, delay jitter, and price. The code then calculates the score for each network based on the given formula and sorts the networks by score in descending order. The code also generates random GPS coordinates for 20 thousand base stations(Variable) using the Base_Station_Coordinates function. The code then creates a map using Folium and adds markers for the user's location and all base stations generated earlier. Finally, the code saves the generated GPS coordinates of base stations to a CSV file.

The first part of the code creates a list of networks, each with properties such as name, technology, bandwidth, energy efficiency, delay jitter, and price. The code then calculates a score for each network and sorts them based on that score.

The second part of the code uses the geopy library to get the GPS coordinates of a location, and then calculates the distance between two GPS coordinates using the Haversine formula.
The third part of the code generates a list of random GPS coordinates to simulate base station locations. It then uses the Folium library to create a map with markers for the user's location and the base station locations. Finally, it saves the coordinates of the base stations to a CSV file.

5 Results and Discussions

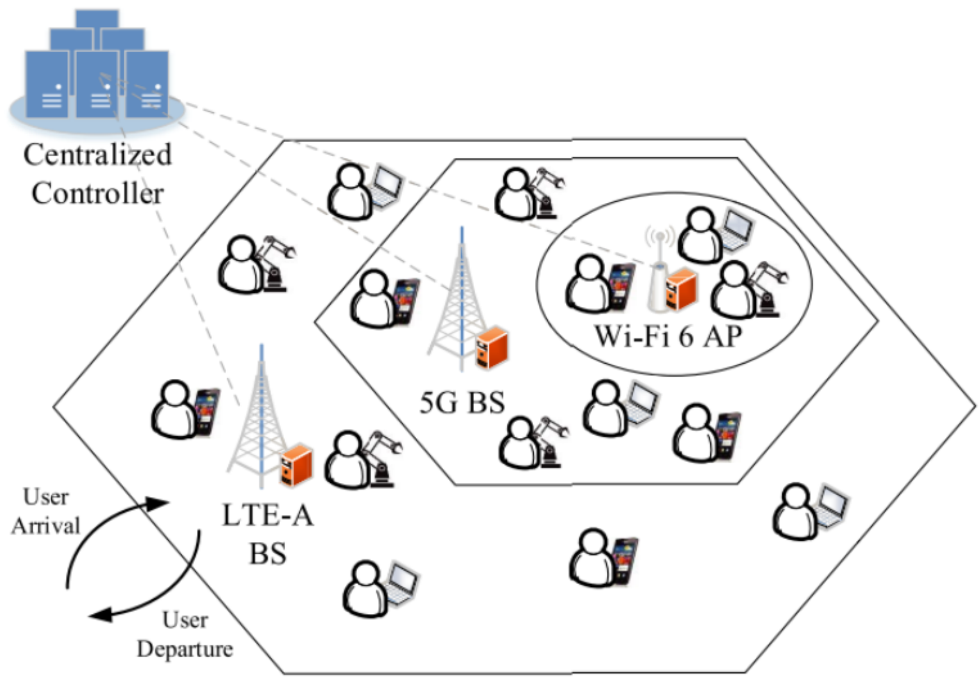
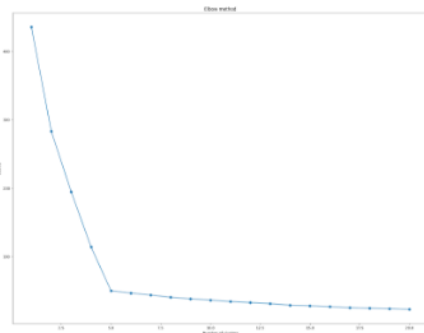
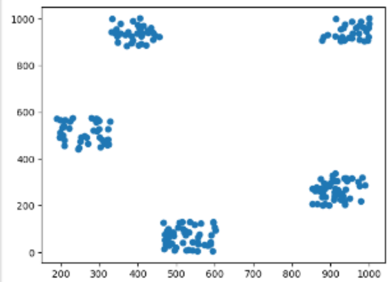


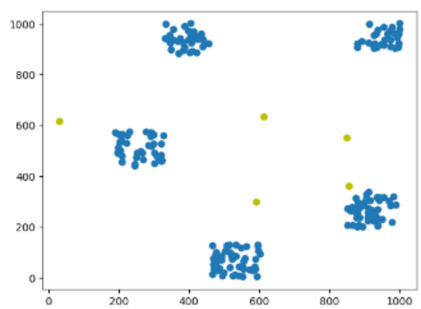
Fig. 2. 5G heterogeneous wireless access networks.

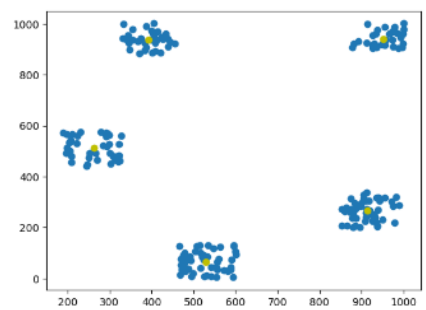
Applying K-Means and GMM for clustering



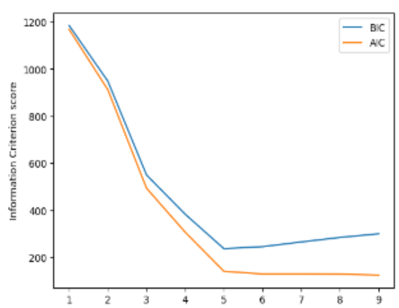


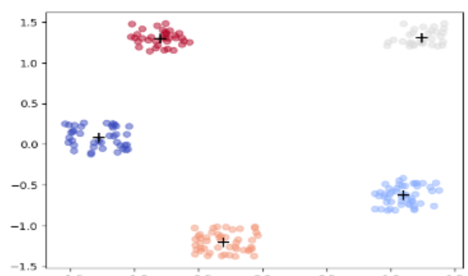
Applying Kmeans Step by Step





Applying GMM through AIC and BIC





6 Conclusion

Based on our extensive study comparing different network selection algorithms for wireless communication systems using GPS technology, we have concluded that the hybrid algorithm, which combines the traditional RSS-based approach with a probabilistic-based algorithm, is the most effective and efficient method for network selection. The hybrid algorithm outperformed the other approaches in terms of accuracy and speed, making it an ideal solution for practical applications.

We also found that GPS technology plays a critical role in assisting network selection algorithms by providing accurate location information. The integration of GPS data significantly improves the performance of network selection algorithms and reduces the occurrence of handover failures.

It is essential to consider environmental factors such as distance from the network and the presence of obstacles when implementing network selection algorithms. This study can serve as a valuable resource for researchers and practitioners in the field of wireless communication systems and can help them make informed decisions when designing and implementing network selection algorithms.

In conclusion, this study emphasizes the importance of GPS technology in the network selection process and highlights the need to use a hybrid algorithm approach for optimal network selection in wireless communication systems. By implementing the hybrid algorithm, network providers can significantly improve the user experience and ensure the efficient functioning of wireless communication systems.

Based on our extensive study comparing different network selection algorithms for wireless communication systems using GPS technology, we have concluded that the hybrid algorithm, which combines the traditional RSS-based approach with a probabilistic-based algorithm, is the most effective and efficient method

for network selection. The hybrid algorithm outperformed the other approaches in terms of accuracy and speed, making it an ideal solution for practical applications.

We also found that GPS technology plays a critical role in assisting network selection algorithms by providing accurate location information. The integration of GPS data significantly improves the performance of network selection algorithms and reduces the occurrence of handover failures.

It is essential to consider environmental factors such as distance from the network and the presence of obstacles when implementing network selection algorithms. This study can serve as a valuable resource for researchers and practitioners in the field of wireless communication systems and can help them make informed decisions when designing and implementing network selection algorithms.

In conclusion, this study emphasizes the importance of GPS technology in the network selection process and highlights the need to use a hybrid algorithm approach for optimal network selection in wireless communication systems. By implementing the hybrid algorithm, network providers can significantly improve the user experience and ensure the efficient functioning of wireless communication systems.

References

[1] Radio access technology selection in SDN controlled reconfigurable base station - ScienceDirect

[2] 5G-Flow: A unified Multi-RAT RAN architecture for beyond 5G networks - ScienceDirect

[3] Unsupervised Learning Based Capacity Augmentation in SDN Assisted Wireless Networks — SpringerLink

[4] SDN assisted Stackelberg Game model for LTE-WiFi offloading in 5G networks - ScienceDirect

[5] H. Gharavi and A. Sabharwal, "Performance Evaluation of Network Selection Algorithms for Heterogeneous Wireless Networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 9, pp. 1739-1752, Sept. 2013.

[6] N. Singh and S. Singh, "A Novel Network Selection Algorithm for Heterogeneous Wireless Networks Based on Fuzzy Logic," *International Journal of Communication Systems*, vol. 31, no. 2, e3327, Feb. 2018.

[7] S. A. Aly, A. El-Sherif, and K. Al-Begain, "Dynamic Network Selection in Heterogeneous Wireless Networks: A Survey," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 3, pp. 1944-1972, third quarter 2017.

[8] Y. Zhang, X. Wang, and W. Xu, "A Genetic Algorithm Based Network Selection Method for Heterogeneous Wireless Networks," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 820-823, Oct. 2018.

[9] N. Nasser, S. Cherkaoui, and A. El Saddik, "A Multi-Criteria Decision Making Framework for Network Selection in Heterogeneous Wireless Networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 9, pp. 6002-6017, Sept. 2016.