Coordinated Frequency Allocation Problem in Wi-Fi Networks

Ozan Karaali, Murat Arisli

I. Introduction

A Wi-Fi network is managed by an access point (AP) which selects a single operating frequency among several possible frequencies. Due to the extremely saturated condition of the Wi-Fi frequencies, correct selection of this operating frequency becomes very important for the performance of the Wi-Fi network. This leads to “frequency allocation problem”. This paper will discuss a solution for frequency allocation problem.

II. Motivation:

In recent years the popularity of wireless LAN hotspots has increased drastically. This is the reason why, today a lot of public places such as airports, cafeterias, metro stations and even complete city centres are equipped with many APs in order to offer wireless connectivity everywhere.

For instance, we can look at the Figure 1 which shows the hotspot development areas in the city of Hamburg, Germany. Hamburg represents itself on its official website as a hotspot city. [2] It offers one hour free surfing in the internet at many locations. Further locations like the tourist centres around the port will be equipped with hot spots in near future.

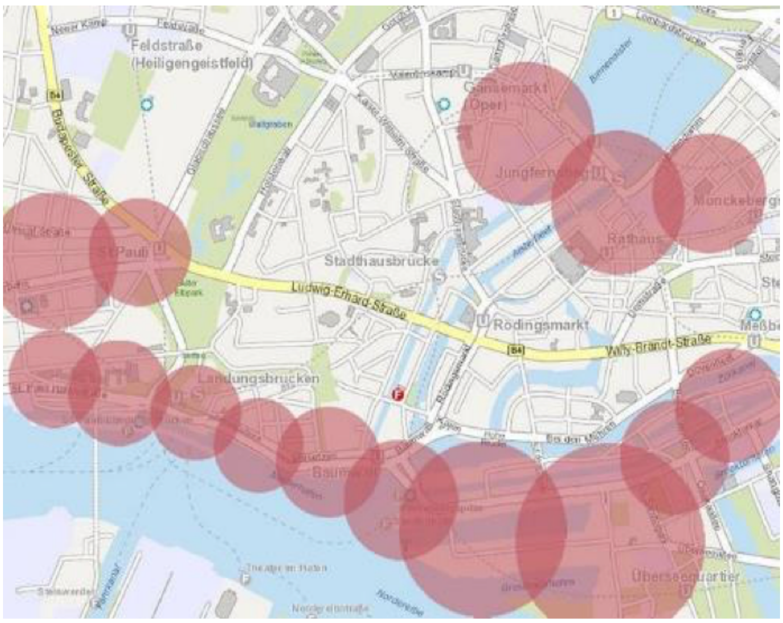

Figure 1: Hotspot planning in city of Hamburg [1]

III. Challenge:

As shown in the motivation section, as the wireless LAN hotspots become more popular at the same time the increased density of access points has started to highlight the shortcomings of the IEEE 802.11 standards. Since no standard frequency allocation method exists for APs, a large majority of APs is using default channels settings which is leading to inefficient use of the already crowded spectrum in the ISM (**I**ndustrial, **S**cientific and **M**edical) bands. The situation is especially critical in the 2.4 GHz band due to the small number of non-overlapping channels available and the coexistence problems with microwave ovens, cordless phones, baby monitors many other wireless technologies which operate on the same channel.

In Figure 2 the available channels in the 2.4 GHz band are illustrated. In 2.4GHz networks, channels are defined 1 to 14, that mapped to 2.412 GHz to 2.484 GHz. The number of possible frequency channels varies from country to country due to each country's regulations. For instance, while in most european countries 1 up to 13 channels are available, in Japan all of these 14 channels are possible for the selection.

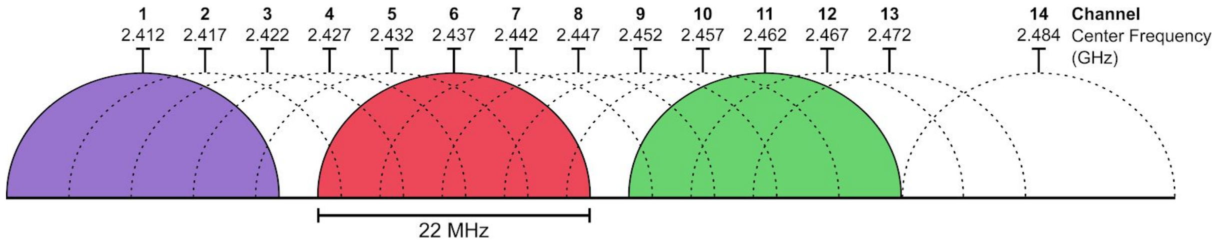

Figure 2: Channels in 2.4 GHz band

In crowded areas such as urban, different networks can easily interfere with each other easily, when especially to avoid multi-channel overlapping generally 1st, 6th and 11st channels are chosen. While in the left network example a) of Figure 3 the three APs use

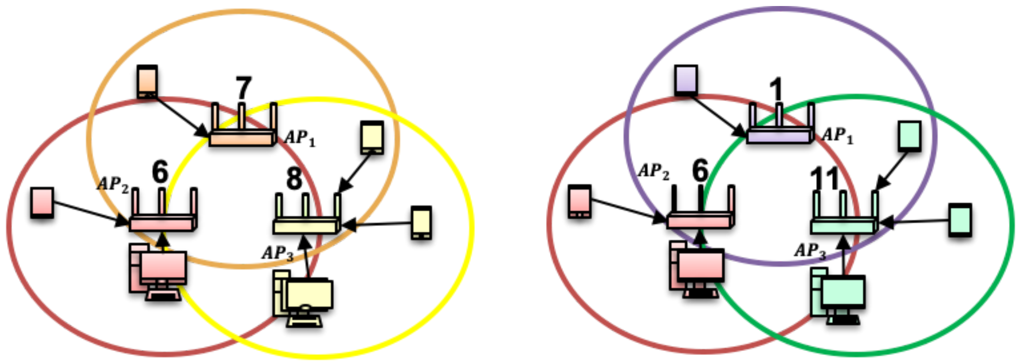

Figure 3: a) Interfering channels *b) Non-interfering channels*

partially overlapped channels, in the right example b) these APs use three non-overlapping channels as their operating frequencies. The network performance of the right hand example b) is significantly better, because there is less interference.

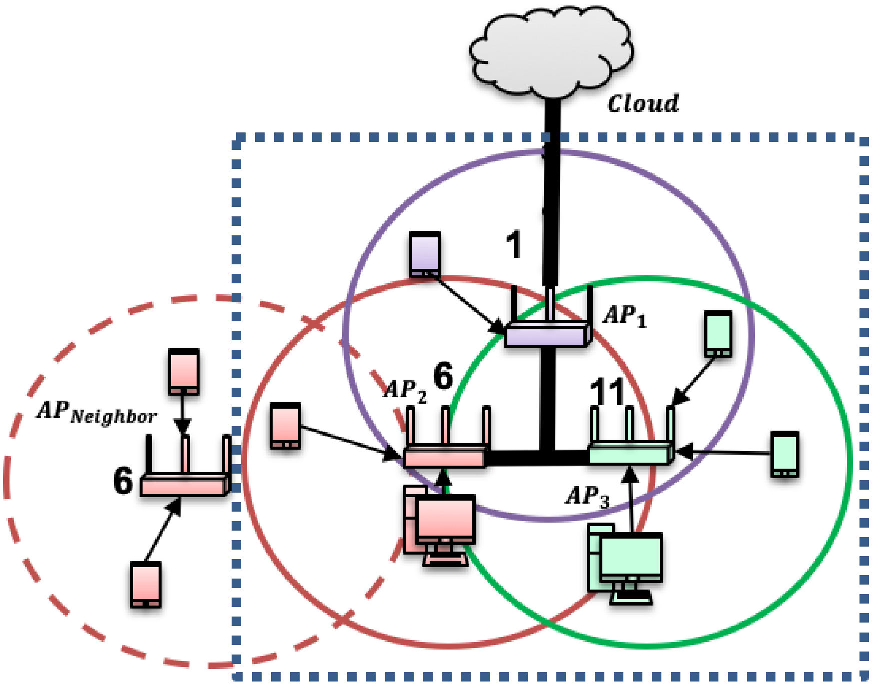

Figure 4: Centrally managed network

In Figure 4, we have a centrally managed network where the APs inside the blue dotted box are managed by one administration which is illustrated by the cloud. The challenge here is to manage the operating frequency channels of the APs carefully by considering other unmanaged APs (like the neighbouring AP from different administration). Since the neighboring AP is using the same channel as one AP of the managed network, they experience inter-cell interference which result in performance degradation. This problem of selecting the correct channel is well-known and it is also called as the Frequency Allocation Problem (FAP).

IV. Algorithm

In literature and in practise several algorithms are available in order to solve the Frequency Allocation Problem. For this reason, the task in the first meetings was to find, compare and evaluate different algorithms proposals regarding our needs. In the fourth meeting, we decided to take a centrally managed and measurement-based algorithm which is called Global-Coordination. [3] This algorithm outperform the other published ones in many terms because it uses a physical interference model rather than a binary model and it has also the ability to deal with rogue interferers. There two different kind of radio frequency (RF) interferers. On the one hand, there are the intentional RF interferers which are basically the unmanaged APs using the same operating channel. On the other hand, there are the unintentional RF interferers which are for example microwave ovens and other wireless devices that operate on the same channel.

Our measurement-based FAP-algorithm is based on the Global-Coordination algorithm. The FAP-algorithm has an iterative nature. At each point in time which can be predefined, randomly chosen or determined at runtime, one iteration of channel switching takes places. Let's go through the main steps of the algorithm how the channel switching is done:

1. The APs measure interference power on each channel periodically only when the APs are idle. The averaging period is a design of choice and the idle state of the APs means that the APs are neither transmitting or receiving. The measurements of the APs are the clear channel assessment (CCA) measurements which reflect how clear a channel is. CCA also considers noise and therefore the measurements are lower bounded by noise floor.
2. The APs send their measurements and additional information like the average packet loss rate of their associated stations to the cloud. This average packet loss rate is used as a weighting factor for the channel assignment.
3. Based on the measurements the channel assignment is determined by Mixed Integer Programing (MIP) in the cloud. The mathematical formulation of the MIP is defined as follows:

$$MIN \sum_{i=0}^{ChannelSet} \sum_{j=0}^{APset} x_{ij} W_{ij}$$

$$subject\ to:$$

$$\sum_{i}^{ChannelSet} x_{ij} = 1, \forall j \in APset$$

4. In the last step, the cloud will output the new set of channels for the APs if and only if the overall weighted interference of the network is lower than before.
   This is how the condition is mathematically defined:
   So, this algorithm $\sum_{n:f_n=k} W_k^n(\vec{f}) > \sum_{n:f'_n=k'} W_{k'}^n(\vec{f'})$ considers the sum weighted interference over all APs on

each channel which is illustrated by Table 1 for example. The W-CCA values describe the weighted measurements of one AP at different channels from 1 to 13.

APs	Channels		
	1	…	13
AP_1	W-CCA_1^1	…	W-CCA_1^{13}
…	…	…	…
AP_N	W-CCA_N^1	…	W-CCA_N^{13}
All	W-CCA_{ALL}^1	…	W-CCA_{ALL}^{13}

Table 1: Overview of the weighted interference of each channel

The FAP-algorithm is implemented in Python using CPLEX to solve the MIP problem for the channel assignment. Its input is a JSON file with a network setup including the measurements of the APs and the average racket loss rate of their stations. Besides these information, the JSON file includes the mac address and the current operating channel number of the AP. The output of the FAP-algorithm is also a JSON file including the mac addresses of the APs and their calculated channel numbers.

V. Simulation

To simulate the results NS3 Network Simulator is used. NS3 can be installed from https://www.nsnam.org/. NS3 uses C++ (and Python available) as programming language to write simulation scenario. For the best compatibility, Linux is used while developing our simulation. In addition, we have used NetAnim to visualize the network that defined in scenarios. We have developed simulation with iterations of every two weeks.

For the first iteration, the goal for simulation side was getting familiar with NS3 and development environment.

Goals of the second and third meetings were setting a network that have two access points and two stations that are on the same channel to create interference and getting metrics such as delay and throughput of the network. After third meeting, we had the first 10 second running simulation that consists of the basic two access points transferring data to their stations with free space loss model and we were successful to get NetAnim (Figure 5). End of this iteration, the problems were defining two different physical layers that does not interfere at all with undefined 802.11 connection, that leads to 10 Mbits/second data transfer for each access point.

In fourth meeting, the goals were defining the “three AP / one neighbor AP” (Figure 4) network scenario. We had correctly working 802.11n 2.4 GHz simulation that have refactored code with free space path loss model (Figure 6). In fourth simulation, we had the network of three AP’s of our user, different amount of station nodes connected to them and one neighbor AP that connected to station nodes. The networks were giving 54 Mbits/second divided by the amount of stations without interference.

For the fifth meeting the goal was creating an interference for the networks that are defined, to meet this requirement a loss model needed to be defined. For a realist scenario, we decided to use “Log Distance Propagation Loss Model” and defined it in the simulation, adjusted mobility distances with adjustable distance option (default = 5 meter).

In sixth and seventh meetings (which are done by a week), we enabled PCAP to get package data and additionally we have implemented an example sniffer callback to measure delay/jitter, delay/jitter estimators, dropped package counters, a flow monitor prototype. The problem in this week was the sniffers needed to be defined for each AP and ambiguity of the metrics such as RX power or CCA.

For seventh meeting due to my (Ozan’s) finals week, and ambiguity of the project I (Ozan) handed the simulator in to Murat. The goal of this meeting was measuring CCA’s at the access points to get channel clarity data, feed into the algorithm, take the output and change the channels for AP and get new channel clarity data to be sure algorithm works correctly and adjusts network.

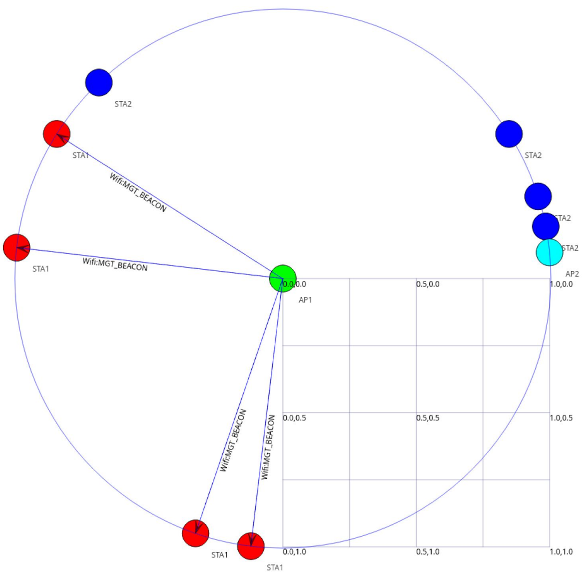

Figure 5: Simple wireless environment consisting of 2 AP’s and their STA’s in NetAnim.

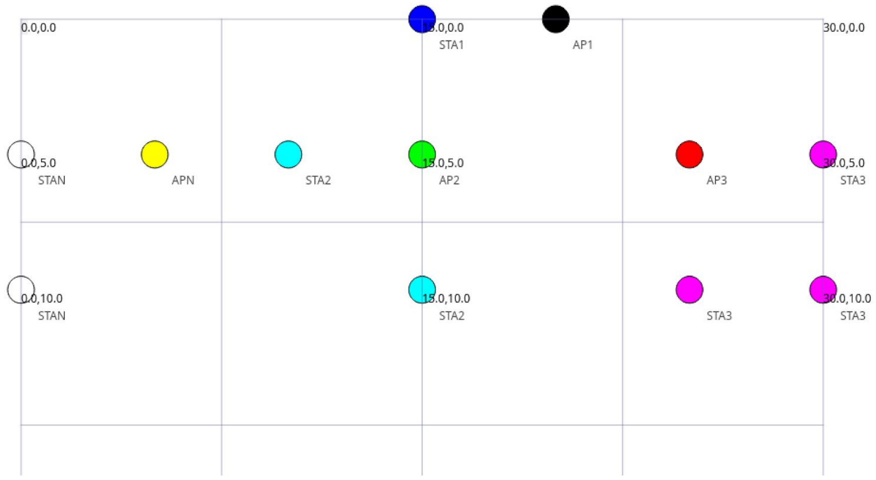

Figure 6: 3-1 AP positioning, simulated in NS3 and pictured with NetAnim

VI. Reflections and Evaluation

Since ns-3 does not provide ready-made mechanisms to measure the the CCA levels on different channels, we had a great challenge how to simulate the CCA measurements. With the help of the supervisors we are able to measure the CCA levels at the APs only on their operating channels by installing some sniffer stations in the network. Since we need to measure the CCA levels on every channel (1 to 13), a dynamic channel surfing mechanism is needed. Again, ns-3 does not offer ready-made solution, so we have tried different mechanisms to get the measurements. Unfortunately we could not master this challenge in a short time. In order to deliver some results, we decided to assign the CCA measurements by hand and fed it the algorithm. So, the communication between the simulation and the FAP-algorithm is done manually.

In order to evaluate the performance of the algorithm we simulated the scenario of Figure 4. Regarding the APs constellation we assume the CCA measurements as in Table 2 and fed this information to the FAP-algorithm. As shown in Table 2 the value of CCA measurements in the brackets are never 0 due to the noise floor. The higher the CCA values (max 255) the more the channel is unclear. The value outside the brackets represents the weight which is the average packet loss rate of their associated stations. The higher the weight of one AP the more this AP will contribute to the overall weighted interference.

APs	Channels												
	1	2	3	4	5	6	7	8	9	10	11	12	13
AP_1	20(50)	20(80)	20(100)	20(150)	20(180)	20(200)	20(180)	20(150)	20(160)	20(180)	20(200)	20(180)	20(160)
AP_2	50(200)	50(180)	50(170)	50(165)	50(180)	50(200)	50(180)	50(170)	50(170)	50(180)	50(200)	50(180)	50(160)
AP_3	20(200)	20(180)	20(160)	20(170)	20(180)	20(200)	20(180)	20(150)	20(120)	20(80)	20(50)	20(80)	20(70)
All	20(50) + 50(200) + 20(200)	...	...	...	...	20(200) + 50(200) + 20(200)	...	...	...	...	20(200) + 50(200) + 20(50)	...	...

Table 2: Assumed weighted interference of each channel before first iteration

Based on the assumed weighted measurements the FAP-algorithm outputs the new calculated channels after the first iteration as follows:

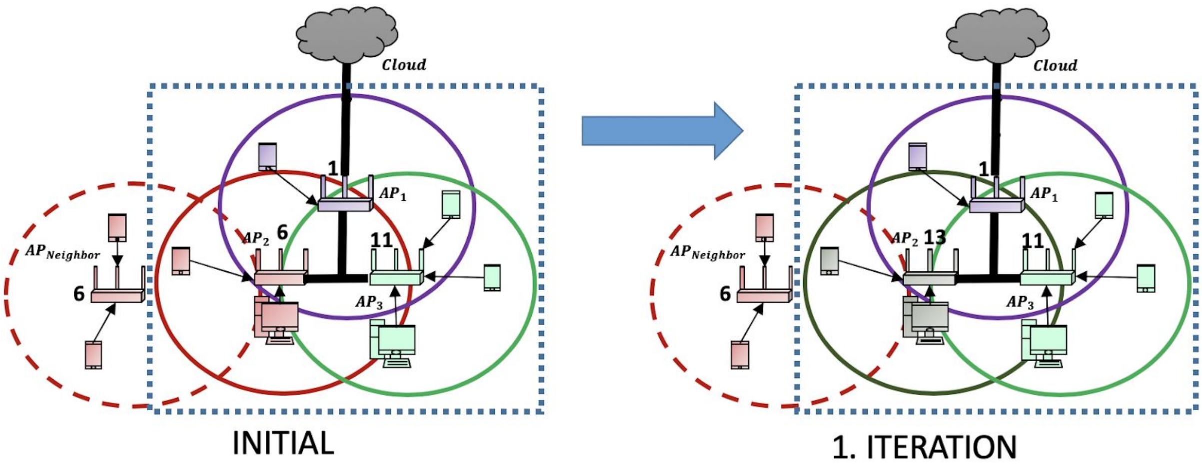

Figure 7: FAP-algorithm output after first iteration

It can be seen from Figure 7 that the channel of AP_2 is changed from 6 to 13 in order to avoid the inter-cell interference. For the next iteration again different weighted CCA measurements are assumed which is illustrated in Table 3.

APs	Channels												
	1	2	3	4	5	6	7	8	9	10	11	12	13
AP_1	20(50)	20(60)	20(60)	20(65)	20(65)	20(70)	20(65)	20(150)	20(170)	20(180)	20(240)	20(200)	20(240)
AP_2	50(200)	50(170)	50(160)	50(165)	50(180)	50(200)	50(180)	50(170)	50(170)	50(180)	50(200)	50(180)	50(160)
AP_3	20(200)	20(180)	20(160)	20(150)	20(120)	20(60)	20(50)	20(60)	20(80)	20100)	20(160)	20(180)	20(200)
All	20(50) + 50(200) + 20(200)	...	...	...	...	20(70) + 50(200) + 20(60)	...	...	...	...	20(240) + 50(200) + 20(160)	...	20(240) + 50(160) + 20(200)

Table 2: Assumed weighted interference of each channel before second iteration

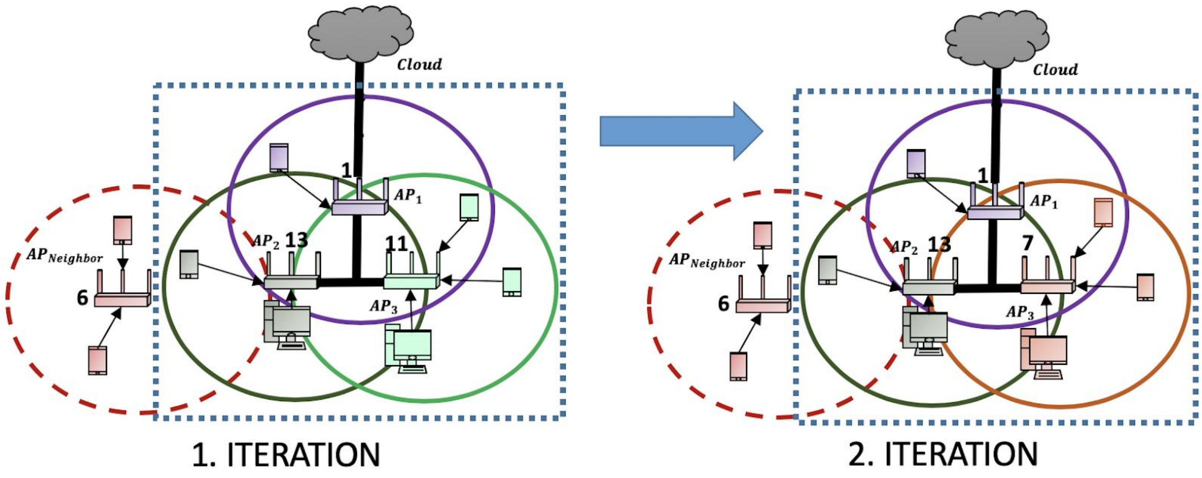

Figure 8: FAP-algorithm output after second iteration

After the second iteration the algorithm changes the channel of AP_3 from 11 to 7 in order to avoid the use of partially overlapped channels which will result in improved network performance. This algorithm will continually improve the network performance regarding the measurements of the APs (even if there is a low improvement). However, this can be changed by saying that the APs switch their channels by at least 5% or 10% improvement of the network performance.

In Figure 9, it can be seen that the aggregated throughput of the network get better after each iteration.

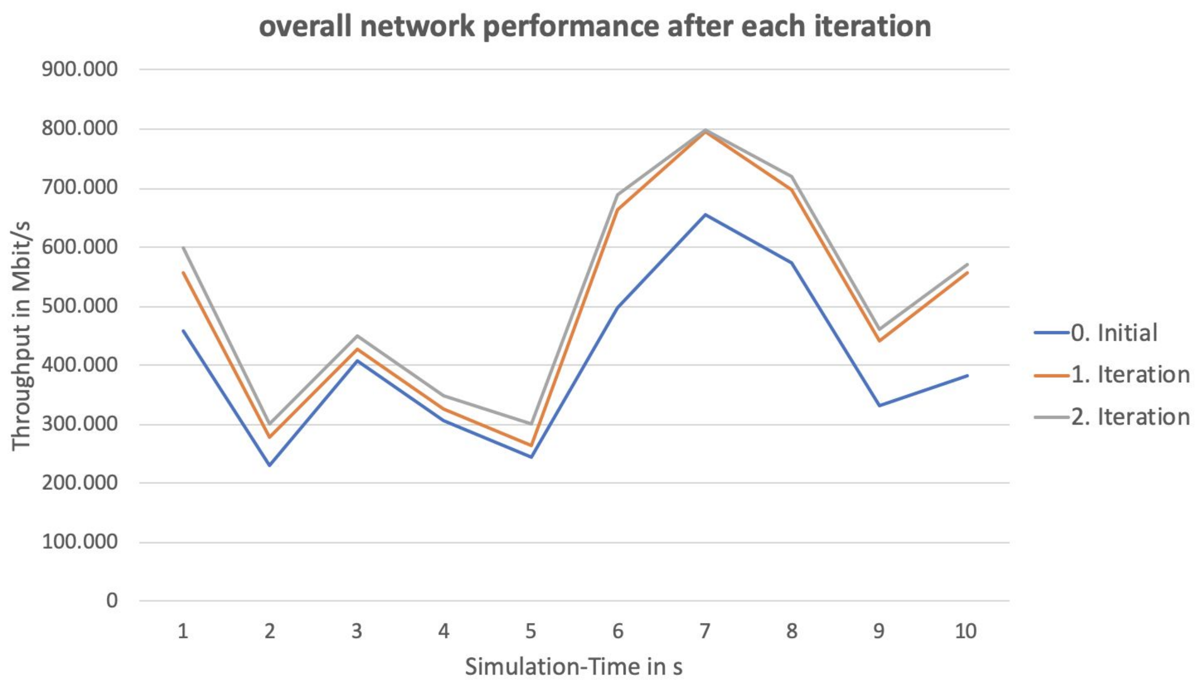

Figure 9: Simulation results after each iteration

Erasmus+ EPIC gave us the opportunity to work on NS3 simulator which is one of the most important networking simulator in the industry. We have also learned a lot of valuable information regarding the wireless communication networks and their management. For example, we have learned that it is better in industrial point of view to use a centrally managed algorithm instead of a distributed algorithm. The cloud saves computation power on each APs and thus saves money for expensive hardware.

VII. Conclusion

In this project, the frequency allocation problem tried to be solved with adapted Global-Coordination algorithm. Basically, this algorithm uses clear channel assessment from access points and takes receives packet drop rates as input to compare different channels using CPLEX to solve FAP problem. The project has been simulated in NS3 network simulator with defining different scenarios which uses log distance propagation loss model, such as in real-life loss. In theory NS3 simulator gives the CCA results to algorithm and takes the new channels as input. To meet the requirement, we have fed the inputs manually and completed our research.

VIII. References

[1] Hamburg WLAN Hafencity Bild. (n.d.). Retrieved from https://www.hamburg.de/image/4102048/4x3/690/518/5f04fee8d5dd3ce8c9963fd75dc485af/tA/w-lan-hafencity-bild.jpg

[2] Hotspots Hamburg WLAN. (n.d.). Retrieved from https://www.hamburg.de/social-media-in-hamburg/4101990/wlan-hafencity-artikel/

[3] J. K. Chen, G. D. Veciana, and T. S. Rappaport, “Improved Measurement-based Frequency Allocation Algorithms for Wireless Networks,” in Proc. IEEE GLOBECOM’07, Washington, DC, USA, Nov. 2007.