Traffic Engineering

Project Report

Optimal Relay Selection in LTE/4G/5G UAV Networks

Institute of Communication Networks

Dr. rer. Nat. Phuong Nga Tran

Konrad Fuger

Winter Term 2022

Submitted By:

Shahriar Malik

30.11.2022

1 Introduction

1.1 UAV Communication

Unmanned Aerial Vehicle (UAV) technology are creating their ways into different scope of applications which promises bringing revolutions in many industries. UAV's help us to reach areas which are difficult to access because of the lack of infrastructure. As a result, they are the likely candidates to be taken over in critical operations such as rescue, surveillance, remote inspection and transportation. Although mechanical capabilities of UAVs would allow those applications for many years already, the communication capabilities are still lacking behind. While some pilot projects with single UAVs show promising results, a reliable and efficient communication network is necessary to enable the large scale deployment of UAVs in urban environments. Currently, many companies rely on the current available generation of cellular networks (LTE/4G/5G) to control and monitor their UAVs as it is the only available infrastructure that covers their entire range of operation and delivers the required Quality of Service (QoS), even in rural areas. While this works well if only a few UAVs are located in every cell, once urban skies are swarming with UAVs, the amount of radio resources used by UAVs must be minimized so that enough resources for other users remain.

Link budget: Link budget accounts for all gains and losses in the communication link. We can calculate link budget (Signal level at receiver) and then compare it with Rx reception sensitivity. Then you will understand Radio Channel Status (Pass or Fail) and Cell Radius.

1.2 LTE/4G/5G

The cellular network is one of the advanced communication networks in existence. Since the LTE standard, a TDMA/FDMA approach is used to subdivide the available spectrum into resource block (RB). A RB is defined as a pair of {0.5 ms, 180 kHz}. Each RB is further subdivided into 7 · 12 resource elements. In each resource element one symbol is transmitted carrying bits according to Table 1.2. There are different methods of assigning resources to users, but each user always gets an integer number of RBs.

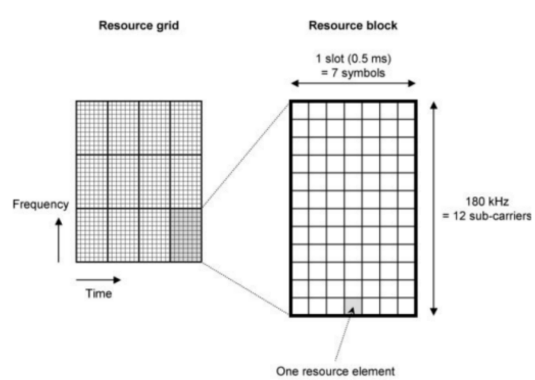

Figure 1: LTE resource grid

2. Problem description

One important feature of LTE/4G/5G communication is adaptive modulation and coding. Here, devices measure the radio channel quality to their communication partner and are able to either use more robust modulation schemes (MCS) with a lower throughput or less robust schemes with a high throughput. The selected MCS defines the number of bits that can be sent in a single resource element. Further, the channel quality is expressed with the Channel Quality Indicator (CQI). The CQI is a number between 0 and 15 where 15 denotes the highest quality and 0 means no communication is possible. To calculate the CQI for a channel between two users, we can perform a link budget calculation (Figure 2). Here, we assume a transmit power P_{tx} of 23 dBm, a frequency f of 1500 MHz and free-space path loss. We summarize all other losses with a constant value of LE = 28 dB the signal power at the receiver P_{rx} can now be expressed as

$P_{rx} = P_{tx} - FSPL - L_E$ …… (1)

With $FSPL = 20\log_{10}(d) + 20\log_{10}(f) - 147.55$ | dB ……………………………………………………………………….. (2)

Where d is the distance between sender and receiver. Which MCSs can be used for a given P_{rx} now depends on the receiver sensitivity S_{rx}. This is a quantity describing for each MCS the minimal P_{rx} so that a signal can be received successfully. The receiver sensitivities for all MCSs are given in the following Table. We always select the highest MCS so that $P_{rx} \geq S_{rx}$.

Table 1: MCS Table

CQI	MCS	Coding rate	Bits per symbol	Receiver sensitivity [dBm
0	n/a	0.00	0.00	$-\infty$
1	QPSK	0.08	0.15	−107.11
2	QPSK	0.12	0.23	−104.91
3	QPSK	0.19	0.38	−103.71
4	QPSK	0.30	0.60	−103.01
5	QPSK	0.44	0.88	−100.01
6	QPSK	0.59	1.18	−97.71
7	16-QAM	0.37	1.48	−96.51
8	16-QAM	0.48	1.91	−95.81
9	16-QAM	0.60	2.41	−94.11
10	64-QAM	0.46	2.73	−90.71
11	64-QAM	0.55	3.32	−89.81
12	64-QAM	0.65	3.90	−89.21
13	64-QAM	0.75	4.52	−86.71
14	64-QAM	0.85	5.12	−84.51
15	64-QAM	0.93	5.55	−83.41

3. Calculation of total number of RB used:

Here in the problem, we have some given parameters. Transmit power, P_{tx} = 23 dBm, Frequency, f = 1500 MHz, Other Losses, L_e = 28 dB

For example, we are trying to calculate RB used between Base station and UAV 1. Distance (d) here is 500m.

So FSPL = 20 log10 (d) + 20 $\log_{10}$ (f)- 147.55 dB = 20 $\log_{10}$(500) + 20 $\log_{10}$ (1500* 10^6) – 147.55 = 89.95

Now, Power at the receiver, $P_{rx} = P_{tx-}$ FSPL- L_e = 23- 89.95- 28 = -94.95

From the MCS table (Fig 1), we see that the CQI is 8 which can transfer 1.91 Bits per symbol.

From Fig. 1 we know our RB contains 7*12 Symbols per resource block.

We can transfer 12*7*1.91 = 160.44 bits/ RB

We need to send 100B= 100*8= 800 bits of data

So, Total no. of RB used = 800/160.44= 4.98 = 5 RB (rounding off to integer)

In this way we can calculate the rest of the RBs for all the distances.

Table 2: Required RBs for different distance

Distance	RB	Distance	RB
100	2	800	11
200	2	900	16
300	4	1000	16
400	4	1100	16
500	5	1200	16
600	9	1300	32
700	11	1500	42

4. Our Task

In our task, we assume that all channel qualities are known by a central scheduler. This scheduler has to calculate the optimal resource assignment. We consider a scenario where UAVs are uniformly random distributed within a circle of radius 1500 m around a ground station. All UAVs are trying to send 100 B of data to the ground station. Here they may communicate directly with the ground station or use relays. The goal is to minimize the total amount of RBs required to send all data.

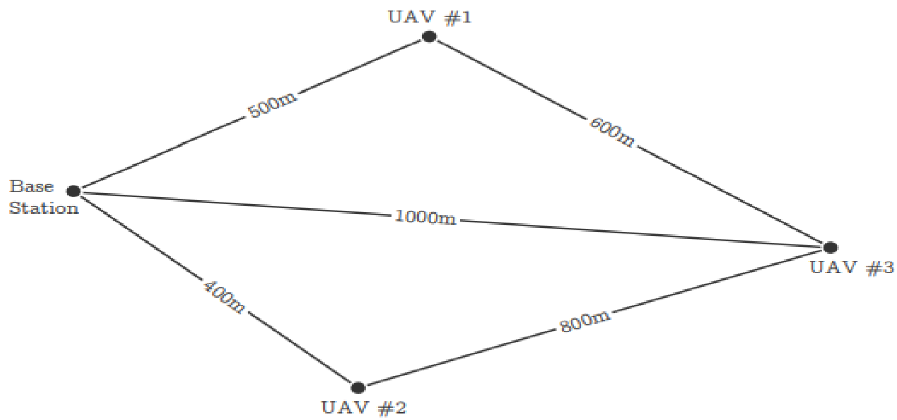

Figure 2: UAVs and Ground Station Network

5. Mathematical Formulation

Here all UAVs are trying to send 100 B of data from UAVs to the ground station. Network = (V, E) where V is a set of nodes (UAVs & Base Station) and E is a set of links.

5.1 Parameters

Required Resource block: R = {$r^{i,j}$} where (i, j ϵ E) ; **r,** represents required resource block between two nodes (i, j) where links exist.

5.2 Decision Variables

$$x_{i,j}^{s,d} = \begin{cases} 1 & \text{If data flows from source (UAVs) to destination (ground station) through link (I, j)} \\ 0 & \text{Otherwise} \end{cases}$$

5.3 Objective

Our objective is to minimize total number of RB for the network.

$$Min \sum_{s,d,i,j} r^{i,j} * x_{i,j}^{s,d}$$ Where i, j is the link between the UAVs and ground station.

For every source (UAVs) and destination (ground station) pair, the routes which requires minimum resource block is considered.

5.4 Constraints

Multi-commodity flow constraints:

$$\sum_{i,j\in \mathrm{E}} x_{i,j}^{s,d} - \sum_{j,i\in \mathrm{E}} x_{j,i}^{s,d} = \begin{cases} 1 & i = s \\ -1 & j = d \\ 0 & \text{Otherwise} \end{cases} \quad \forall \mathrm{s, d, i} \in \mathrm{V}$$

This Multi-commodity flow constraint is required to make sure that data will be routed in our network from UAVs to the base station. It ensures that, if a UAV is not a ground station or destination either, the total data received by the UAV is equal to the total data leaving that UAV.

6. Verification of Mathematical formulation:

In Table 1, required resource block between two nodes based on the distance are shown. We have implemented our mathematical formulation to find minimum total number of RB in a network using Gurobi Solver (Python API). For network in Figure 2, the minimum required number of RBs will be 23.

Table 3: Verification of Mathematical Formulation and Implementation

Data flow from source to destination	Link (i, j) used	Required RB
$x_{0,1}^{1,0}$	0,1	5
$x_{0,2}^{2,0}$	0,2	4
$x_{0,1}^{3,0}$	0,1	5
$x_{1,3}^{3,0}$	1,3	9

From the above table we can see that, the minimum number of RB required for the network is 23 which we have found using Gurobi solver. The code for network in Figure 2 can be found in *optimization_Gurobi_UAV#3.py* file. From UAV#3 to ground station, UAV#1 has been used as a relay. If UAV#1, receive packets from UAV#3 and it also wants to send data by itself to the ground station, then UAV #1 has sufficient RB so that it can concatenate additional packets.

7. Implementation of Randomized network:

We have implemented a randomized network setup with arbitrary (n = 2, 3, 4, 7) numbers of UAVs. Since the transfer data is fixed 100B, the execution time increases with the increment of number of UAVs. We can see from the Figure 3, the computation time increases as the number of UAVs increases. The execution time varies slightly every time we run the code.

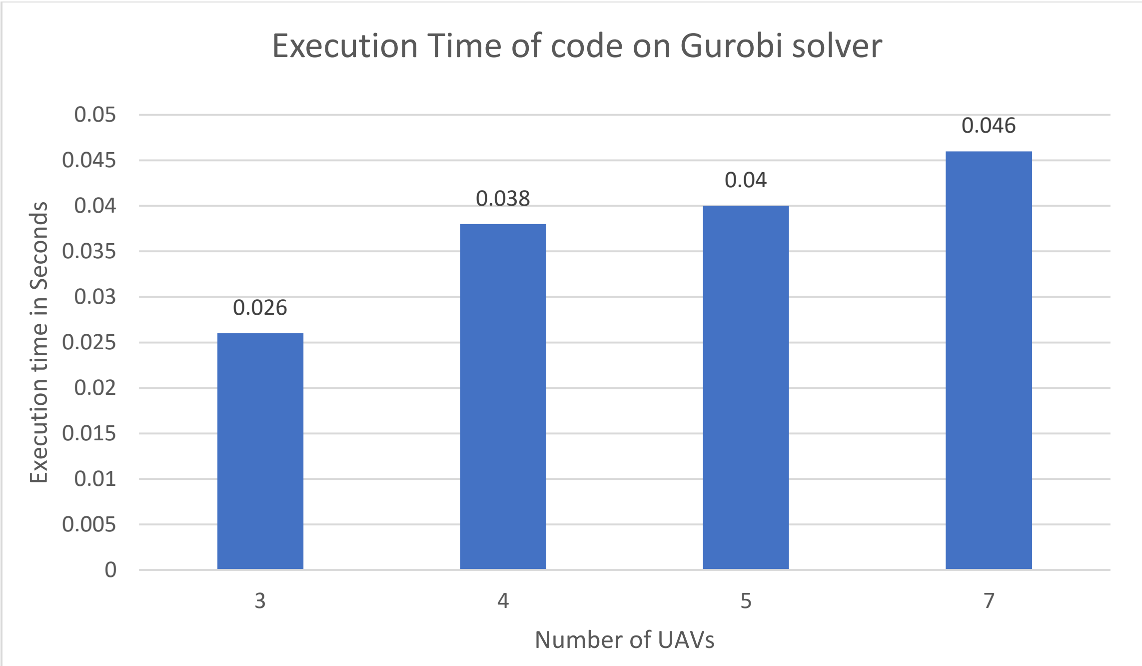

Figure 3: Effect of increasing the number of UAV on code execution time

8. Implementation of heuristic approach:

In the task, we were asked to develop a heuristic algorithm to solve our problem. We have implemented Genetic Algorithm (GA). GA is inspired by nature's ability to evolve individuals influenced by adaptation to the environment and it has provided efficient, effective techniques for optimization applications. Following are the steps for GA:

- Encoding
- Genetic operators
- Parents selection-mechanism
- Survivors' selection-mechanism

Encoding: First, we have calculated a set of routes for each source and destination pair. We did not consider all possible path, but only a few of them (in terms of the number of hops). We mapped our problem in a vector format that contains binary numbers 0 & 1. For the network in Figure 3, there are three source and destination pair, i.e. (UAV #1 - Ground Station, UAV #2 - Ground Station, and UAV #3 - Ground Station). We have considered first three path (route 1: 00, route 2: 01, route 3: 10), in this network the total number of route for each pair is also three. For each pair, we encoded two bits which represent the selected route. From randomly generated solution in a vector format, we have selected route randomly. An example is shown in Table 4. We then generated a set population with size 70.

Table 4: Encoding of solution

UAV #1 - Ground Station	UAV #2 - Ground Station	UAV #3 - Ground Station
01	10	00

Genetic Operators: we have iterated randomly generated solution for 10000 times. In each iteration, to increase diversity by exploring solution spaces, we have used crossover and mutation technique. Crossover (recombination) produces some new individuals that inherit something from their parents and Mutation enables offsprings to have different genes as those from their parents.

Parent' selection mechanism: we have considered solutions from our previous population from which new solutions are created. In our case, we selected arbitrarily new solutions which are generated by crossover and mutation operations.

Survivor's selection mechanism: In our implementation, we always keep the best solutions from the previous populations and the newly generated solutions.

In each iteration for each solution, we validate our solution. During validation, we made sure that at least one of the source and destination pair use relay because of the nature our network. After validating the solution, we calculated the total RB for the network. If we find optimal solution, then we stop the program immediately.

6.1 Computation time GA

We have generated random solution, that's why we haven't notice any trend in execution time. We evaluated our implementation in different scenarios as shown in Table 5. In the Table 5, we can see the time in seconds that genetic algorithm took in a particular run in which it found the optimal solution or the solution that is close to optimal solution.

Table 5: GA Execution Time

Number of UAVs	Time (S)	Found Optimal Solution
3	0.007	Yes
4	8.26	No
5	8.01	No
7	9.42	No

6.2 Comparison of computation time between GA and optimal solution

From the computation time from above results, we can see that Gurobi Solver took much less time compare to GA. But it should be the opposite because we reduced the search space. One reason could be that we have used very high number of iteration for GA and our implemented network was very small. For heuristic implementation, we haven't noticed any trend but for Gurobi Solver we have seen the time for computation increase proportionally with the number of UAVs.