Performance Evaluation of Computer Systems and Networks

Project 9 - Documentation

Kevin Boni
Filippo Lucchesi
Martina Speciale

University of Pisa
Department of Information Engineering
Academic Year 2023/2024

Contents

Chapter 1

The Model

1.1 Base Station

The *Base Station* of a cellular network transmits data packets to N users. Transmissions from the *Base Station* occur in *time slots* of fixed and constant duration. On each *timeslot*, once the *Base Station* has received the *CQIs* from all the users in the network, it composes a *frame* of $M = 25RBs$ by scheduling traffic from the users' queues, and sends the frame to all the users, i.e. removing the corresponding packet(s) from the users' queues. A packet that cannot be transmitted entirely will not be scheduled. Each *Resource Block* can carry data for only *one* user, while two or more packets from the *same* user can share the same RB. A frame can include packets for different users.

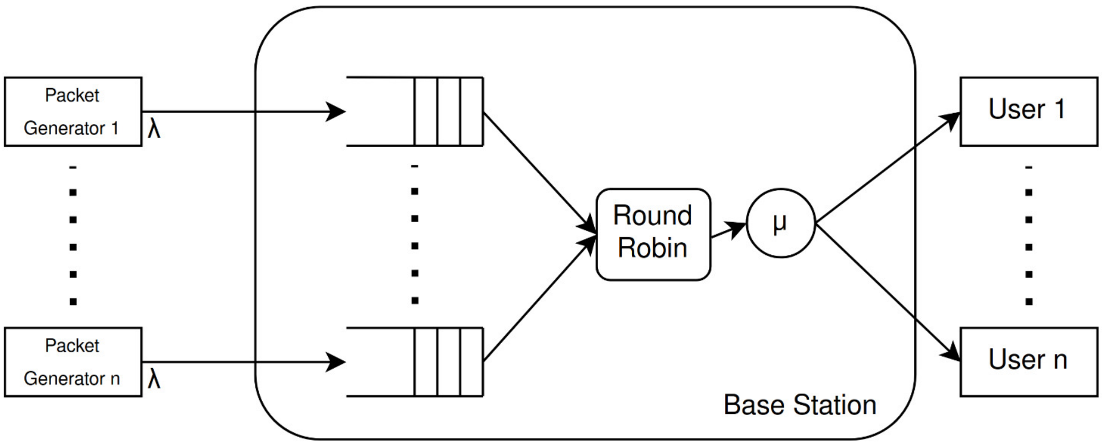


Figure 1.1: Representation of the model used for the simulation.

1.2 User

Each user has its own dedicated FIFO queue at the *Base Station*. Packets of size B (generated by the *Packet Generator*) arrive at the queue with inter-arrival times T. On each *timeslot* users report to the *Base Station* a *Channel Quality Indicator* (CQI), i.e. a number ranging from 1 to 15, which determines the number of bytes that the *Base Station* can pack into a *Resource Block* (RB), according to the table 1.1. They specify the quality

CQI	1	2	3	4	5	6	7	8	9	10	11	12	18	14	15
Bytes	3	3	6	11	15	20	25	36	39	50	63	72	80	93	93

Table 1.1: Possible values of *Channel Quality Indicator* (CQI)

of the trasmission channel, then they receive frames from the *Base Station*.

1.3 Packet Generator

They create packets of random size (uniformly distributed) for each user and send them to the *Base Station*.

1.4 Round Robin Policy

The *Base Station* serves its users using a Round Robin policy: backlogged users are served cyclically, in a fixed order. When a user is considered for service, the BS empties the user' queue if the number of unallocated RBs in the frame is large enough. Otherwise, if the remaining part of the frame is too small to transmit all the backlog of the user being served, the remaining backlog will be transmitted in the next round of the Round Robin policy.

Chapter 2

Case Study

We'll study the throughput and delay of the above system with a varying workload in the following scenarios:

- exponential inter-arrival times, uniform packet sizes (the maximum packet size is such that it fits a frame at the minimum CQI), uniform CQIs
- same as above, with binomial CQIs, chosen so that the mean CQI of different users are sensibly different

2.1 Assumptions

The delay between the generation of a packet and its arrival to the *BS* is assumed to be zero. The same goes for the delay between the generation of a frame and its arrival to the users.

2.2 CQI

We studied 2 cases for the CQI: one where it was picked from a uniform distribution, and one where it was picked from a binomial distribution. In the binomial case, the mean CQI of each user had to be sensibly different from the others. The probability p that defines the binomial was therefore formulated as follows:

$$p = \frac{UserID}{N}$$

where N is the total number of users in the simualtion and *UserID* ranges from 1 to N. Users with lower IDs will end up with a lower CQI average, while users with higher IDs will get higher CQI average. Figures 2.1, 2.2 support this claim.

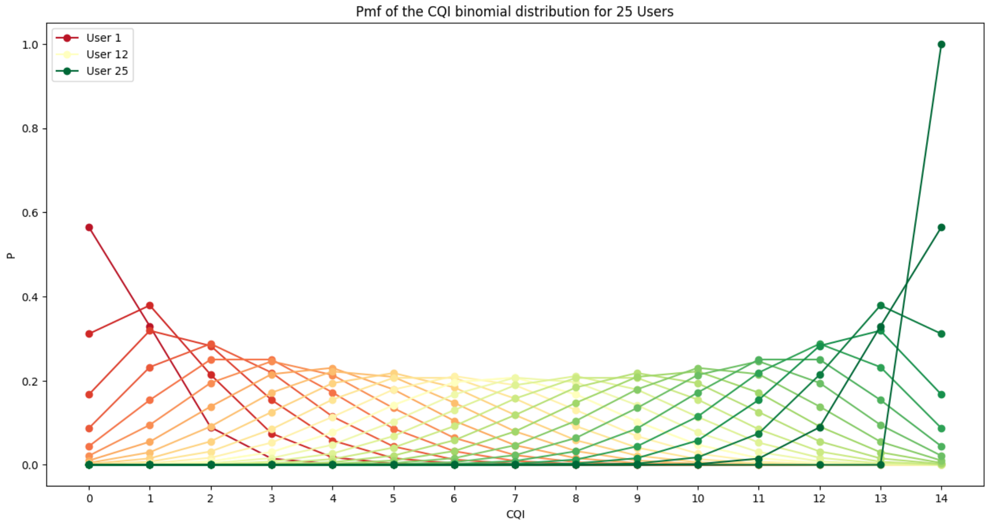


Figure 2.1: PMF of the CQI binomial distribution for 25 users.

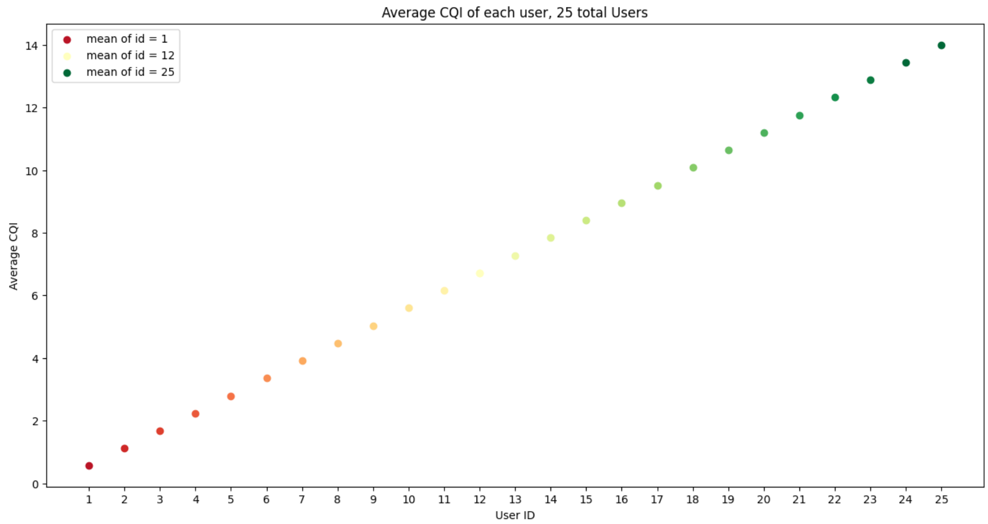


Figure 2.2: Average CQI for each user in the binomial case, with 25 total users.

Chapter 3

Verification

To ensure the correctness of the system during the design phase, four different types of tests were conducted:

- Degeneracy
- Scheduling algorithm
- Continuity
- Consistency

3.1 Degeneracy Test

For the degeneracy test, the system was simplified: we assumed to have a single user, fixed-size packets, and an exponential distribution of packet arrivals. The Channel Quality Indicator (CQI) was fixed, allowing the system to send at most one packet to the single user. Under these conditions, the system behaves as an M/D/1 (Figure 3.1).

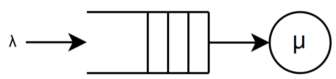


Figure 3.1: M/D/1 queue

The average number of packets in the queue is expected to follow:

$$E[Nq] = \frac{\rho^2 + \lambda^2 \cdot Var(t_S)}{2(1-\rho)} = \frac{\rho^2}{2(1-\rho)}$$

$Var(t_S)$ is equal to zero because t_S is deterministic and constant. Figure 3.2 shows how the experimental results obtained from the simulation align with the ideal curve.

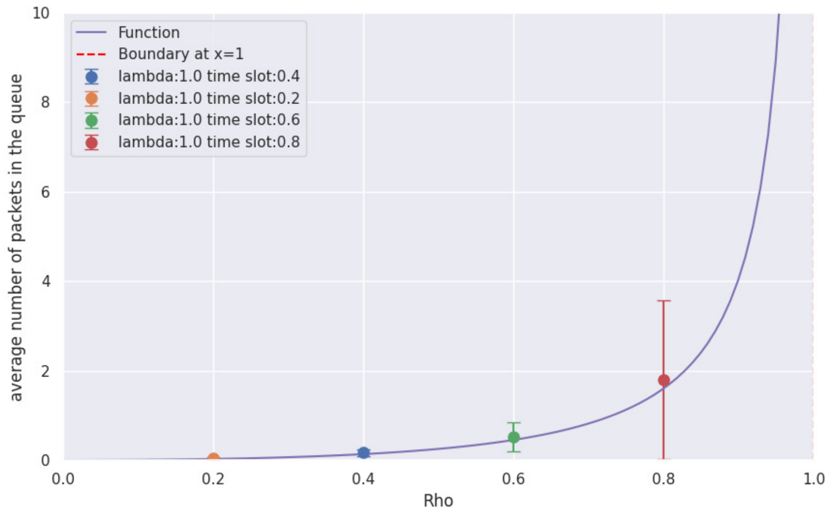


Figure 3.2: Number of Packets in the Queue

Experimental data were collected through simulations lasting 70 seconds, with variations in λ and timeslot to achieve ρ within a deviation of 0.2.

3.2 Scheduling Algorithm Test

To test the Round Robin algorithm, deterministic values for both sent packets' size and users' RB were set: each packet, with a fixed size of 25 *bytes*, was sent every $0.02s$. The system, with a timeslot of $0.1s$, was not able to empty the users' queues. The three users in the simulation had RB values of 25, 50, and 75 bytes respectively, for each timeslot. Finally, the frame had a fixed length of 1 RB. With these assumptions, we expect the

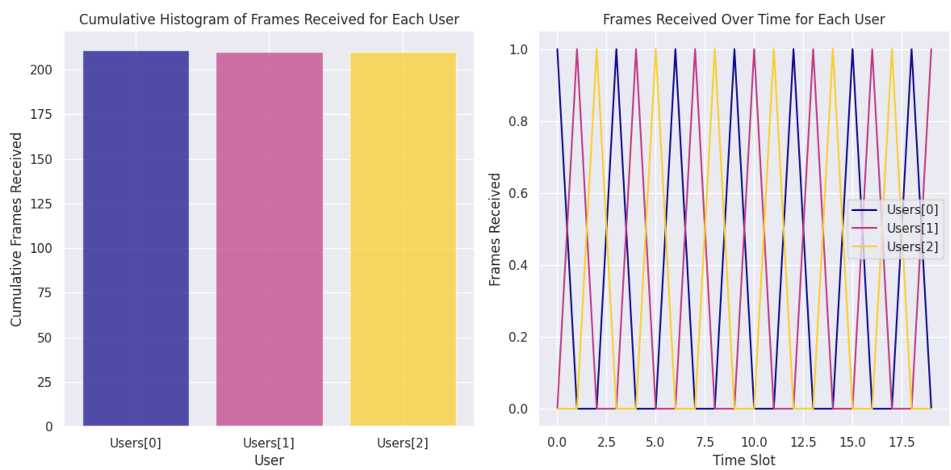


Figure 3.3: Frames Received

number of frames sent to be the same for all users, and for the algorithm to send one frame to each user at each timeslot, as shown by the experimental result in Figure 3.3 (we consider a frame to be received by a user if the frame contains at least one packet intended for them). As for the number of packets received we expect *User[0]* to receive

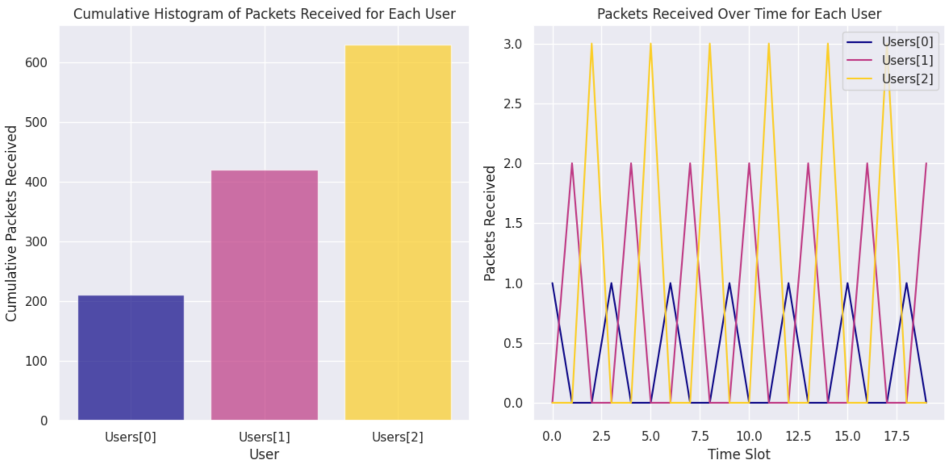


Figure 3.4: Packets Received

one packet per frame, *User[1]* to receive two packets per frame, *User[2]* three, and so on, as clearly shown in Figure 3.4.

3.3 Continuity Test

For the continuity test, the system was analyzed with 10 active users, evenly distributed CQI, and a fixed timeslot of 0.01s. We varied λ to observe changes in the value of the throughput under different configurations. Figure 3.5 shows that as λ increases, the throughput increases as well. The various configurations do not however exhibit significantly different trends.

3.4 Consistency Test

For the consistency test, λ and the inverse of the timeslot were varied inversely to maintain a constant ratio of 1. In this scenario, the system is expected to maintain consistent throughput values.
Figure 3.6 illustrates that the recorded values remain stable despite variations in λ and the inverse of the timeslot.

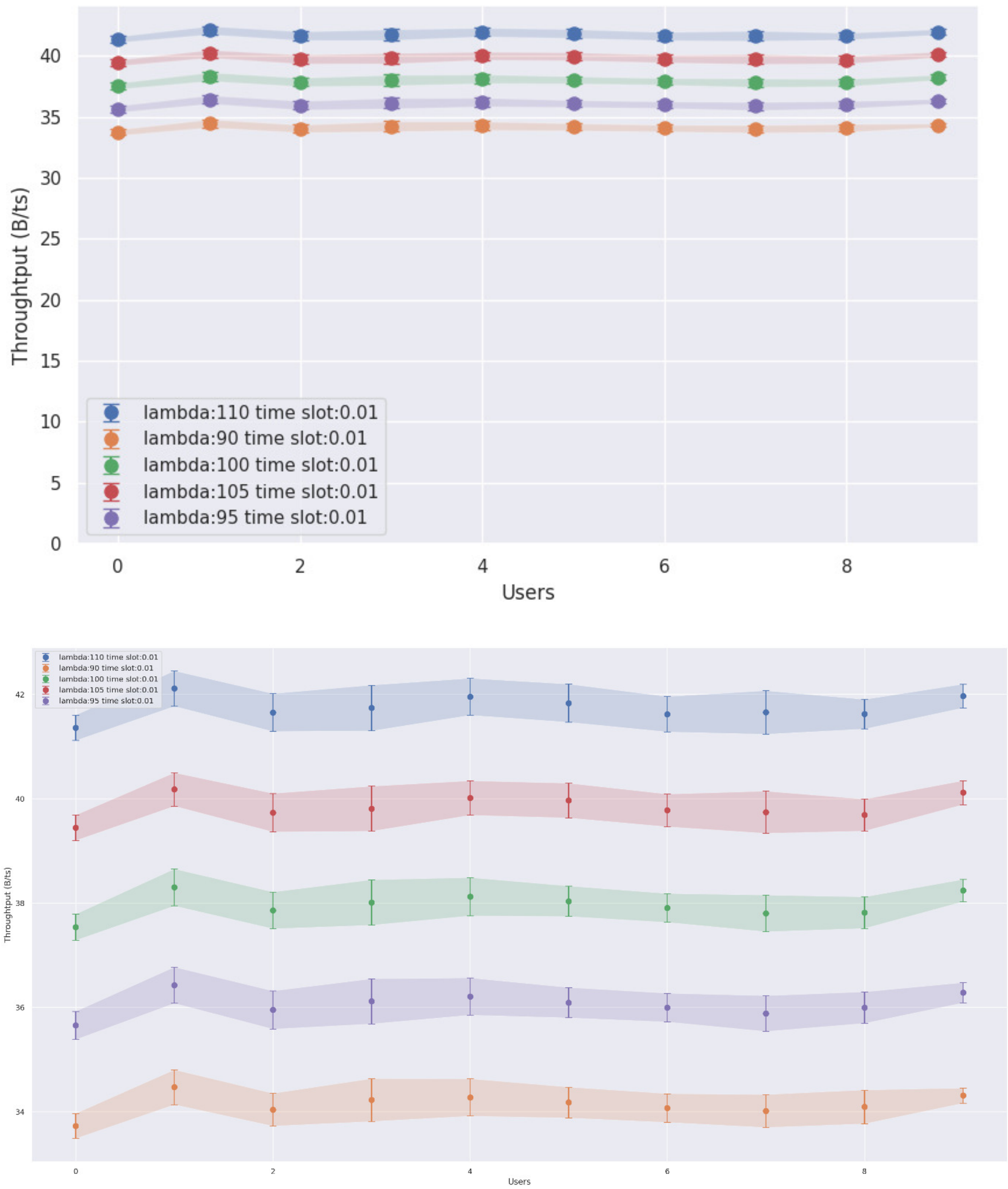
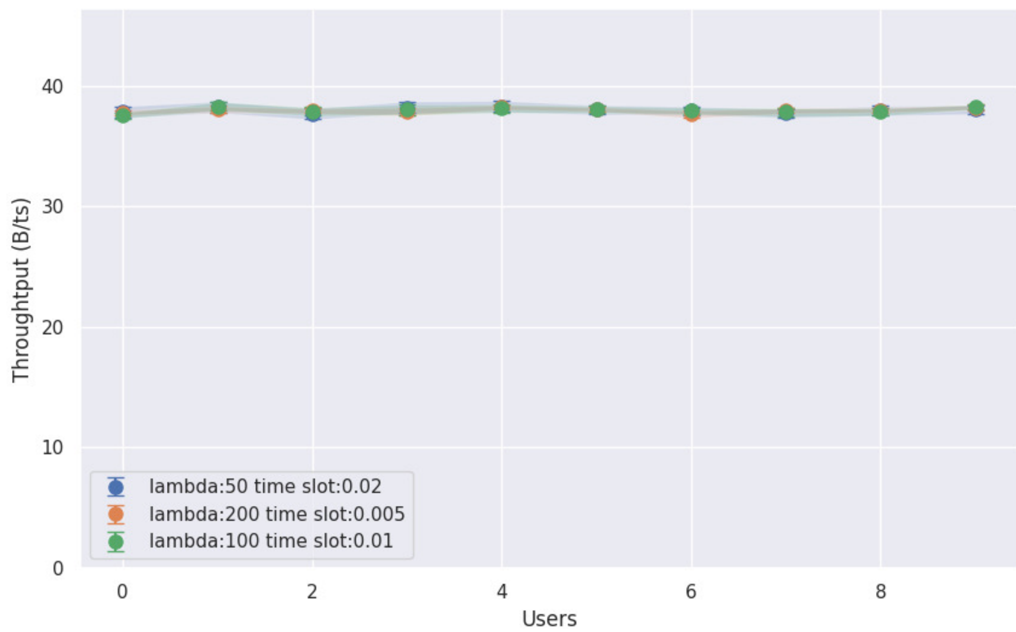


Figure 3.5: Throughput at Different λ Values (the second plot is the zoomed version of the first one)



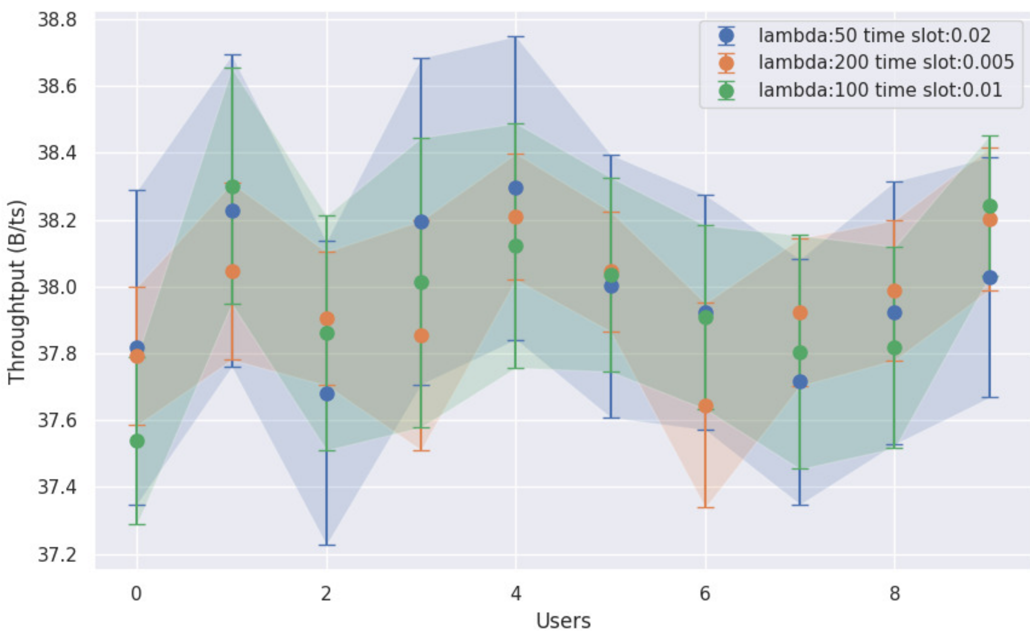


Figure 3.6: Throughput with different values for λ and the inverse of the timeslot (the second plot is the zoomed version of the first one)

Chapter 4

Result Analysis

4.1 Stability

Uniform CQI					
$nUsers$ \ λ	25	50	100	150	200
10	✓	✓	✓	✓	✓
25	✓	✓	✓	✗	✗
50	✓	✓	✗	✗	✗
100	✓	✗	✗	✗	✗

Binomial CQI					
$nUsers$ \ λ	25	50	100	150	200
10	✓	✓	✓	✗	✗
25	✓	✗	✗	✗	✗
50	✗	✗	✗	✗	✗
100	✗	✗	✗	✗	✗

Table 4.1: ✓: stable configuration ✗: unstable configuration

We approached the problem of stability by considering the evolution over time of the total *backlog* at the *Base Station*, counted by summing up all of the users' individual backlogs. We consider a model to be stable (precisely its *combination of parameters*) if we observe the existence of a constant that bounds this value over time (to establish that a constant actually exists, we guess it by looking at data over a certain amount of time and then checking that our guess still holds by doubling the simulation time). We reported some of the stable combinations observed for the studied parameters during simulation, for both the uniform and the binomial case (table 4.1). We have also noticed that, in the uniform case, all queues reach instability at the same lambda value. In the binomial case, as lambda grows, queues reach instability one after the other, starting from the lowest ID (Figure 4.2).

4.2 Warm-up

To estimate the warm-up time, we studied the time it takes for the total number of packets in all the queues to reach a steady state. We ran 30 repetitions of a simulation with a specific configuration of parameters $[nUsers = 50, \lambda = 50, CQI \sim uniform]$, which was chosen because it could reach a steady state while also having a saturated frame (i.e. the average number of occupied slots is 25). We computed the *sliding window average* (*window size* $= 100$, graph 4.1). We chose 10 *seconds* as *warm-up time*, after which we can start looking at the behavior of the system once the transient phase is extinguished.

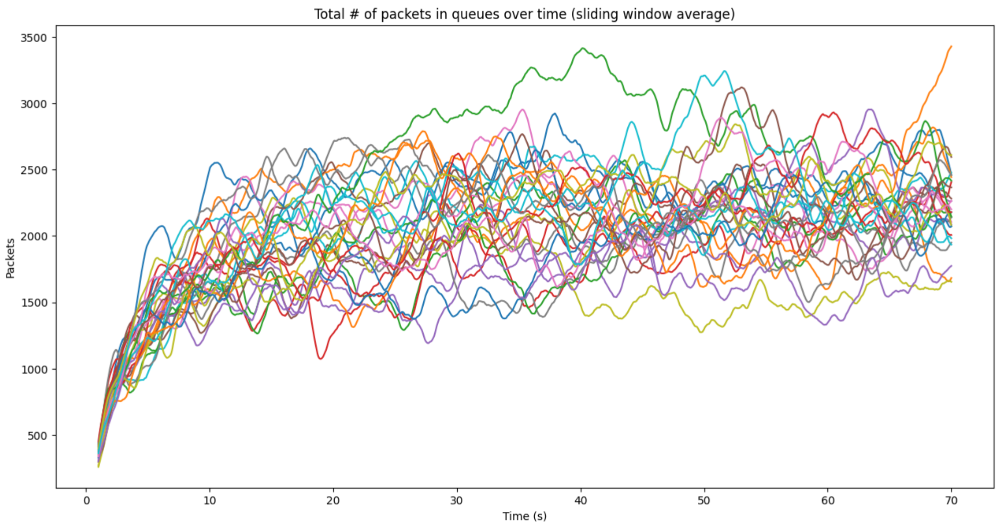


Figure 4.1: sliding window average

4.3 Saturation

First off, we looked at the level of saturation of the frame, as shown in the table 4.2
The cells highlighted by a scale of gray refers to the case with *binomial CQI*, while the pink ones regard the case with *uniform CQI*. As can be seen, the two cases are quite similar.

$nUsers$ \ λ	25		50		100		150		200	
10	6,932	7,799	12,89	13,665	21,461	21,236	24,511	24,427	24,82	24,976
25	17,96	16,92	24,798	24,574	24,987	25	25	25	25	25
50	24,93	24,94	25	25	25	25	25	25	25	25
100	25	25	25	25	25	25	25	25	25	25

Table 4.2:
: Binomial CQI
: Uniform CQI

After this we looked at the throughput for every user, defined as the number of bytes received per timeslot, and the delay between the generation of a packet and its arrival at the destination.

The *Central Limit Theorem* states that, regardless of the population's distribution, the distribution of the sample means approximates a normal distribution as the sample gets larger. Sample sizes equal or greater than 30 are often considered sufficient for the CLT to hold and that's the reason behind the 30 repetitions we executed for every simulation run. The more repetitions, the more the results will converge to the expected values.

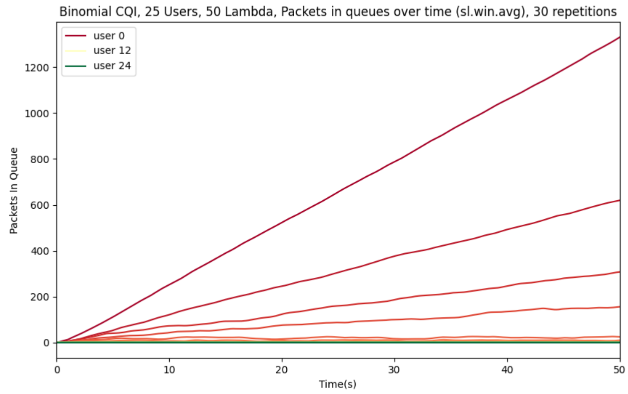
We are also confident that the variance of both the throughput and the packet delay is finite; to test this, we checked the variance for multiple sizes of our samples, and we saw that the variance did not increase with the sample size.

By doing this we were able to exploit the Normal distribution to compute the *confidence intervals* for the average values of both the *throughput* and the *delay*.

4.4 Throughput

In the uniform CQI case, we notice that the throughput is roughly the same for all users. This is reasonable, since no user has privilege in the scheduling and everyone has the same odds of having a 'good' or 'bad' CQI. As λ increases, so does the throughput, up to a max. After exceeding a certain λ threshold, the user that is served first tends to occupy nearly the entire frame. Consequently, there is a restricted number of Resource Blocks (RBs) available to other users at each round. As a result, the latter will transmit a reduced number of packets during their round, leading to a substantial decrease in their individual throughput. This pattern repeats itself every round, for different users, significantly increasing variance.

In the binomial case, things are different: users with higher ID have better throughput. This is because users with a high ID number have much greater odds of having a very high CQI than users with a low ID. Hence the BS will be able to store more packets in



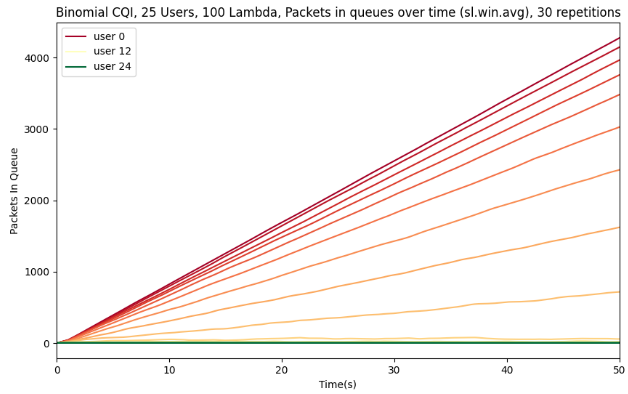


Figure 4.2: Graphs of the number of packets in queues over time.

the RBs, leading to a more efficient use of the frame.

Looking at parameters [nUsers = 50, λ = 200] and [nUsers = 100, λ = 100], we noticed an unusual throughput decrease for high CQI users, and an increased confidence interval.

The reason for this behaviour lies in the scheduler: one user's queue with very high CQI is very often scheduled right after another user's queue with very high CQI. This means that the latter will only be able to take advantage of a few RBs, the majority of which will be used by the first one, who has the whole frame at its disposal. On the next timeslot, the situation repeats itself, with the first user taking full advantage of the frame while the second has only the 'scraps' of the frame left. This then creates the unusual behaviour that we see.

Graph 4.6 shows the average number of slots that are already occupied when the time of scheduling for a particular user comes. As we can see from the 'zig-zag' pattern, most of the times one user sees an high number of already occupied slots, while the next sees an empty frame.

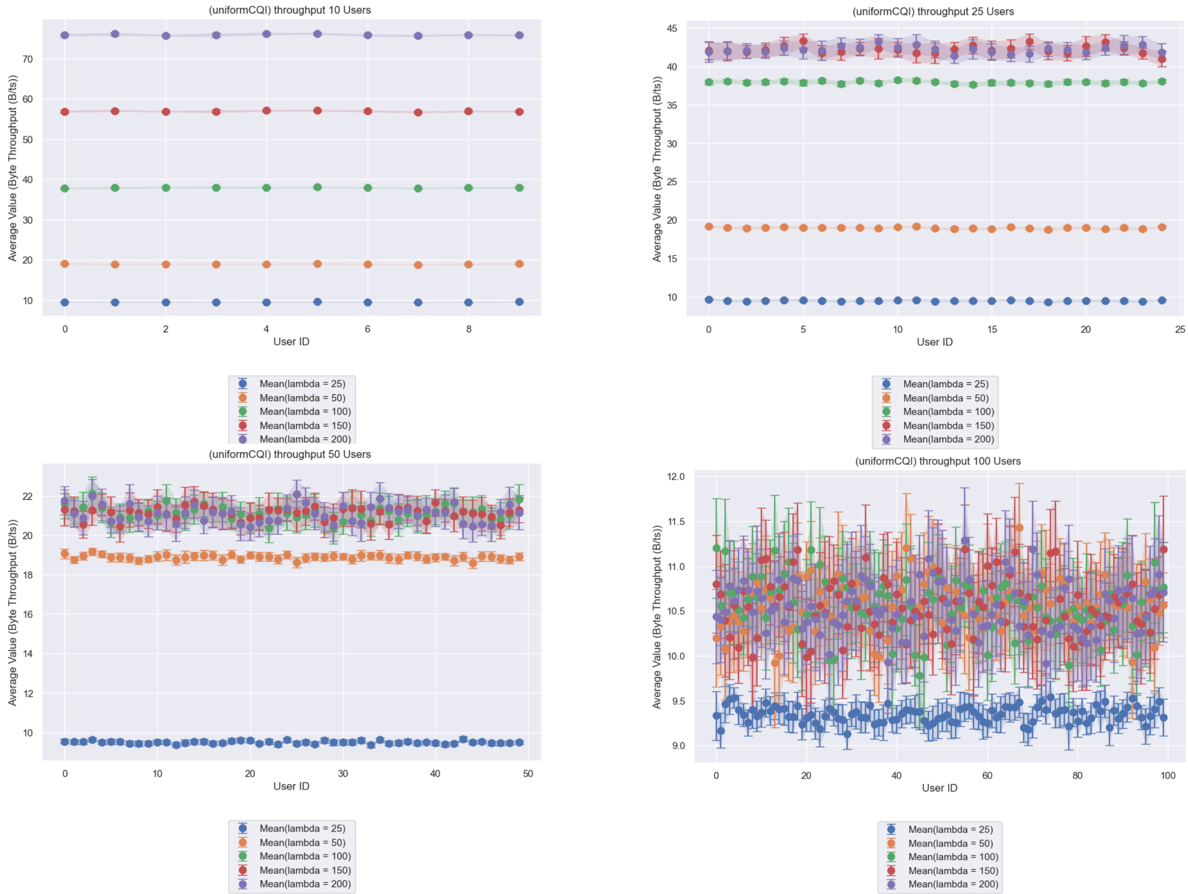


Figure 4.3: Throughput results for the uniform CQI case.

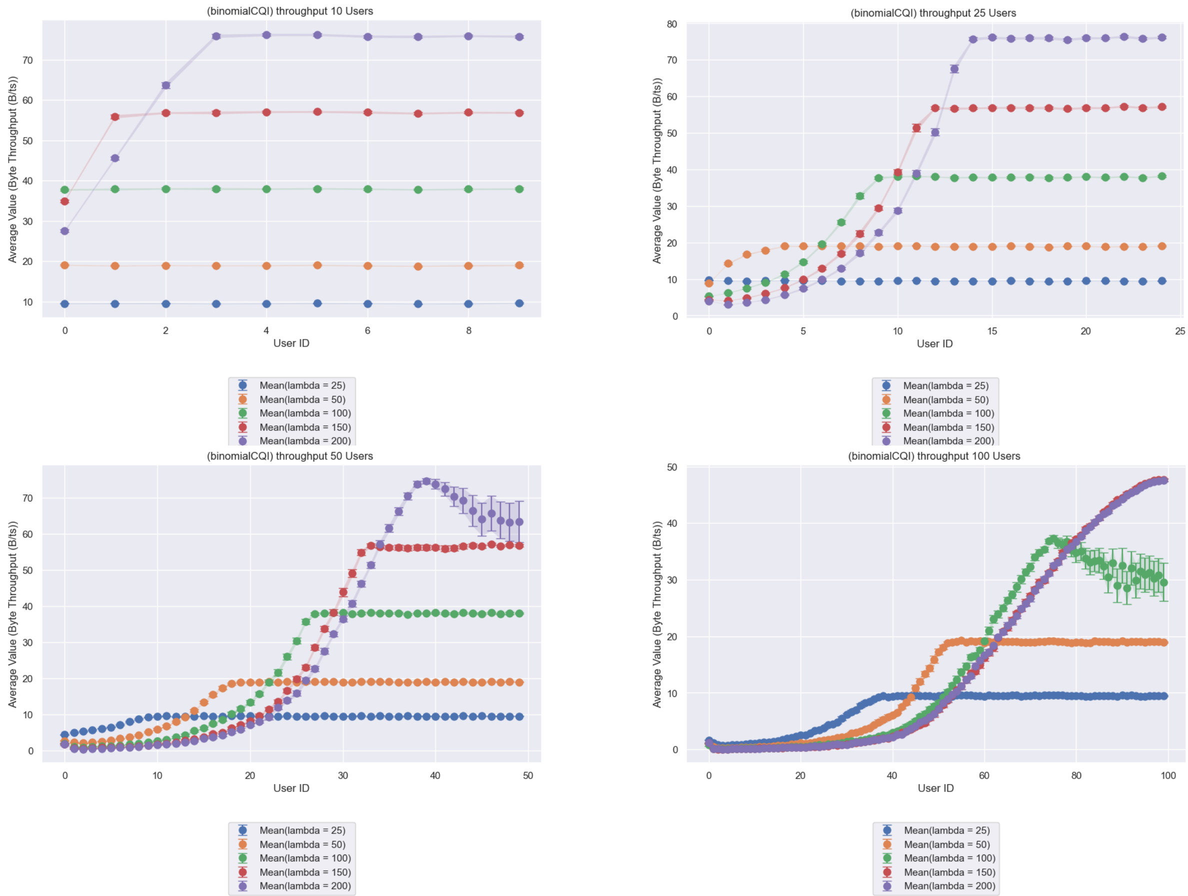


Figure 4.4: All results for the binomial CQI case.

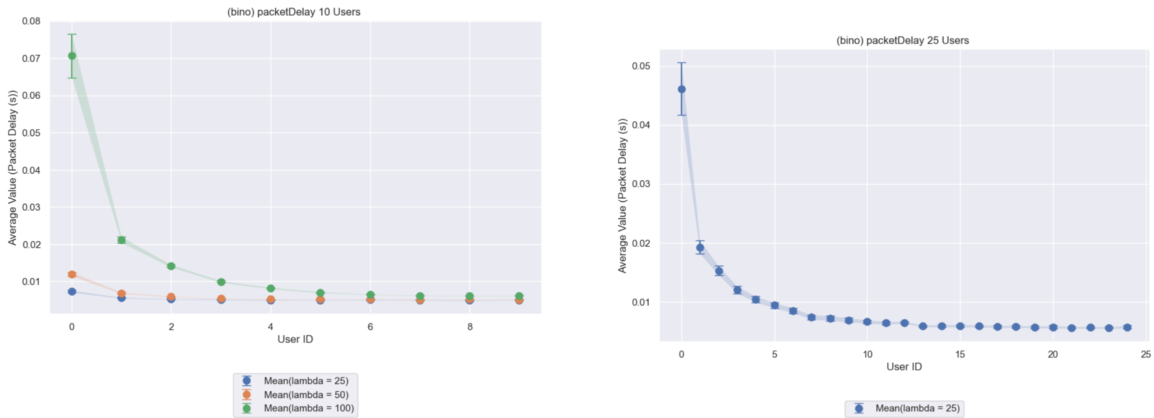


Figure 4.5: Delay results for stable configurations in the binomial CQI case.

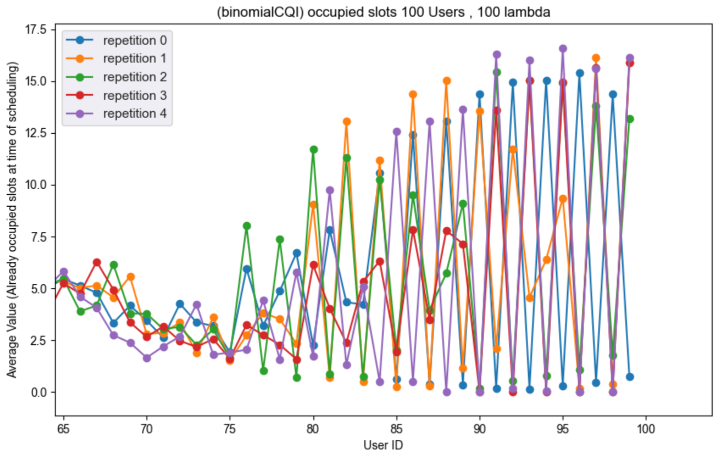


Figure 4.6: Graph of already occupied frame slots at time of scheduling.

4.5 Packet Delay

First off, we suspected that the observations for packet delay would not be independent, since the more packets are in a queue, the more delayed a packet is going to be. This was confirmed by the Autocorrelation graph (figure 4.7). In order to obtain independent observations we applied to our original sample a subsampling technique that includes points with a probability p. The chosen value for p is the one that yields the larger amount of observations that turn out to be independent:

$$p = \frac{1}{2^4}$$

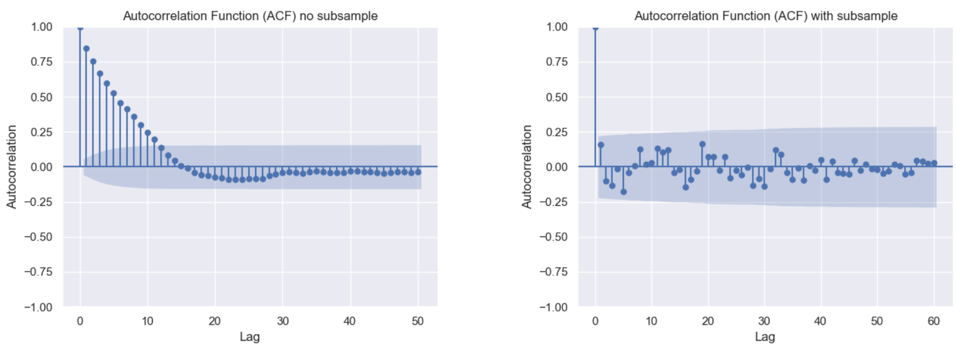


Figure 4.7: Graph of ACF of packet delay of a user with and without subsampling.

When the model is stable, so will be the packet delay: every packet will eventually arrive at its destination. For some of these stable cases, we plotted the average delay that the users

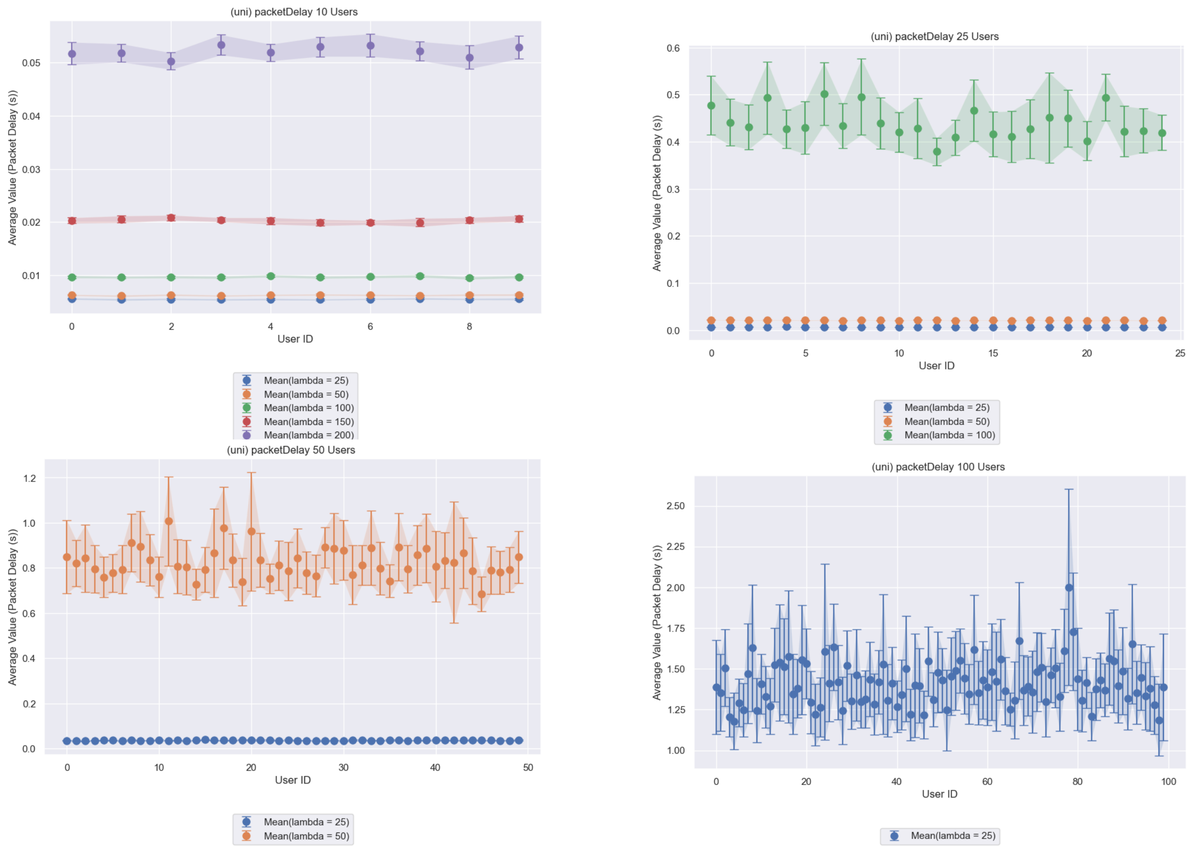


Figure 4.8: Delay results for stable configurations in the uniform CQI case.

in our model experience (figure 4.8). On the other hand, when the model is unstable, we got at least one users' queue that grows indefinitely. As a logical consequence we expect the packet delay of that (or those users) to diverge just as well. For these cases we plotted the packet delay experienced by each user over time (figure 4.9) As we can see, for the uniform case the packet delay is the same for every user, for the same reason as the throughput. For the binomial, instead we can see that users with higher CQI experience much less delay than low CQI users for lower values of λ, but as λ grows, the delay becomes more similar among all users.

4.6 Fairness

Evaluating the fairness of the model requires a distinction between the two scenarios examined: the uniform scenario and the binomial scenario. In the first case, the Lorentz curve (figure 4.10) shows a maximum level of fairness. This result can be attributed to the equal distribution of resources among users, ensuring that each is equally likely to benefit from a high CQI. In contrast, in the binomial case, disparities are observed with some of the users that benefit from more resources (consequently better CQI), while

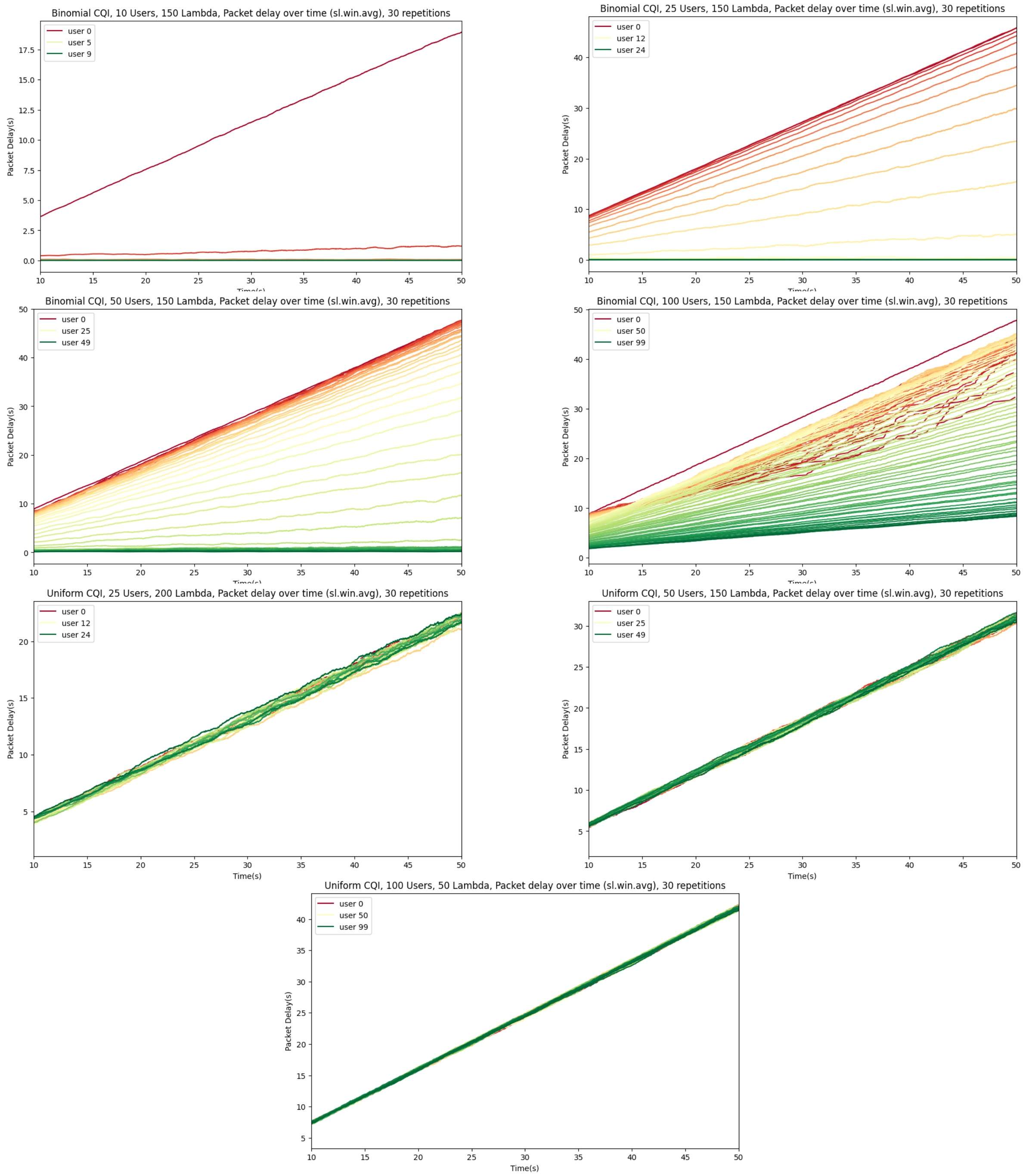


Figure 4.9: Significant results for unstable cases.

others send significantly fewer packets per slot. In particular, with low λ values, we expect higher fairness. This phenomenon occurs because, with a limited number of packets, even disadvantaged users manage to empty their queue efficiently, just like the privileged ones. In this context all users send a similar number of packets. However, as λ increases, we notice an increase in the amount of packets in the queues. Therefore only privileged users will be able to empty their queues efficiently (with a resulting high throughput). On the other hand, users with low CQI will send a similar number of packets as observed with low λ. This dynamic is clearly illustrated in the graphs in Figure 4.11, where the curves with higher values of λ deviate significantly from the line of maximum fairness.

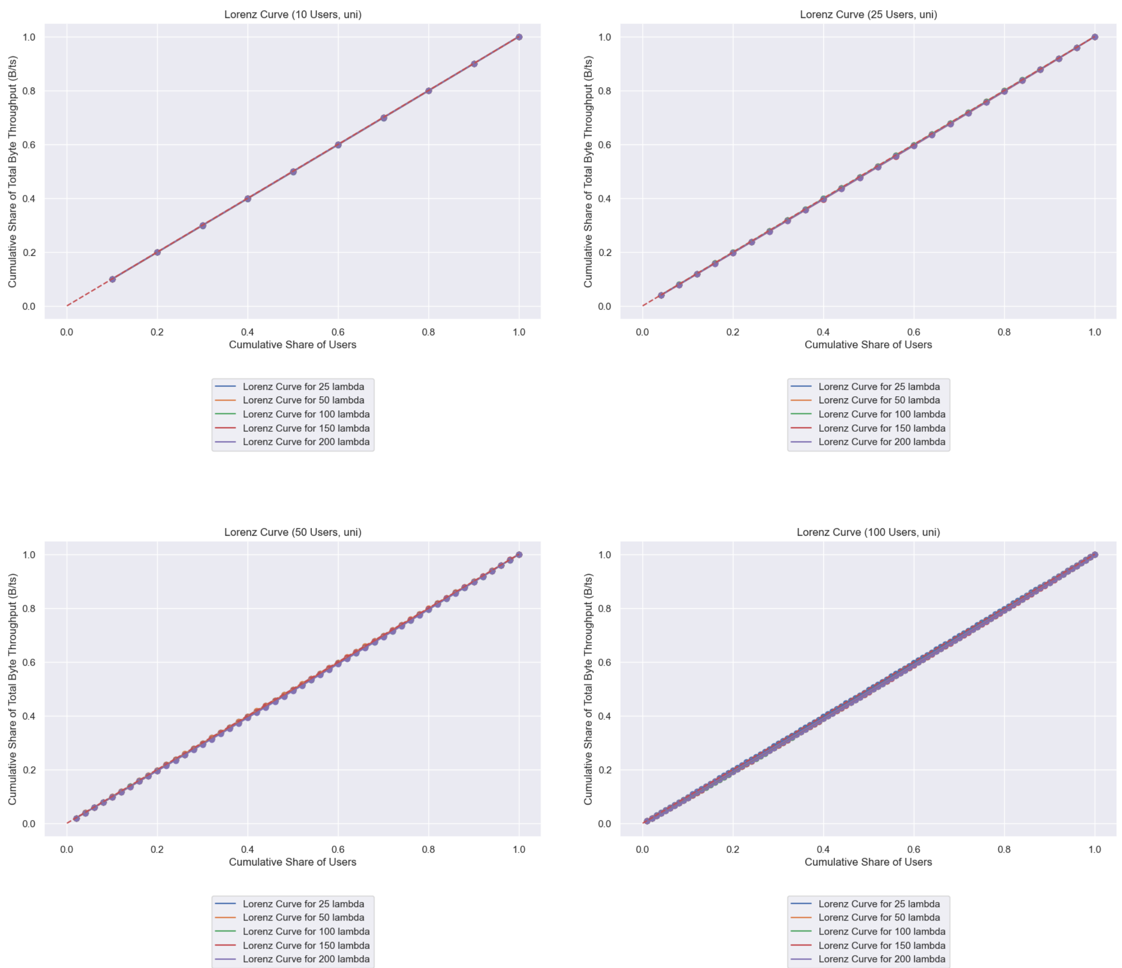


Figure 4.10: The Lorenz Curves reflect maximum fairness in the *uniform CQI* case.

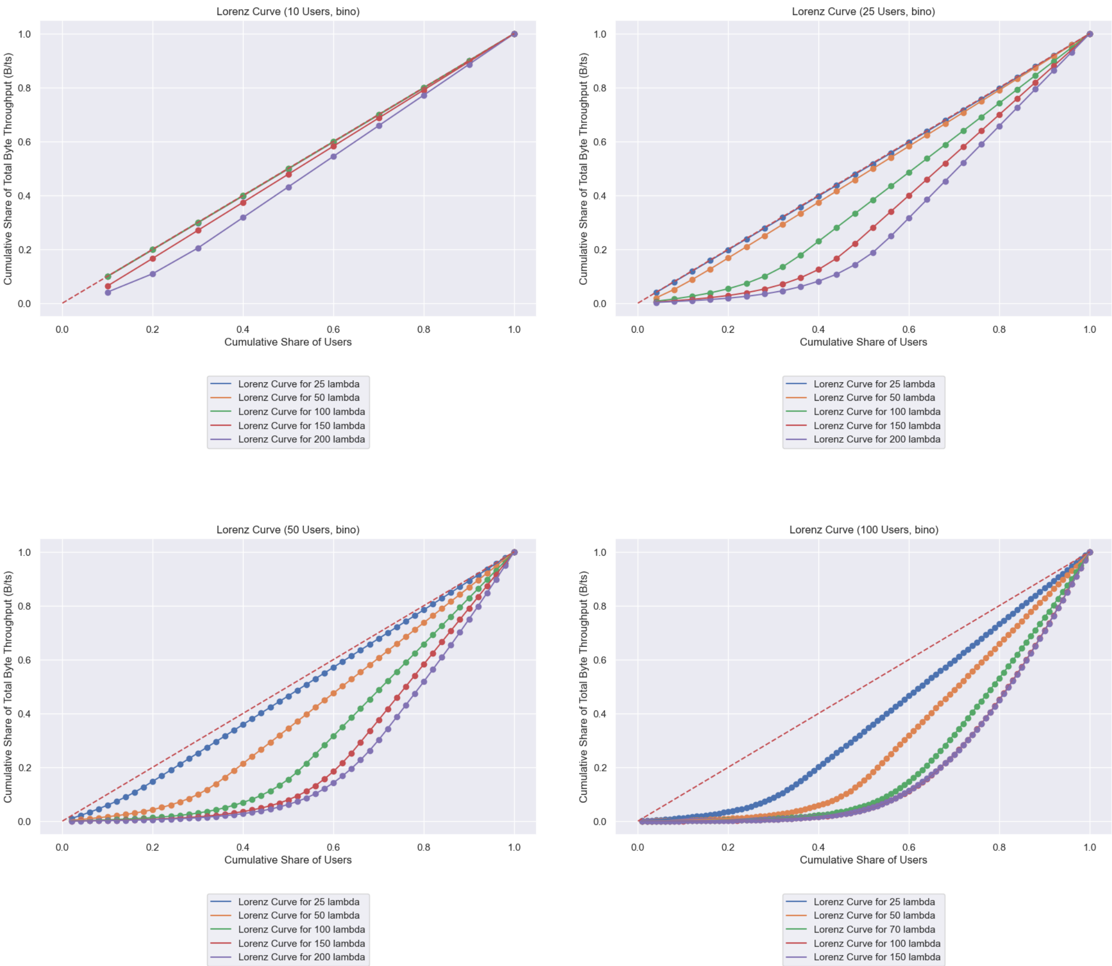


Figure 4.11: Lorenz Curves for the *binomial CQI case*

Chapter 5

Conclusions

The system we studied through the project shows optimal performances when the users are able to receive on average the same number of packets, making the Round-Robin scheduling a valid implementation choice in its simplicity.
Whenever a notable disparity in CQI values between users is expected, however, the system does not appear to be the more appropriate choice. In this scenario significant differences in throughput would indeed occur, with some users benefiting from high throughput and others, disadvantaged, with very low throughput. Moreover, it would require queues of considerable sizes to be provided for the users with lower CQI values.