PERFORMANCE EVALUATION AND APPLICATIONS

Project presentation

Project D: Performance of the LAN in a Cloud Datacenter

Matteo Makovec
Student ID: 10782774
QR-code ID: 24499815

POLITECNICO
MILANO 1863

CONTENTS

1. INTRODUCTION AND ANALYSIS

A Cloud Datacenter works on a three-layer architecture:

- Access layer
- Aggregation layer
- Core layer

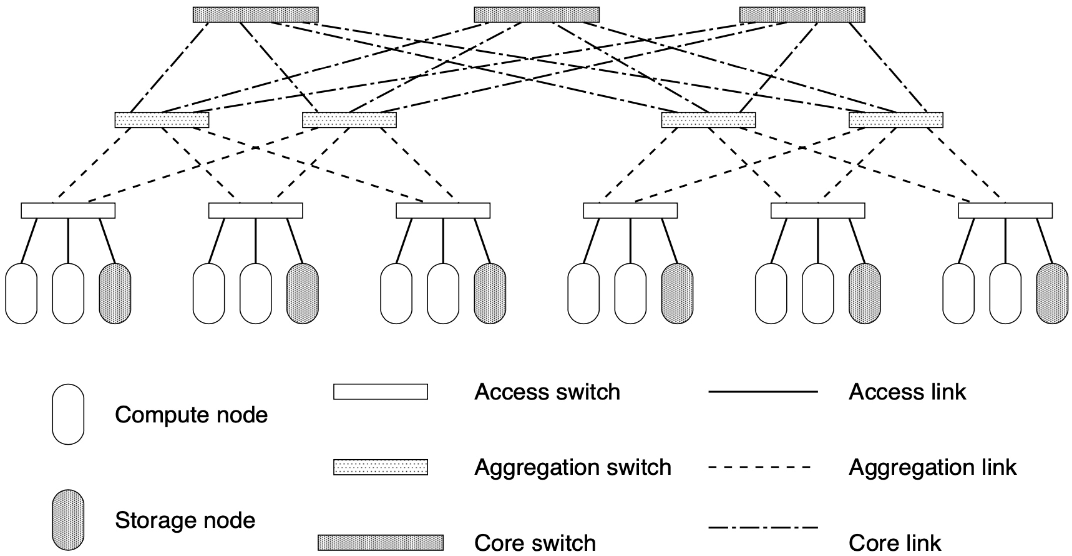

Figure 1: Schema of the Cloud Datacenter

Switches on different layers **process data blocks** differently:

- Access layer: 112 MB/s
- Aggregation layer: 280 MB/s
- Core layer: 1.12 GB/s = 1.12 * 1024 MB/s = 1147 MB/s

Every layer has two switches, and every switch has a **finite capacity of 16 MB.** Since a data block is 1 MB size, nodes can be modelled with a finite capacity of 16 jobs, with drop policy in case of exceeds.

There are two types of **traffic**:

1. <u>Background traffic</u>
   It is intrinsic in the layer to which the switch operates, and it represents the traffic inherited by the bottom layers.
   Every layer has different processing times:
   - Access layer: $\lambda 1 = \lambda 2 = 40$ MB/s
   - Background layer: $\lambda 3 = \lambda 4 = 180$ MB/s
   - Core layers: $\lambda 5 = \lambda 6 = 600$ MB/s

2. Traffic A, B, C, D described by log traces
   Traffic coming from nodes attached to switches in the Access layer. Those traffics are described by data traces provided with the system description, from TraceD-A.txt to TraceD-D.txt.

Talking about the **routing policy**, every Access switch sends packets A, B, C, D to one of the above switches, while the Aggregation switches only to the right above Core switch (thus Aggregation1 to Core1 and Aggregation2 to Core2).
Background traffic is routed to the two upstream nodes.
Thus, there are 4 types of routing:
- A: 1→3→5
- A: 1→4→5
- B: 2→4→6
- B: 2→3→6

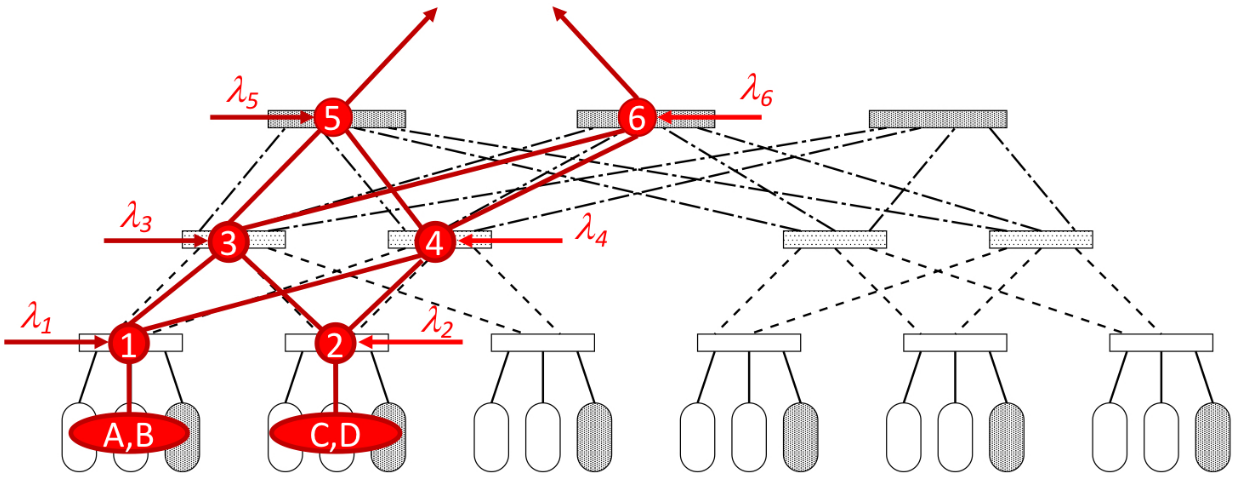

Figure 2: Routing policies

2. MODEL OF THE SYSTEM

As modelling tool, it has been chosen JMT (Java Modelling Tool).

The system can be modelled as a **queueing network**, where the queueing stations are the switches of the different layers (Access, Aggregation and Core).
It is **open model**, since traffic comes from outside the system and, once processed, it leaves the system.
Every switch has a finite **queue capacity of 16.** Once requests exceed, a **drop policy** is going to be applied.
Service time distributions have been modelled with an exponential distribution, and lambda coefficient depends on the layer to which the switch belongs.
Therefore:

- An **Access switch** would have a Service time distribution **exp (112)**
- An **Aggregation switch** would have a Service time distribution **exp (280)**
- A **Core switch** would have a Service time distribution **exp (1147)**

This service time distribution value has been applied only to the classes actually processed by the station. If a traffic class cannot reach that station, the service time distribution is set to disabled.

Traffic has been modelled with different Sources.
Traffics A, B, C, D belong to the common source component, and they are linked to the Access layer.
The background traffic has been modelled with different sources, one of each lambda (λ1, λ2, ..., λ6), and every lambda has been linked with the respective switch.

Classes Characteristics
Define type (Open or Closed), name and parameters for each customer class.
Closed Classes: If a **ClassSwitch** is in the model, then **all** the closed classes must have the **same** reference station.
Open Classes: An open class that has **Fork**, **ClassSwitch**, **Scaler** or **Transition** as the reference station is **not** generated by **any** Source.
Priorities: A larger value implies a higher priority.

Add Class

Classes: 10

Color	Name	Type	Priority	Population	Interarrival Time Distribution		Reference Station
	A	Open	0		exp(20.084)	Edit	Arrivals
	B	Open	0		exp(10.066)	Edit	Arrivals
	C	Open	0		exp(19.882)	Edit	Arrivals
	D	Open	0		exp(34.928)	Edit	Arrivals
	Lambda1	Open	0		exp(40)	Edit	Lambda 1
	Lambda2	Open	0		exp(40)	Edit	Lambda 2
	Lambda3	Open	0		exp(180)	Edit	Lambda 3
	Lambda4	Open	0		exp(180)	Edit	Lambda 4
	Lambda5	Open	0		exp(600)	Edit	Lambda 5
	Lambda6	Open	0		exp(600)	Edit	Lambda 6

Figure 3: Multiclass model

To proper create the different Sources, a new class has been added for every type of traffic we want to describe.

Since **lambdas** are Poisson processes, they have been modelled with an Exponential distribution:

- **λ1 = λ2 = exp (40)**
- **λ3 = λ4 = exp (180)**
- **λ5 = λ6 = exp (600)**

Regarding the traffics A, B, C, D, deeper considerations should be done.
Log files of the four data traces have been provided. These can be used to discover the distribution of the traces.
To do so, it has been used MATLAB. ***[see file tracesAnalysis.m]***
Every data trace has been open, the interarrivals have been calculated from the timestamps, and, using the method of moments, different distributions have been fitted (uniform, exponential, hyper-exponential, and hypo-exponential).
Then plotting, the distribution that better approximates the trace can be visually identified.

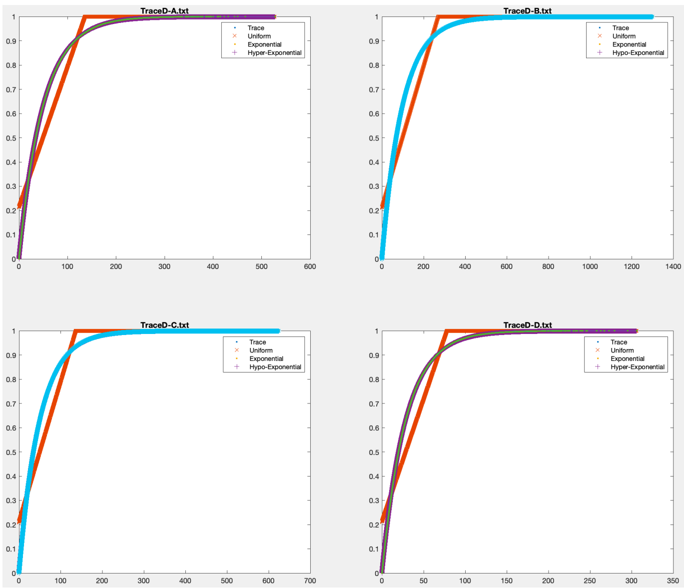

Figure 4: data trace and distributions

Looking at the plots, it can be seen that the trace can be approximated both from an exponential distribution and a hyper/hypo exponential depending on the cases (traceA and traceD with hyper-exponential, traceB and traceC with hypo-exponential).
Since the values of the coefficient of variance (cv) can be truncated to 1, we can choose to model all the four **data traces** with an **Exponential distribution.**
With computations, we have found:

- **λ(A) = 20.084**
- **λ(B) = 10.066**
- **λ(C) = 19.882**
- **λ(D) = 34.928**

Last but not least, **routing policies.**
Traffics A, B communicates with the switch Access 1, while traffics C, D with Access 2.

[All the considerations below are taken for one switch of a layer. The routing policies of the other switch in the same layer are specular]

<u>Access 1</u>
All the traffic coming into the station (A, B and λ1) is going to be routed with equal probability to the two switches of the layer above.

Routing Options

Destination	Probability
Aggregation 1	0.5
Aggregation 2	0.5

Figure 5: Switch Access 1

<u>Aggregation 1</u>
Lambdas coming into the station (λ1, λ2, λ3) are going to be equally routed to Core 1 and Core 2.

Routing Options

Destination	Probability
Core 1	0.5
Core 2	0.5

Figure 6: Switch Aggregation 1 - lambdas

Different routing policies has been configured for traffics A, B, C, D.
To allows only the four types of routing expressed in the previous section, traffics A and B are going to be routed only to Core 1, while C and D to Core 2.

Routing Options

Destination	Probability
Core 1	1.0
Core 2	0.0

Figure 7: Switch Aggregation 1 - A, B

Routing Options

Destination	Probability
Core 1	0.0
Core 2	1.0

Figure 8: Switch Aggregation 1 - C, D

Routing policies for the **Core layer** are not important, since every packet exiting this layer would exit the system.

Finally, the model of the system would be:
[see file system_model.jsimg]

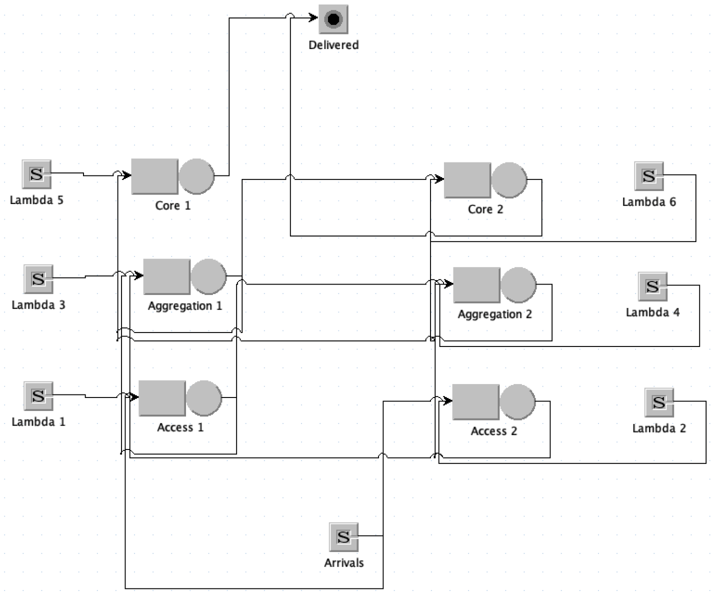

Figure 9: Model of the system

3. EVALUATION OF THE SYSTEM

For the evaluation of the system in terms of performances, it has been used the simulation functionality of JMT.
It is reasonable, of course, to set performance indices to measure the drop rate of the whole system and of every single stations. Another useful measurement to perform is the evaluation of the utilization of the stations, since there is a drop policy when the finite queue of 16 jobs is exceed.

[all the evaluations and values that follow respect a confidence interval of 0.99, as highlighted by the green tick alongside the JMT results in the images]

Performing the simulation, it has been registered an average value for the drop rate of the entire system of **22**.

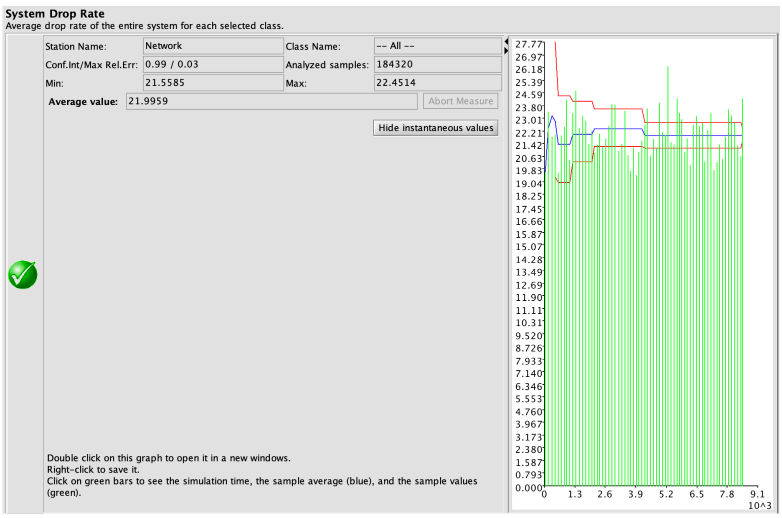

Figure 10: Drop Rate of the whole system

Investigating more on the results, it is possible to highlights the fact that switches in the Aggregation layer have the biggest drop rate values.

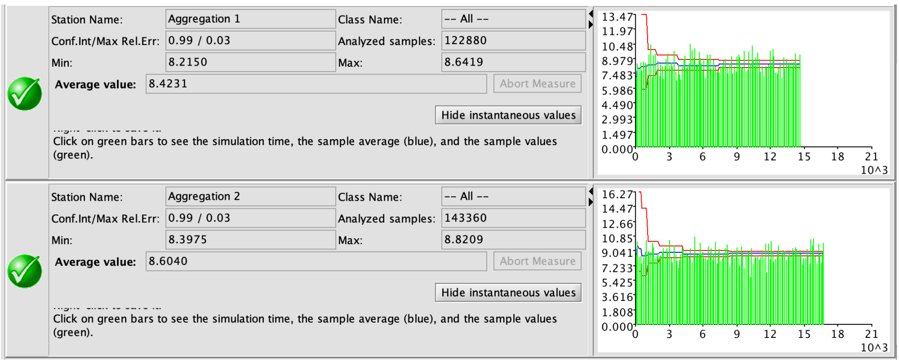

Figure 11: Aggregation switches Drop Rate

This fact can be justified by high utilization values.

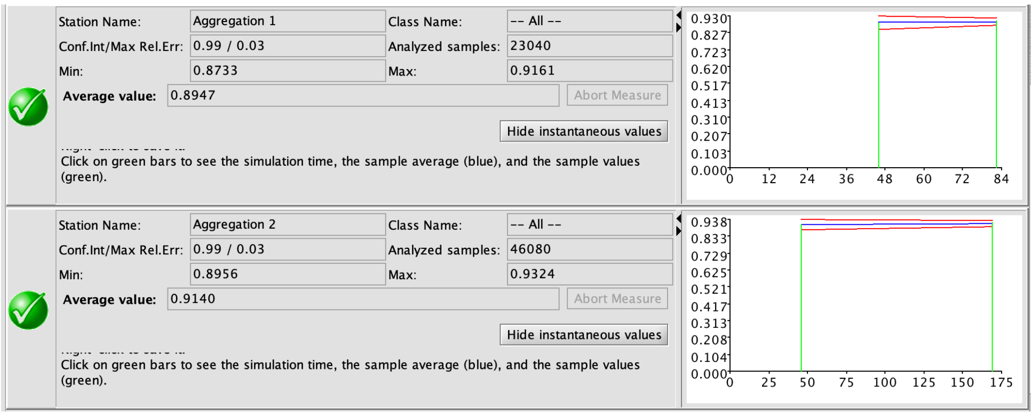

Figure 12: Aggregation switches Utilization

4. TOWARDS THE FINAL MODEL

Since the manager allows to add one switch to decrease the drop rate of the entire system, intuitively it might support the routing in the Aggregation layer.

Discussions can be raised about the routing policies to apply.

One solution could be to equally route all the traffic from the Access layer to the three switches of the Aggregation layer (thus traffics A, B, C, D, λ1, λ2 have equal probability to reach one of the three Aggregation switches).

Routing Options

Destination	Probability
Aggregation 1	0.333333333...
Aggregation 2	0.333333333...
Aggregation_new	0.333333333...

Figure 13: Towards the final model - Routing Policies

The resulting model is:
[see file towards_final_model.jsimg]

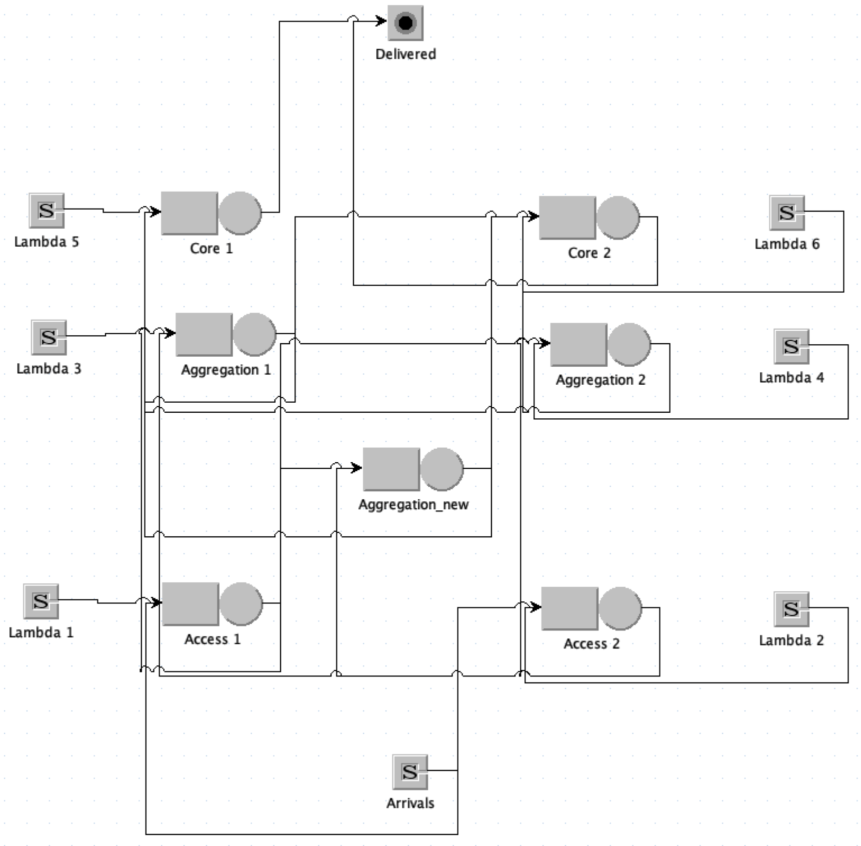

Figure 14: Towards the final model

This leads to a drastically decrease of the drop rate, to the value of **10.18**.

Figure 15: Towards the final model - Drop Rate

However, the system is still not fully optimized.
In fact, the utilization of the switches Aggregation 1 and Aggregation 2 is really unbalanced with respect to the utilization of the new switch.

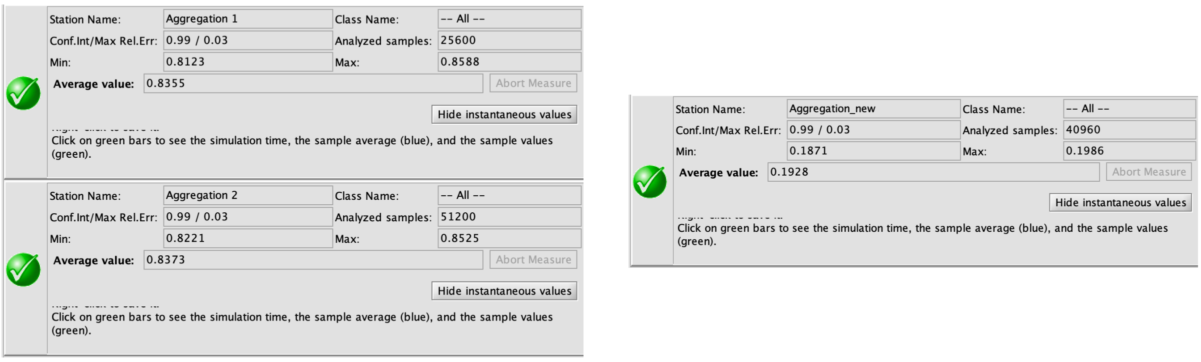

Figure 16: Towards the final model - Aggregation1, Aggregation2 and Aggregation_new Utilizations

This suggests that better results can be achieved exploiting more the processing capabilities of the new switch.

5. SOLUTION: FINAL MODEL

To exploit even more the new switch, it is reasonable to, starting from the model found on the previous section (figure 14), route also a fraction of the background traffic affecting the switches Aggregation 1 and Aggregation 2 to the new station.

Therefore, routing policies for the traffics A, B, C, D, λ1, λ2 remain the same as the one expressed in figure 13.

What changes is the routing policies of the sources λ3 and λ4.
The former has been set with a probability of 0.66 to reach Aggregation1, and of 0.33 to reach Aggregation_new.
Similarly, λ4 has 0.66 of probability to reach Aggregation2, and 0.33 to reach the new switch.

Routing Options

Destination	Probability
Aggregation 1	0.666666666...
Aggregation_new	0.333333333...

Figure 17: Final solution - Routing Policy λ3

Routing Options

Destination	Probability
Aggregation 2	0.666666666...
Aggregation_new	0.333333333...

Figure 18: Final solution - Routing Policy λ4

The final model would be then:
[see file final_solution.jsimg]

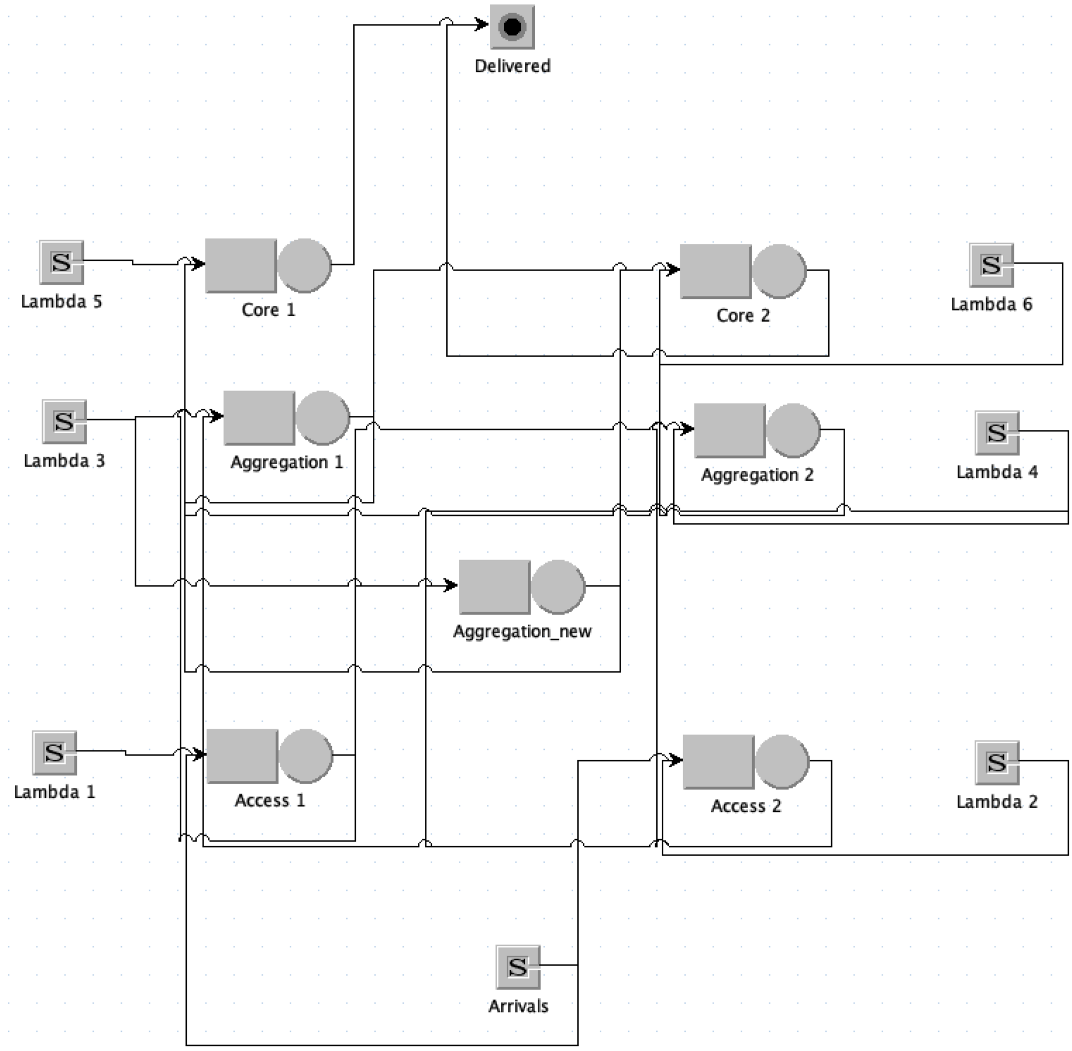

Figure 19: Final solution - Model

This change in the routing policies have balanced the utilizations of the three switches in the Aggregation layer (comparisons can be made with figure12, that is the initial model of the system, and figure16, that is the first step towards the final solution).

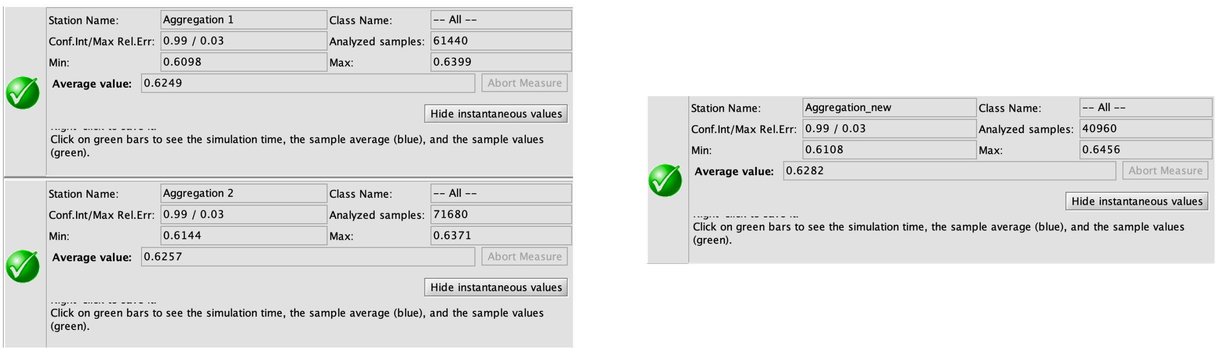

Figure 20: Final solution - Aggregation1, Aggregation2 and Aggregation_new Utilizations

This set up allows to decrease the drop rate of the system again.
The average value reached is **5.75.**

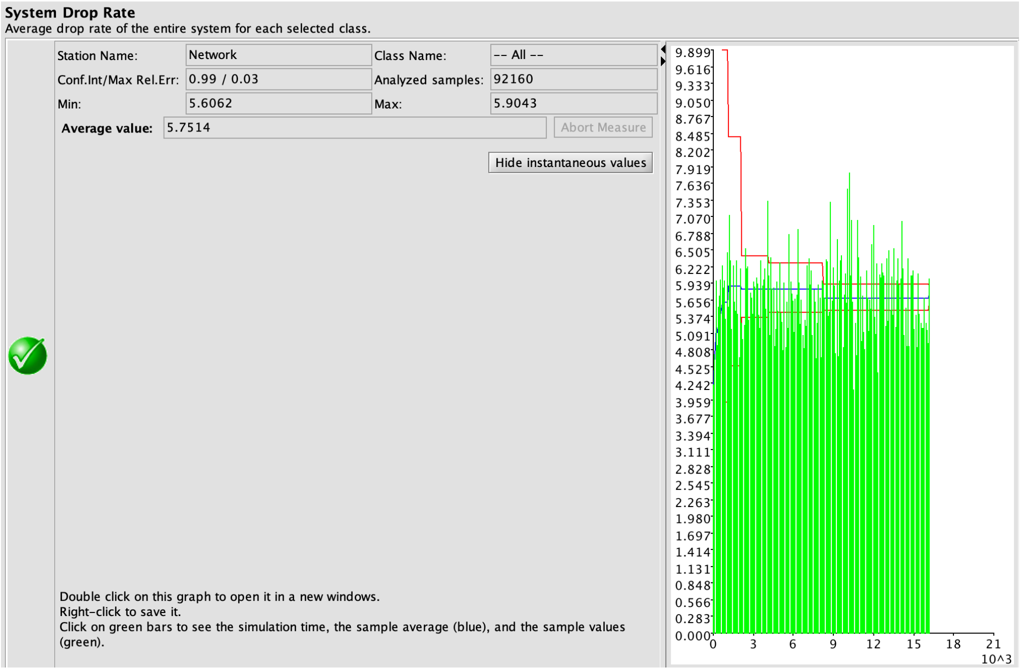

Figure 21: Final solution - Drop Rate

The final solution is a good optimization in terms of performance of the initial model, since it is characterized by a decrease in the drop rate of the entire system of 76.86%.

6. FURTHER CONSIDERATIONS

Another possible final solution, that reaches results similar to the ones reached in the final model of figure19, is in the JMT file *model_ABCDL1L2.jsimg*.
Here, the traffics A, B, C, D, λ1 and λ2 have all been routed to the new switch. Switches Aggregation1 and Aggreagation2 only deal with their background traffic λ3 and λ4.
This solution has not been proposed as the final one since this introduces a bottleneck in the system. In fact, if the switch Aggregation_new goes down, the entire system will collapse.

A more balanced solution would be the one described in the JMT file *model_Lambdas.jsimg*.
Here, the traffics A, B, C, D have been equally routed to Aggregation1 and Aggregation2, while λ1 and λ2 have been routed only to Aggregation_new.
Finally, λ3 (and λ4) has been set with a probability of 0.7 to reach Aggregation1 (and respectively Aggreagation2) and of 0.3 probability to reach Aggregation2.
Given that the drop rate reached is slightly higher than the one of the final model (figure21) and that the configuration is a little unbalanced in the type of traffic that is processed in each station, the more balanced final model described in the section 5 has been preferred.

7. INSTRUCTIONS FOR THE PROJECT

The MATLAB program used to analyze the data traces is *tracesAnalysis.m*, which uses *HyperExp.m* and *HypoExp.m*.

The initial model of the system is in the JMT file *system_model.jsimg*.

The first modification of the initial model is in the JMT file *towards_final_model.jsimg*.

The final model, which is the solution of the project, is in the JMT file *final_solution.jsimg*.

Other solutions, justified in the section 6, are in the JMT files *model_ABCDL1L2.jsimg* and *model_Lambdas.jsimg*.

POLITECNICO
MILANO 1863