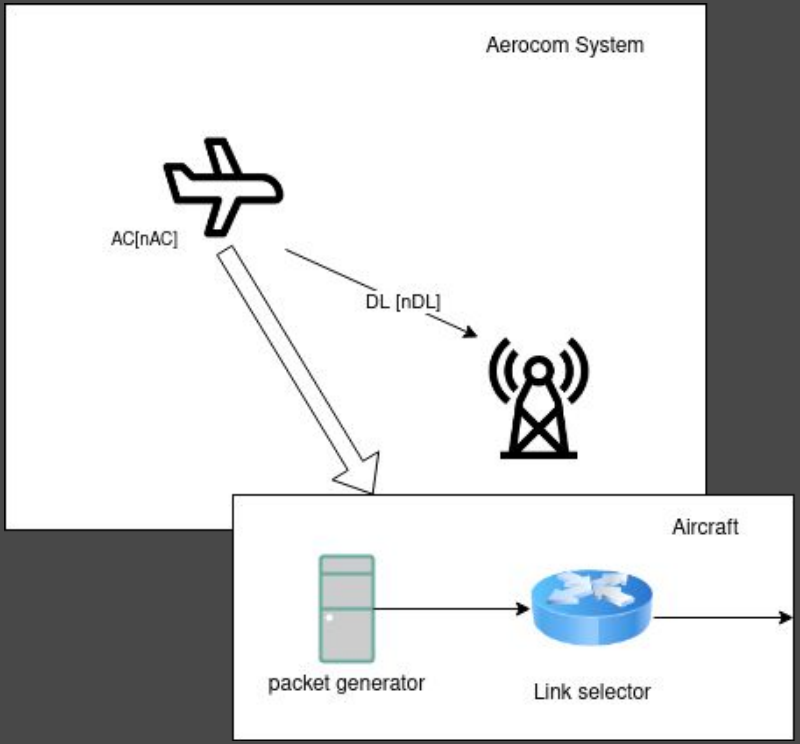
Aerocom System 2

Alessio Di Ricco
Edoardo Geraci
Mariella Melechì

Modelling



- **Aircraft (AC)**: Main module of the analysis composed of:
  - **Packet Generator**
  - **Link Selector**
- **Data Link (DL)**: A DL is the link between ACs and the CT. There is a fixed number of DLs available to all the ACs.
- **Control Tower (CT)**: Receives packets and drops them.
- **Packet**: Created by the packet generator, it's sent to the CT.

Main model informations

Main factors:

- Capacity malus: X;
- Number of aircrafts: nAC;
- Mean packet generation: kMean;
- Number of Data Links: nDL;
- Setting capacity time: t;
- Operation mode: monitored, unmonitored.

Assumptions:

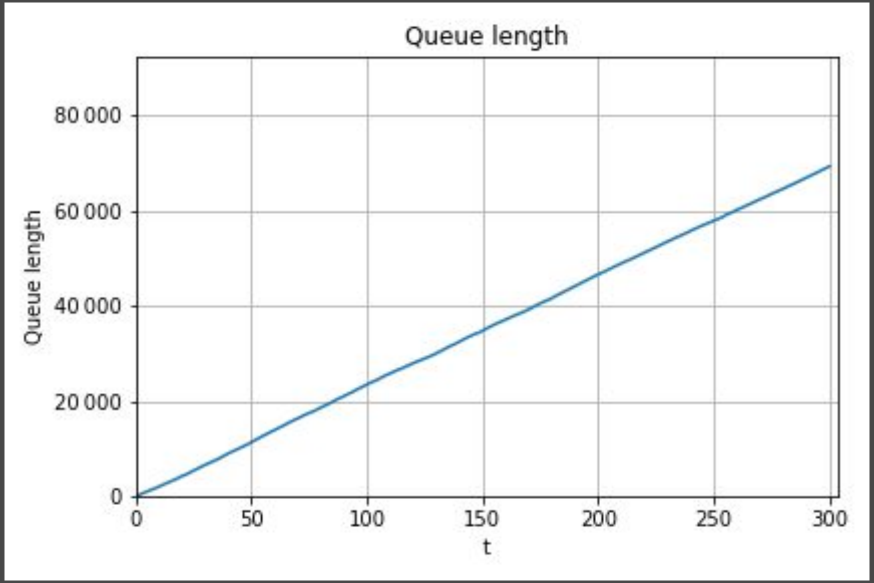
- DLs are ideal channels;
- ACs are independent;
- Fixed packet size of 100 byte;
- 3 different RNGs;
- Infinite queues.

Two modes of operation:

- **Monitored**: the Data Link with the highest capacity is selected whenever a new message needs to be sent
- **Unmonitored:** At the start of a simulation a random Data Link will be selected

Stability Condition

The system is stable for values of kMean > 0.009. The value chosen for the simulation is kMean = 0.05, to avoid any limit condition possible.



queue length with kMean = 0.003

To validate the model, the following test were performed:

- Memory analysis
- Degeneracy test
- Packet loss test
- Variation of the number of ACs
- Continuity test
- Little's law test

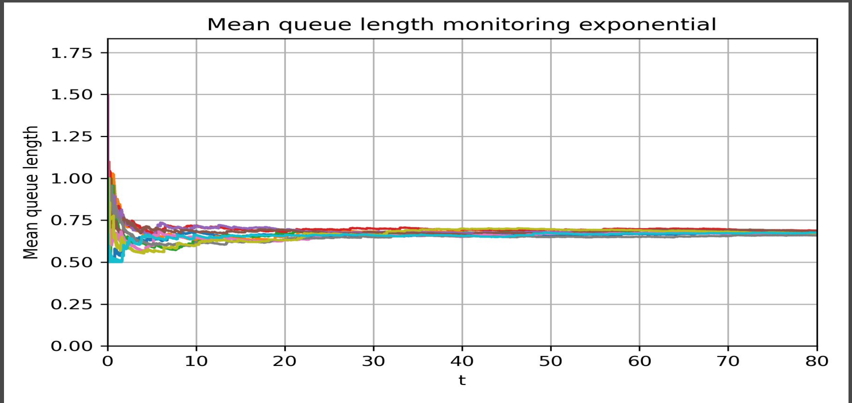
kMean	Service Time	Mean Response Time	Mean Queue Length
0.003	0.008407	31.294345	~∞
0.009	0.008567	0.387522	43.141522
0.02	0.008606	0.021300	1.301352
0.05	0.008626	0.018495	0.729679

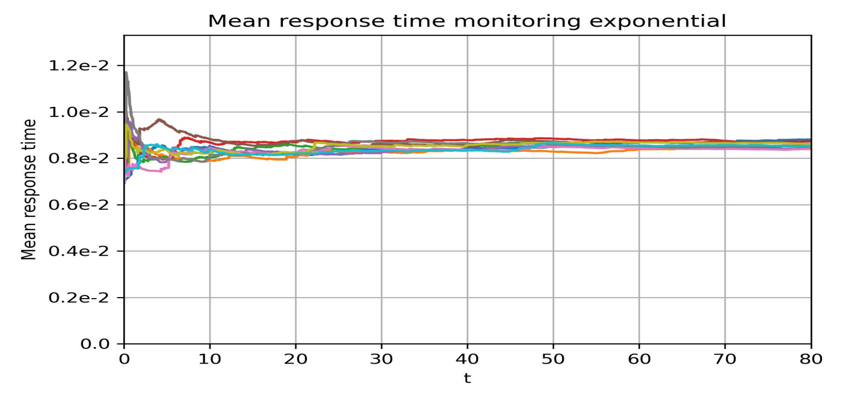
Warmup time and continuity test

In order to define the warm-up period, two main factors need to be considered: **queue length** and **response time**.

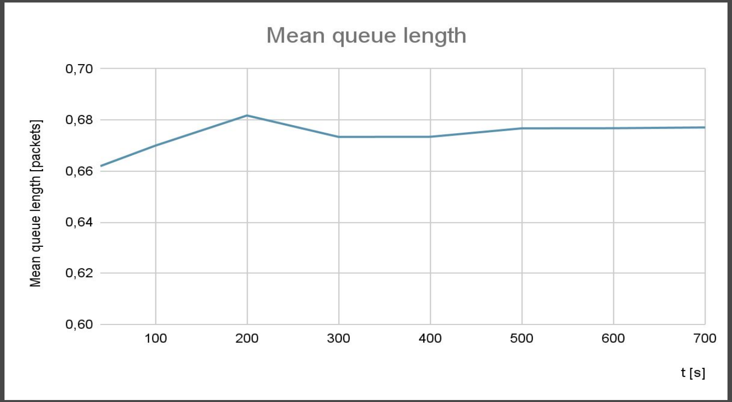
The plots were obtained from simulations which consisted of 10 repetitions each. This number of repetitions is a good enough indicator of warm-up behavior.We can see that after 10s the behavior is stable, so we can set our warm-up time like so.

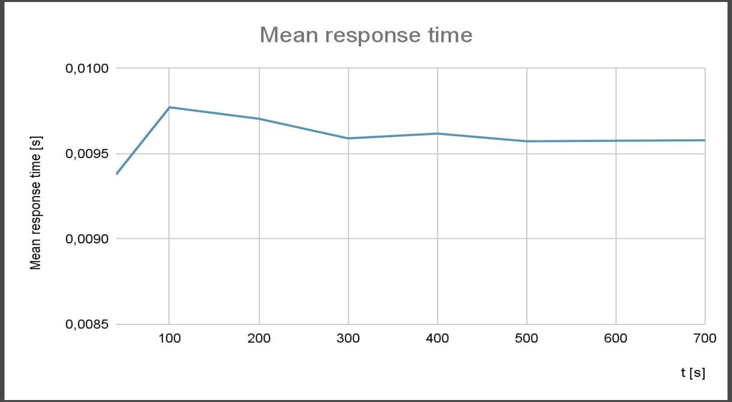
After having chosen the kMean, a number of simulations were performed to prove that, after reaching stability, no critical conditions or unexpected behaviors are encountered.





Simulation Length Analysis



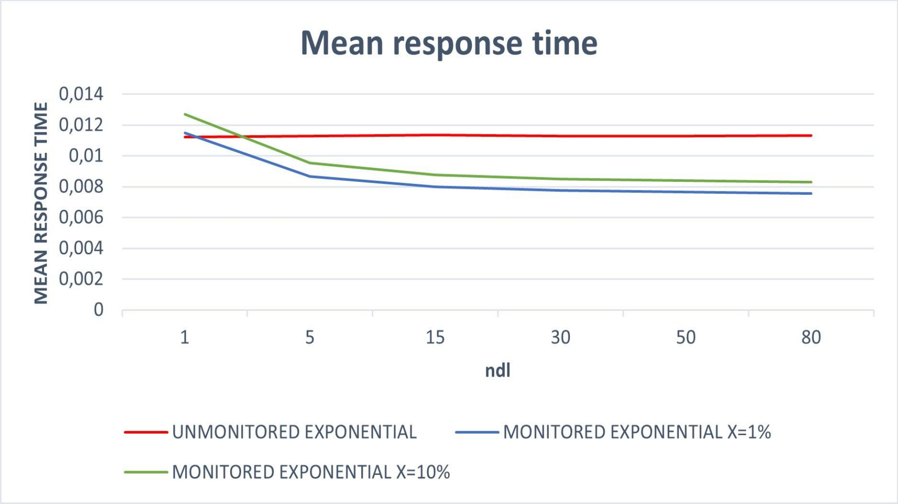


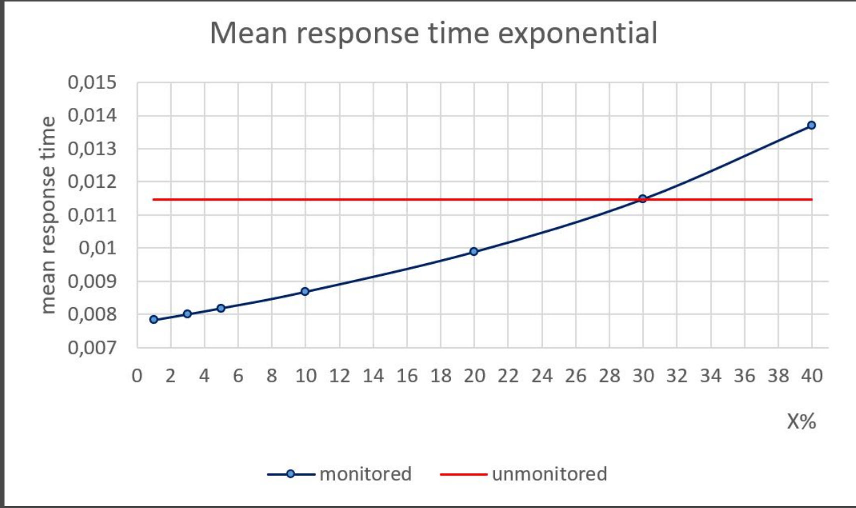
The behavior of the system was tested with different simulation lengths. The mean value and the standard deviation of *queue length* and *response time* vary when the simulation length gets increased, up to ~300s. The amount of samples collected in this interval of time seems to be enough to represent the system variability.

In the graphs are reported the *queue length* and *response time* behaviors related to the duration of the simulation in monitored mode, with exponential time generation

Analysis varying X% and nDL

To achieve better performance in monitored mode, compared to the unmonitored one, given that nDL=15, a malus lower than the 30% of the *DL* capacity is needed. Otherwise the system wastes more capacity monitoring than the one it gains by selecting the optimal *DL*. With lower values of *nDL*, for instance with values lower than 3, there’s no gain at all in using a monitored system, even with a malus of only 10% of the transmission capacity.

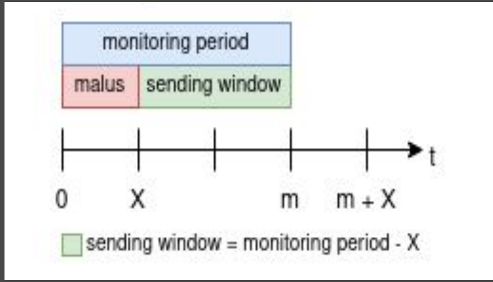


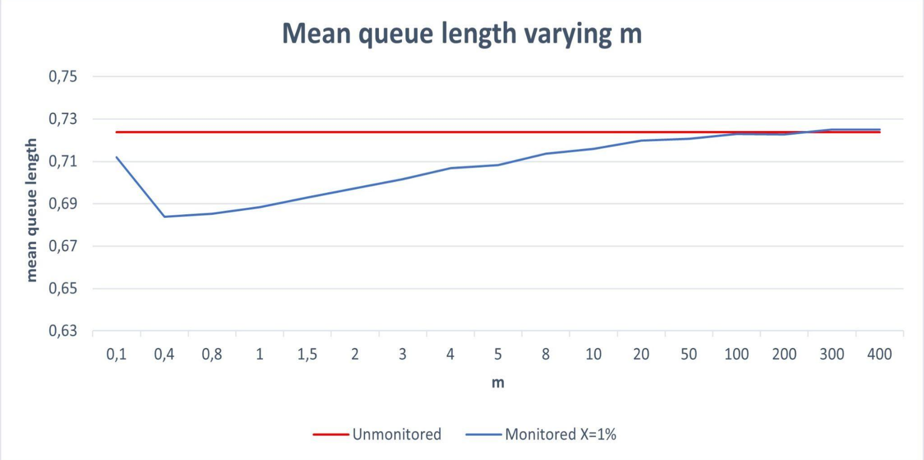


Different implementation

Another possible interpretation of the *malus* and of the overall monitored mode is discussed. In this version, there isn't a constant *X%* malus on the capacity, but instead an *X* seconds monitoring time (malus) every *m* seconds, where *m* is the monitoring period of the system.

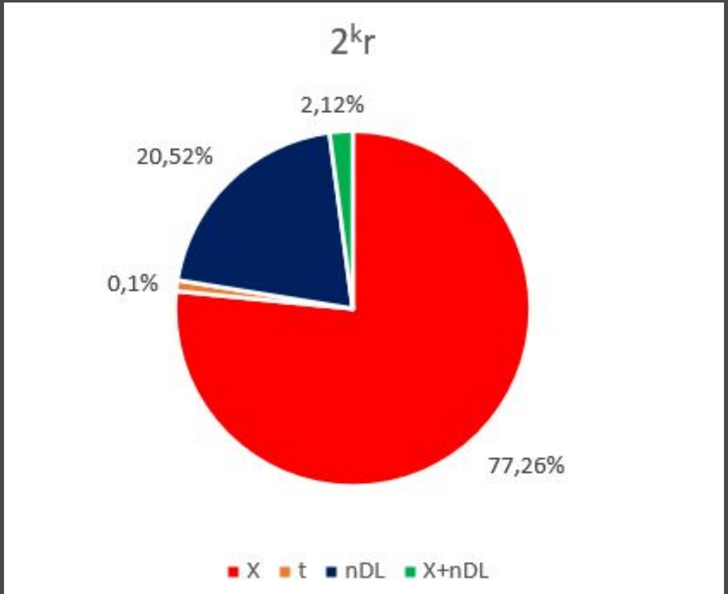
The system monitors the capacity of all the *DLs* every *m* seconds, and the monitoring takes *X* seconds to complete, where X = k + nDL*c ;





Factorial analysis

As expected, and as can be seen in the graph, t mean is negligible, while X and nDL are the main parameters influencing the results. Moreover, X is clearly the most important one, as well as the easiest to work on.



In a real implementation of such a system, *nDL* won't probably vary that much.

As such, the best way to achieve better performance would be trying to reduce the capacity occupancy of the monitoring stream of data as much as possible, either by reducing the size of packets as much as possible or by finding a less capacity consuming monitoring algorithm.

Conclusion

We can conclude that the choice between the two implementations is mostly influenced by the value of *nDL* and *X%*.

To achieve better performance in monitored mode, compared to the unmonitored one, given that nDL = 15, a malus lower than the 30% of the *DL* capacity is needed. Otherwise the system wastes more capacity monitoring than the one it gains by selecting the optimal *DL*.

With lower values of *nDL*, there’s no gain at all in using a monitored system, even with a malus of only 10% of the transmission capacity.

The simulated system was built using the infinite queue abstraction, but, with a malus below 30%, the queue stabilizes around a size of 4 packets.