BUFFER DIMENSIONING IN THE AFDX CONTEXT

Benammar Nassima, Henri Bauer, Frédéric Ridouard, Pascal Richard

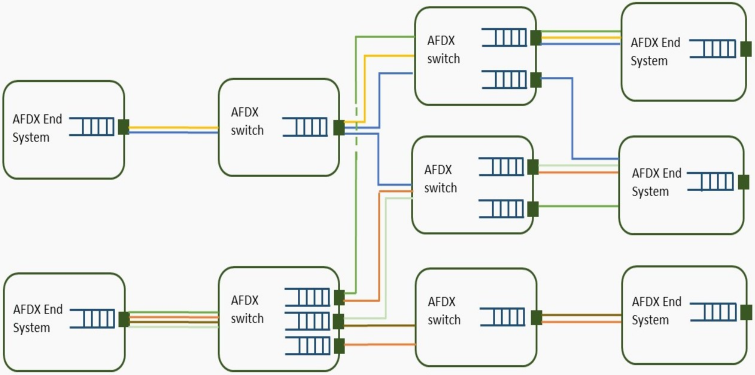
July 5, 2016

LIAS/ISAE-ENSMA - Université de Poitiers

Table of Contents

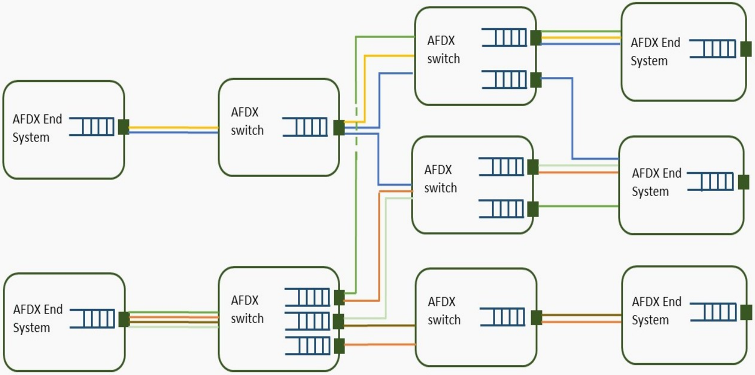
MOTIVATIONS

Motivations



- Asynchronous components → Competing frames in each buffer.

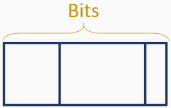
Motivations



- Asynchronous components → Competing frames in each buffer.
- Buffer dimensioning for certification reasons

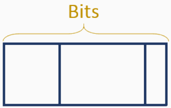
Buffer Design

- In terms of bits: dynamic memory allocation.

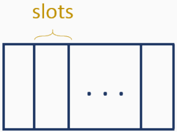


Buffer Design

- In terms of bits: dynamic memory allocation.

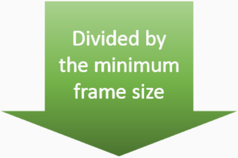


- In terms of number of frames: fixed size buffer slots (static design).



Naive method

Buffer occupancy in terms of bits.

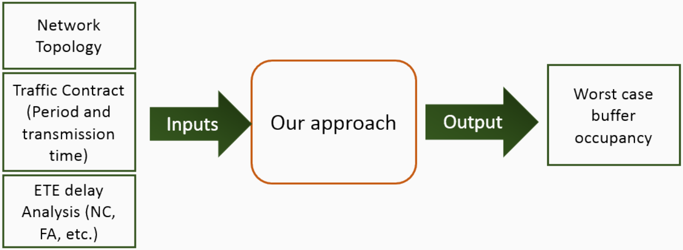


Buffer occupancy in terms of number of frames.

About Buffer Dimensioning

- Buffer size requirements derived from an ETE delay Method (Network Calculus (NC) [Boudec and Thiran, 2001]).
- Buffer occupancy in terms of number of competing frames using the Trajectory Approach (ETE Delay Analysis) with fixed frame sizes [Coelho et al., 2015].

Buffer dimentioning inputs



Note: Forward ETE Delay Analysis (FA) [Kemayo et al., 2014].

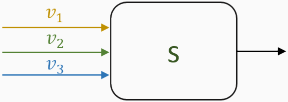
PROBLEMATIC

Problematic

The maximum number of frames is not necessarily obtained at time when the backlog is maximized:

Problematic

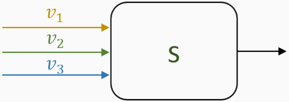
The maximum number of frames is not necessarily obtained at time when the backlog is maximized:



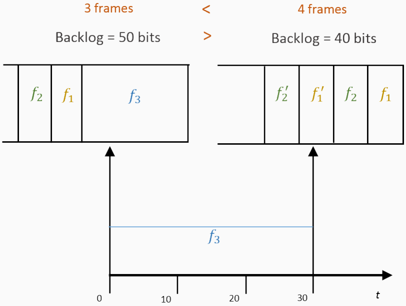
	$C_i(\mu s)$	$T_i(\mu s)$
v_1, v_2	10	30
v_3	30	100

Problematic

The maximum number of frames is not necessarily obtained at time when the backlog is maximized:



	$C_i(\mu s)$	$T_i(\mu s)$
v_1, v_2	10	30
v_3	30	100



Note: the servicing rate is 1 bit/μs.

BUFFER DIMENSIONING

Using the **FIFO** policy is difficult to maximize the number of pending frames :

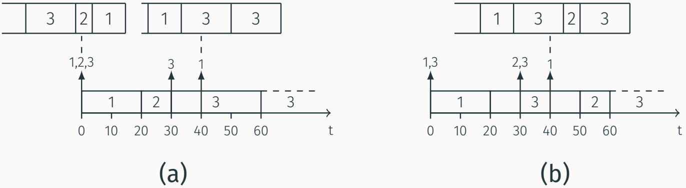
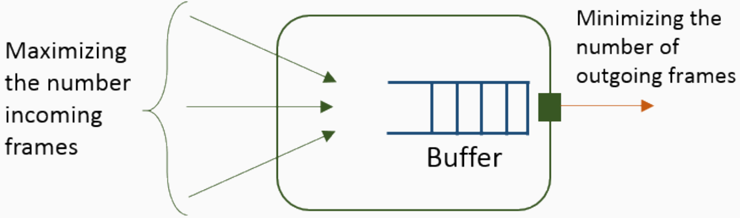


Figure: Arrival scenarios considering FIFO buffer.

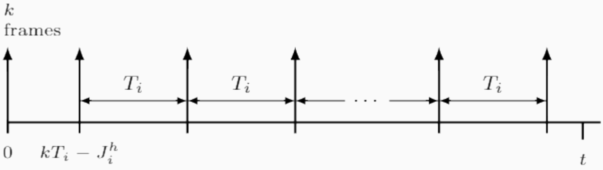
Principle



Maximizing the number incoming frames
Buffer
Minimizing the number of outgoing frames

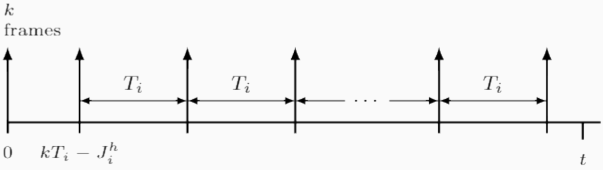
Incoming Frames

For every flow v_i crossing a node h, the incoming frames follow the scenario bellow:



Incoming Frames

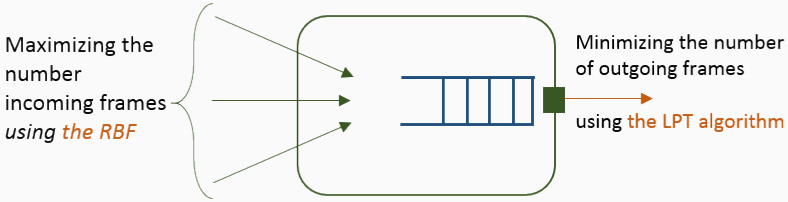
For every flow v_i crossing a node h, the incoming frames follow the scenario bellow:



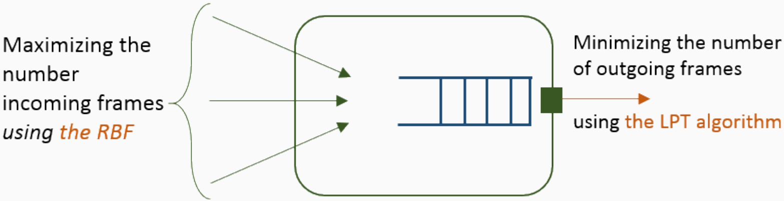
- $RBF_i^h(t) = \left(1 + \left\lfloor \frac{t+J_i^h}{T_i} \right\rfloor\right) C_i$, $RBF_i^h(0) = \underbrace{\left(1 + \left\lfloor \frac{J_i^h}{T_i} \right\rfloor\right)}_{\text{k frames}} C_i$;
- $(k-1)T_i \leq J_i^h < kT_i$;
- After that, all the frames arrive periodically.

The jitter J_i^h is obtained using an ETE delay analysis.

Outgoing Frames



Outgoing Frames



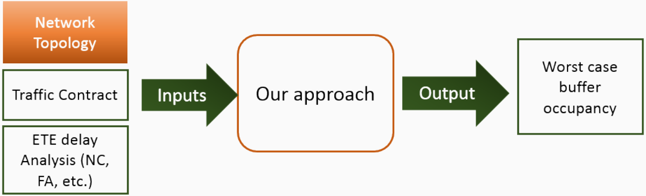
The Longest Processing Time algorithm [Graham, 1969] is optimal to minimize the number of the outgoing frames (proof: interchanged argument).

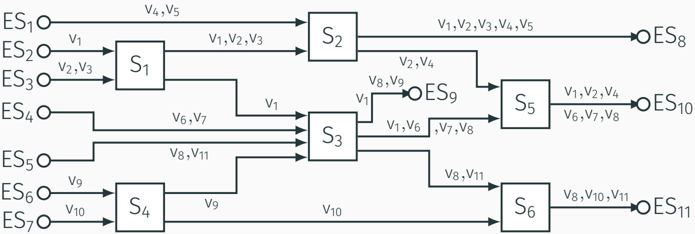
The number of frames present simultaneously at each time equals the Vertical Distance between two curves:

- Cumulative arrival curve following the scenario of incoming frames (RBF).
- Service curve following the algorithm LPT.

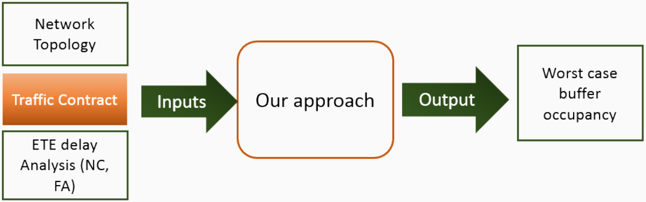
EXPERIMENTATION

TOPOLOGY


Network Topology
Traffic Contract
ETE delay Analysis (NC, FA, etc.)
Inputs
Our approach
Output
Worst case buffer occupancy


ES_1
ES_2
ES_3
ES_4
ES_5
ES_6
ES_7
ES_8
ES_9
ES_{10}
ES_{11}
S_1
S_2
S_3
S_4
S_5
S_6
v_4, v_5
v_1
v_2, v_3
v_1, v_2, v_3
v_1, v_2, v_3, v_4, v_5
v_2, v_4
v_8, v_9
v_1
v_1
v_6, v_7
v_1, v_6
$, v_7, v_8$
v_1, v_2, v_4
v_6, v_7, v_8
v_8, v_{11}
v_8, v_{11}
v_9
v_9
v_{10}
v_{10}
v_8, v_{10}, v_{11}

Traffic Contract



	$v_1, \ldots, v_5$	v_6	v_7	v_8	v_9	v_{10}	v_{11}
C_i	10	38	12	22	64	22	22
T_i	60	320	150	80	126	48	320

ETE Delay Analysis

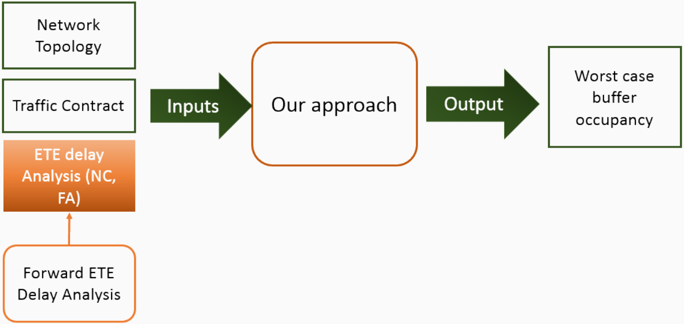


Illustration: maximum number of pending frames in the output buffer of port 1 from switch 3

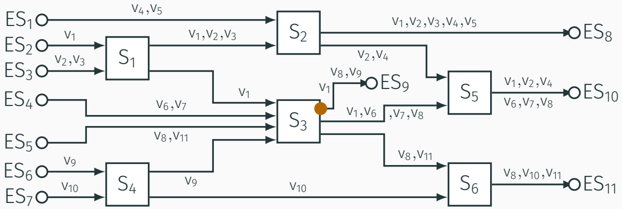
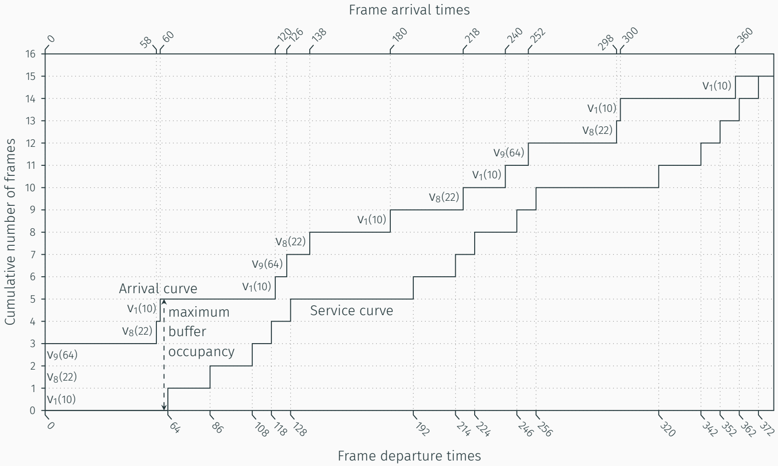


Illustration: maximum number of pending frames in the node S_{31}

Motivations
Problematic
Buffer Dimensioning
Experimentation
Case Study
Results
Conclusion



Results

	Naive approach		Our approach
Node	Backlog (bits)	Backlog (frames)	Backlog (frames)
ES_1	2000	2	2
ES_2	1000	1	1
ES_3	2000	2	2
ES_4	5000	5	2
ES_5	4400	2	2
ES_6	6400	1	1
ES_7	2200	1	1
S_{11}	3000	3	3
S_{12}	1000	1	1
S_{21}	5000	5	5
S_{22}	2000	2	2
S_{31}	9600	10	5
S_{32}	8200	9	4
S_{33}	4400	2	2
S_{41}	6400	1	1
S_{42}	2200	1	1
S_{51}	13400	14	13
S_{61}	6600	3	3

Table: Per bits and per frames approaches for buffer dimensioning in the configuration from using the FA method.

Results

	Naive approach		Our approach
Node	Backlog (bits)	Backlog (frames)	Backlog (frames)
ES_1	2000	2	2
ES_2	1000	1	1
ES_3	2000	2	2
ES_4	5000	5	2
ES_5	4400	2	2
ES_6	6400	1	1
ES_7	2200	1	1
S_{11}	3000	3	3
S_{12}	1000	1	1
S_{21}	5000	5	5
S_{22}	2000	2	2
S_{31}	9600	10	5
S_{32}	8200	9	4
S_{33}	4400	2	2
S_{41}	6400	1	1
S_{42}	2200	1	1
S_{51}	13400	14	13
S_{61}	6600	3	3

Table: Comparison of the two approaches for determining worst-case buffer occupancy in terms of frames.

Results

	Naive approach		Our approach
Node	Backlog (bits)	Backlog (frames)	Backlog (frames)
ES_1	2000	2	2
ES_2	1000	1	1
ES_3	2000	2	2
ES_4	5000	5	2
ES_5	4400	2	2
ES_6	6400	1	1
ES_7	2200	1	1
S_{11}	3000	3	3
S_{12}	1000	1	1
S_{21}	5000	5	5
S_{22}	2000	2	2
S_{31}	9600	10	5
S_{32}	8200	9	4
S_{33}	4400	2	2
S_{41}	6400	1	1
S_{42}	2200	1	1
S_{51}	13400	14	13
S_{61}	6600	3	3

Table: Comparison of the two approaches for determining worst-case buffer occupancy in terms of frames.

Results

	Naive approach		Our approach
Node	Backlog (bits)	Backlog (frames)	Backlog (frames)
ES_1	2000	2	2
ES_2	1000	1	1
ES_3	2000	2	2
ES_4	5000	5	2
ES_5	4400	2	2
ES_6	6400	1	1
ES_7	2200	1	1
S_{11}	3000	3	3
S_{12}	1000	1	1
S_{21}	5000	5	5
S_{22}	2000	2	2
S_{31}	9600	10	5
S_{32}	8200	9	4
S_{33}	4400	2	2
S_{41}	6400	1	1
S_{42}	2200	1	1
S_{51}	13400	14	13
S_{61}	6600	3	3

Table: Comparison of the two approaches for determining worst-case buffer occupancy in terms of frames.

CONCLUSION

Summary

- **Buffer dimensioning for AFDX switch buffers in terms of frames, given different frame sizes.**
- Our approach requires: a network topology, traffic contracts and an ETE delay Analysis.
- Using FIFO, it is difficult to maximize the number of frames $\rightarrow$ analyzing the incoming frames and the outgoing frames separately using resp. the RBF and the LPT algorithm.
- Experimentation $\rightarrow$ Tighter results besides the Naive computation.

Thank You. Questions?

References I

Boudec, J.-Y. L. and Thiran, P. (2001).
Network calculus: A Theory of Deterministic Queuing Systems for the Internet.
Springer Verlag.

Coelho, R., Fohler, G., and Scharbarg, J.-L. (2015).
Dimensioning buffers for afdx networks with multiple priorities virtual links.
In Digital Avionics Systems Conference (DASC), 2015 IEEE/AIAA 34th. IEEE.

Graham, R. L. (1969).
Bounds on multiprocessing timing anomalies.
SIAM Journal on Applied Mathematics, 17(2):416–429.

References II

Kemayo, G., Ridouard, F., Bauer, H., and Richard, P. (2014).
A forward end-to-end delays analysis for packet switched networks.
In 22nd International Conference on Real-Time Networks and Systems. RTNS 2014.

Annex (1)

The Request Bound Function computes the amount of backlog generated by flow v_i crossing a node S:

$$RBF_i^s(t) = \left(1 + \left\lfloor \frac{t + J_i^s}{T_i} \right\rfloor\right) C_i \quad (1)$$

For a non-preemptive sporadic flow v_i, the maximum number of frames generated during $[t_0, t_1]$ (with $t_1 - t_0 = t$) is: $\left(1 + \left\lfloor \frac{t_1 - t_0}{T_i} \right\rfloor\right)$. However, if $[t_0, t_1]$ is the time interval to consider in s, the corresponding interval in the source node of each flow v_i expands to: $[t_0 - Smax_i^s, t_1 - Smin_i^s]$, where $Smax_i^s$ and $Smin_i^s$ are respectively the longest and the shortest times needed for a frame from v_i to reach s from its source node. The jitter is defined as $J_i^s = Smax_i^s - Smin_i^s$.

Annex (2)

The worst-case traversal time of a flow from the source node to the destination node is split into two parts:

- **Constant part:** propagation delay.
- **Variable part:** waiting time in the buffer due to interfering frames. The worst-case backlog computation in FA is based on the RBF of each flow, accounting the periodicity, the maximum frame size and the maximum jitter [Kemayo et al., 2014].

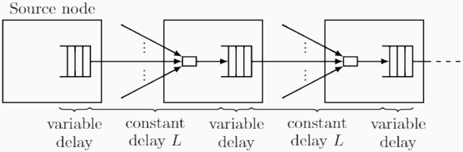


Figure: Element of ETE delay.

Annex (3)

The iterative computation of the traversal time of a flow v_i to reach a node $h+1$, denoted $Smax_i^{h+1}$, depend on the worst-case traversal time to reach the previous node h, denoted $Bklg_i^h$, the waiting time in node h to be processed and the propagation delay L.

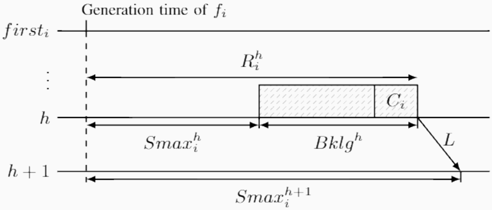


Figure: Iterative computation of the delay.

Note: R_i^h is the worst-case traversal delay for a frame of a flow v_i from its ingress node to a given node h.