Modelling avionics communicating systems: successes, failures, challenges

Marc Boyer

ONERA – The French Aerospace Lab

Dagstuhl Seminar on Network Calculus
March 8-11, 2015

ONERA
THE FRENCH AEROSPACE LAB

r e t o u r s u r i n n o v a t i o n

Disclaimer

“some perspectives on the application modelling side, what is required from NC, what is still missing, what are success and failure stories”

Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

Challenges

- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

Outline

The core technology: AFDX

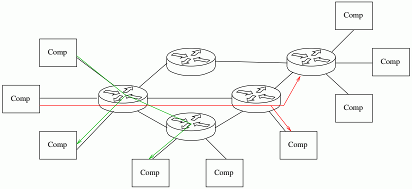
Success: modelling AFDX in network calculus

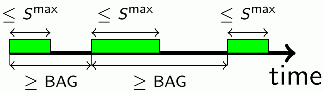
Failure: modelling spacewire/whormhole

Challenges
- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

AFDX: Avionic Full DupleX

- Standard ARINC 664 P7
- Ethernet tailored for avionic needs
  - Flows: Virtual links
    - static routing
    - static priority
    - flow control: minimal inter-arrival distance (BAG) , maximal packet size (S^{max})
  - Network: Full duplex, SP/FIFO





ONERA THE FRENCH AEROSPACE LAB

Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

Challenges
- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

ONERA
THE FRENCH AEROSPACE LAB

Modelling AFDX in network calculus

Modelling the arrival curves:

- fluid token bucket
- stair-case function

Modelling server impact:

- Static Priority/FIFO: residual service
- Grouping/Shaping: maximal service / shaper

Handling arrival curves/service curves:

- sum, minus, convolution, deconvolution....

Topology analyse:

- kind of mix between SFA/TFA handling maximal service

AFDX accuracy

Realistic configuration

- $\approx$ 6-8 switches
- $\approx 10^4$ virtual links flows

Impact of modelling:

1. start from token-buckets curves, local FIFO analyse
2. add maximal service/shaping
   - switch to concave/convex piecewise linear functions
   - gain: $\approx 40\%$
3. switch to stair-case functions: gain of 6%

Performance (RTaW-PEGASE)

- computing time: $\approx 1 - 10s$
- accuracy: $\approx 20\%$

Future of AFDX modelling

Exact FIFO delays:

- ☺ exact delay
- ☹ computation time
- ☹ implementation complexity

Future of AFDX modelling

Exact FIFO delays:

- ☺ exact delay
- ☹ computation time
- ☹ implementation complexity

Modelling end-system behaviour:

- ☺ gain of $\approx 20\%$
- ☹ implementation complexity
- ☹ implementation dependant

Future of AFDX modelling

Exact FIFO delays:

- ☺ exact delay
- ☹ computation time
- ☹ implementation complexity

Modelling end-system behaviour:

- ☺ gain of $\approx 20\%$
- ☹ implementation complexity
- ☹ implementation dependant

No *current* industrial interest: implementation cost vs accuracy gain

Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

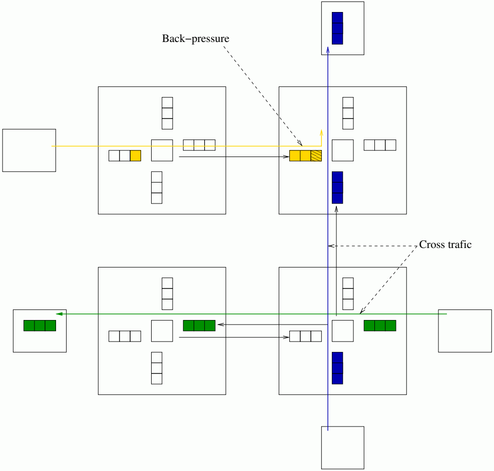
Failure: modelling spacewire/whormhole

Challenges
- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

Spacewire I

- Spacewire: a spatial ESA standard (ECSS-E-ST-50-12C, 2003)
- Topology: switches, full duplex links
- Throughput: 2Mb/s - 200Mb/s
- Flow control: Wormhole
  - small buffer
  - blocking/back-pressure

Spacewire II



Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

Challenges
- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

Challenges

- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

Always more scheduling policies

Next embedded networks?

- GPS, Deficit Round Robin
- AVB, TSN (AVB 2.0)
- TTEthernet
- TDMA
- ...

Hierarchical scheduling: (SP/DRR/FIFO, SP/AVB)

- generic β service
- residual service

Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

Challenges

- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

Packet/Event model I

Industrial case study: gateway

- connecting two nets
- packet reception releases a forwarding task
- CPU shared between forwarding tasks and computing tasks
- task execution time may depend on packet size, or not

Cumulative curves:

- amount of data/bits (network/real-time calculus), A
- number of packets/events (event stream) E
- packet curve: $P(A) = E$

On going work:

- three bounding curves $(A \leq A * \alpha, E \leq E * \eta, P \leq P * \pi)$
- a theory to bring them all and in the same model bind them

Packet/Event model II

Expected benefits:

- better links with scheduling analyses
- heterogeneous networks
- heterogeneous analyses (state-less and state-based)
- application to application delay

Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

Challenges

- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

Network on chip

Hardware evolution

- From 1 to 4 to 64 cores
- From bus to network on chip (NoC)

$\Rightarrow$ can network calculus handle it?

Network on chip

Hardware evolution

- From 1 to 4 to 64 cores
- From bus to network on chip (NoC)

$\Rightarrow$ can network calculus handle it?

Obstacles founds:

- get the NoC model
- back pressure behaviour (wormhole)

Outline

The core technology: AFDX

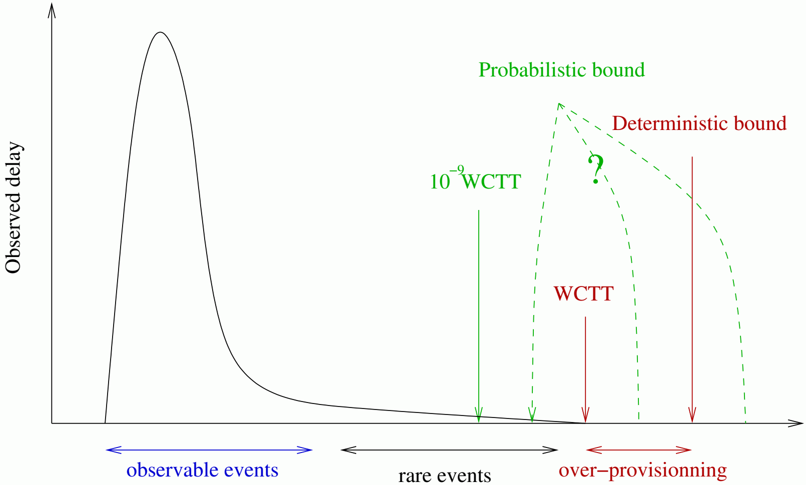
Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

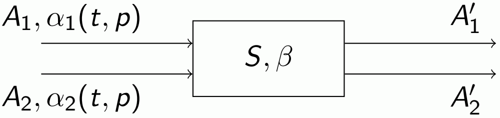
Challenges

- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

Probabilistic bounds for critical systems I



Probabilistic bounds for critical systems II



Naive questions:

- how to get input probabilities?
- what if arrivals are not independent?
- are 10^{-9} stoch. bounds lesser than deterministic ones

Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

Challenges

- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

New notion of delay: cumulative delay

- critical network is often in a control/command loop
- performances of control/command law are based on delay upper bound
- a new contract Δ, "Delay density" can be defined [1],
  Let d_i be the delay of i-th message

$$D(n) = \sum_{i=1}^{n} d_i$$

$$\forall p, q \in \mathbb{N} : D(p+q) - D(p) \leq \Delta(q)$$

- can network calculus compute such bound?

[1] A Delay Density Model for Networked Control Systems, *Tobias Bund and Frank Slomka*, Proc. of the 21st Int. Conf. on Real-Time Networks and Systems (RTNS '13),

Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

Challenges

- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

Design help

- network calculus computes bounds from configuration
- can we compute configuration from bounds?
    - routing
    - priority allocation
    - minimal topology
    - task/CPU allocation

Outline

The core technology: AFDX

Success: modelling AFDX in network calculus

Failure: modelling spacewire/whormhole

Challenges

- Always more scheduling policies
- Packet/Event model
- Network on chip
- Probabilistic bounds for critical systems
- New notion of delay
- Design help
- Formal correctness proofs

Formal correctness proofs

Can you trust the results?

- is the theory correct?
- is the implementation bug-free?

Approach

- model NC in formal proof assistant (Isabelle/HOL, Coq)
- generate a proof at each computation

Conclusion

Successes	1
Failures	1
Challenges	7
Questions	?