Frame latency evaluation: when simulation and analysis alone are not enough

Nicolas Navet, INRIA / RTaW
Aurélien Monot, LORIA / PSA
Jörn Migge, RTaW

WFCS 2010 – Industry Day
Nancy, 19/05/2010
Outlook

1. **Context:** early design phases where only simulation and analysis are available

2. **Goal:** see how simulation and analysis compare and point out their pitfalls

3. **Method:** insight from experiments on Controller Area Network
Why timing verification is required

- Verify that performance requirements are met: deadlines, jitters, throughput
- Select the hardware / software components: optimize costs
- Meet some certification level: e.g., avionics, railway systems, power plants, etc

Timing models: trade-off to be found between accuracy / complexity / computing time
RTaW mission: help designers build truly safe and optimized systems

- **Activities**: Model-Based Design, dependability, formal and temporal verification
- **Communications systems**: CAN, AFDX, FlexRay, SpaceWire, industrial Ethernet, TTP, etc ...
- **Verification techniques**: schedulability analysis, network-calculus, model-checking and simulation
- **Domains**: aerospace, automotive and industry at large

In our experience, 2 cases for timing verification:
- ✓ Certification is mandatory (e.g., DO178B - DAL A): well accepted
- ✓ No certification: various practices / levels of acceptance
Type A: no timing validation whatsoever (early in the V-cycle)

Practice: Carry-over of existing (proven in use) systems with domain-specific rules:

“The load on an automotive CAN network must not be higher than 30%”

“A frame pending for transmission for more than 30ms is cancelled out”

etc…

Sub-optimal design: e.g., does one really need 5 (or more) distinct CAN buses in a car?!

Potentially unsafe design with problems that are hard to reproduce and are costly to repair later...
Type B: simulation is enough, worst-case never occurs anyway!

Practice: software simulations, then simulations with HiL (Hardware in the Loop) as the ECUs become available ...

- Hardware resources (too?) well optimized
- Unsafe results because the worst-case sometimes occurs (and may even last for a long time, see preliminary results later in the presentation)

A question that remain mainly open in timing verification:
“How often does the worst-case actually occur?“
First, get some insight with experimentations ...
Type C: analysis says the system is safe, so we are covered ...

Practice: use some black-box software that implements worst-case timing analysis and concludes about the feasibility of the system.

- Sub-optimal design sometimes because overestimations / pessimistic assumptions add up
- Potentially unsafe design :
  - software are error-prone,
  - not everything is accurately modeled
  - analytic models – especially unpublished complex ones – can be wrong
Experimental setup
Requirements on temporal verification:

- handle 150+ frames
- ≠ waiting queue policy at the microcontroller level
- limited number of transmission buffers
- handle frame offsets
Principle: desynchronize transmissions to avoid load peaks

Algorithms to decide offsets are based on arithmetical properties of the periods and size of the frame.
## Network configuration

<table>
<thead>
<tr>
<th>Network</th>
<th>Controller Area Network 125 kbit/s</th>
</tr>
</thead>
</table>
| **Set of messages**                        | Automotive body network generated with NETCARBENCH [8]  
http://www.netcarbench.org |
| **# ECUs**                                 | 15 |
| **# frames**                               | 145 |
| **Workload**                               | 50.5% |
| **Periods**                                | [50,2000ms] with distributions from an existing car |
| **Frame offsets**                          | Optimized with DOA algorithm [3] |
| **Inter-ECU offsets**                      | All offsets are possible (clock drifts, ECU reboots, ECU boot sequence depends on sleep mode, etc) |
| **ECU clock drifts**                       | 3 cases: no drift, ±1ppm, ±1000ppm |
RTaW software used in the study

RTaW-Sim: Fine-Grained Simulation of Controller Area Network with Fault-Injection Capabilities

NETCAR-Analyzer: Timing Analysis and Resource Usage Optimization for Controller Area Network (© Inria/Inpl)

WCRT is not overestimated!

RTaW-Sim freely available at http://www.realtimeatwork.com starting from June 2010
On why we should not trust **analytic models** for worst-case frame latency evaluation
Types of results achievable with worst-case analysis

Max buffer utilization

Worst-case inter-ECU offsets

Frame worst-case response times

- non-optimized solution
- optimized solution 1 and 2
- lower-bound

Frames by decreasing priority
Analytic models need to be fine-grained frame offsets overlooked here ...

Analysis Setup:
- Frame offsets: DOA algorithm [3]
- Worst-case results whatever clock drifts and inter-ECU offsets

To the best of our knowledge, there are no (usable) published results on this
Analytic model needs to be fine grained
Frame waiting queue is FIFO on ECU1
the OEM does not know or software cannot handle it ...

Analysis Setup:
- Frame offsets: DOA algorithm [3]
- Worst-case results whatever clock drifts and inter-ECU offsets
- FIFO waiting queue on ECU1

Many high-priority frames are delayed here because a single ECU (out of 15) has a FIFO queue ...
There is a gap between WCRT analytic models and reality IMHO

- Traffic is not always well characterized and/or well modeled
e.g. aperiodic traffic?! see [5] for some solution
- Implementation choices really matter
  and standards do not say everything, eg. Autosar drivers
- Analytic models are often much simplified abstraction of reality
  - optimistic (=unsafe): FIFO queue, hardware limitations such as
    non-abortable transmissions [4,7], etc
  - overly pessimistic: e.g. overlooking frame offsets, aperiodic traffic
    modeled as sporadic, etc
- Analytic models are prone to errors
  remember “CAN analysis refuted, revisited, etc” [6] ?!

Bottom line: do not blindly trust analytic models!
Systems should be conceived so as to be analyzable in temporal domain
On why we should not trust simulation models for worst-case frame latency evaluation
Are simulation results (max) close to worst-case response times? Well ...

Simulation Setup:
- Random inter-ECU offsets
- no ECU clock drift

≈ 40ms!
Are simulation results (max) close to worst-case response times? with clock drifts

**Simulation Setup:**
- Random inter-ECU offsets
- Slow and fast clock drifts
- Sim duration: vehicle lifetime

Whatever you do, you have little chance with simulation to find the worst-case!
Are simulation results (max) close to worst-case response times? with clock drifts

**Simulation Setup:**
- Same as previous slide

Increasing the clock drift rate is not enough ...
Knowing the analysis results – including here worst-case inter-ECU offsets for each frame - simulation becomes more useful
Simulation helps validate assumptions made, correctness and tightness of WCRT analysis.

**Simulation Setup:**
- worst-case inter-ECU offsets for frame 61 given by NETCAR-Analyzer
- no ECU clock drift

Difference comes here from the blocking factor that is not explicitly simulated.
How often does the worst-case occurs: very often on certain trajectories ...

Simulation Setup:
- worst-case inter-ECU offsets for frame 61 given by NETCAR-Analyzer
- no ECU clock drift

Average value is "close" to maximum on the worst-case trajectory!
Distribution of response times for frame 61 with and without clock drifts

Even with clock drift, unusually large response times occur during more than 30mn!
Conclusion: in the context of dependability constrained systems ...

- Simulation is not enough and analytic models are usually much simplified, often pessimistic and sometimes even wrong
- **Simulating the worst-case trajectory (and neighbours):**
  - helps to validate analytic models: latencies, buffer occupation, etc
  - tells us about how long we stay in the worst-case situation
- Our ongoing work: how often does the worst-case actually occur? do we really need to dimension for the worst-case for a given a SIL level?
- Application to CAN, AFDX and switched Ethernet in aerospace, power plant and automotive domains
References
References


Questions / feedback?

Please get in touch at:

nicolas.navet@realtimeatwork.com
aurelien.monot@mpsa.com
jorn.migge@realtimeatwork.com