Verification of automotive networks - what to expect (and not expect) from each technique

Nicolas NAVET – nicolas.navet@uni.lu
“Automotive Bus systems + Ethernet”
Stuttgart, Germany, December 9-11, 2013.
Early-stage timing verification of wired automotive buses – illustration on CAN

A historical perspective of verification techniques

Sets of message and verification techniques along the development cycle

Comparing early stage technique: schedulability analysis versus simulation
Verification techniques and their use along the development cycle

If the workload submitted is bounded and the resources are deterministic, then it is always possible to provide timing guarantees.
Schedulability analysis
“mathematic model of the worst-case possible situation”

\[ K_i^k(t) \overset{\text{def}}{=} \left[ \frac{J_i^k + \varphi_i^k(\phi)^i}{T_i^k} \right] + \left[ \frac{t - \varphi_i^k(\phi)^i}{T_i^k} \right] + 1 \]

max number of instances that can accumulate at critical instants
max number of instances arriving after critical instants

😊 Upper bounds on the perf. metrics → Safe (really?! – TBD)
😊 Analysis is known to be correct → Safe (really?! – TBD)
😊 Pessimistic → over-dimensioning
😊 Gap between models and real systems!
😊 Do not provide much information since a single trajectory is studied

Simulation
“program that reproduces the behavior of a system”

😢 Models close to real systems
😢 Fine grained information
😢 Upper bounds are out of reach! → Unsafe (really?! – TBD)
😢 Model correctness is unsure
Historical development of verification techniques – personal perspective

- **Technologies**: CAN, TTP/C, FlexRay, Gateways, Ethernet, CAN-FD, ...

- **1994**: « Smart » real-time monitoring tools & trace analysis

- **1997**: Probabilistic analysis (sub-system, eg: CAN)

- **2005**: « Worst-case » deterministic analysis (sub-system, eg CAN)

- **Today**: « correctness by construct » and optimal configuration

- **Mostly ahead of us!**

- **Simulation tools** (software, HIL, sub-system, system level)
Sets of messages and verification techniques along the development cycle

“Early stage”
- “Virtual” set of messages derived from existing ones
- Architecture design & technological choices
- Coarse-grained verification
- System will be able to grow? Add frames, ECU, clusters?
- Workload generator
- Simulation & analysis techniques

“Project”
- Set of messages as specified by the designer
- Configuration: offsets, priorities, frame packing, round, routing, etc
- Fine-grained verification .. but model-based
- Configuration optimization
- Simulation & analysis

“Real”
- Set of messages as seen in the car
- errors, aperiodic, ECU clock drifts,
- Specifications are met?
- Impact of non-conformance?!
- Monitoring tools
- Trace analysis
- Simulation & analysis with real traffic monitored

[Netcarbench & RTaW-Sim] [RTaW-Sim / RTaW-Pegase] [RTaW-TraceInspector]
Analyzing communication traces: are there departures from the specifications?

Check comm. stack implementation, periods, offsets, jitters, model for aperiodic traffic and transmission errors, clock drifts, etc..

Priority inversion here because frames are not queued in the order of priority.
2

Early-stage verification techniques: schedulability analysis versus simulation
Main performance metric: frame response time \( \approx \) communication latency

“Time from transmission request until frame received by consuming nodes”

Synthetic metrics at the bus level: eg. Max (response time / deadline)

Software delay
Waiting time in software queue
Arbitration delay
Transmission time
End-to-end response time verification has to handle for heterogeneous networks, task scheduling, gateways, etc.

But tasks and messages scheduling are often decoupled in the design ...
Frame response time distribution

Q1: pessimism of schedulability analysis ?!

Q2: distance between simulation max. and WCRT ?!

Automotive Bus systems + Ethernet

12/11/2013 - 12
Q1: Pessimism of CAN schedulability analysis?
Q2: distance with simulation?

Case 1: ideal communication stacks + no gateway →
the computed upper-bound can occur (and be re-simulated)

Re-simulating the worst-case scenario
Q1: Pessimism of CAN schedulability analysis?
Q2: distance with simulation?

Case 2: perfect communication stacks + gateway →
the computed upper-bounds do not occur for forwarded frames
in the general case

WCRT is pessimistic for forwarded frames

End-to-end
Response times

Frames by decreasing priority

Simulation max

WCRT

[RTaW-sim screenshot]
Beware of verification models!

“Schedulability analysis ensures safety!”
Our view: it might not be so...

1. Analytic models are pessimistic (except in the “ideal” case)
2. Analytic models are unrealistic (except in the “ideal” case)
3. Analytical models and their implementation can be flawed

“Simulation cannot provide firm guarantees”
Our view: it might not be so...

4. It is possible to verify correctness of simulation models
5. User-chosen guarantees can be enforced with proper methodology, e.g. with quantiles
Assumptions made by analytical models may not always be realistic
Possible departures from assumptions made:
communication stack – illustration on CAN

1. Non-prioritized waiting queues [5,6]
2. Frame queuing not done in priority order by communication task
3. Non abortable transmission requests [9]
4. Not enough transmission buffers [8,10]
5. Delays in refilling the buffers [11]
6. Delay data production / transmission request

…
Possible departures from assumptions made:
frame transmission patterns

7 code upload or segmented messages

8 Autosar-like mixed transmission models

9 Diagnostics requests

10 Transmission errors (probabilistic model ?! [1])

11 Aperiodic traffic (probabilistic model ?! [2])

12 Gatewayed traffic

...
If the analytical model does not capture accurately all the characteristics of the system, then the results will be wrong ... in an unpredictable manner.

Afaik, on CAN there is no schedulability analysis published yet for both frame offsets and FIFO queues ...

Many high-priority frames are delayed here because a single ECU (out of 15) has a FIFO waiting queue ... could propagate through gateways.

Frames by decreasing priority
Good news: many works try to bridge the gap between analytic models and real systems [Ref.1 to 12]

- However – not everything is covered, no integrated framework (first step in [6])
- And - many existing analyses are conservative (= inaccurate), thus hardly usable for highly-loaded systems.
- Alas - comprehensive and exact analysis would be overly complex (e.g. as in [9]) and intractable!

Personal view : both accurate and comprehensive analyses are out of reach ... if you need analysis, you have to conceive the systems accordingly
Why should we trust verification models?
Models and software can be flawed ...

✓ Schedulability analyses are complex and error prone. remember “CAN analysis refuted, revisited, etc” [14]?! → peer-review of the WCRT analyses and no black-box software

✓ Schedulability analysis implementations are error prone: analyses complexity, floating-point arithmetic!, how to check correctness?, not many end-users, cost-pressure, etc …

✓ Easier to validate a simulator? Yes …

  o Cross-validation by re-simulating worst-case situation from schedulability analysis (when possible)
  
  o Cross-validation by comparison with real communication traces: e.g., comparing inter-arrival times distribution
  
  o Checking a set of correctness properties on simulation traces
Simulation can provide guarantees with proper methodology
Using quantiles means accepting a **controlled risk**

**Quantile** $Q_n$: $P[\text{response time} > Q_n] < 10^{-n}$

- **Upper-bound with schedulability analysis**
- **Simulation max.**

- Probability $< 10^{-5}$
- **one frame every 100 000**

- ✓ No extrapolation here, won't help to say anything about what is too rare to be in simulation traces
1) How often performance objectives can be violated regarding frame criticality?

<table>
<thead>
<tr>
<th>Quantile</th>
<th>One frame every …</th>
<th>Mean time to failure Frame period = 10ms</th>
<th>Mean time to failure Frame period = 500ms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q3</td>
<td>1000</td>
<td>10 s</td>
<td>8mn 20s</td>
</tr>
<tr>
<td>Q4</td>
<td>10 000</td>
<td>1mn 40s</td>
<td>≈ 1h 23mn</td>
</tr>
<tr>
<td>Q5</td>
<td>100 000</td>
<td>≈ 17mn</td>
<td>≈ 13h 53mn</td>
</tr>
<tr>
<td>Q6</td>
<td>1000 000</td>
<td>≈ 2h 46mn</td>
<td>≈ 5d 19h</td>
</tr>
</tbody>
</table>

Warning: successive failures in some cases might be temporally correlated, this must be ruled out …
2) Determine the minimum simulation length

- time needed for quantile convergence
- reasonable # of values: a few tens ...

Tool support can help here:
e.g. numbers in gray should not be trusted

Reasonable values for Q5 and Q6 (with periods <500ms) are obtained in a few hours of simulation (with a high-speed simulation engine) – e.g. 2 hours for a typical automotive setup
3
Concluding remarks
Simulation vs analysis

There might be a gap between assumptions made for analytic models and the real system

- pessimistic at best, can be unsafe
- no dramatic improvements in sight
- “analyzability” should be a design constraint if needed

Simulation is a practical alternative even for critical systems .. with the proper methodology

- Determine quantile wrt criticality, and simulation length wrt to quantile
- Simulator and models validation
- High-performance simulation engine needed for higher quantiles
Increasingly complexity & higher load level calls for

1. More constraining specifications, or conservative assumptions → a single node can jeopardize the system

2. Combined use of verification techniques:
   - Refinement of traffic knowledge over time
   - Simulation and/or analysis, and trace inspection
   - None of them alone is sufficient

✓ No verification model & tool can be trusted blindly – always question assumptions

✓ If schedulability analysis is required, the (sub-)system should be conceived accordingly, otherwise simulation is - in our view - a better option
Interested in this talk and simulation methodology?

Please consult our appear at ERTSS’2014: “Timing verification of automotive communication architectures using quantile estimation” co-authored with Shehnaz LOUVART (Renault), Jose VILLANUEVA (Renault) and Jörn MIGGE (RealTime-at-Work).
References


