FORMAL METHODS AND TOOLS IN INDUSTRIAL APPLICATIONS: THE ASTRAIL APPROACH

SAFETY OF FUTURE SYSTEMS: SCIENCE MEETS INDUSTRY
Call: S2R JU Call for Proposals 2017

Topic: S2R-OC-IP2-01-2017

Research and Innovation Action

**SATellite-based Signalling and Automation SysTems on Railways** along with formal Method ad Moving Block Validation (ASTRail)

A specific call was issued asking to analyse the suitability of formal methods in supporting the transition to the **next generation of ERTMS/ETCS** signalling systems, which will include satellite-based train positioning, moving block distancing and automatic driving.
Enhance signalling and automation system leveraging cutting-edge technologies from different sectors and taking in particular care safety and performance issues.
OBJECTIVES

To examine the safety level of a Moving Block system in view of complete removal of trackside detection

1. To prepare a model of the moving block signalling system suitable for Hazard Analysis
2. To define system use cases to be analysed
3. To identify hazards derived from possible system errors and faulty states in main operative conditions
4. To assess the resulting risk level derived from identified hazards (risk qualifying)
5. To evaluate resulting safety level of a Moving Block signalling system operating without trackside train detection
6. To define Safety Related Application Conditions (operational procedures to be applied in normal or degraded conditions, according to GoAx, operational maintenance activities)
The H2020/ Shift2Rail initiative considers formal methods to be fundamental to the provision of safe and reliable technological advances to increase the competitiveness of the railway industry.

........ and

CENELEC EN 50128 for railway control and protection systems mentions FMs as Highly Recommended practices for SIL3-4 systems

But which FM method or which tools???

• Several FM methods exist (more than 100), with available support tools
• Several research experience on using FM in railways, and some success stories
Formal Methods and Railways Systems

• In the last decades, Formal and Semi-Formal Model Based Design has been largely adopted in the railway signalling industry, achieving lower costs and better quality software.

• The B method has been successfully applied to railway signaling systems, especially by MATRA Transport and ALSTOM, mainly in France.

• Simulink/Stateflow or SCADE have been used for modelling, exploiting the offered possibilities of automatic code generation for the production of high quality code.

• Model checking have been employed for verifying portions of an interlocking system (symbolic model checker NuSMV, explicit model checker SPIN, SAT/SMT-based Bounded MC).

• Model Based Testing has been adopted for verification, either together with more traditional testing techniques, or with model checking or abstract interpretation.
OUR OBJECTIVES IN ASTRAIL

• Identify the most suitable semi-formal and/or formal language and formal method to be applied in the railway field

1. review of the main formal modelling and verification languages and tools used in industrial railway applications

2. refine and rank identified languages and tools according to their use in the different phases of the development process of a railway application

3. evaluate usability and applicability of selected languages and tools in the railway domain by modelling the Moving Block signalling system

4. validate the identified languages and tools by modelling the integration of the Moving Block system with Automated Driving Technologies
OUR APPROACH

- Retrieve **structured information** about the state-of-the-art and the state-of-the-practice of FM in railways

- Systematic Literature Review
- Projects Review
- Questionnaire with Practitioners
- Tool Trial

**Ranking function**
CURRENT RESULTS - SLR
SLR GOALS

• Identify most used **formal** and **semi-formal** techniques used in **railways**
• Identify languages and **tools**
• Identify **tasks** performed
• Identify **phase** addressed
• Identify **industry readiness**

**A typical paper reviewed:**
• takes a railway system (e.g., entire ERTMS/ETCS)
• models a sub-system with a **semi-formal** language (e.g., UML)
• transforms the model into a **formal** model (e.g. UPPAAL, SPIN, NuSMV)
• verifies properties on the model
SLR – TECHNIQUES AND TOOLS

![Bar chart showing the usage of various tools and techniques in SLR studies. The tools listed are Simulink, UPPAAL, NuSMV/SMV, Atelier B, SPIN, Prover, ProB, Statemate, IBM Rational Software Architect, Rodin, Eclipse, Rhapsody, PHAVer, and CPN Tools. The usage level is indicated on a scale from 0 to 10.](image-url)
In most of the cases it is Architecture, and not design.

Papers do verification in the design phase, so we should not select Validation.
CURRENT RESULTS – TOOLS
TOOL REVIEW

SUBTITLE

Evaluation Features

SLR

12 Tools

Deadlock model

Moving-block model

Usability Assessment with industrial partner

Trial 2

Trial 1

6 Tools that may be acceptable for railway developers

Questionnaires

12 Tools

Tools that may be acceptable for railway developers

Trial 1

SLR

Presentation Title

Location, DD/MM/YYYY

14
TOOLS

• Tools are selected from SLR and from Questionnaires

• Selected tools for Trial 1 (deadlock avoidance model):
  – **Done (10):** CPN Tools, Atelier B/ProB, UPPAAL, NuSMV, SPIN, UMC, FDR4, mCRL2, CADP, TLA+
  – **TODO (2):** Simulink/Stateflow, SCADE

• Selected tools for Trial 2 (moving-block model):
  – **Done (2):** UPPAAL, Atelier B/ProB
  – **TODO (4):** NuSMV, SPIN, Simulink/Stateflow, SCADE
<table>
<thead>
<tr>
<th>Framework</th>
<th>Supported Verification Techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>UMC</td>
<td>model checking CTL-like, state-event based logics</td>
</tr>
<tr>
<td>SPIN</td>
<td>model checking LTL, fairness requirements</td>
</tr>
<tr>
<td>NuSMV/nuXMV</td>
<td>LTL, CTL, PSL [1], SMT model checking, fairness requirements</td>
</tr>
<tr>
<td>CADP</td>
<td>MCL [22], Parametric Mu-Calculus model checking, equivalence checking</td>
</tr>
<tr>
<td>UPPAAL</td>
<td>MITL [3], time-related, and probability related properties</td>
</tr>
<tr>
<td>TLA+</td>
<td>LTL, Theorem Proving, Proof Validations</td>
</tr>
<tr>
<td>ProB</td>
<td>LTL, CTL model checking, constraints based checking</td>
</tr>
<tr>
<td>mCRL2</td>
<td>Parametric Mu Calculus model checking, equivalence checking</td>
</tr>
<tr>
<td>FDR4</td>
<td>Refinement Checking, fairness requirements</td>
</tr>
<tr>
<td>CPN</td>
<td>CTL, custom ML properties</td>
</tr>
<tr>
<td>Framework</td>
<td>Range of evaluation times</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>UMC</td>
<td>38 - 86 seconds</td>
</tr>
<tr>
<td>SPIN</td>
<td>13 - 47 seconds</td>
</tr>
<tr>
<td>NuSMV/nuXVM</td>
<td>2.9 - 43 seconds</td>
</tr>
<tr>
<td>CADP</td>
<td>29 seconds</td>
</tr>
<tr>
<td>UPPAAL</td>
<td>16 seconds</td>
</tr>
<tr>
<td>TLA+</td>
<td>3 minutes</td>
</tr>
<tr>
<td>ProB</td>
<td>32 minutes</td>
</tr>
<tr>
<td>mCRL2</td>
<td>2 minutes - 19 minutes</td>
</tr>
<tr>
<td>FDR4</td>
<td>15 seconds - 20 minutes</td>
</tr>
<tr>
<td>CPN</td>
<td>unable to deal with the state-space size</td>
</tr>
</tbody>
</table>
Formal methods are often suggested to be used for system validation but verification tools are far from being themselves validated at the same safety level of the software they should verify. Using “diversity” in formal modelling, i.e. more than one model and tool, might mitigate this problem.

A lot of variability comes out when trying to use the existing tools. Depending on the tool specific choices made, the result can be orders of magnitude more efficient (or inefficient).

You really need to be an “expert” on a tool for reasonably use it.

When it comes to the analysis of likely incorrect system designs, the “feedback” provided by the tools, is often very inadequate.