How formal modeling addresses advanced issues in railway signalling

Alessandro Fantechi

University of Florence
Dipartimento di Ingegneria dell'Informazione
Florence, Italy

January 28, 2015 *Univ. of Surrey*

Railway Signalling equipments

Formal methods have traditionally focused on (*safety* issues in) particular classes of signalling equipments, e.g.:

- *train control systems*, that guarantee safe speed and braking control for trains, along the line (e.g. ERTMS/ETCS)
- *interlocking systems*, that establish safe routes through the intricate layout of tracks and points of a station.

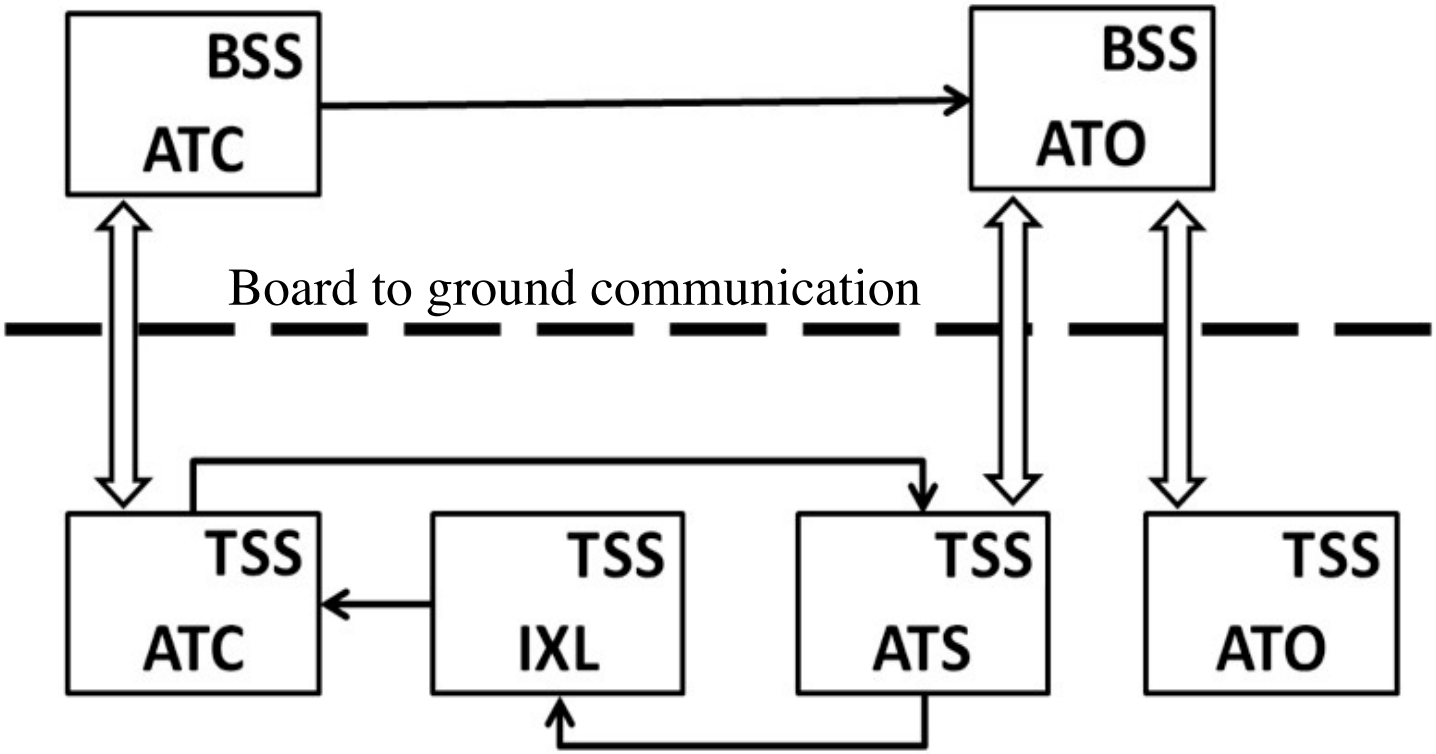
- *Trend towards greater integration of the two classes, with equipments providing both types of functions, or demanding stronger attention to interfaces among different types of equipment.*
- *Looking at the big picture:* **Systems of Systems** *of high complexity*

An example: CBTC (Communications-Based Train Control) systems; new generation control systems for urban railways (metros, light rails,...) that allow for a high degree of automation (e.g. driverless metros)

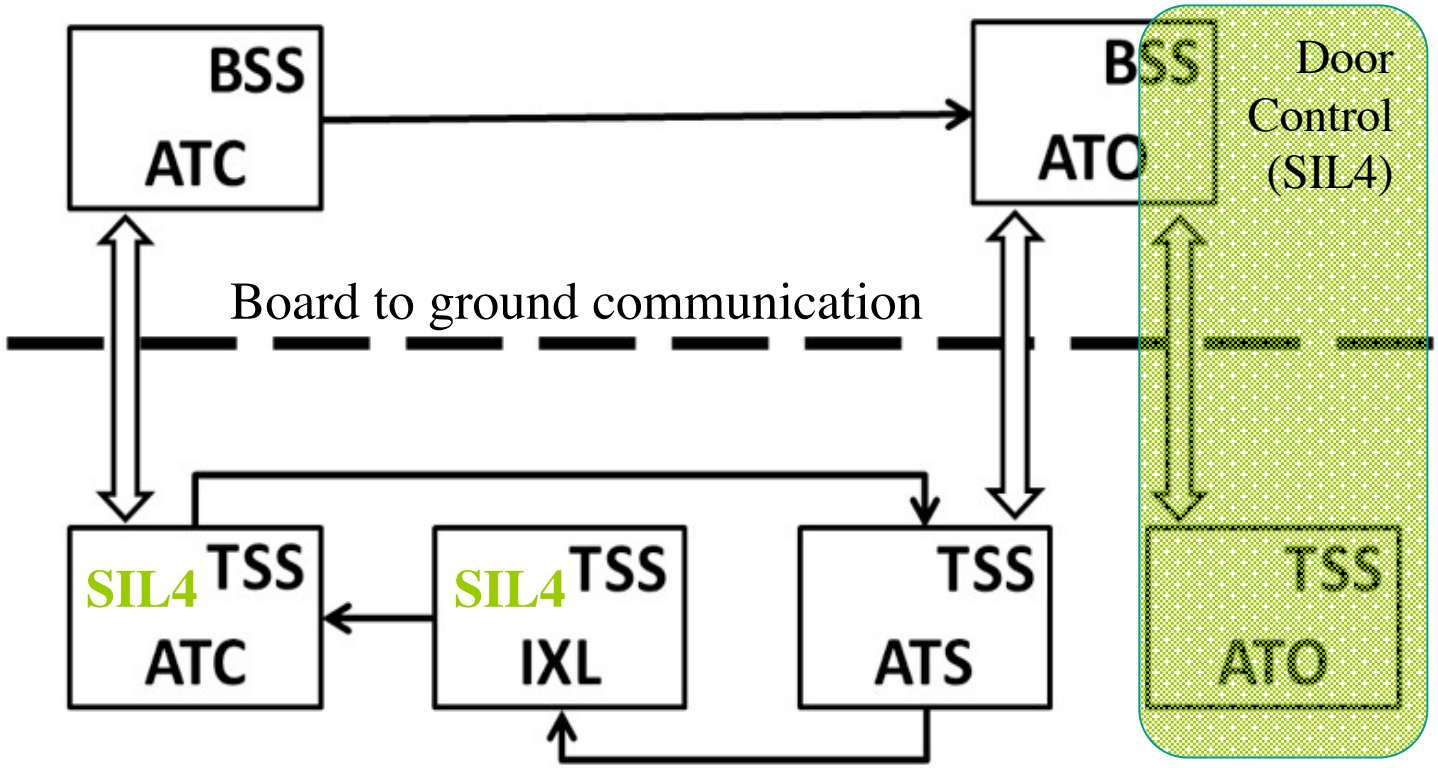
CBTC

- For CBTC systems the reference standards are IEEE 1474.1-2004 and IEC 62290. Although these documents do not constitute an industry standard for CBTC system architecture and function allocation, they provide a recommended practice for the design CBTC products to be launched into the market.
- The CBTC control systems are constituted by *onboard* equipments and *wayside* equipments. The former are installed on the trains and form the on-board subsystem (BSS). The latter are located at a station or along the line and form the trackside subsystem (TSS).
- The major identified components are ATC (Automatic Train Control), ATS (Automatic Train Supervision), ATO (Automatic Train Operation) and IXL (interlocking system).

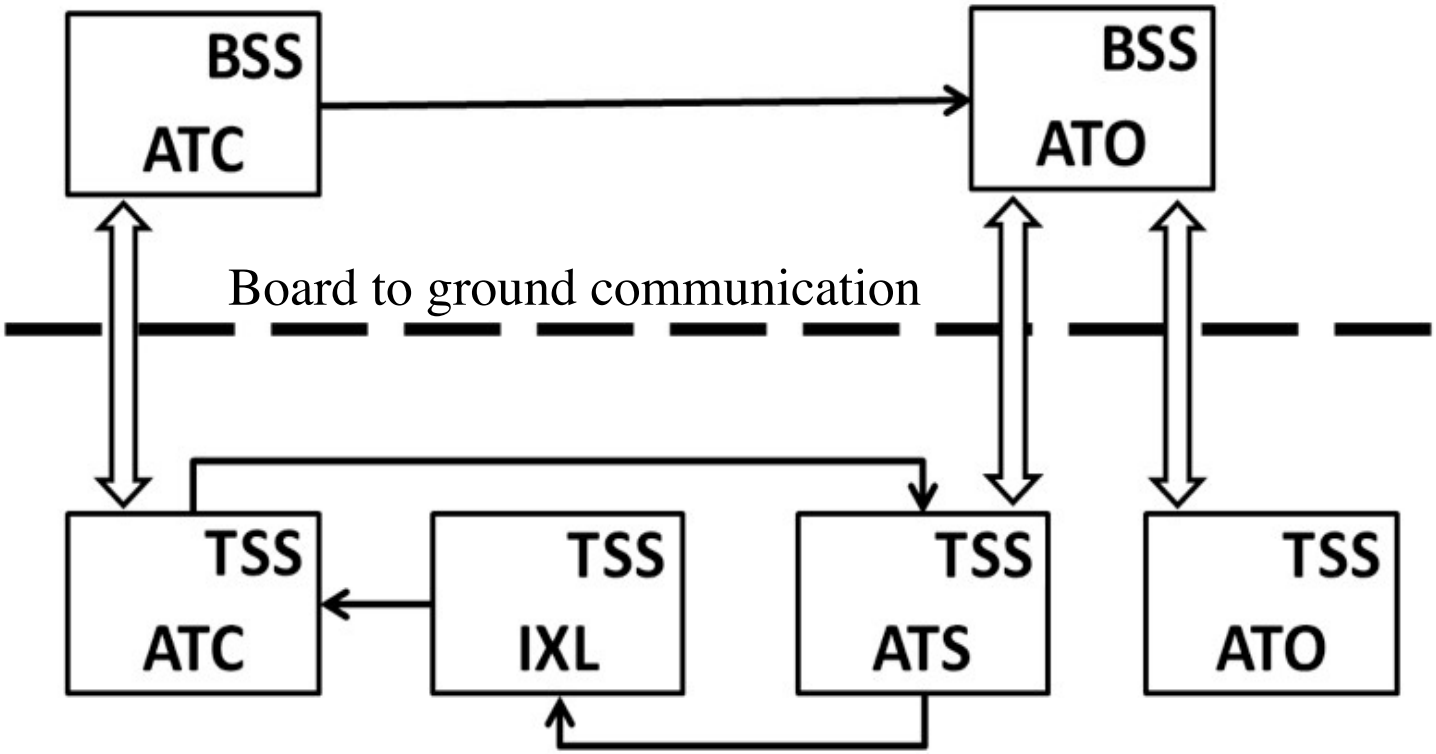
CBTC


BSS
ATC
BSS
ATO
Board to ground communication
TSS
ATC
TSS
IXL
TSS
ATS
TSS
ATO

CBTC

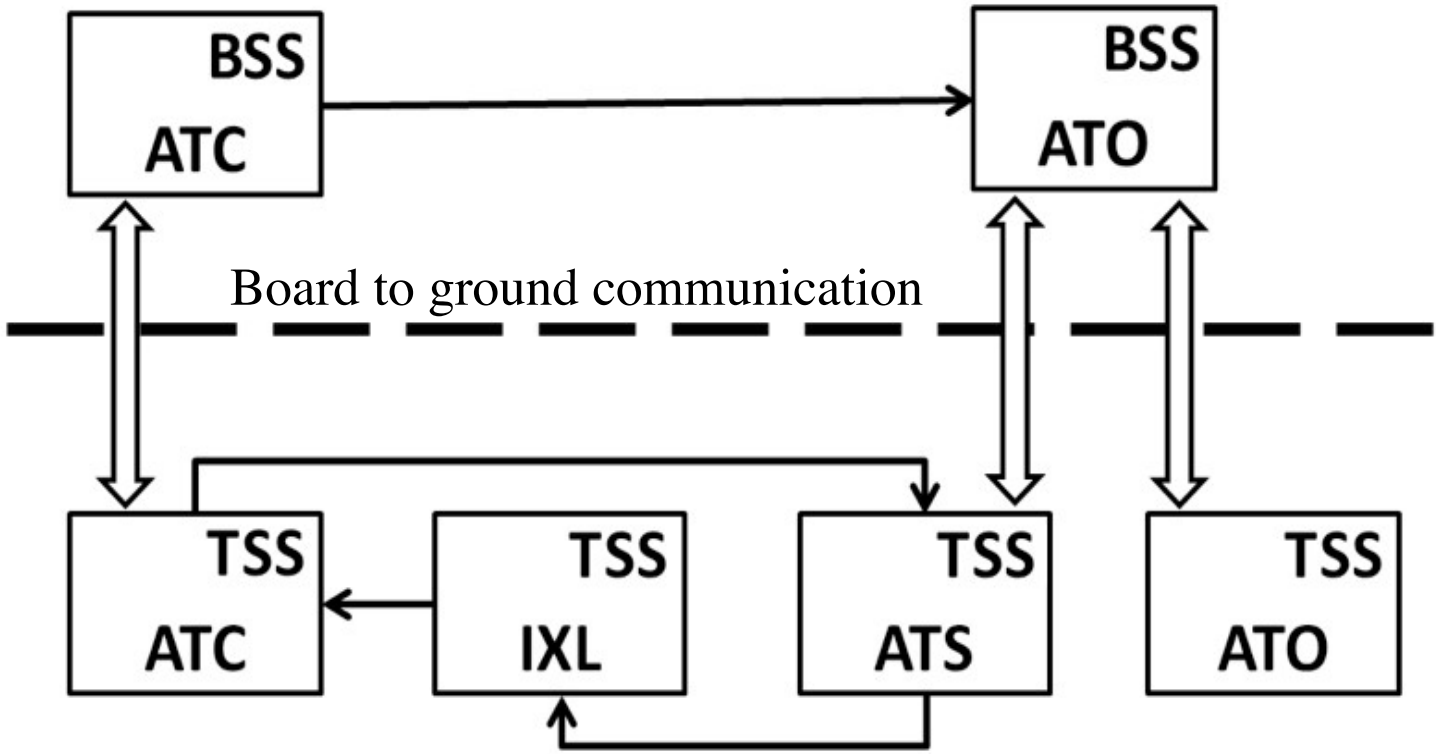

BSS
ATC
BSS
ATO
Door
Control
(SIL4)
Board to ground communication
SIL4 TSS
ATC
SIL4 TSS
IXL
TSS
ATS
TSS
ATO

CBTC


BSS
ATC
BSS
ATO
Board to ground communication
TSS
ATC
TSS
IXL
TSS
ATS
TSS
ATO

CBTC

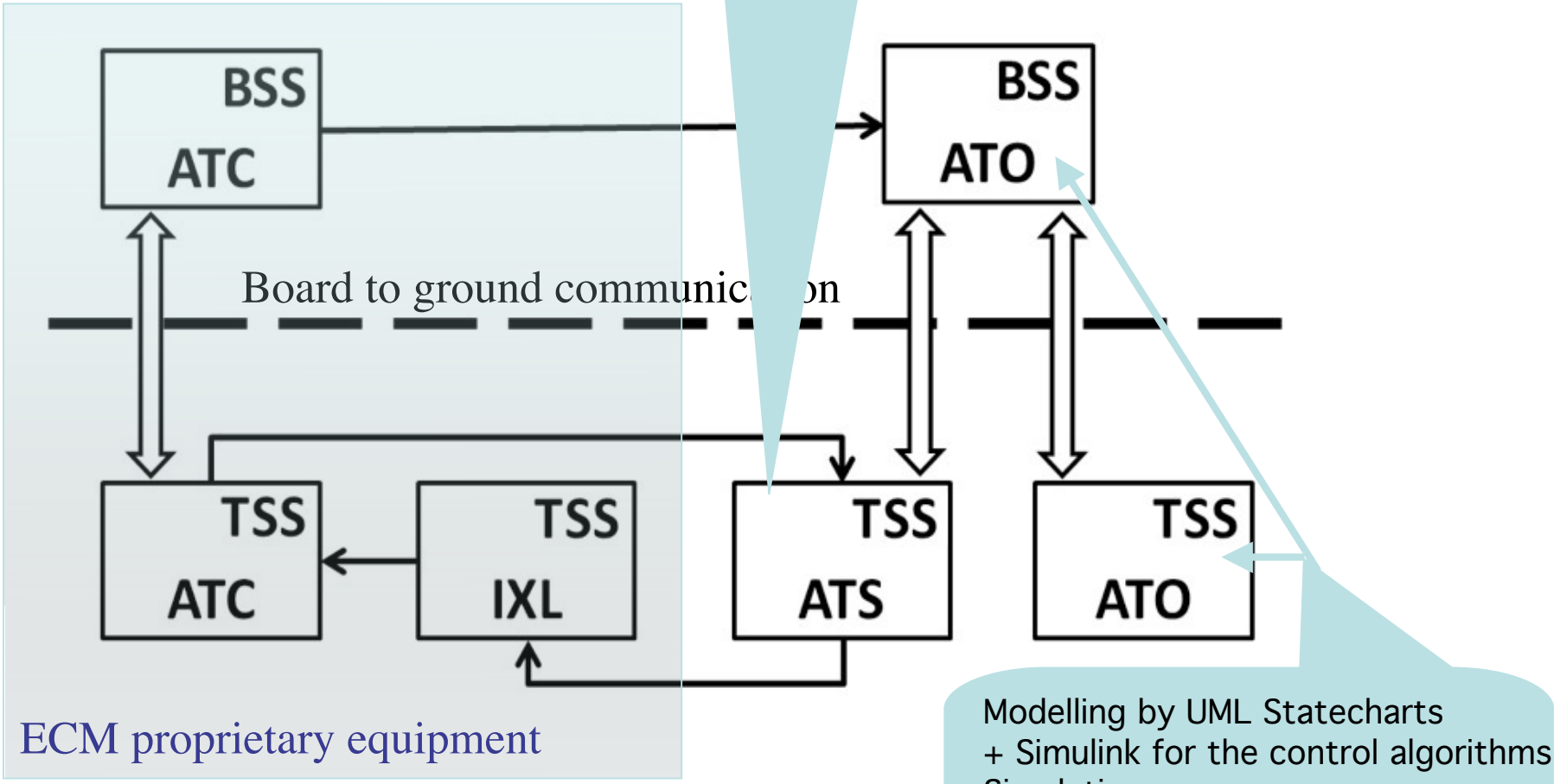
1 instance per train



1 instance per track zone

CBTC
Formal modelling
the TraceIt project

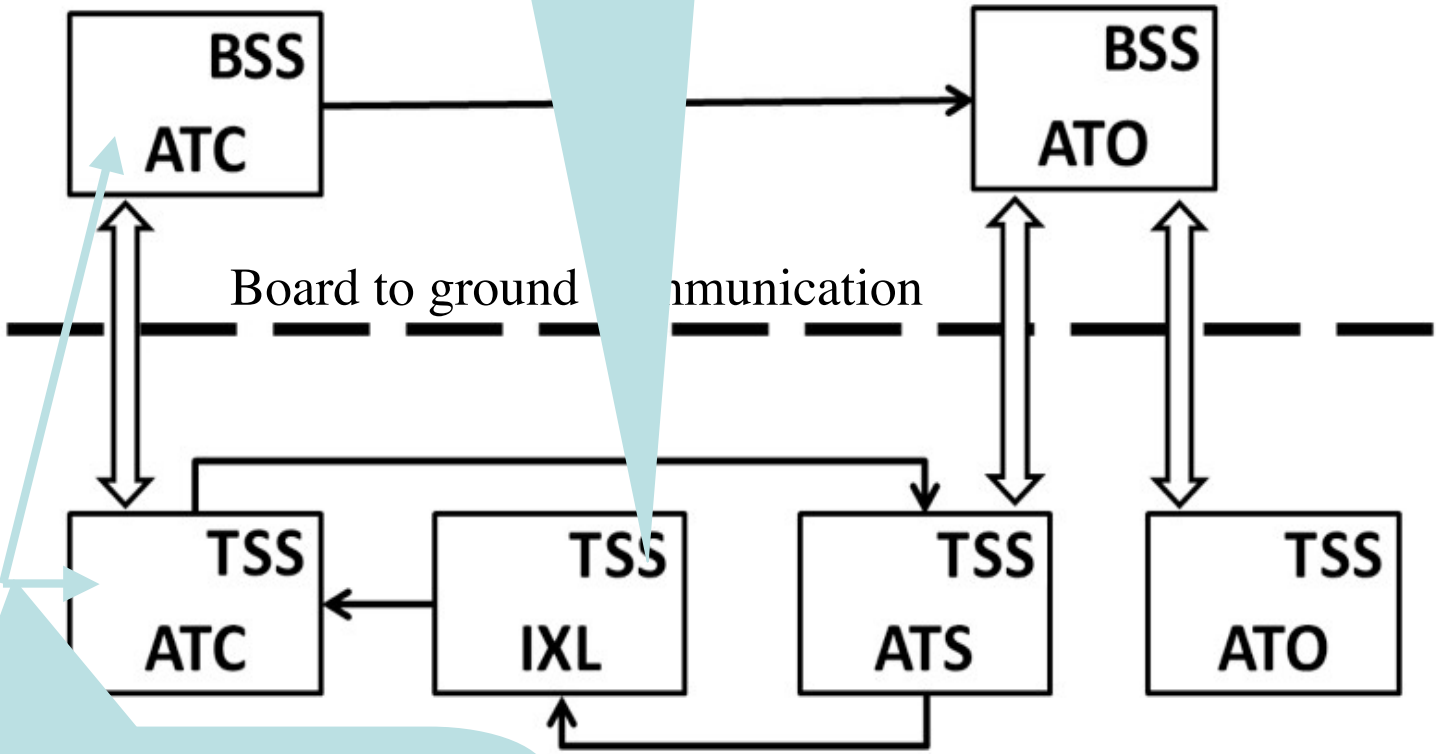
Train scheduling Deadlock avoidance
by UMC model checking
(ISTI - FMICS2014)
Capacity analysis by SAN
(ISTI - SERENE2013)



Modelling by UML Statecharts
+ Simulink for the control algorithms
Simulation
Auto-code generation
(UNIFI - ITSC2014)

CBTC
Formal modelling
other experiences

Several experiences of UNIFI, ISTI,
in collaboration with Ansaldo,
ALSTOM, GETS,...
Model checking, model based testing...



Model-based design and testing
UNIFI - GETS
IEEE Software 2013

Business as usual....

- Model-based disciplines applied in industry
- Formal methods sometimes applied in industry (e.g. the B "school")
- Formal verification sometimes applied in industry
- State explosion problems (trend to use SAT/SMT-based verification)
- Multi-level modelling (e.g., fine-grained modelling for safety model checking, coarse grained modeling for availability/capacity)
- Much work still to be done in these directions...

Geographic/Distributed approach

IXL traditionally seen as centralized systems, that manage a whole station, or a part of it in case of large ones:
a monolithic set of equations interpreted by a reasoner engine.

- Interlocking logic made up by composition of small elements that take care each of the control of a physical element (point, track circuit, signal), connected by means of predefined composition rules, mimicking the topology of the specific layout.
- Elements configured as a set of distributed communicating processes: each process controls a given layout element.
- The route is instead a global notion: a route has to be established by proper cooperation of the distributed elements.
- The communication among processes follows the physical layout of the station/yard and a route is established by the status of the elements that lie along the route.

Some experiments conducted at UNIFI on modelling the logic of a distributed interlocking
Different formalisms such as SCADE, Stateflow and UML State Diagrams.
Route establishment protocol based on the classical two-phase commit (2PC) protocol

Parallel verification

Distributed interlocking can be used just as a concept to optimize verification

- The preliminary results show that even for medium size interlockings, state spaces appear to be affordable, provided few trains are modeled, due to the "locking" properties of the system. Limiting trains to two limits concurrent behaviour. Easier to show independence from the number of trains:
- Proving no-collision for all conflicting pairs of routes with two trains requesting each route of the pair, is enough to prove global no-collision properties. Indeed, it appears that there is no possibility of a three train collision without a two-train collision first: hence proving that no two-train collision occur is enough.
- In the end, in order to prove global no-collision we need to perform a number of tractable verification runs, of the order at most of the square of the number of conflicting routes
- Much space for parallel verification here (e.g. distributing such computations on a grid formed by all of a company's computers).

Actual distribution of an IXL *(Systems of Systems of Systems)*

- The safety logic and the computing power is distributed according to the geographical distribution of the controlled elements, even considering the extreme scenario in which a controller is associated to each physical entity.
- and communicates with the adjacent controllers by means of safe, possibly wireless, communication.
- The distributed computing elements, according to the systems of systems paradigm, are autonomous and collaborate to the achievement of the main system functionalities.

Main problems to solve

- distributed initialization
- distributed configuration
- distributed re-configuration

Fine-grained distributed interlocking, including configuration and reconfiguration, is not something that is currently in practice nor in the foreseeable future (5 to 10 years) of the railway industry.

SaRDIn project proposal - Safe Reconfigurable Distributed Interlocking

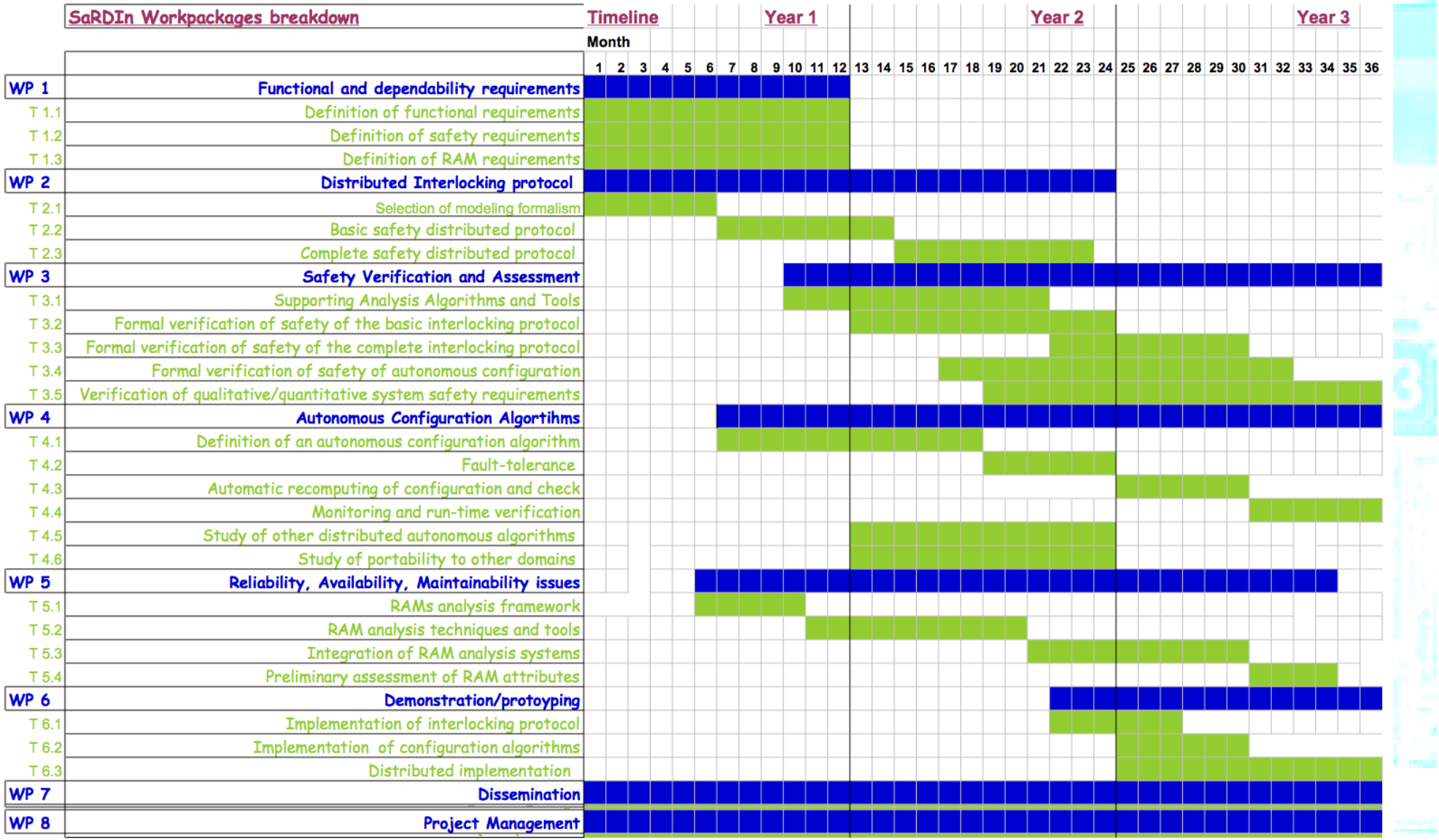
From the proposal statement:

The breakthrough of SaRDIn consists of a fundamentally new vision on the design and certification of safety-critical Systems of Systems, by pursuing the following new ideas:

- plug-and-play configuration algorithms based on a fully distributed safety layer;
- the configuration algorithms lead to compositional certification of safety-critical systems of systems;
- the configuration algorithms are demonstrated by formally verified distributed railway interlocking protocols.

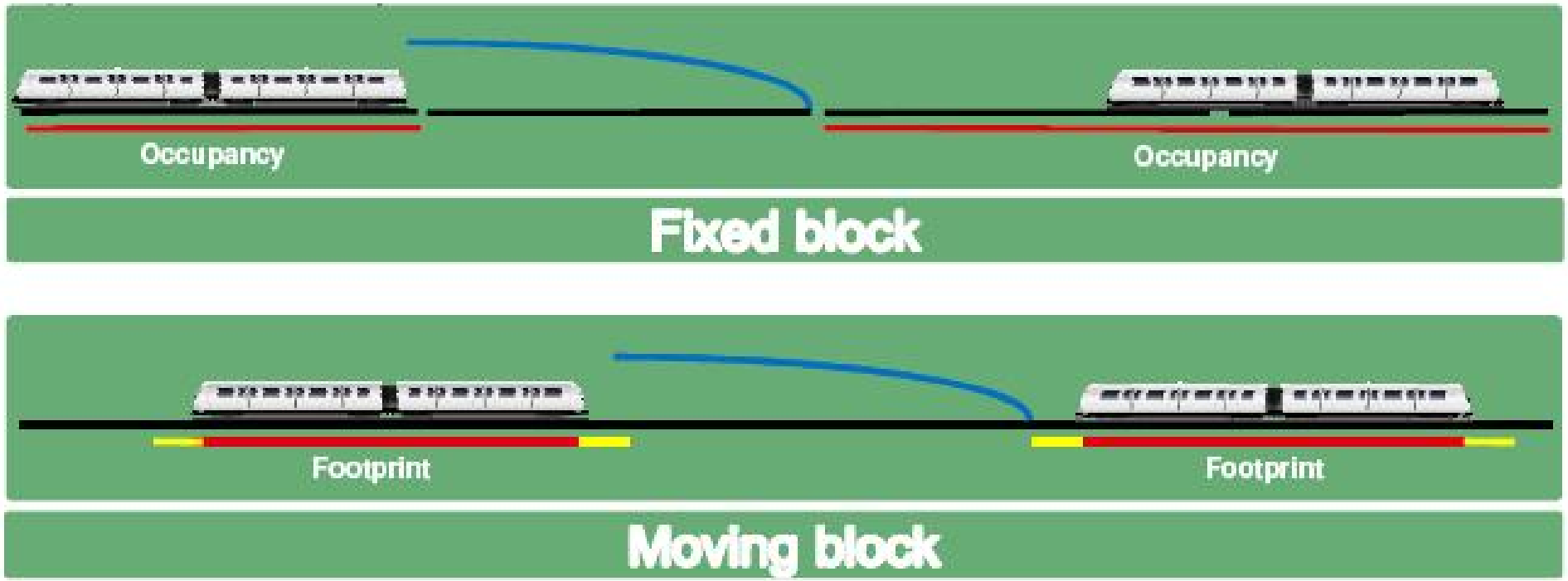
Certification aspects

- The configuration and verification process is expensive, monolithic and not easily repeatable.
- Moreover, certification has to be repeated for every deployed system, since the track layout changes from station to station.
- Hence, in case of modifications to the layout, reconfiguring the system on a new layout may be very expensive also for small layout changes.
- The configuration and verification process itself is often proprietary, and therefore an infrastructure company can become locked to the vendor for any modification of the track layout.
- But formal verification of the safety of the interlocking logic can take advantage from separating the common interlocking distributed protocol from the local logic, driven by local data about the adjacent nodes of the network, allowing for a *divide et impera* approach.


SaRDIn Workpackages breakdown
Timeline
Year 1
Year 2
Year 3
Month
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
WP 1 Functional and dependability requirements
T 1.1 Definition of functional requirements
T 1.2 Definition of safety requirements
T 1.3 Definition of RAM requirements
WP 2 Distributed Interlocking protocol
T 2.1 Selection of modeling formalism
T 2.2 Basic safety distributed protocol
T 2.3 Complete safety distributed protocol
WP 3 Safety Verification and Assessment
T 3.1 Supporting Analysis Algorithms and Tools
T 3.2 Formal verification of safety of the basic interlocking protocol
T 3.3 Formal verification of safety of the complete interlocking protocol
T 3.4 Formal verification of safety of autonomous configuration
T 3.5 Verification of qualitative/quantitative system safety requirements
WP 4 Autonomous Configuration Algortihms
T 4.1 Definition of an autonomous configuration algorithm
T 4.2 Fault-tolerance
T 4.3 Automatic recomputing of configuration and check
T 4.4 Monitoring and run-time verification
T 4.5 Study of other distributed autonomous algorithms
T 4.6 Study of portability to other domains
WP 5 Reliability, Availability, Maintainability issues
T 5.1 RAMs analysis framework
T 5.2 RAM analysis techniques and tools
T 5.3 Integration of RAM analysis systems
T 5.4 Preliminary assessment of RAM attributes
WP 6 Demonstration/protoyping
T 6.1 Implementation of interlocking protocol
T 6.2 Implementation of configuration algorithms
T 6.3 Distributed implementation
WP 7 Dissemination
WP 8 Project Management

- Reliability, Availability, Maintainability determine Capacity, Performance indexes, Serviceability....
- Quantitative evaluation of stochastic *-abilities:* SAN, STPN, Simulation, Probabilistic Model Checking (w.r.t. deterministic, say, Capacity definitions)
- Coarse grained modelling where Safety rules act as basic constraints. Again, the looser are these constraints, the higher are *-abilities* indexes...

Fixed block vs. Moving block



- *Fixed block*: line segmented into blocks, total occupancy of the leading train includes the whole block which the train is located on. Only allows the following train to move up to the last unoccupied block's border.
- *Moving block*: the train position and its braking curve is continuously calculated by the trains, and then communicated via radio to the wayside equipment, which establish protected areas, each one called Limit of Movement Authority (LMA), up to the nearest obstacle (tail of the train in front).
- *ERTMS: ETCS Lev. 2 has fixed blocks, ETCS Lev. 3 will have moving blocks.*

Safety paradigm shift

- Trains move within a safety envelope that is dynamically moving as well in front of the train.
- Capacity is higher the smaller is the safety envelope.
- The envelope depends on speed, hence capacity depends on speed
- The concept of dynamic safety envelope can be pushed to make dynamic all the static safety constraints that are traditional in Railways. For example, the concept of *route* that must be set free in front of a train and gives its movement envelope:
  - routes are currently predetermined in terms of a sequence of track elements
  - routes could be dynamically generated in front of the train, allowing for last minute choice according to optimization strategies
  - local (on board) choices basing on the knowledge on the next "payload" stop position and time
  - *need to balance autonomy of choice with safety and global optimization*

Visionary projects

- **Virtual Coupling:** considered in Shift2Rails, aims at having trains that run attached one to the other without being physically coupled
- **Dynamic connections:** Parallel running trains that allow for a safe exchange of passengers from one to the other routes while running - in order to minimize delays to AV trains for serving intermediate stations.
- **Very Long Freight Trains:** less visionary, currently in experimentation in France, VLFTs have the potential of blocking in a form of deadlock current interlocking systems

Conclusions

There are a lot of challenging issues out there for our "not-Leverhulme" network!!!

Conclusions

There are a lot of challenging issues out there for our "not-Leverhulme" network!!!

Why funding should not be there as well?