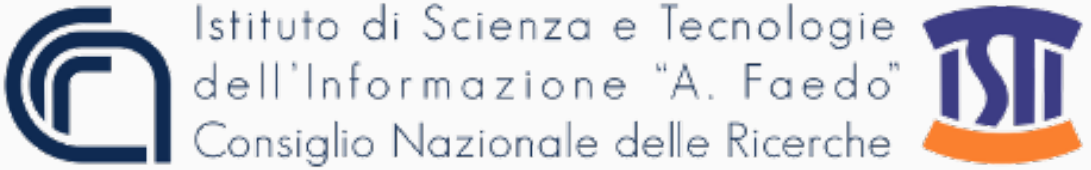
Models for formal methods and tools

The case of railway systems

Maurice ter Beek
FMT, CNR–ISTI, Pisa, Italy



MoDELS'23, Västerås, Sweden, October 4th, 2023

Outline

- Railway research
  - Sustainable mobility
  - Industrial standards
- Formal methods and tools
  - Models in railways
  - Tools in railways
- Some success stories
  - Rail scheduling
  - Railway signalling
  - Smart railways
- Concluding remarks

This is my first-ever MoDELS!

But I was told you like models . . .

Railway research

Global community effort for green transportation

EU green deal initiative

→H2020 Shift2Rail Joint Undertaking (JU): €920 million (2014–2020)

"*formal methods are fundamental for safe and reliable technological advances to increase the competitiveness of the European rail industry*"

→Horizon Europe Europe's Rail JU: €1.2 billion (2020–2027)

"*to ensure the safety, quality, and applicability of specifications, formal methods can provide significant benefits. Therefore, formal methods will be used for [. . .] to build the necessary confidence in the systems*"

Global community effort for green transportation

EU green deal initiative

→H2020 Shift2Rail Joint Undertaking (JU): €920 million (2014–2020)

"*formal methods are fundamental for safe and reliable technological advances to increase the competitiveness of the European rail industry*"

→Horizon Europe Europe's Rail JU: €1.2 billion (2020–2027)

"*to ensure the safety, quality, and applicability of specifications, formal methods can provide significant benefits. Therefore, formal methods will be used for [. . .] to build the necessary confidence in the systems*"

UK Rail Research and Innovation Network (UKRRIN)

Global community effort for green transportation

EU green deal initiative

→H2020 Shift2Rail Joint Undertaking (JU): €920 million (2014–2020)

"*formal methods are fundamental for safe and reliable technological advances to increase the competitiveness of the European rail industry*"

→Horizon Europe Europe's Rail JU: €1.2 billion (2020–2027)

"*to ensure the safety, quality, and applicability of specifications, formal methods can provide significant benefits. Therefore, formal methods will be used for [. . .] to build the necessary confidence in the systems*"

UK Rail Research and Innovation Network (UKRRIN)

China State Key Laboratory of Rail Traffic Control and Safety

FMT involved in EU, national and regional projects

Formal Methods and Tools Lab

ASTRail SAtellite-based Signalling and Automation SysTems on Railways along with Formal Method and Moving Block Validation (2017–2019)

→Requirements analysis plus safety, hazard and performance analyses of moving block signalling scenarios with the most suitable formal methods and tools

4SECURail Formal methods and CSIRT for the railway sector (2019–2021)

→Demonstrator to evaluate cost, benefits and required learning curve of using formal methods for rigorous specification of a railway signalling infrastructure

MOST CENTRO NAZIONALE PER LA MOBILITÀ SOSTENIBILE Sustainable Mobility Center: "Rail Transportation" (2022–2025)

→Learning formal models for predictive monitoring and maintenance

FMT involved in EU, national and regional projects

Formal Methods and Tools Lab

SATellite-based Signalling and Automation SysTems on **Rail**ways along with Formal Method and Moving Block Validation (2017–2019)

→Requirements analysis plus safety, hazard and performance analyses of moving block signalling scenarios with the most suitable formal methods and tools

Formal methods and CSIRT for the railway sector (2019–2021)

→Demonstrator to evaluate cost, benefits and required learning curve of using formal methods for rigorous specification of a railway signalling infrastructure

Sustainable Mobility Center: "Rail Transportation" (2022–2025)

→Learning formal models for predictive monitoring and maintenance

RT STINGRAY: SmarT station INtelliGent RAilwaY (2018–2021)

RT SmaRIERS: Smart Railway Infrastructures: Efficiency, Reliability and Safety (2021–2023)

FMT involved in EU, national and regional projects

Formal Methods and Tools Lab

SATellite-based Signalling and Automation SysTems on **Rail**ways along with Formal Method and Moving Block Validation (2017–2019)

→Requirements analysis plus safety, hazard and performance analyses of moving block signalling scenarios with the most suitable formal methods and tools

Formal methods and CSIRT for the railway sector (2019–2021)

→Demonstrator to evaluate cost, benefits and required learning curve of using formal methods for rigorous specification of a railway signalling infrastructure

Sustainable Mobility Center: "Rail Transportation" (2022–2025)

→Learning formal models for predictive monitoring and maintenance

RT STINGRAY: SmarT station INtelliGent RAilwaY (2018–2021)

RT SmaRIERS: Smart Railway Infrastructures: Efficiency, Reliability and Safety (2021–2023)

PRIN ADVENTURE: ADVancEd iNtegraTed evalUation of Railway systEms (2023–2025)

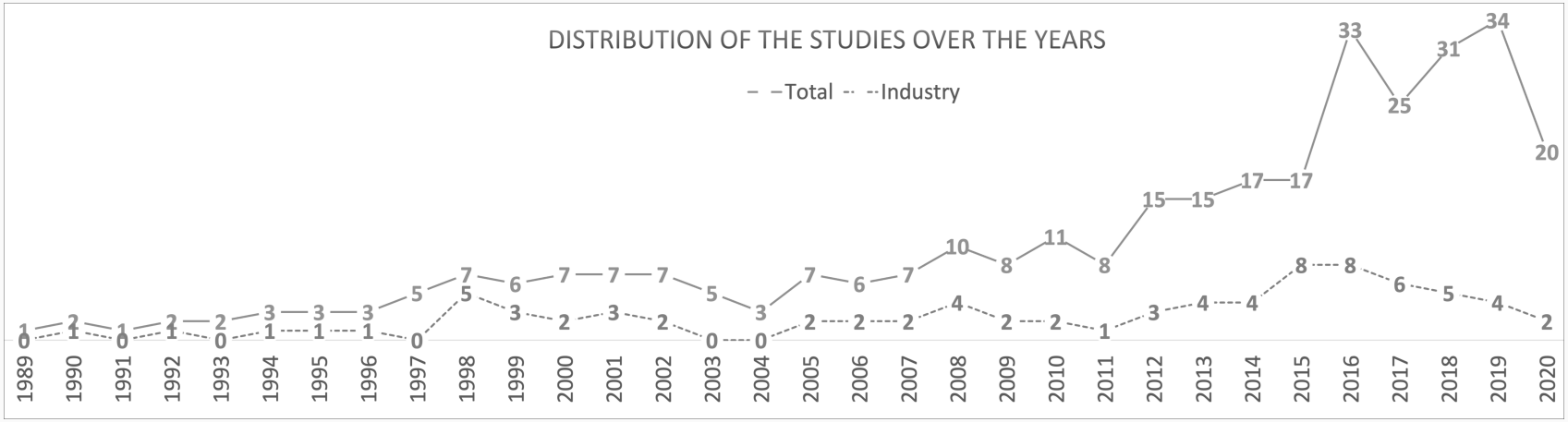
CENELEC EN 50128

EU standard for the development of software for the railway industry: highly recommends formal methods for the design and verification of products that need to meet the highest Safety Integrity Levels

CENELEC EN 50128

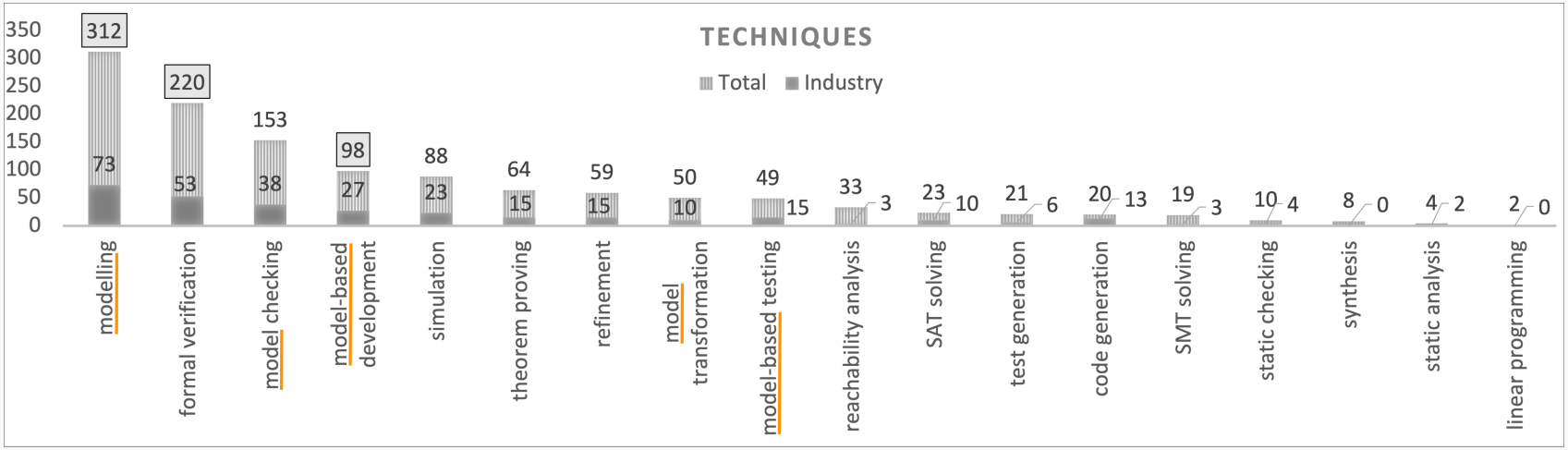
EU standard for the development of software for the railway industry: highly recommends formal methods for the design and verification of products that need to meet the highest Safety Integrity Levels

In fact, formal methods in railways is a thriving research field with strong industrial ties: 143 studies (44% of a total of 328) published solely in the last 5 years, while 79 studies (24%) involved industry

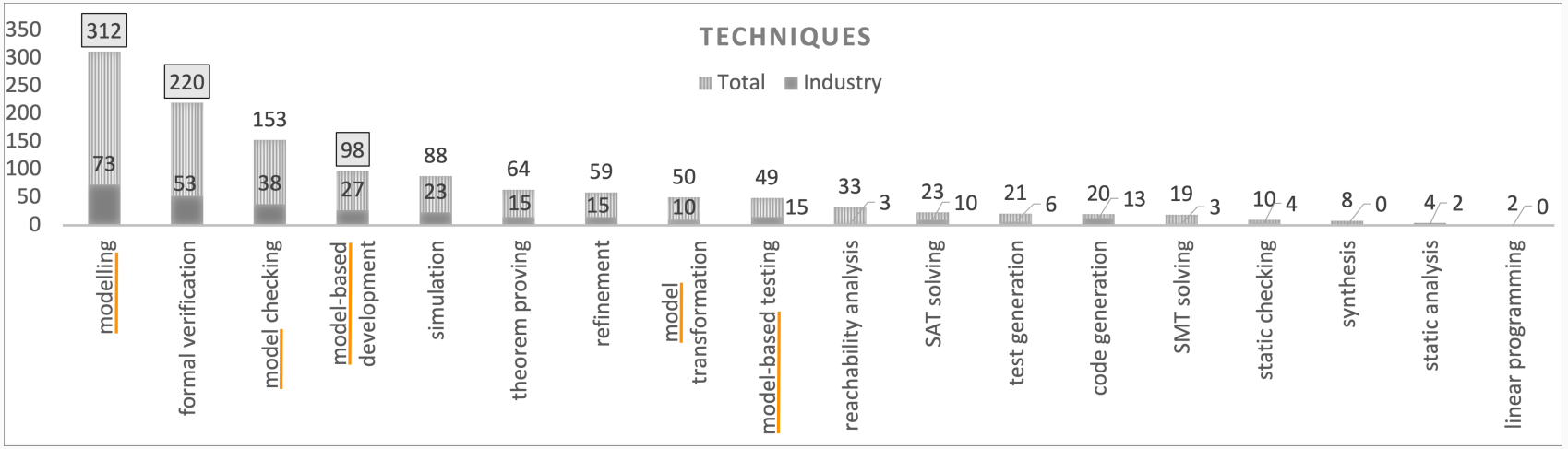


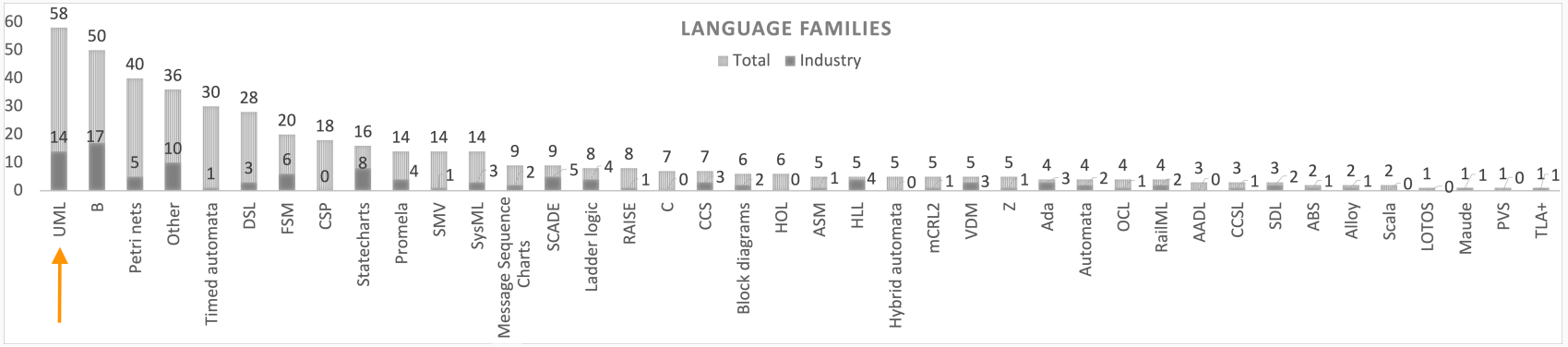
[CSUR22] A. Ferrari & M.H. ter Beek, Formal Methods in Railways: A Systematic Mapping Study. *ACM Computing Surveys* (2022)

Models are at the basis!

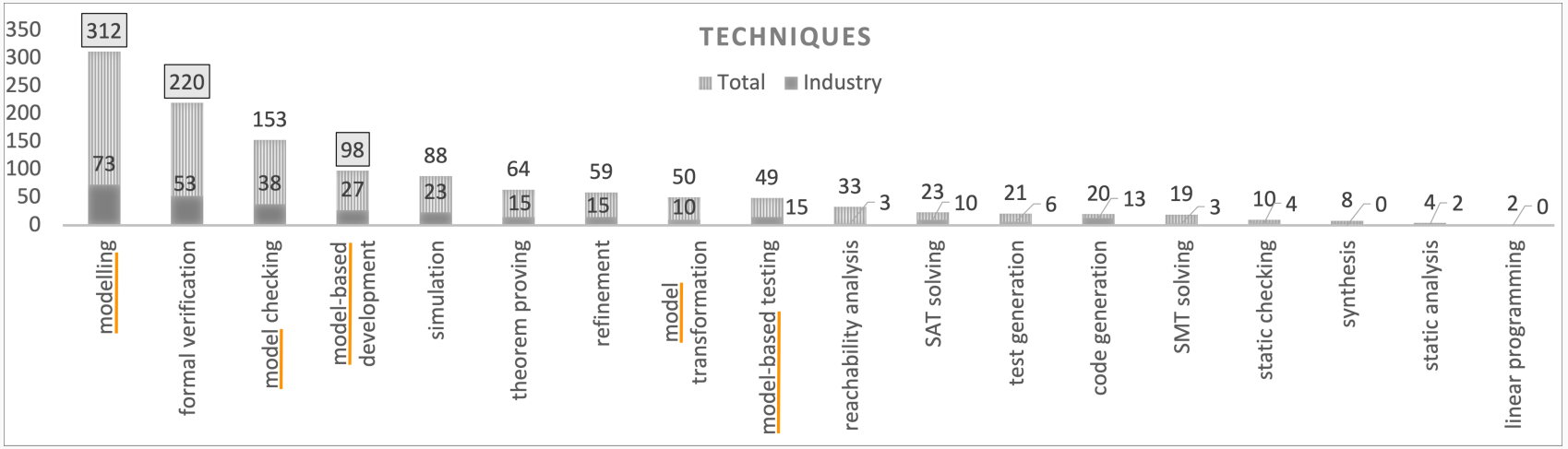


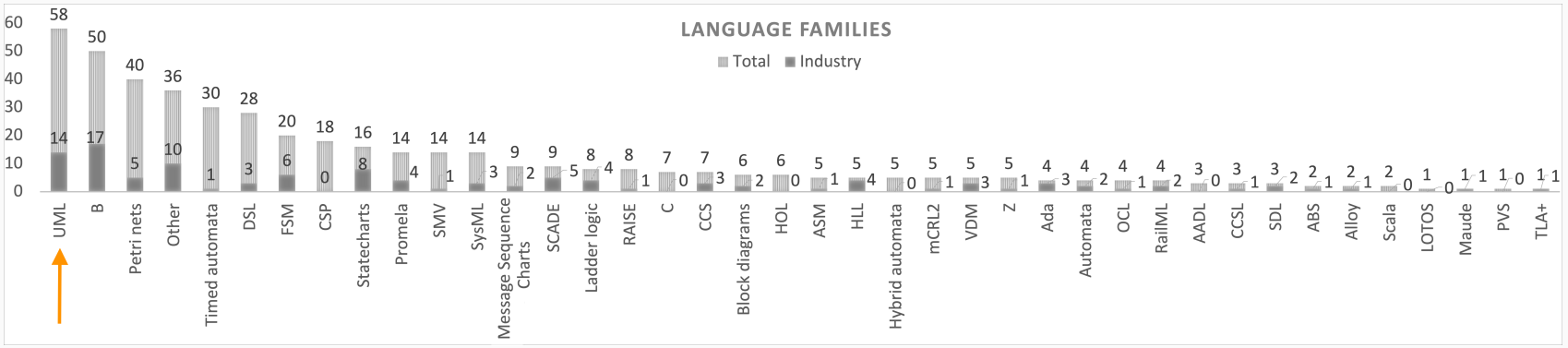
Models are at the basis!

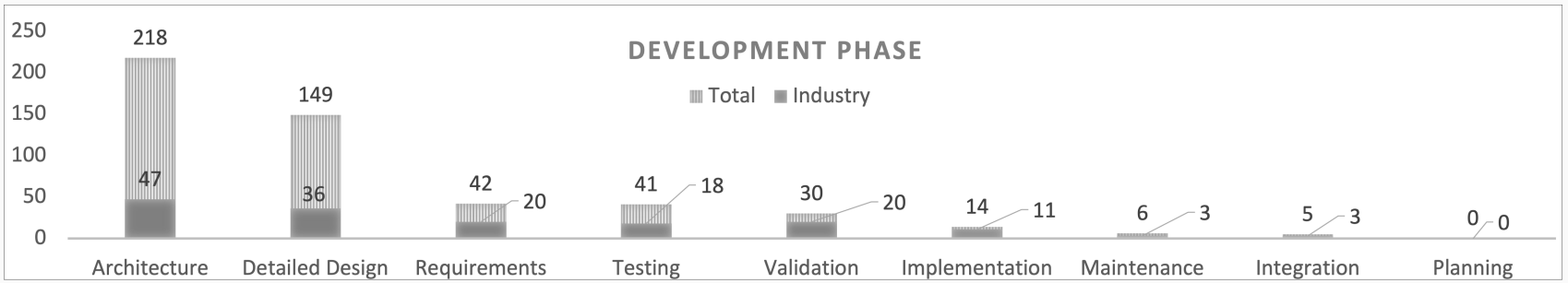




Models are at the basis!







(Formal) models in railways

Formal methods in railways: search process

Search String

Inclusion Criteria (*Abstract/Title*)

	ACM Digital Library			IEEE Xplore			Science Direct			SpringerLink - CS			SpringerLink - ENG		
Primary Search	I	II	III	I	II	III	I	II	III	I	II	III	I	II	III
Retrieved	72	9	321	631	88	205	321	86	141	1780	243	318	124	4	3
Included	34	0	6	163	15	28	74	12	8	298	22	55	8	1	0

Exclusion Criteria (*Full Text*)

Quality Criteria

Primary Search	I (X - 2017)	II (2017-2018)	III (2018 - 2020)	**Secondary Search**
Selected	198	27	67	36

	TOTAL
Retrieved	4346
Included	724
Selected	**328**

Formal methods . . .

Formal Methods and Tools Lab

"formal" OR "model check*" OR "model based" OR "model driven"
OR "theorem prov*" OR "static analysis"

AND

"railway*" OR "CBTC" OR "ERTMS" OR "ETCS" OR "interlocking"
OR "automatic train" OR "train control" OR "metro" OR "CENELEC"

Formal methods . . .

"formal" OR "model check*" OR "model based" OR "model driven"
OR "theorem prov*" OR "static analysis"

AND

"railway*" OR "CBTC" OR "ERTMS" OR "ETCS" OR "interlocking"
OR "automatic train" OR "train control" OR "metro" OR "CENELEC"

Formal methods are rigorous mathematics-based techniques and tools for the specification (modelling) and manual or automated verification (analysis) of software or hardware systems (designs)

Formal methods . . .

"formal" OR "model check*" OR "model based" OR "model driven"
OR "theorem prov*" OR "static analysis"

AND

"railway*" OR "CBTC" OR "ERTMS" OR "ETCS" OR "interlocking"
OR "automatic train" OR "train control" OR "metro" OR "CENELEC"

Formal methods are rigorous mathematics-based techniques and tools for the specification (modelling) and manual or automated verification (analysis) of software or hardware systems (designs)

Formal verification concerns verifying (e.g., by model checking) that functional properties (e.g., absence of deadlocks) or critical system properties related to safety are satisfied

Formal methods . . .

"formal" OR "model check*" OR "model based" OR "model driven"
OR "theorem prov*" OR "static analysis"

AND

"railway*" OR "CBTC" OR "ERTMS" OR "ETCS" OR "interlocking"
OR "automatic train" OR "train control" OR "metro" OR "CENELEC"

Formal methods are rigorous mathematics-based techniques and tools for the specification (modelling) and manual or automated verification (analysis) of software or hardware systems (designs)

Formal verification concerns verifying (e.g., by model checking) that functional properties (e.g., absence of deadlocks) or critical system properties related to safety are satisfied $\Rightarrow$ requires formal models **with a precise semantics** . . .

Formal methods . . .

"formal" OR "model check*" OR "model based" OR "model driven"
OR "theorem prov*" OR "static analysis"

AND

"railway*" OR "CBTC" OR "ERTMS" OR "ETCS" OR "interlocking"
OR "automatic train" OR "train control" OR "metro" OR "CENELEC"

Formal methods are rigorous mathematics-based techniques and tools for the specification (modelling) and manual or automated verification (analysis) of software or hardware systems (designs)

Formal verification concerns verifying (e.g., by model checking) that functional properties (e.g., absence of deadlocks) or critical system properties related to safety are satisfied $\Rightarrow$ requires formal models **with a precise semantics** . . . UML models are semi-formal!

Formal methods . . .

"formal" OR "model check*" OR "model based" OR "model driven"
OR "theorem prov*" OR "static analysis"

AND

"railway*" OR "CBTC" OR "ERTMS" OR "ETCS" OR "interlocking"
OR "automatic train" OR "train control" OR "metro" OR "CENELEC"

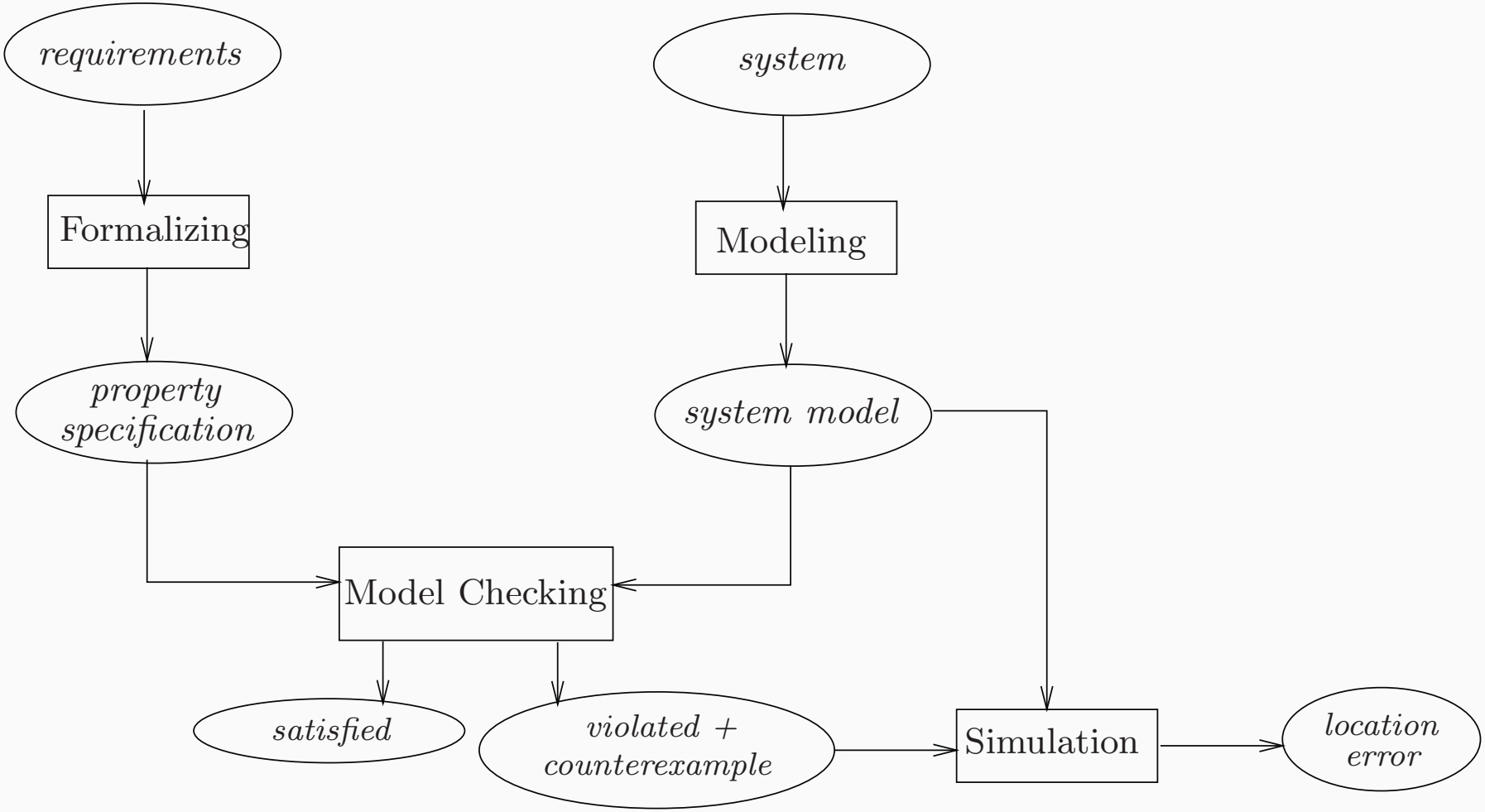
Formal methods are rigorous mathematics-based techniques and tools for the specification (modelling) and manual or automated verification (analysis) of software or hardware systems (designs)

Formal verification concerns verifying (e.g., by model checking) that functional properties (e.g., absence of deadlocks) or critical system properties related to safety are satisfied $\Rightarrow$ requires formal models **with a precise semantics** . . . UML models are semi-formal!

"testing can be used to show the presence of bugs,
but never to show their absence!" – Edsger W. Dijkstra

Intermezzo: model checking

What is model checking?



C. Baier & J.-P. Katoen, *Principles of Model Checking*, MIT, 2008

Strengths and weaknesses

- General verification approach, applicable to a wide range of applications (e.g., embedded systems, software engineering, hardware design)
- Supports partial verification, i.e. properties can be checked individually, allowing to focus on essential properties first (no complete requirement specification needed)
- Not vulnerable to likelihood that an error is exposed; this contrasts with testing and simulation that are aimed at tracing the most probable defects
- Provides diagnostic info in case a property is invalidated: very useful for debugging
- **Potentially a "push-button" technology**, requiring neither a high degree of user interaction nor a high degree of expertise
- Enjoys a rapidly increasing interest by industry: several companies have started their in-house verification labs (e.g., Amazon), frequent job offers with required skills in model checking, and commercial model checkers have become available
- Easy integration in existing development cycles: its learning curve is not very steep, empirical studies indicate that it may lead to shorter development times
- Sound and mathematical underpinning, based on theory of graph algorithms, data structures and logic

Strengths and weaknesses

- General verification approach, applicable to a wide range of applications (e.g., embedded systems, software engineering, hardware design)
- Supports partial verification, i.e. properties can be checked individually, allowing to focus on essential properties first (no complete requirement specification needed)
- Not vulnerable to likelihood that an error is exposed; this contrasts with testing and simulation that are aimed at tracing the most probable defects
- Provides diagnostic info in case a property is invalidated: very useful for debugging
- **Potentially a "push-button" technology**, requiring neither a high degree of user interaction nor a high degree of expertise
- Enjoys a rapidly increasing interest by industry: several companies have started their in-house verification labs (e.g., Amazon), frequent job offers with required skills in model checking, and commercial model checkers have become available
- Easy integration in existing development cycles: its learning curve is not very steep, empirical studies indicate that it may lead to shorter development times
- Sound and mathematical underpinning, based on theory of graph algorithms, data structures and logic

Strengths and weaknesses

- Mainly appropriate to control-intensive applications and less suited for data-intensive applications as data typically ranges over infinite domains
- Applicability subject to decidability issues: for infinite-state systems or reasoning about abstract data types (which requires undecidable or semi-decidable logics), model checking is in general not effectively computable
- Verifies a system model, and not the actual system (product or prototype) itself: **any obtained result is thus as good as the system model** (need complementary techniques, like testing, to find fabrication faults (for HW) or coding errors (for SW)
- Checks only stated requirements, i.e. **no guarantee of completeness**, since the validity of properties that are not checked cannot be judged
- Suffers from state space explosion problem: number of states needed to model the system accurately may easily exceed the amount of available computer memory (despite the development of several very effective methods to combat this problem, models of realistic systems may still be too large to fit in memory)
- Its usage requires some **expertise** in finding appropriate abstractions to obtain smaller system models and to state properties in the logical formalism used
- Correct results not guaranteed: like any tool, it may contain software defects
- Does not allow checking generalizations: in general, checking systems with an arbitrary number of components, or parameterized systems, cannot be treated

Strengths and weaknesses

- Mainly appropriate to control-intensive applications and less suited for data-intensive applications as data typically ranges over infinite domains
- Applicability subject to decidability issues: for infinite-state systems or reasoning about abstract data types (which requires undecidable or semi-decidable logics), model checking is in general not effectively computable
- Verifies a system model, and not the actual system (product or prototype) itself: **any obtained result is thus as good as the system model** (need complementary techniques, like testing, to find fabrication faults (for HW) or coding errors (for SW)
- Checks only stated requirements, i.e. **no guarantee of completeness**, since the validity of properties that are not checked cannot be judged
- Suffers from state space explosion problem: number of states needed to model the system accurately may easily exceed the amount of available computer memory (despite the development of several very effective methods to combat this problem, models of realistic systems may still be too large to fit in memory)
- Its usage requires some **expertise** in finding appropriate abstractions to obtain smaller system models and to state properties in the logical formalism used
- Correct results not guaranteed: like any tool, it may contain software defects
- Does not allow checking generalizations: in general, checking systems with an arbitrary number of components, or parameterized systems, cannot be treated

. . . and tools

Model Checking

- Automatically check whether a formal model satisfies a temporal logic property (LTL, CTL) and provide a counterexample if it does not
- Exhaustive, but suffers from the state space explosion problem
- BLAST, CADP, JPF, mCRL2, PRISM, (Nu)SMV, SPIN, UMC, UPPAAL, . . .

. . . and tools

Model Checking

- Automatically check whether a formal model satisfies a temporal logic property (LTL, CTL) and provide a counterexample if it does not
- Exhaustive, but suffers from the state space explosion problem
- BLAST, CADP, JPF, mCRL2, PRISM, (Nu)SMV, SPIN, UMC, UPPAAL, . . .

Probabilistic/Stochastic Model Checking (PMC)

- Model check whether a stochastic model satisfies a temporal logic property (PCTL, CSL) with a probability greater than a set threshold
- Model uncertainty/performance; do quantitative analysis (QoS, . . .)
- CADP, LiQuor, MRMC, PARAM, PRISM, Storm, UPPAAL PRO, . . .

. . . and tools

Model Checking

- Automatically check whether a formal model satisfies a temporal logic property (LTL, CTL) and provide a counterexample if it does not
- Exhaustive, but suffers from the state space explosion problem
- BLAST, CADP, JPF, mCRL2, PRISM, (Nu)SMV, SPIN, UMC, UPPAAL, . . .

Probabilistic/Stochastic Model Checking (PMC)

- Model check whether a stochastic model satisfies a temporal logic property (PCTL, CSL) with a probability greater than a set threshold
- Model uncertainty/performance; do quantitative analysis (QoS, . . .)
- CADP, LiQuor, MRMC, PARAM, PRISM, Storm, UPPAAL PRO, . . .

Statistical Model Checking (SMC)

- Simulation-based technique to statistically approximate (P)MC
- Highly parallelisable and automatable; tunable preciseness via CI
- PLASMA, PRISM, UPPAAL SMC, (P)VeStA, MultiVeStA, QFLan, . . .

SMC and UPPAAL SMC

SMC: run a controlled number N of (probabilistically distributed) simulations of a system model to obtain a statistical evaluation p' of some formula φ such that

- $\Pr(|p' - p| \leq \epsilon) \geq 1 - \alpha$
- N only depends on α (precision, sub-linear) and ϵ (confidence, quadratic)

SMC and UPPAAL SMC

SMC: run a controlled number N of (probabilistically distributed) simulations of a system model to obtain a statistical evaluation p' of some formula φ such that

- $\Pr(|p' - p| \leq \epsilon) \geq 1 - \alpha$
- N only depends on α (precision, sub-linear) and ϵ (confidence, quadratic)

Pros: avoid full state-space exploration, easy to implement and to parallelise, scalable

Cons: no 100% exact results, difficulties in dealing with rare events

SMC and UPPAAL SMC

SMC: run a controlled number N of (probabilistically distributed) simulations of a system model to obtain a statistical evaluation p' of some formula φ such that

- $\Pr(|p' - p| \leq \epsilon) \geq 1 - \alpha$
- N only depends on α (precision, sub-linear) and ϵ (confidence, quadratic)

Pros: avoid full state-space exploration, easy to implement and to parallelise, scalable

Cons: no 100% exact results, difficulties in dealing with rare events

UPPAAL SMC:

- frequently used to specify and analyse railway systems, modelled as Stochastic Hybrid Automata (SHA), implementing SMC
- graphical interactive simulations, also basic CTL model checking

(Formal) tools in railways

Formal tools in railways: research questions

Systematic evaluation of formal tools for system design in railways

B. Kitchenham, S. Linkman & D. Law, DESMET: A methodology for evaluating software engineering methods and tools. *Computing & Control Engineering Journal* (1997)

Formal tools in railways: research questions

Formal Methods and Tools Lab

Systematic evaluation of formal tools for system design in railways

B. Kitchenham, S. Linkman & D. Law, DESMET: A methodology for evaluating software engineering methods and tools. *Computing & Control Engineering Journal* (1997)

RQ1 *Which are the features to consider for evaluating formal tools?*

RQ2 *How do different tools compare with respect to these features?*

RQ3 *How do different tools compare with respect to their usability?*

[TSE22] A. Ferrari, F. Mazzanti, D. Basile & M.H. ter Beek, Systematic Evaluation and Usability Analysis of Formal Methods Tools for Railway Signaling System Design. *IEEE Transactions on Software Engineering* (2022)

[ICSE20] A. Ferrari, F. Mazzanti, D. Basile, M.H. ter Beek & A. Fantechi, Comparing Formal Tools for System Design: a Judgment Study @ ICSE'20

Formal tools in railways: research questions

Systematic evaluation of formal tools for system design in railways

B. Kitchenham, S. Linkman & D. Law, DESMET: A methodology for evaluating software engineering methods and tools. *Computing & Control Engineering Journal* (1997)

RQ1 *Which are the features to consider for evaluating formal tools?*

RQ2 *How do different tools compare with respect to these features?*

RQ3 *How do different tools compare with respect to their usability?*

[TSE22] A. Ferrari, F. Mazzanti, D. Basile & M.H. ter Beek, Systematic Evaluation and Usability Analysis of Formal Methods Tools for Railway Signaling System Design. *IEEE Transactions on Software Engineering* (2022)

[ICSE20] A. Ferrari, F. Mazzanti, D. Basile, M.H. ter Beek & A. Fantechi, Comparing Formal Tools for System Design: a Judgment Study @ ICSE'20

Selected tools? Top ranked ones in a survey among railway practitioners

[FM19] M.H. ter Beek et al., Adopting Formal Methods in an Industrial Setting @ FM'19

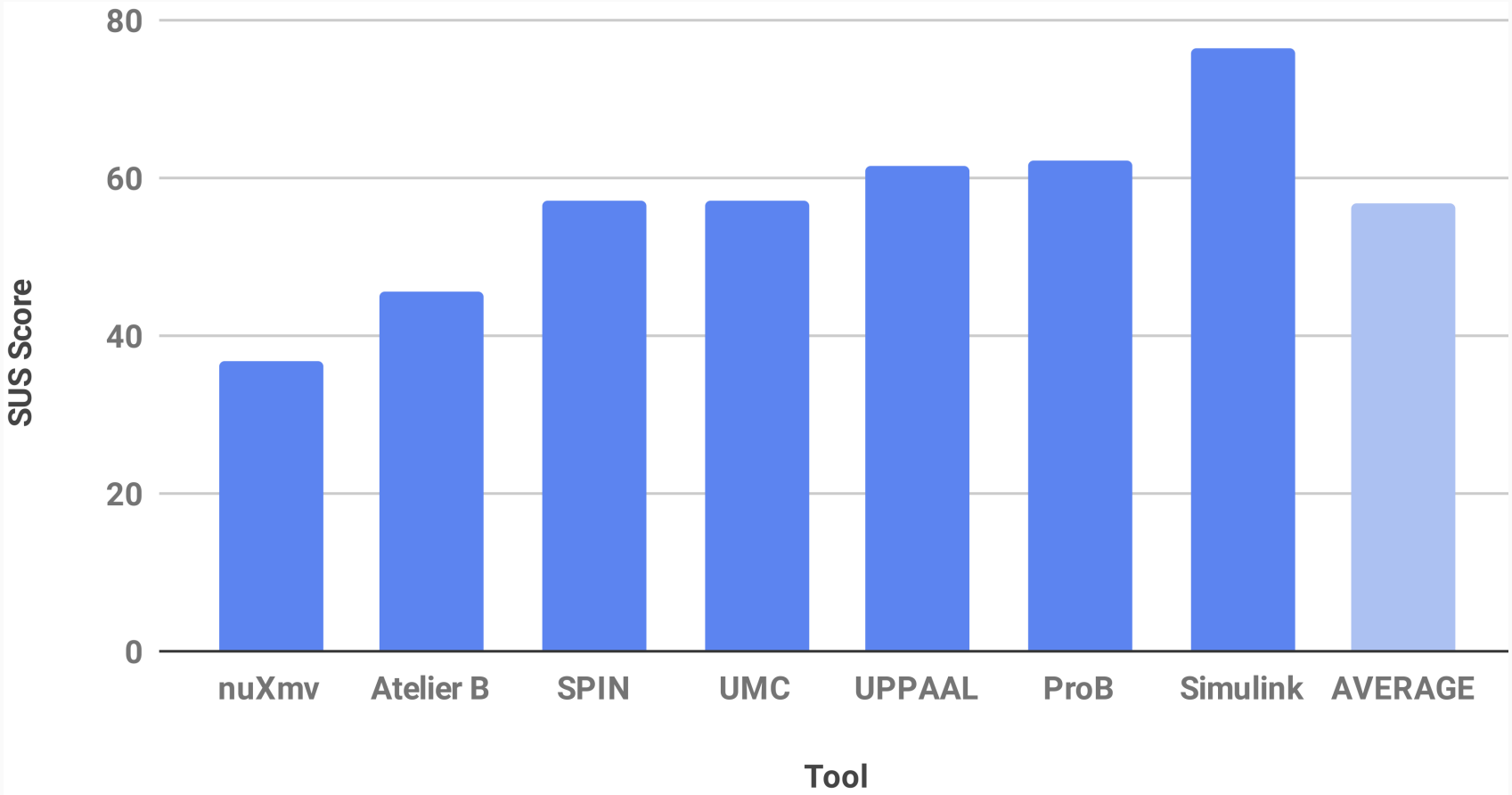
Characteristics and expertise of study participants

ID	Role in Study	Milieu	Main Function	Age	Sex	Years of Experience in		
						Formal Methods (FM)	Railway Industry	FM in Railways
1	assessor	academic	workpackage leader	39	M	> 13	3	13
2	assessor	academic	tool developer	62	M	> 20	0	9
3	assessor	academic	researcher	36	M	$>\ 6$	0	4
4	expert	academic	group leader	48	M	> 15	0	9
5	expert	academic	project leader	66	F	> 30	0	> 25
6	expert	academic	professor	65	M	> 30	0	> 25
7	expert	industry	system engineer	NA	M	0	> 10	0
8	expert	industry	system engineer	52	M	0	> 10	0
9	expert	industry	system engineer	48	M	0	> 10	0
10	expert	industry	software developer	43	M	0	> 10	0
11	expert	industry	product manager	NA	M	0	> 10	0
12	expert	industry	system engineer	48	M	0	> 10	0
13	expert	industry	innovation engineer	NA	M	0	> 10	0
14	expert	industry	software developer	45	M	0	> 10	0
15	expert	industry	innovation engineer	NA	F	0	3 to 10	0

Feature evaluation table

Category	Name	SPIN	Simulink	nuXmv	ProB	Atelier B	UPPAAL	FDR4	CPN Tools	CADP	mCRL2	SAL	TLA+	UMC
Development Functionalities	Specification / Modeling	TEXT	GRAPH	TEXTIM	TEXT	TEXT	GRAPH	TEXTIM	GRAPH	TEXTIM	TEXT	TEXTIM	TEXT	TEXT
	Code Generation	NO	YES	NO	NO	YES	NO	NO	NO	YES	NO	NO	NO	NO
	Documentation / Report Generation	PARTIAL	YES	NO	PARTIAL	PARTIAL	PARTIAL	PARTIAL	NO	PARTIAL	PARTIAL	NO	NO	PARTIAL
	Requirements Traceability	NO	YES	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO
	Project Management	NO	YES	NO	YES	YES	NO	NO	NO	NO	NO	NO	NO	NO
Verification Functionalities	Simulation	TEXT	GRAPH	TEXT	MIX	NO	GRAPH	TEXT	GRAPH	TEXT	TEXT	TEXT	NO	TEXT
	Formal Verification	MC-L	MC-O	MC-L,MC-B	MC-L,MC-B,RF	TP	MC-L,RF	RF	MC-B	MC-B,RF	MC-B,RF	MC-L,TP	MC-L,TP	MC-B
	Large-scale Verification Technique	FLY,POR,PAR	BMC	BMC,SYM	SCT	SCT	SMC,SYM	COM,POR	BMC	COM,PAR	COM	PAR,SCT	SYM,SCT	FLY
	Model-based Testing	NO	YES	NO	YES	NO	YES	NO	NO	YES	NO	YES	NO	NO
Language Expressiveness	Non-determinism	INT	EXT	INT,EXT	INT,EXT	INT,EXT	INT,EXT	INT,EXT	INT	INT,EXT	INT,EXT	INT,EXT	INT	INT
	Concurrency	ASYNCH	NO	SYNCH	NO	NO	SYNCH	ASYNCH	ASYNCH	ASYNCH	ASYNCH	A/SYNCH	ASYNCH	A/SYNCH
	Timing Aspects	NO	YES	YES	NO	NO	YES	YES	YES	NO	YES	YES	NO	NO
	Stochastic or Probabilistic Aspects	NO	NO	NO	NO	NO	YES	NO	NO	NO	YES	NO	NO	NO
	Modularity of the Language	HIGH	HIGH	MEDIUM	LOW	LOW	MEDIUM	HIGH	HIGH	HIGH	HIGH	MEDIUM	MEDIUM	HIGH
	Supported Data Structures	BASIC	COMPLEX	COMPLEX	COMPLEX	COMPLEX	COMPLEX	COMPLEX	COMPLEX	COMPLEX	COMPLEX	COMPLEX	COMPLEX	COMPLEX
	Float Support	NO	YES	YES	NO	NO	YES	NO	NO	NO	NO	NO	NO	NO
Tool Flexibility	Backward Compatibility	LIKELY	LIKELY	LIKELY	LIKELY	MODERATE	LIKELY	MODERATE	LIKELY	LIKELY	LIKELY	MODERATE	MODERATE	MODERATE
	Standard Input Format	OPEN	PARTIAL	OPEN	OPEN	OPEN	PARTIAL	OPEN	PARTIAL	STANDARD	OPEN	OPEN	OPEN	STANDARD
	Import / Export vs. Other Tools	MEDIUM	LOW	MEDIUM	HIGH	MEDIUM	LOW	MEDIUM	MEDIUM	HIGH	HIGH	MEDIUM	LOW	MEDIUM
	Modularity of the Tool	LOW	HIGH	LOW	HIGH	MEDIUM	HIGH	LOW	LOW	HIGH	MEDIUM	LOW	LOW	MEDIUM
	Team Support	NO	NO	NO	NO	YES	NO	NO	NO	NO	NO	NO	NO	NO
Maturity	Industrial Diffusion	HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	MEDIUM	MEDIUM	MEDIUM	MEDIUM	LOW	MEDIUM	LOW
	Stage of Development	MATURE	MATURE	MATURE	MATURE	MATURE	MATURE	MATURE	MATURE	MATURE	MATURE	MATURE	MATURE	PROTOTYPE
Usability	Availability of Customer Support	PARTIAL	YES	PARTIAL	YES	YES	YES	PARTIAL	PARTIAL	PARTIAL	PARTIAL	PARTIAL	PARTIAL	PARTIAL
	Graphical User Interface	LIMITED	YES	NO	PARTIAL	PARTIAL	YES	LIMITED	PARTIAL	LIMITED	PARTIAL	NO	LIMITED	PARTIAL
	Mathematical Background	MEDIUM	BASIC	MEDIUM	MEDIUM	ADVANCED	MEDIUM	ADVANCED	MEDIUM	ADVANCED	ADVANCED	ADVANCED	ADVANCED	MEDIUM
	Quality of Documentation	GOOD	EXCELLENT	GOOD	GOOD	EXCELLENT	GOOD	EXCELLENT	GOOD	GOOD	GOOD	GOOD	GOOD	LIMITED
Company Constraints	Cost	FREE	PAY	MIX	FREE	FREE	MIX	MIX	FREE	MIX	FREE	FREE	FREE	FREE
	Supported Platforms	ALL	ALL	ALL	ALL	ALL	ALL	ALL	Windows	ALL	ALL	ALL	ALL	ALL
	Complexity of License Management	EASY	ADEQUATE	EASY	EASY	EASY	MODERATE	MODERATE	EASY	MODERATE	EASY	EASY	EASY	EASY
	Easy to Install	YES	YES	YES	YES	YES	YES	YES	YES	PARTIAL	YES	YES	YES	YES
Railway-specific Criteria	CENELEC Certification	NO	PARTIAL	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO
	Integration in the CENELEC Process	MEDIUM	YES	MEDIUM	YES	YES	MEDIUM	MEDIUM	MEDIUM	MEDIUM	LOW	LOW	LOW	MEDIUM
		SPIN	Simulink	nuXmv	ProB	Atelier B	UPPAAL	FDR4	CPN Tools	CADP	mCRL2	SAL	TLA+	UMC

System Usability Scale (SUS) evaluation



J. Brooke,
SUS: A 'quick and dirty' usability scale. Usability Evaluation in Industry (1996)
SUS: A retrospective. *Journal of Usability Studies* (2013)

Take-away messages

- Many of the formal tools lack support for development features and process-integration aspects
- Most of the formal tools are independent ecosystems, with unique, non-standard languages and specialised verification capabilities

Results

Take-away messages

- − Many of the formal tools lack support for development features and process-integration aspects
- − Most of the formal tools are independent ecosystems, with unique, non-standard languages and specialised verification capabilities
- \+ Formal tools are mature, as highly desired by the railway industry [iFM18,FM19]
- \+ Most usability aspects appear to be low in principle, but, when the formal tools are assessed by railway practitioners, usability is considered acceptable

[iFM18] D. Basile, M.H. ter Beek, et al., On the Industrial Uptake of Formal Methods in the Railway Domain: A Survey with Stakeholders @ iFM'18

Formal methods & tools in railways: Success stories

Well-known success stories

Using B, Simulink/Stateflow, NuSMV, SPIN, AADL, UPPAAL, ...

- Automatic Train Protection (ATP) system of Paris RER Line A
- Subway Speed Control System (SSCS) of the Calcutta subway
- Paris Metro Line 14 and its derivatives, like NY Canarsie line 1
- Alstom's U400 system (in use in $\pm$ 100 metro lines worldwide)
- Driverless Paris–Roissy Airport shuttle
- Metro control system of Rio de Janeiro
- ERTMS/ETCS standard, e.g. hybrid ERTMS/ETCS Level 3
- Movement Authority (MA) scenario of Chinese Train Control System Level 3 (CTCS-3)
- Autonomous Positioning System (APS) of Florence tramways
- ...

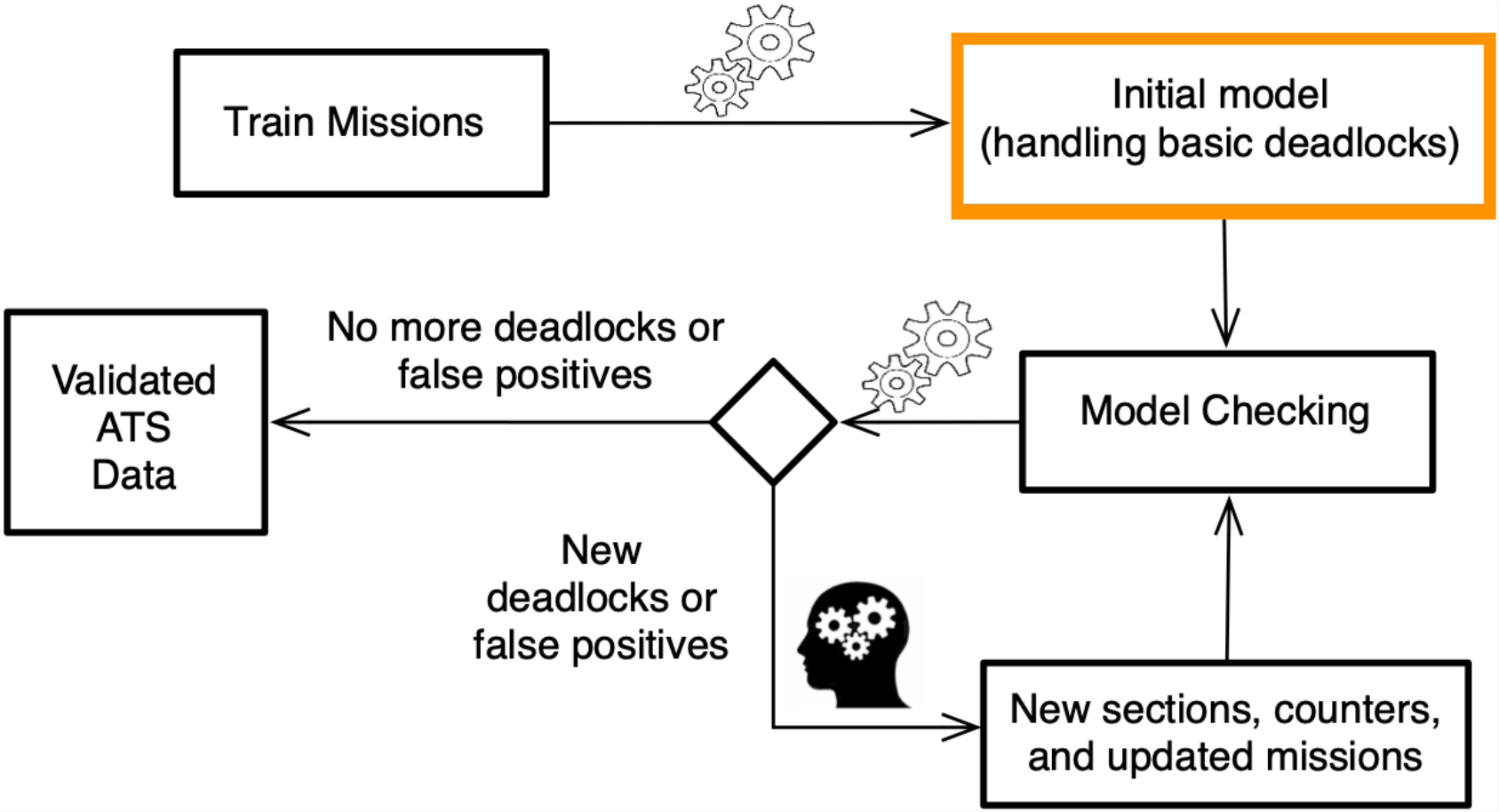
Some of our success stories

A taste of FMT's railway research in projects with industrial partners over the last 10 years:

- Deadlock avoidance in train scheduling
- Next generation railway signalling systems
- Synthesis of autonomous driving strategies
- Smart railway systems & stations of the future

Success story 1

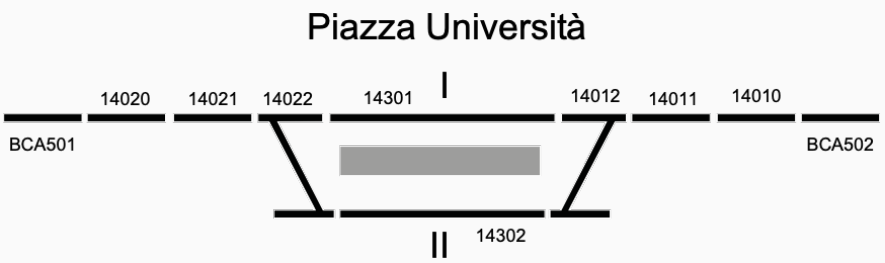
Automatic Train Supervision (ATS): deadlocks?



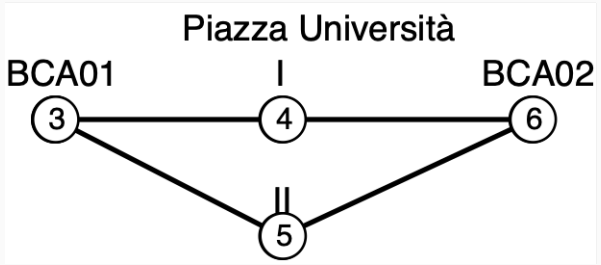
[NFM14] F. Mazzanti, G.O. Spagnolo & A. Ferrari, Designing a Deadlock-Free Train Scheduler: A Model Checking Approach @ NFM14

Model: choose appropriate level of abstraction!

Track circuit:

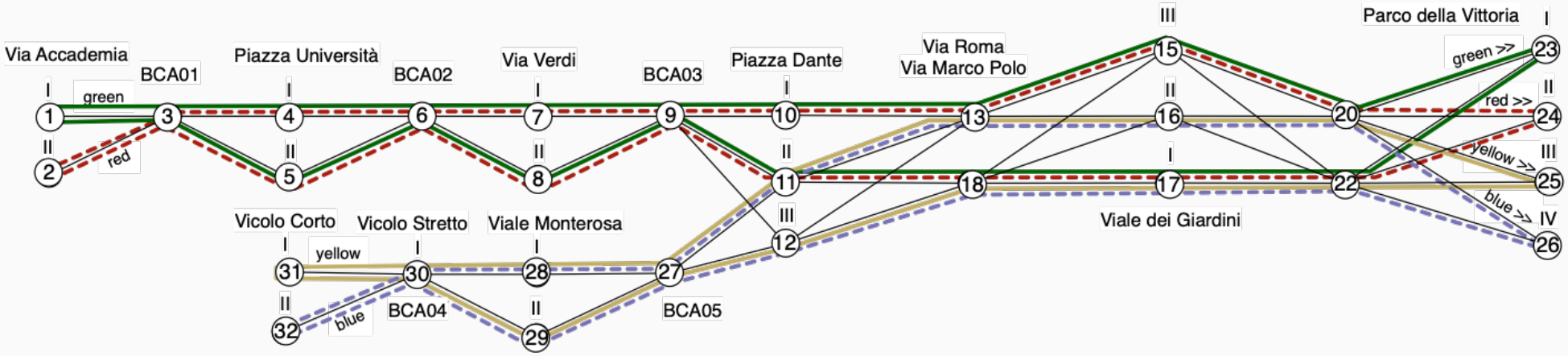


Itinerary:



Model: case study

Metro layout with 4 lines and 8 trains providing circular services:




[1,3,4,6,7,9,10,13,15,20,23,22,17,18,11,9,8,6,5,3,1]



[2,3,4,6,7,9,10,13,15,20,24,22,17,18,11,9,8,6,5,3,2]



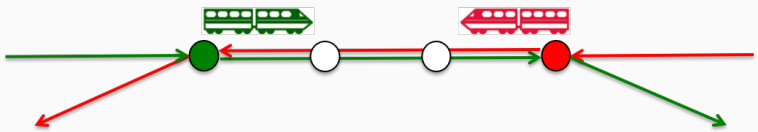
[31,30,28,27,11,13,15,20,25,22,17,18,12,27,29,30,31]



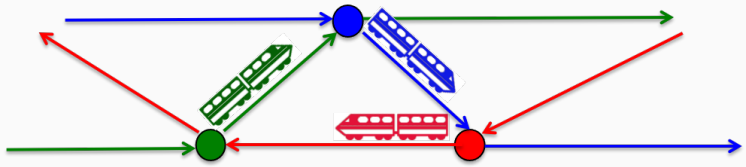
[32,30,28,27,11,13,15,20,26,22,17,18,12,27,29,30,32]


Model: identify deadlocks

Linear deadlock:

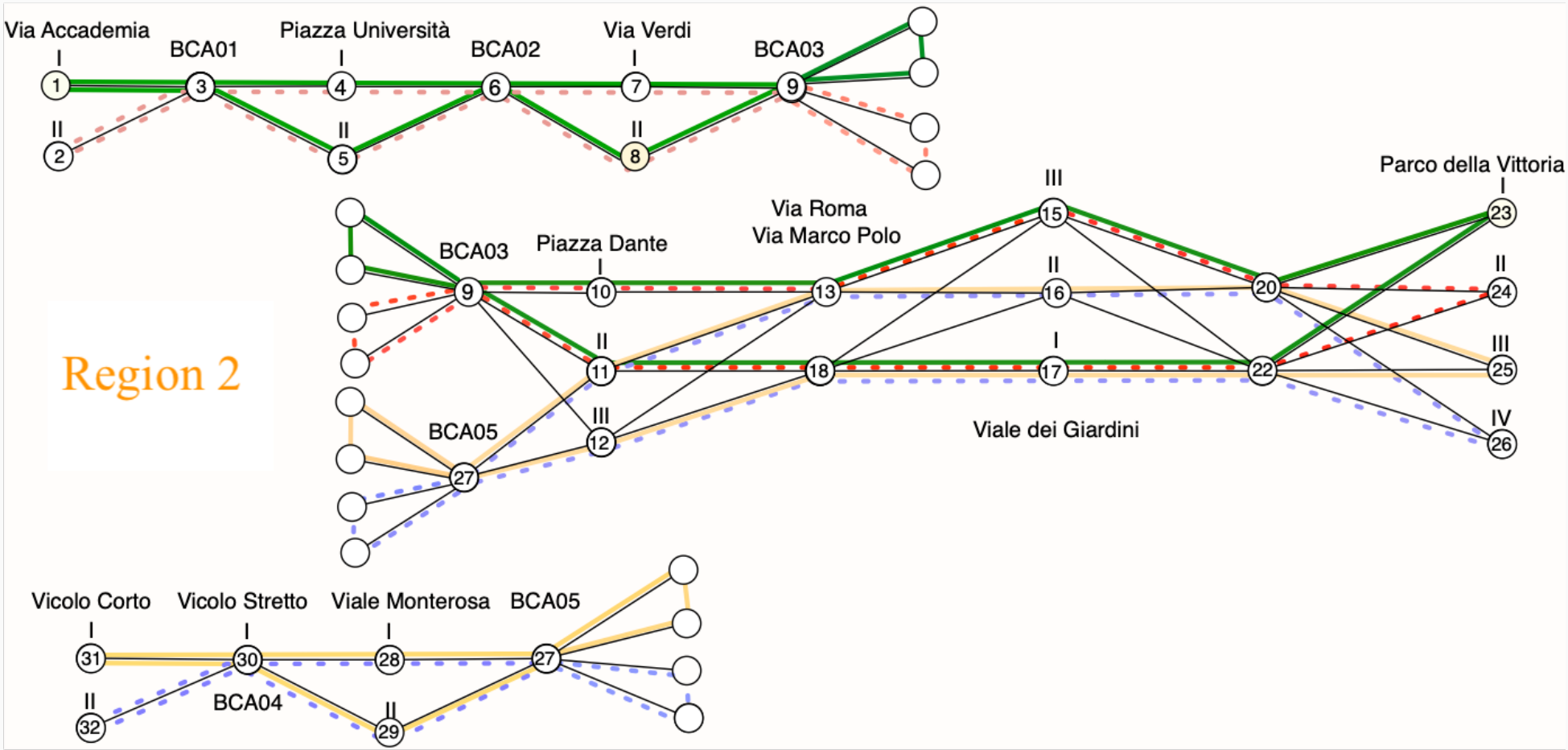


Circular deadlock:



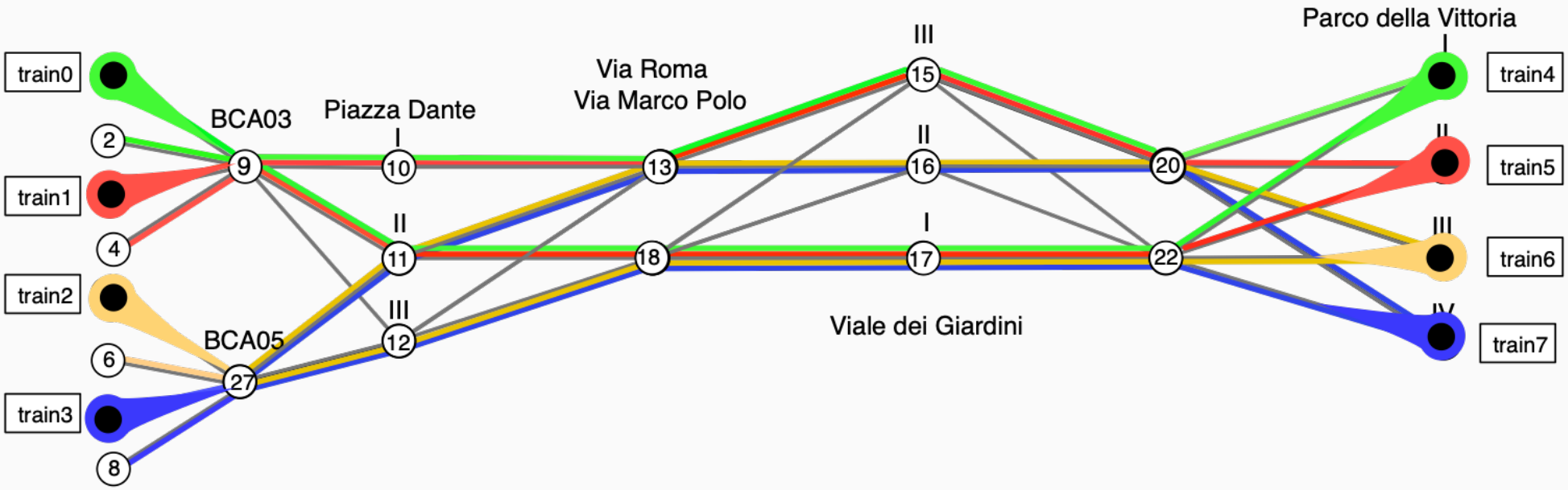
Divide et impera

Region 1

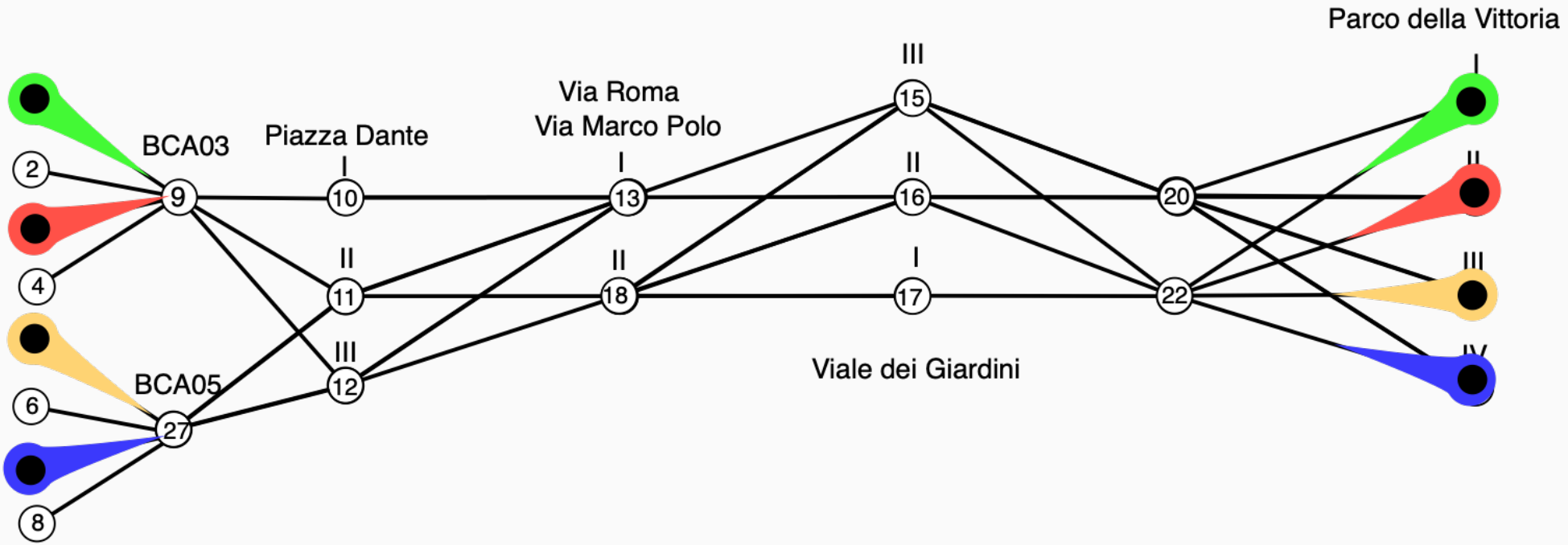


Region 3

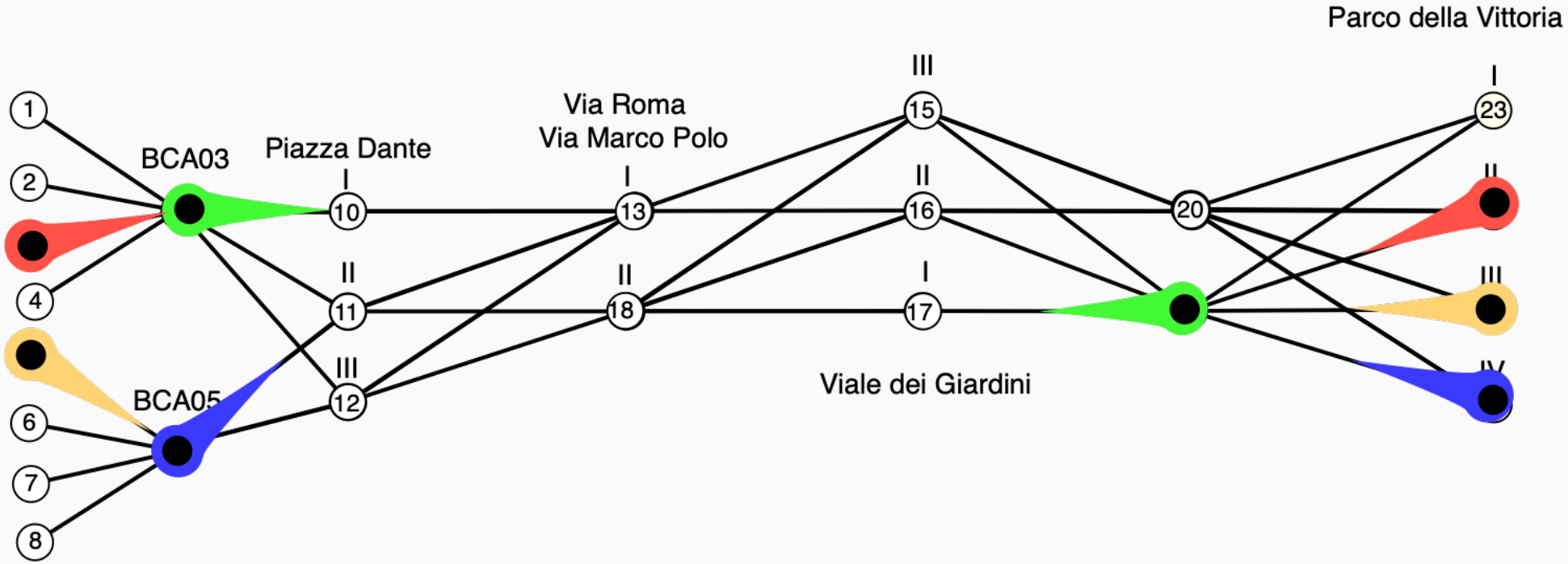
Region 2 model: 8.878.643 states



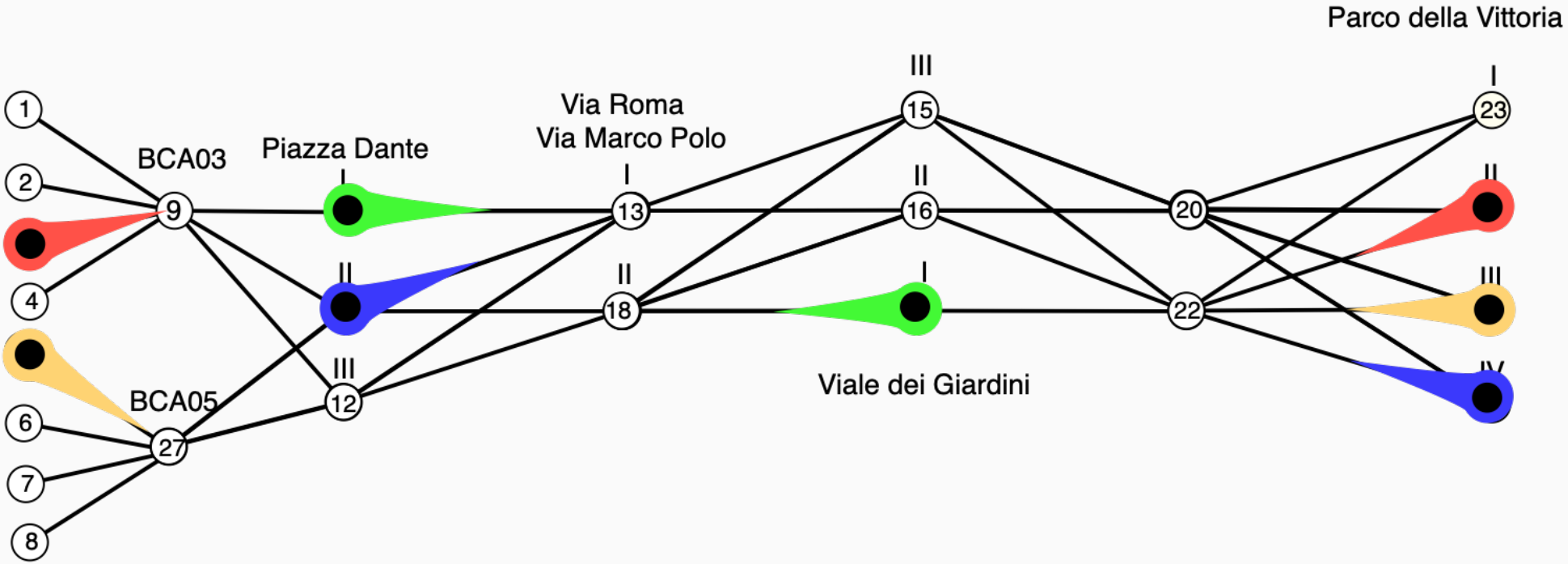
We must dispatch the trains without deadlocks . . .



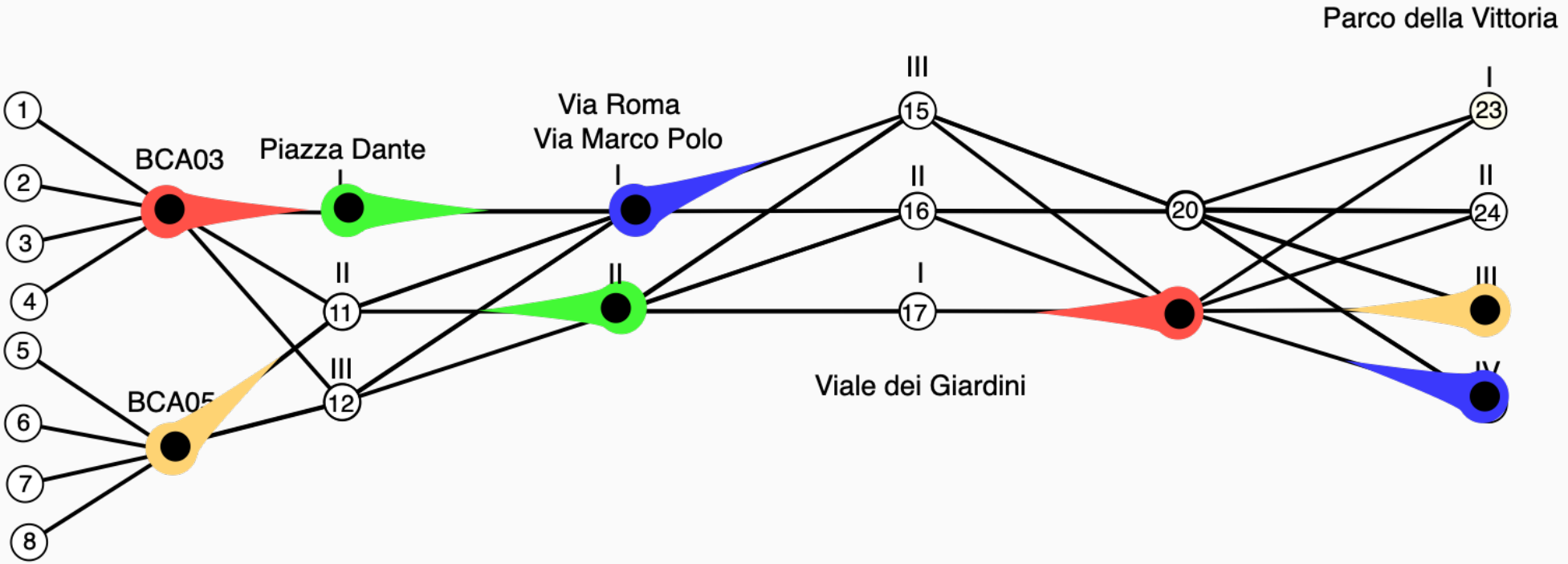
We must dispatch the trains without deadlocks . . .



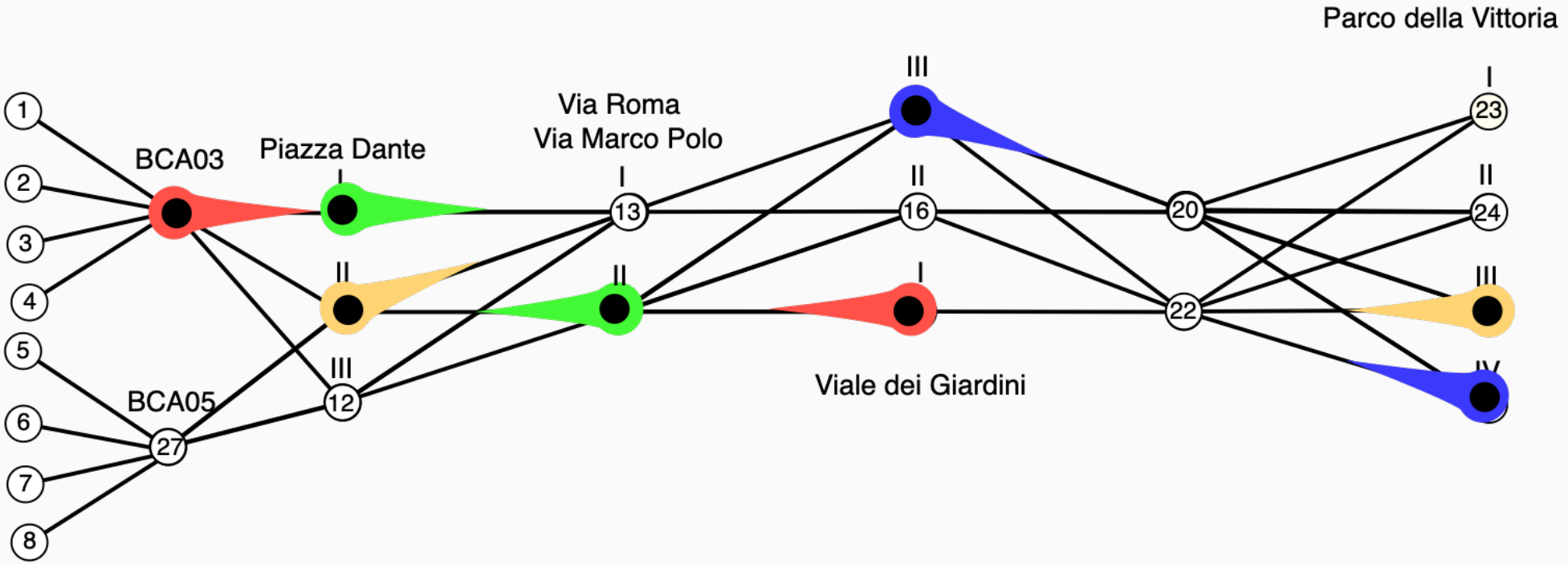
We must dispatch the trains without deadlocks . . .



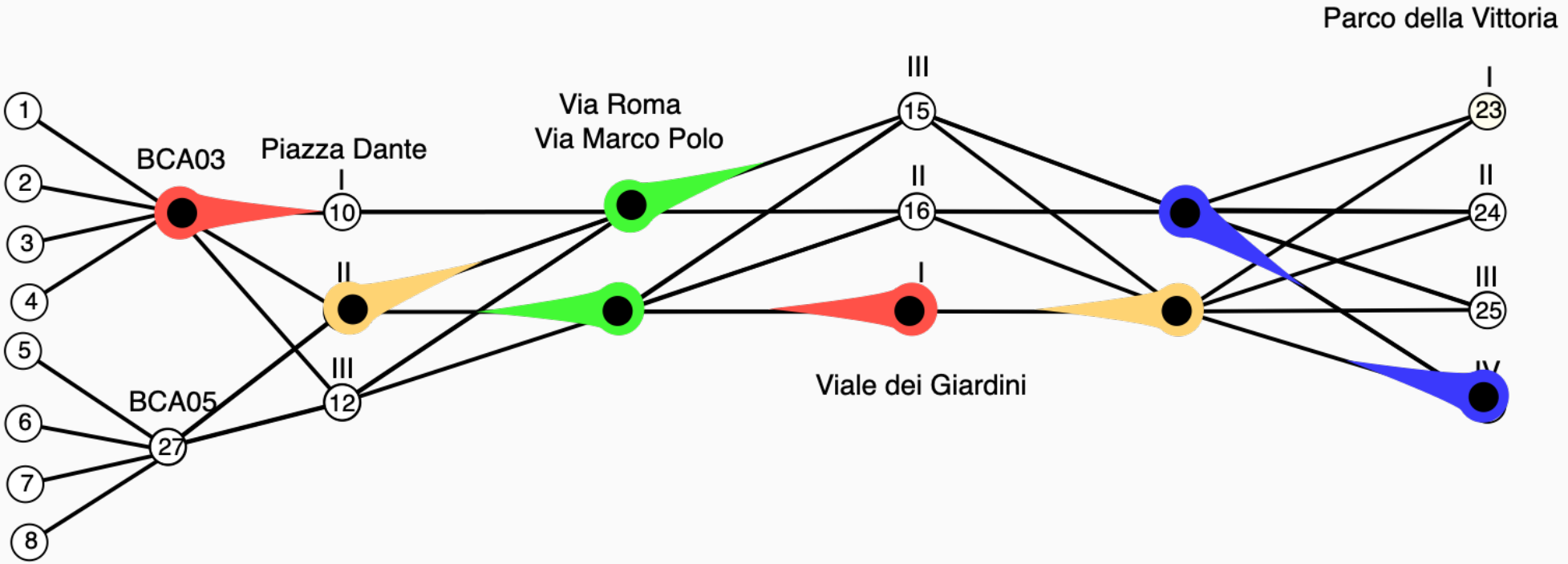
We must dispatch the trains without deadlocks . . .



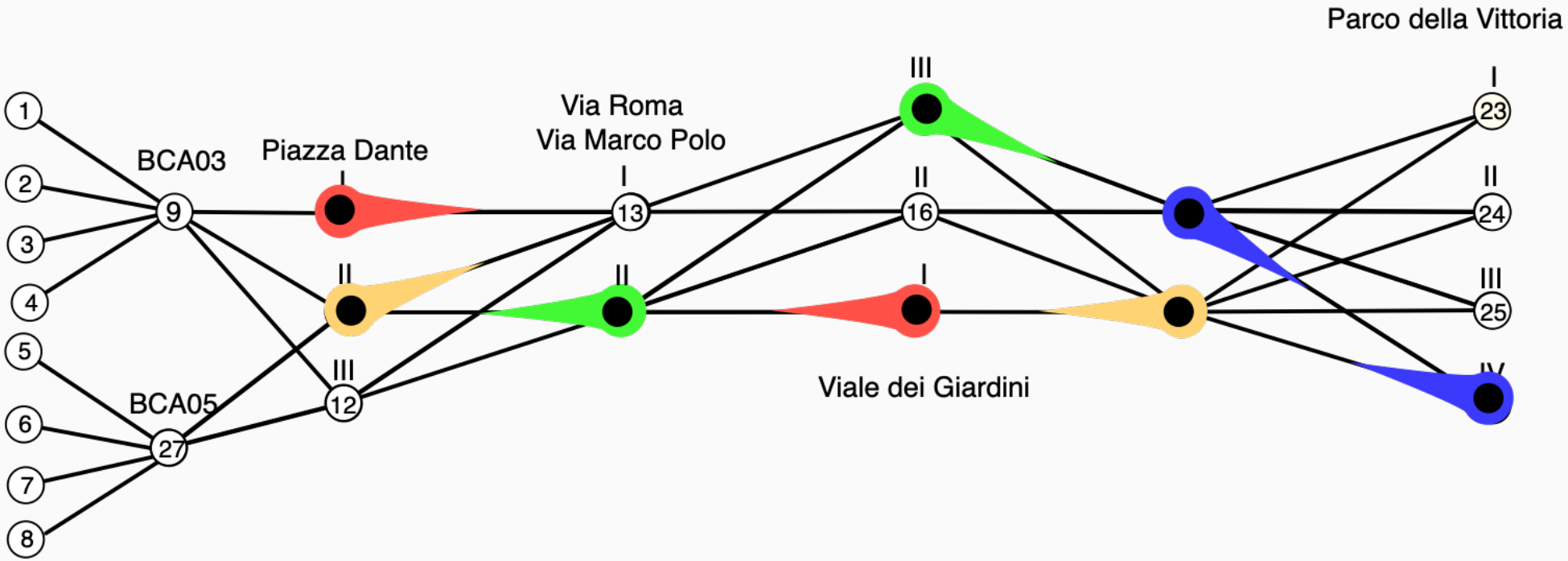
We must dispatch the trains without deadlocks . . .



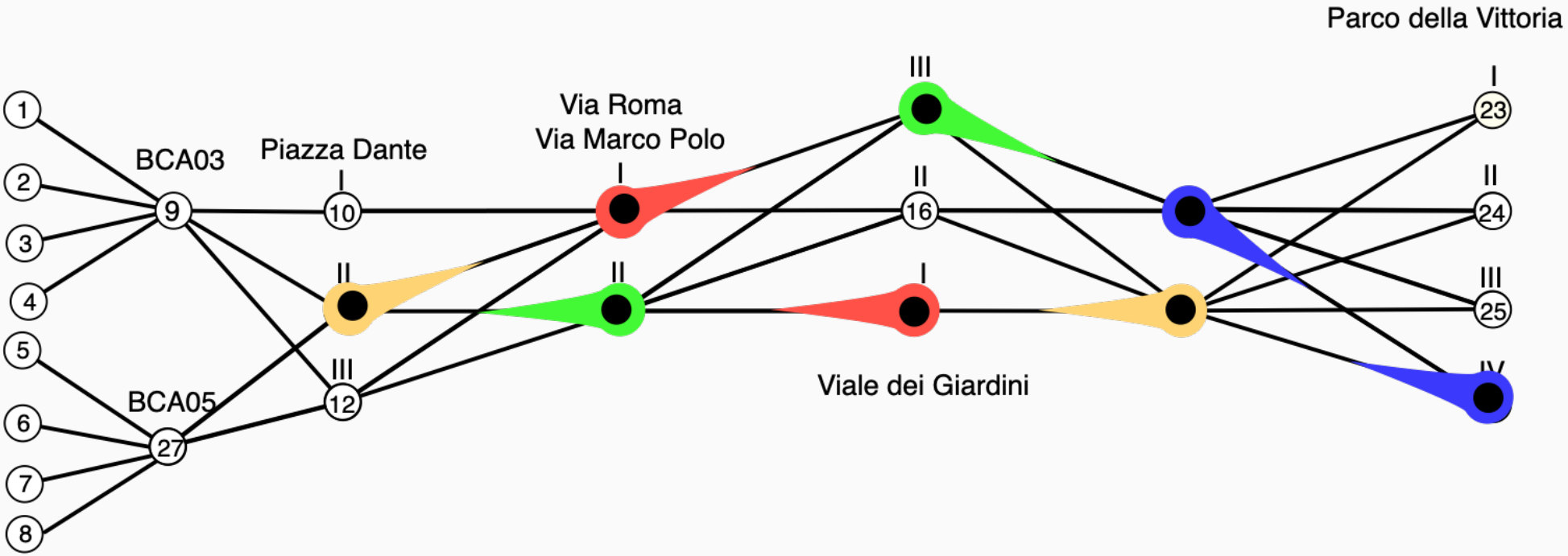
We must dispatch the trains without deadlocks . . .



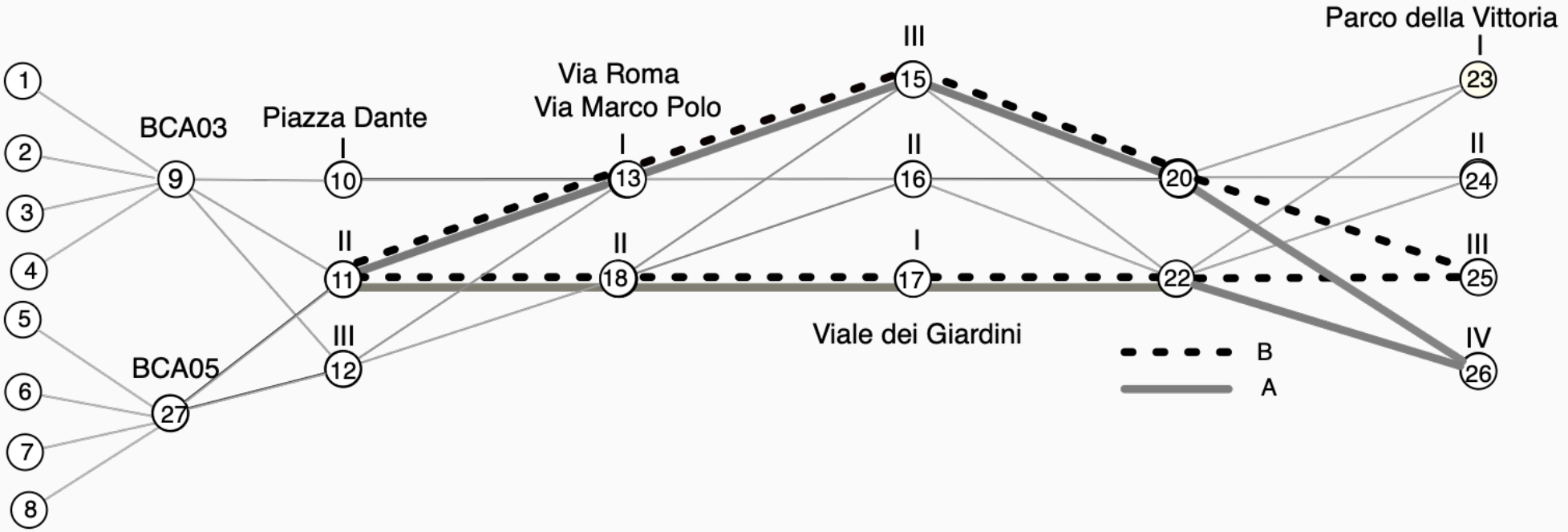
We must dispatch the trains without deadlocks . . .



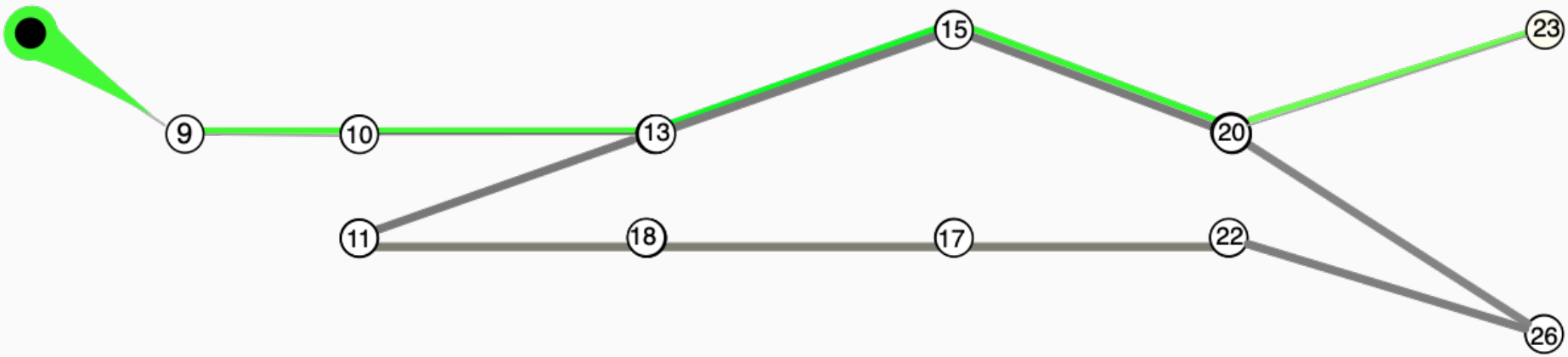
. . . I said without deadlocks!



Critical sections



Critical sections: control access



Section A

Counter for section $\mathrm{A} = RA(init == 1)$

Mission $T_0 = [1, 9, 10, 13, 15, 20, 23]$

T_0 operations on $RA = [0, 0, 0, +1, 0, -1, 0]$

Constraints for section $\mathrm{A} = RA \leq 7$

Results

Different tools require different models! ($\Rightarrow$ different performance)

#	Tool	Description	Performance
1	CADP	Parallel without shared memory / Sequential	29s
2	CPN Tools	Parallel without shared memory	N/A
3	FDR4	Parallel without shared memory / Sequential	15s – 20m
4	mCRL2	Parallel without shared memory / Sequential	2m – 19m
5	ProB	Sequential	32m
6	NuSMV/nuXmv	Sequential	3s – 43s
7	SPIN	Sequential	13s – 47s
8	TLA+	Sequential	3m
9	UMC	Sequential	38s – 86s
10	UPPAAL	Parallel with shared memory / Sequential	16s

Note: many modelling languages are textual (e.g., process algebras)

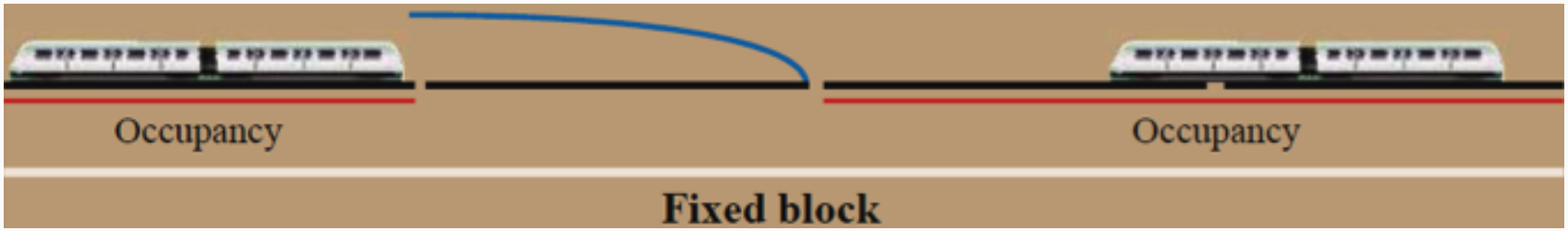
[MARS18] F. Mazzanti & A. Ferrari, Ten Diverse Formal Models for a CBTC Automatic Train Supervision System @ MARS'18

Success story 2

Next generation railway signalling systems

Current ERTMS / ETCS signalling systems max. level 2:

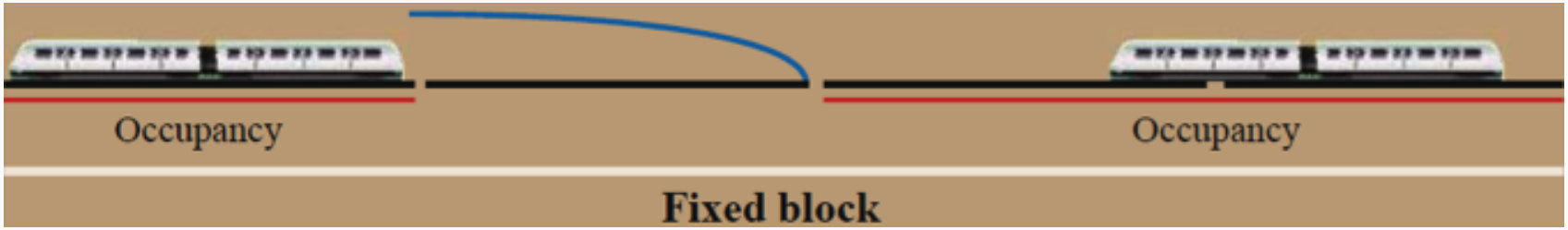
- fixed blocks: based on line's speed limit, train's speed/braking, . . . , thus faster trains imply longer blocks imply lower track occupancy
- trackside equipment for train positioning, with costly maintenance



Next generation railway signalling systems

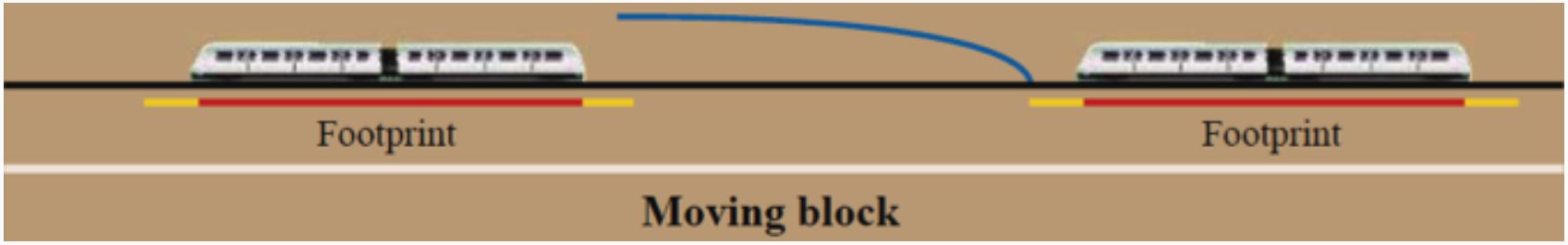
Current ERTMS / ETCS signalling systems max. level 2:

- fixed blocks: based on line's speed limit, train's speed/braking, . . . , thus faster trains imply longer blocks imply lower track occupancy
- trackside equipment for train positioning, with costly maintenance



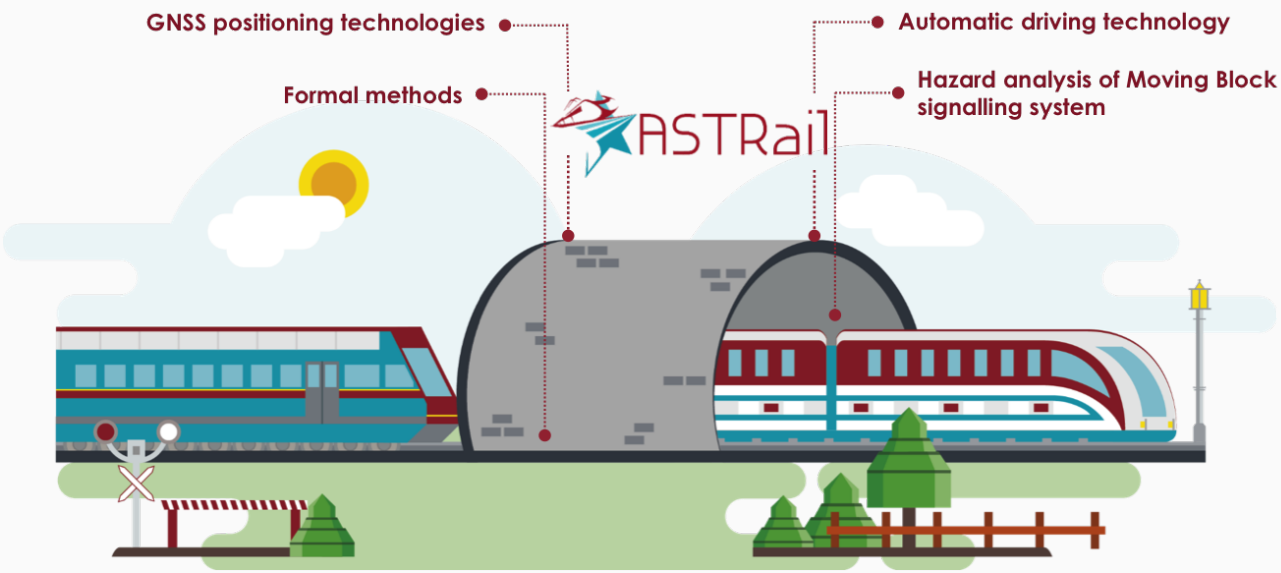
Next generation railway signalling systems from level 3:

- moving blocks: safe zone based on rear position of train ahead, thus reducing trains' headways, in principle to braking distance
- onboard odometry for train positioning (no trackside equipment)



Safety analysis of moving block signalling system

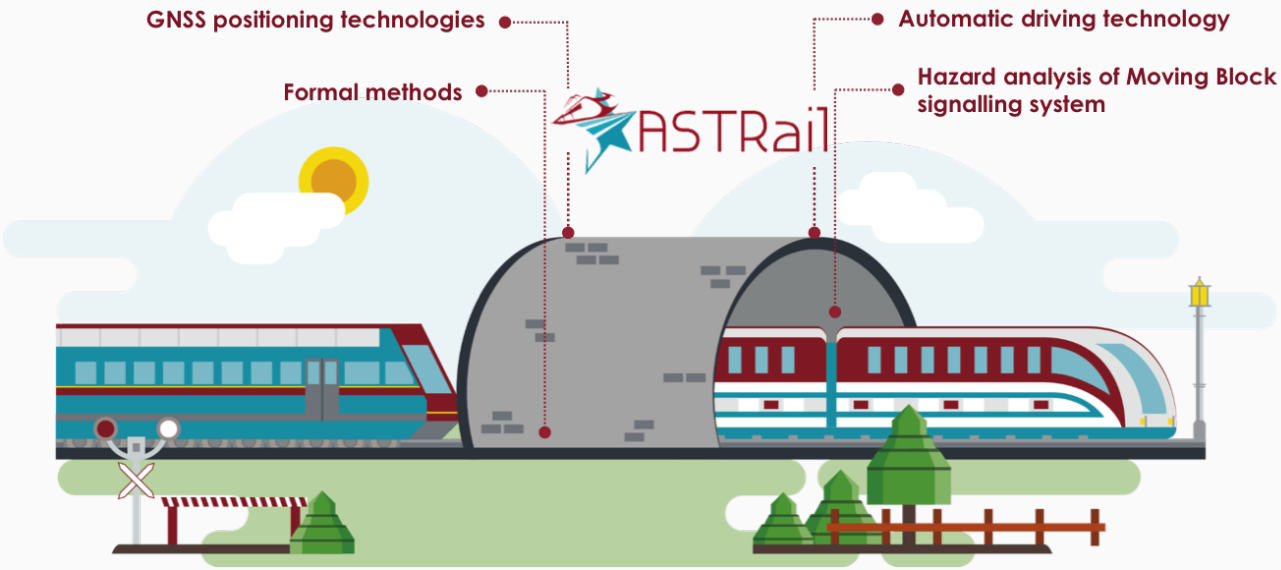
Challenge: effective, precise moving block signalling systems by GNSS-based satellite positioning, leveraging on an integrated solution for signal outages (e.g. tunnels) and multipath interference in dense urban areas



Safety analysis of moving block signalling system

Challenge: effective, precise moving block signalling systems by GNSS-based satellite positioning, leveraging on an integrated solution for signal outages (e.g. tunnels) and multipath interference in dense urban areas

Aim: analyse suitability of formal methods in transitioning to the next generation of ERTMS/ETCS railway signalling systems, with satellite-based positioning, moving block distancing, and automatic driving



Safety analysis of moving block signalling system

Challenge: effective, precise moving block signalling systems by GNSS-based satellite positioning, leveraging on an integrated solution for signal outages (e.g. tunnels) and multipath interference in dense urban areas

Aim: analyse suitability of formal methods in transitioning to the next generation of ERTMS/ETCS railway signalling systems, with satellite-based positioning, moving block distancing, and automatic driving

[STTT22] D. Basile, M.H. ter Beek, A. Ferrari & A. Legay, Exploring the ERTMS/ETCS full moving block specification: An experience with formal methods. *International Journal on Software Tools for Technology Transfer* (2022)

[FORTE20] D. Basile, M.H. ter Beek & A. Legay, Strategy Synthesis for Autonomous Driving in a Moving Block Railway System with Uppaal Stratego @ FORTE'20

[FMICS19] D. Basile, M.H. ter Beek, A. Ferrari & A. Legay, Modelling and Analysing ERTMS L3 Moving Block Railway Signalling with Simulink and UPPAAL SMC @ FMICS'19

[ISoLA18] D. Basile, M.H. ter Beek & V. Ciancia, Statistical Model Checking of a Moving Block Railway Signalling Scenario with Uppaal SMC @ ISoLA'18

Main components L3 moving block signalling system

OBU train's onboard unit measures the train's current speed and verifies the train's integrity

LU train's localisation unit uses a GNSS-based positioning system to determine the train's location

Main components L3 moving block signalling system

OBU train's onboard unit measures the train's current speed and verifies the train's integrity

LU train's localisation unit uses a GNSS-based positioning system to determine the train's location

RBC wayside radio block centre communicates continuously with OBU and LU

- receives data regarding the train's position and the train's integrity from the train
- sends speed restrictions, route configurations, and MAs (movement authorities) to the train
- computes MAs by communicating with neighbouring RBCs and with a Route Management System (RMS) for positions of switches and other trains (head and tail position)

Main components L3 moving block signalling system

OBU train's onboard unit measures the train's current speed and verifies the train's integrity

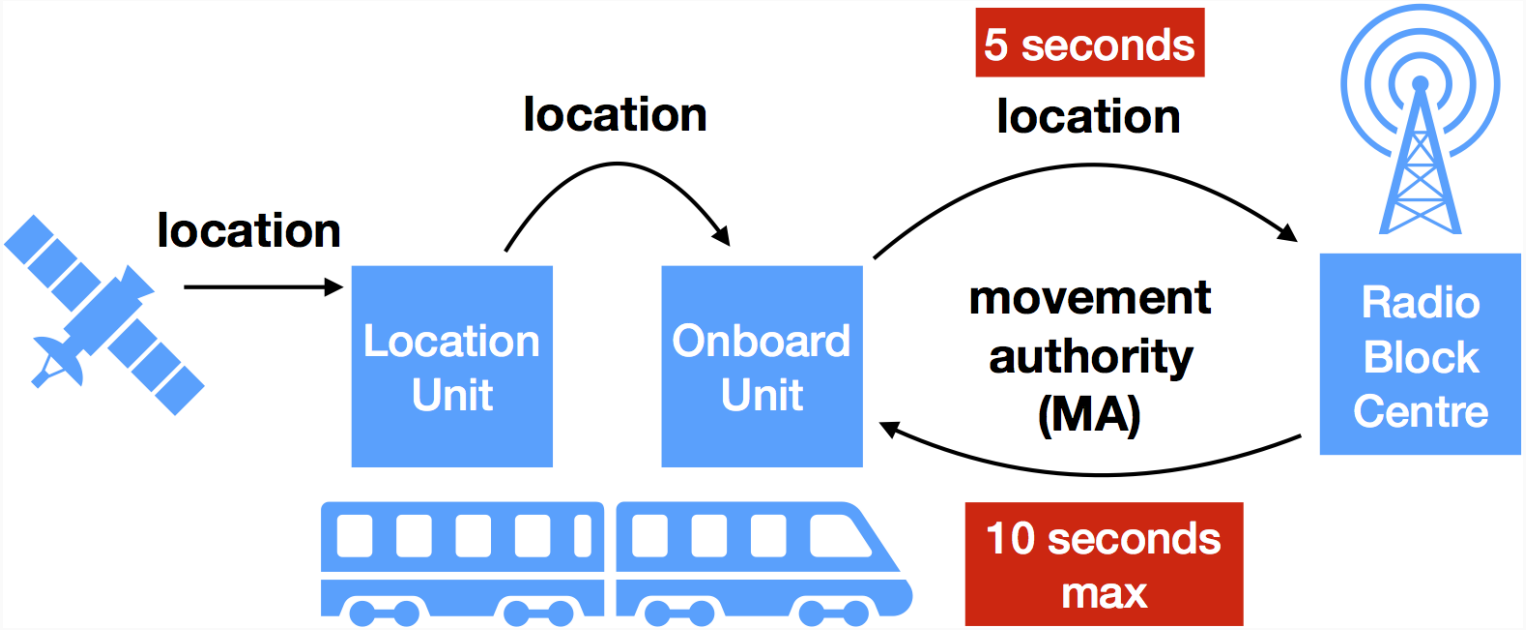
LU train's localisation unit uses a GNSS-based positioning system to determine the train's location

RBC wayside radio block centre communicates continuously with OBU and LU

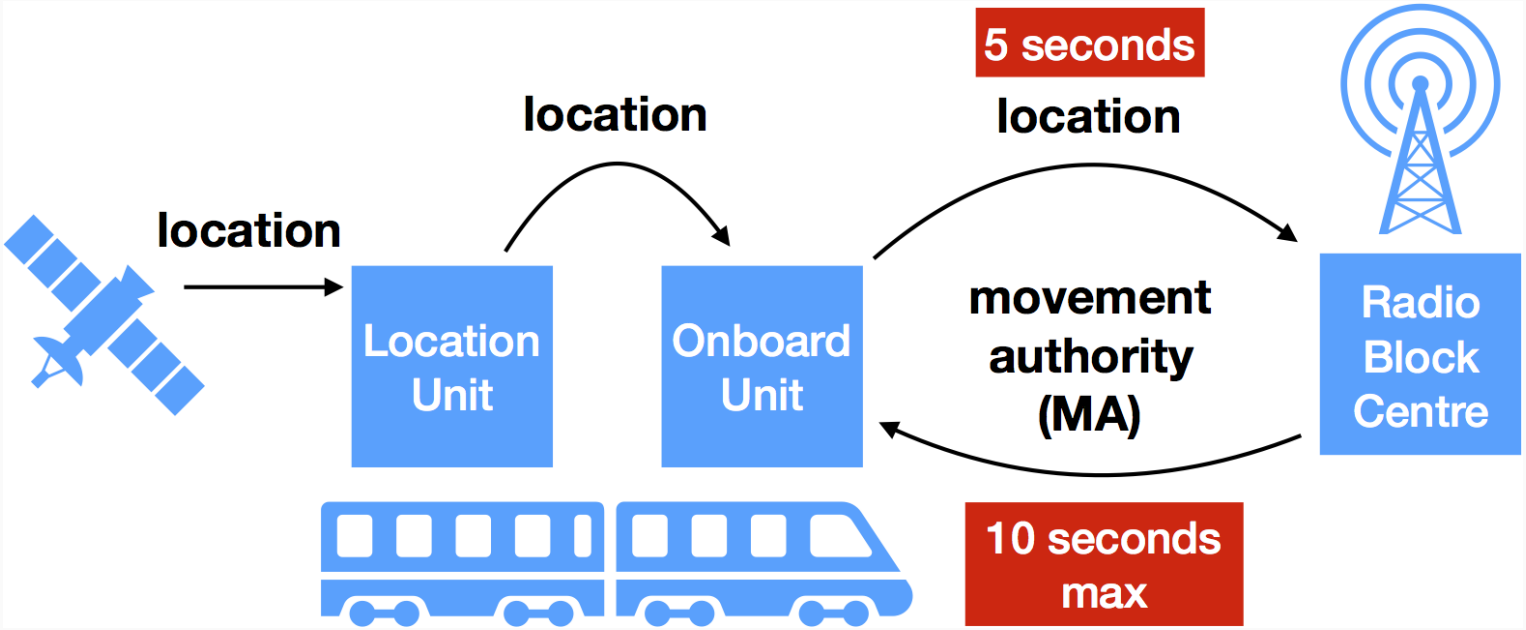
- receives data regarding the train's position and the train's integrity from the train
- sends speed restrictions, route configurations, and MAs (movement authorities) to the train
- computes MAs by communicating with neighbouring RBCs and with a Route Management System (RMS) for positions of switches and other trains (head and tail position)

Model abstraction: assume a train communicates with one RBC, based on seamless handover going from one RBC supervision area to the next

Moving block system architecture



From semi-formal models to formal models



Input: Real-Time UML (RT UML) and Simulink models obtained from/upon requirements elicitation and refinement with industrial partners

Output: UPPAAL SMC model

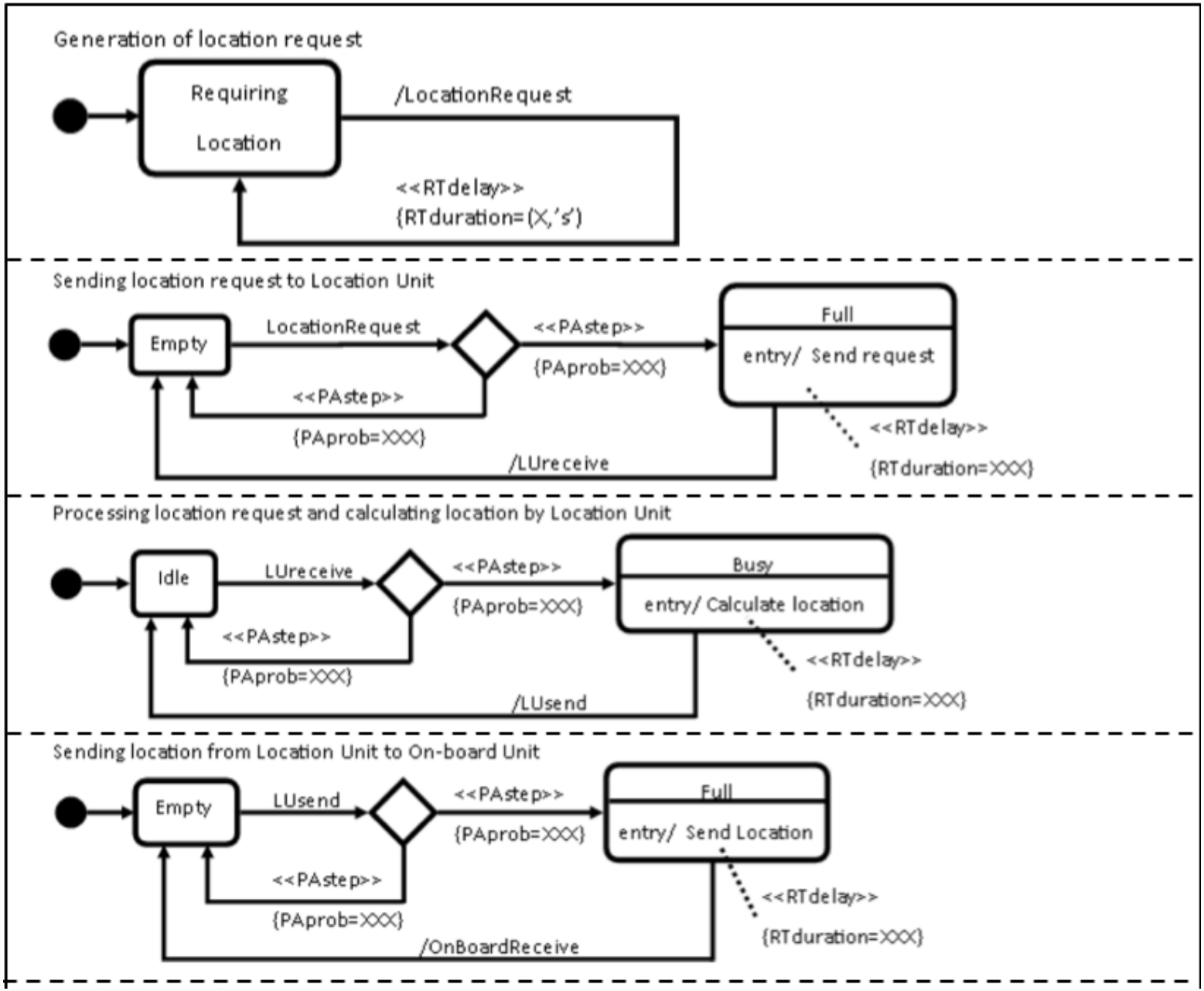
- capable of natively accommodating both real-time and probabilistic aspects
- ± UML state machine diagrams, easing understanding by industrial partners

Model transformation 1: RT UML → UPPAAL

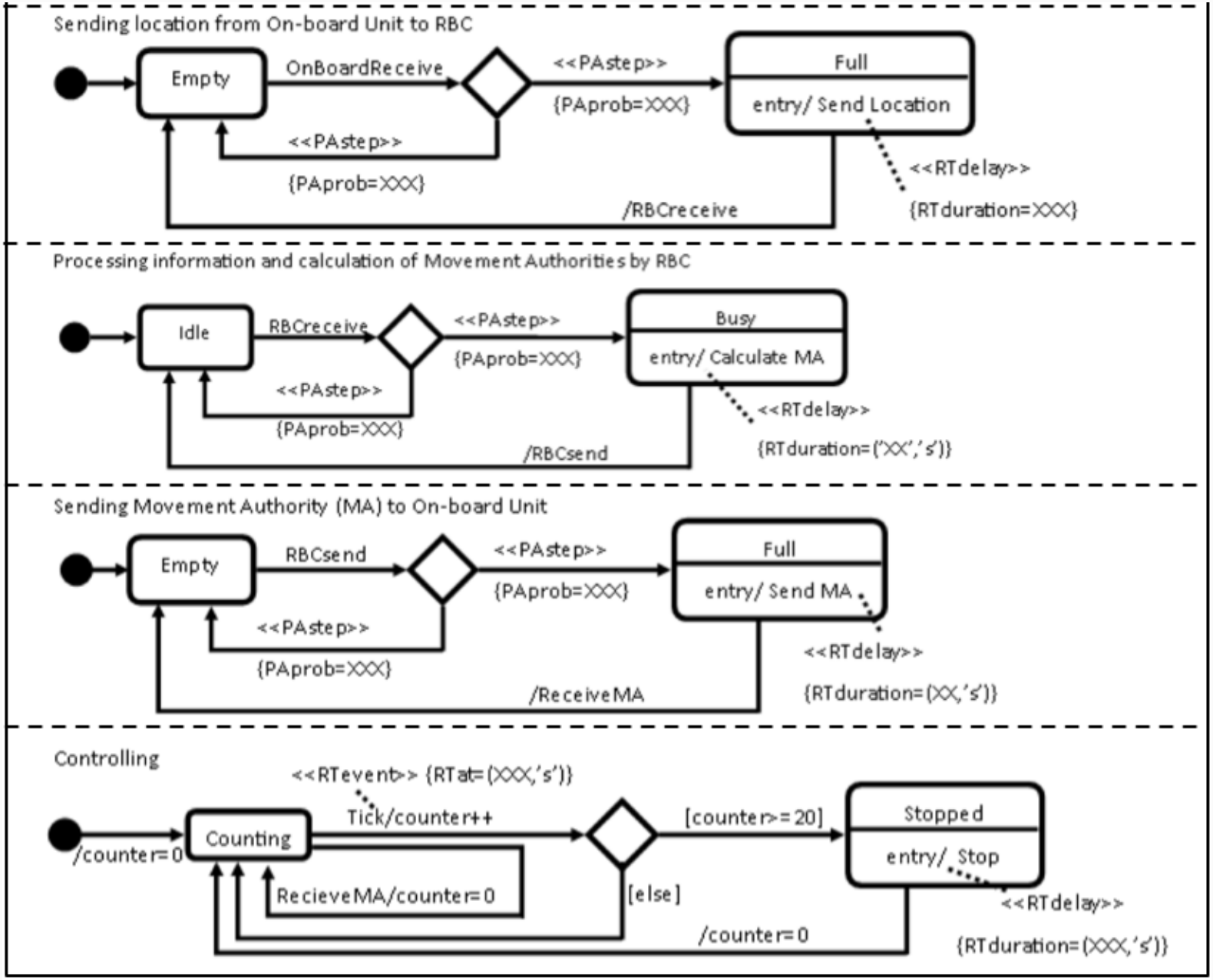
From RT UML state machine diagrams to stochastic timed automata

- each parallel region of RT UML model translated into a separate automaton
- (pseudo) states and (probabilistic) transitions are in 1-1 correspondence
- failure probabilities initially set to placeholder value 10^{-5} (industry tuning)
- guards and triggers modelled as input and output broadcast channels, thus:
  - synchronous communication, discarding messages if receiver not ready to receive
  - fresh MA sent by RBC to OBU will supersede older MA if latter was not yet received
- straightforward, except some time-related modelling choices cleared with partners
  - timed events `RTat` of stereotype `<<RTevent>>`, used to trigger transitions based on event's timing information, modelled as invariant conditions and clock guards, forcing transitions to be executed when the precise moment in time is reached
  - probabilistic delayed events `RTduration` of stereotype `<<RTdelay>>`, used to add durations to actions/transitions, modelled as probabilistic delays: when an action/transition is enabled, the time at which it is fired is probabilistically distributed
- failure probabilities and rates of probabilistic distributions based on industry input

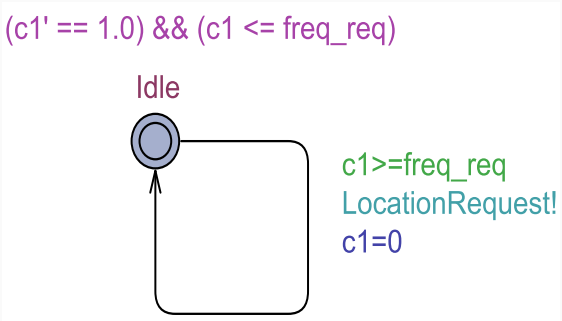
RT UML models from industry (1/2)



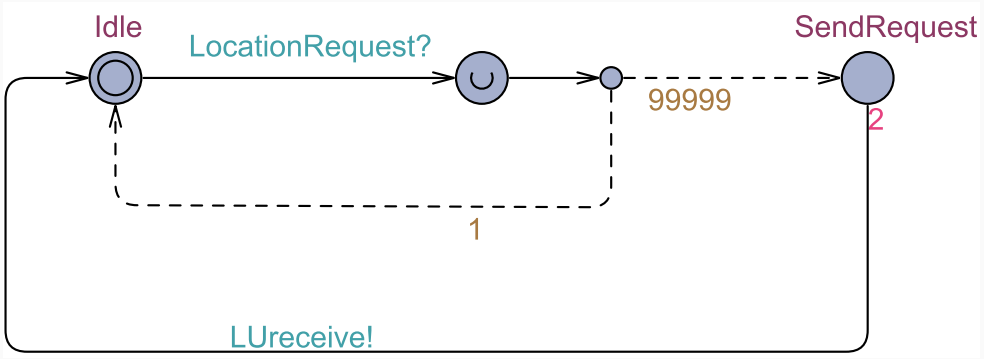
RT UML models from industry (2/2)



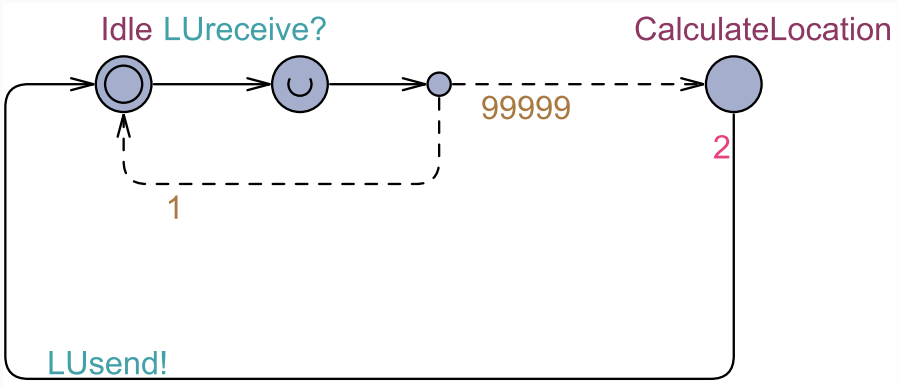
UPPAAL model of moving block signalling scenario (1/2)



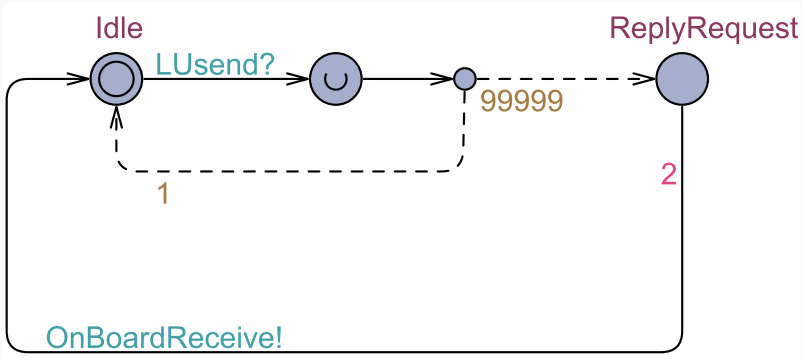
Generate location request



Send location request



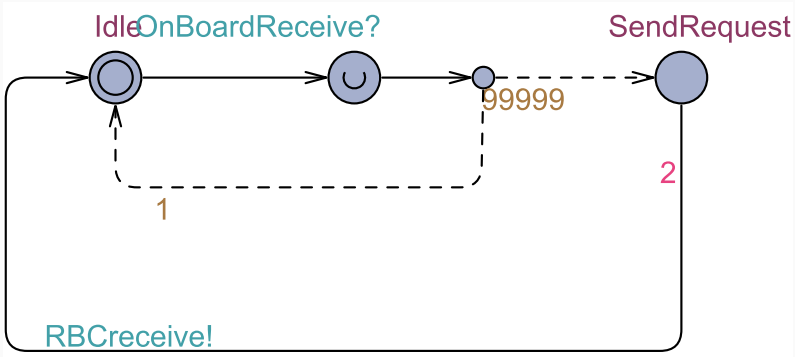
Calculate location



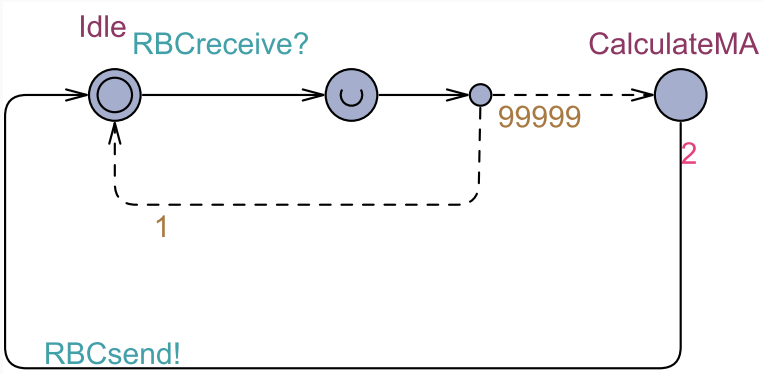
Send location

Industrial partners: `freq_req` = 5 sec., initial value clock `c1` is `freq_req`

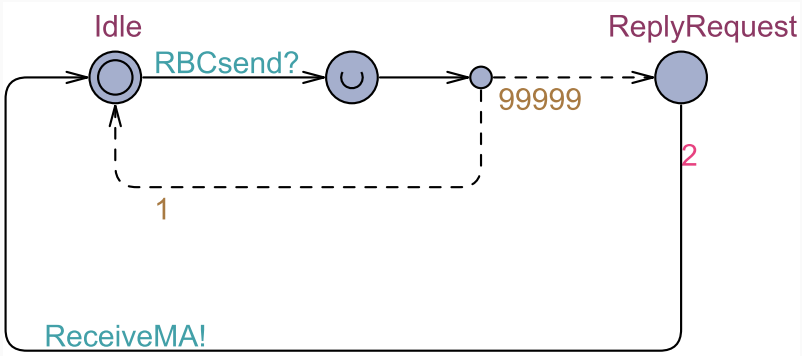
UPPAAL model of moving block signalling scenario (2/2)



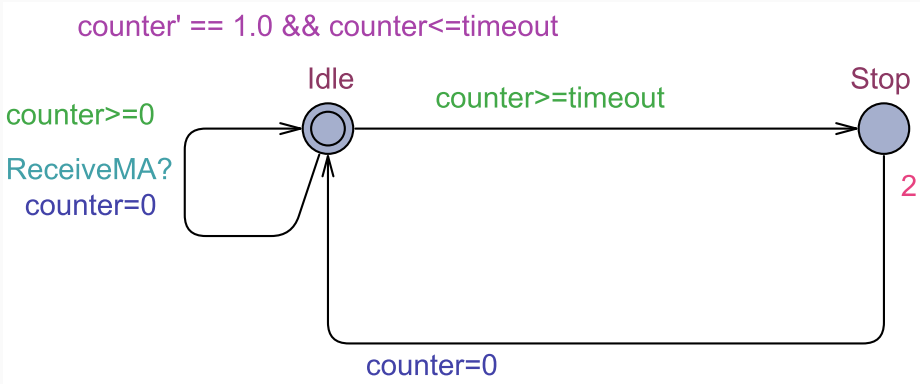
Send MA request



Calculate MA



Send MA



Control MA freshness

Industrial partners: `timeout` $= 3 \times$ `freq_req`, initial value clock `counter` is 5

Preliminary hazard analysis by industrial partners

Goal: evaluate safety level of a moving block signalling system

Procedure: identify and analyse hazards (e.g. GNSS-related errors, communication failures, faulty states)

- risk assessment: probability of occurrence of a hazard and severity of its consequences
- risk qualifying according to CENELEC EN 50126 standard (RAMS: Reliability, Availability, Maintainability and Safety)

Outcome: hazard log

Examples from the hazard log

Requirements:

"Communication between RBC and OBU must be safe and continuously supervised, if the connection is lost an alarm must be triggered."

"OBU device must be SIL 4 device. Once OBU receives the alarm [. . .] it must immediately send an alarm to RBC."

Mitigation: "In case of communication loss enter in safe state mode."

Examples from the hazard log

Requirements:

"Communication between RBC and OBU must be safe and continuously supervised, if the connection is lost an alarm must be triggered."

"OBU device must be SIL 4 device. Once OBU receives the alarm [. . .] it must immediately send an alarm to RBC."

Mitigation: "In case of communication loss enter in safe state mode."

Safety Related Application Conditions:

"If train position cannot be received within the maximum time limit, the OBU shall generate an alarm and must transit to degraded mode."

"If Train Integrity cannot be confirmed within the maximum time limit, the train shall be stopped."

Analyses with UPPAAL SMC

1 It must always be the case that eventually either a MA is received or the train enters a safe state `Stop`:

$$A \Diamond (\texttt{ReplyMA.ReplyRequest} \parallel \texttt{Controlling.Stop})$$

UPPAAL SMC reports that this (reachability) property (expressed in CTL) holds

Analyses with UPPAAL SMC

1 It must always be the case that eventually either a MA is received or the train enters a safe state `Stop`:

$$A \Diamond (\texttt{ReplyMA.ReplyRequest} \parallel \texttt{Controlling.Stop})$$

UPPAAL SMC reports that this (reachability) property (expressed in CTL) holds

2 Probability that the train enters a safe state `Stop` upon a `timeout`:

$$\mathbb{P}_M(\Diamond_{\leq(\texttt{timeout})}\, \texttt{Controlling.Stop})$$

UPPAAL SMC reports that this probability is in interval [0,9.99994e-005], with confidence 0.995 and obtained from 59912 runs in ± 5 min.

UPPAAL SMC: evaluating the freshness of the MA

Requirements: OBU attempts for three times to compute the train's location and receive the MA

Model: first attempt at time 0, after which OBU attempts again each 5 sec. until timeout at time 15

UPPAAL SMC: evaluating the freshness of the MA

Requirements: OBU attempts for three times to compute the train's location and receive the MA

Model: first attempt at time 0, after which OBU attempts again each 5 sec. until timeout at time 15

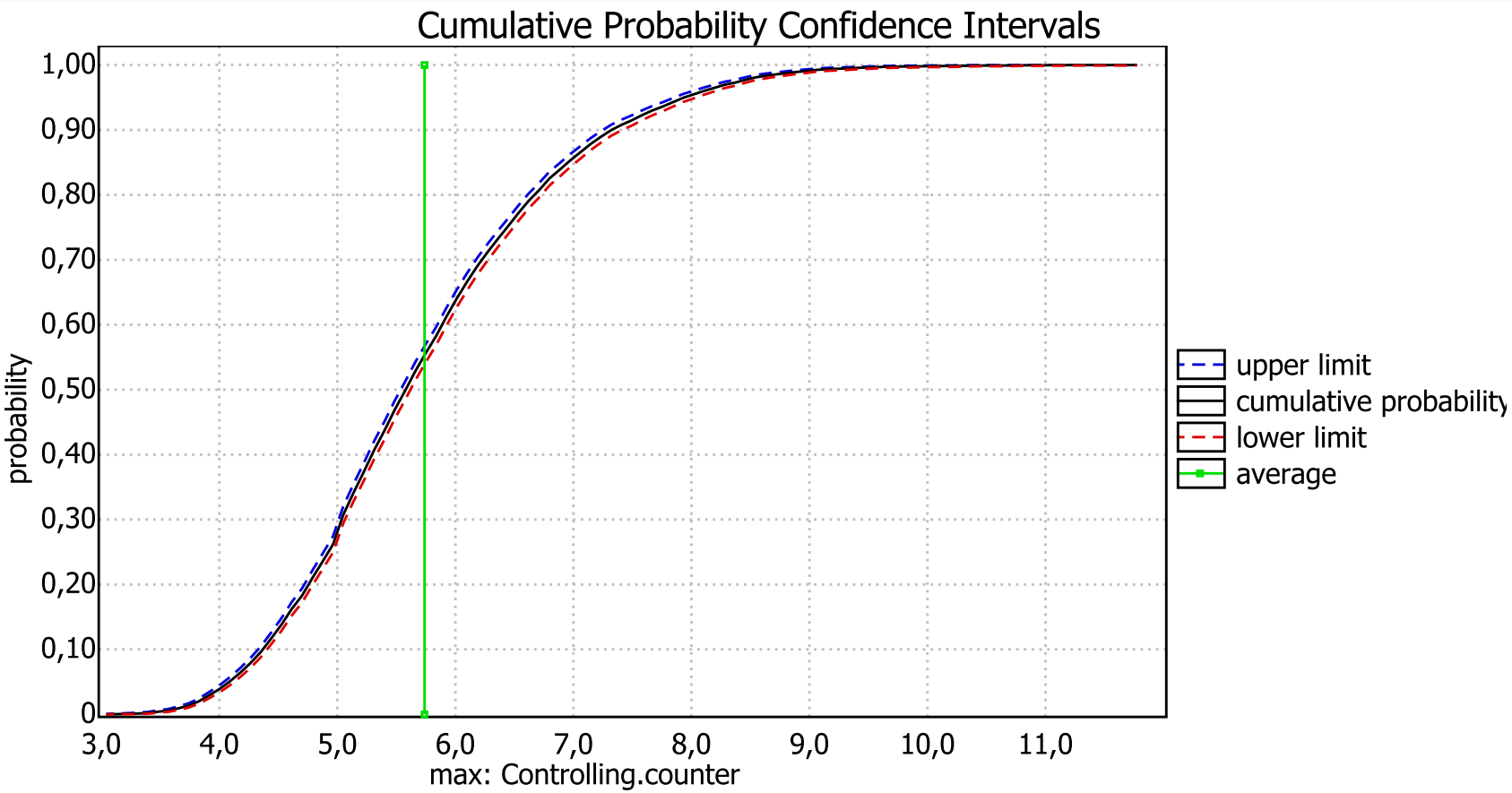
Goal: which of the three attempts has higher probability of success?

$$E[\leq \texttt{timeout}; 10000](\mathit{max} : \texttt{Controlling.counter})$$

This evaluation computes in the interval of time of `timeout` (i.e. 15 sec.) the average of the maximum value of clock `counter`, using 10000 runs ; Since `counter` is reset each time a new MA is received, its average value is the average time in which a new MA is received

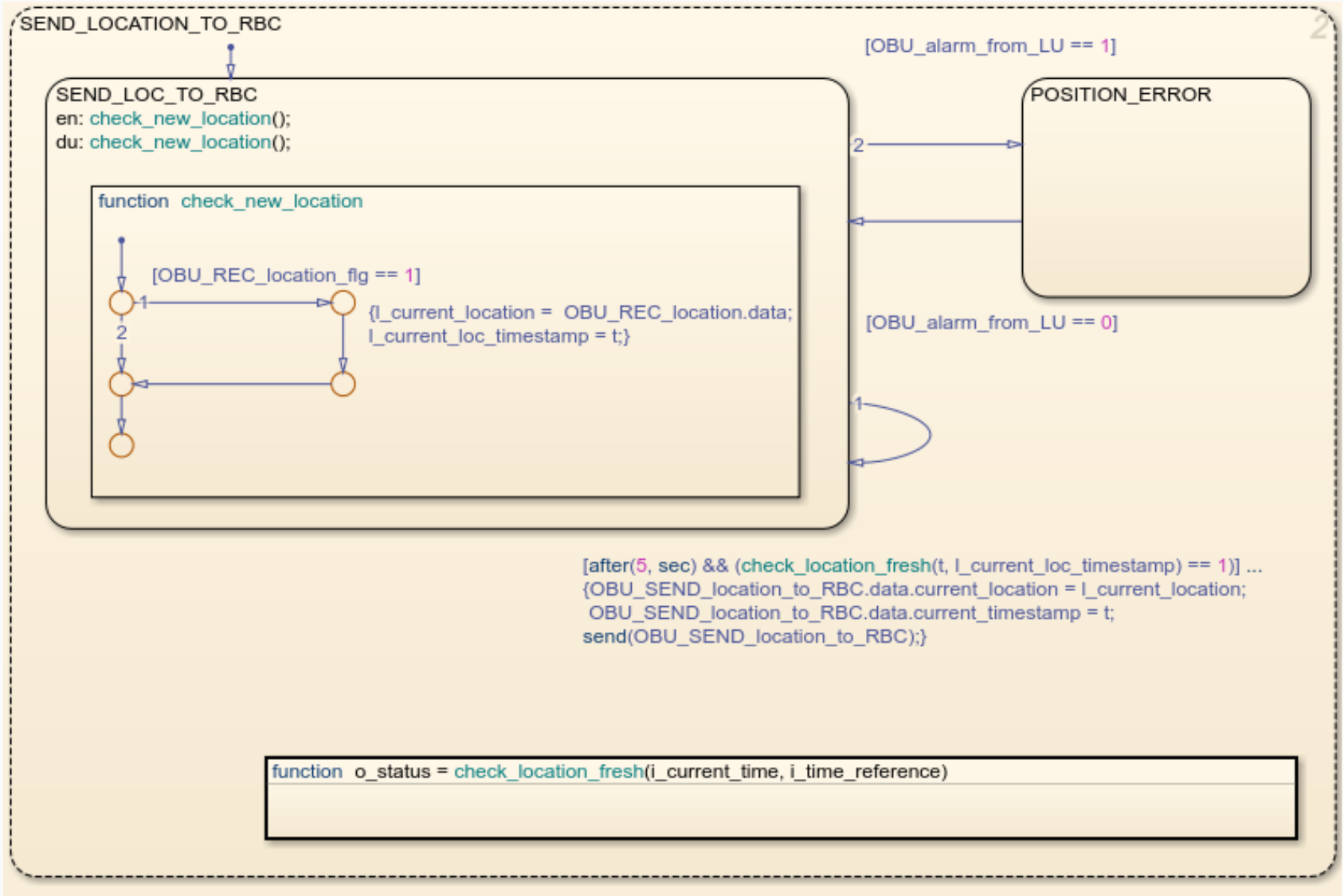
Result: MA messages have a higher probability of being received between the first and the second attempt

UPPAAL SMC: evaluating the freshness of the MA



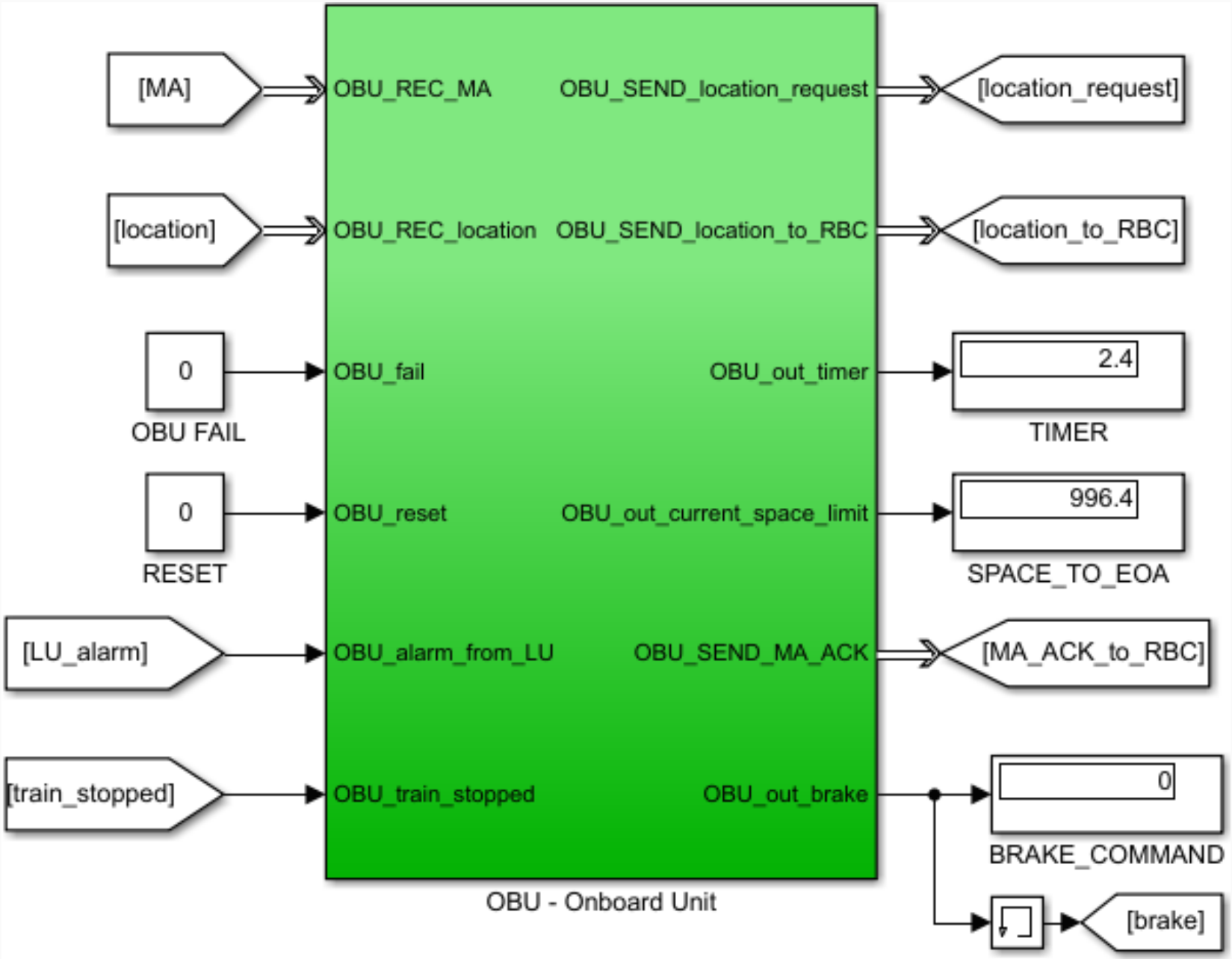
Parameters: α=0.005, ε=0.005, bucket width=0.08733, bucket count=100
Runs: 10000 in total, 10000 (100%) displayed, 0 (0%) remaining
Span of displayed sample: [3.04171764778005, 11.7747301491212]
Mean of displayed sample: 5.73865788065071 ± 0.0327581295234518 (99.5% CI)

Model transformation 2: Simulink → UPPAAL

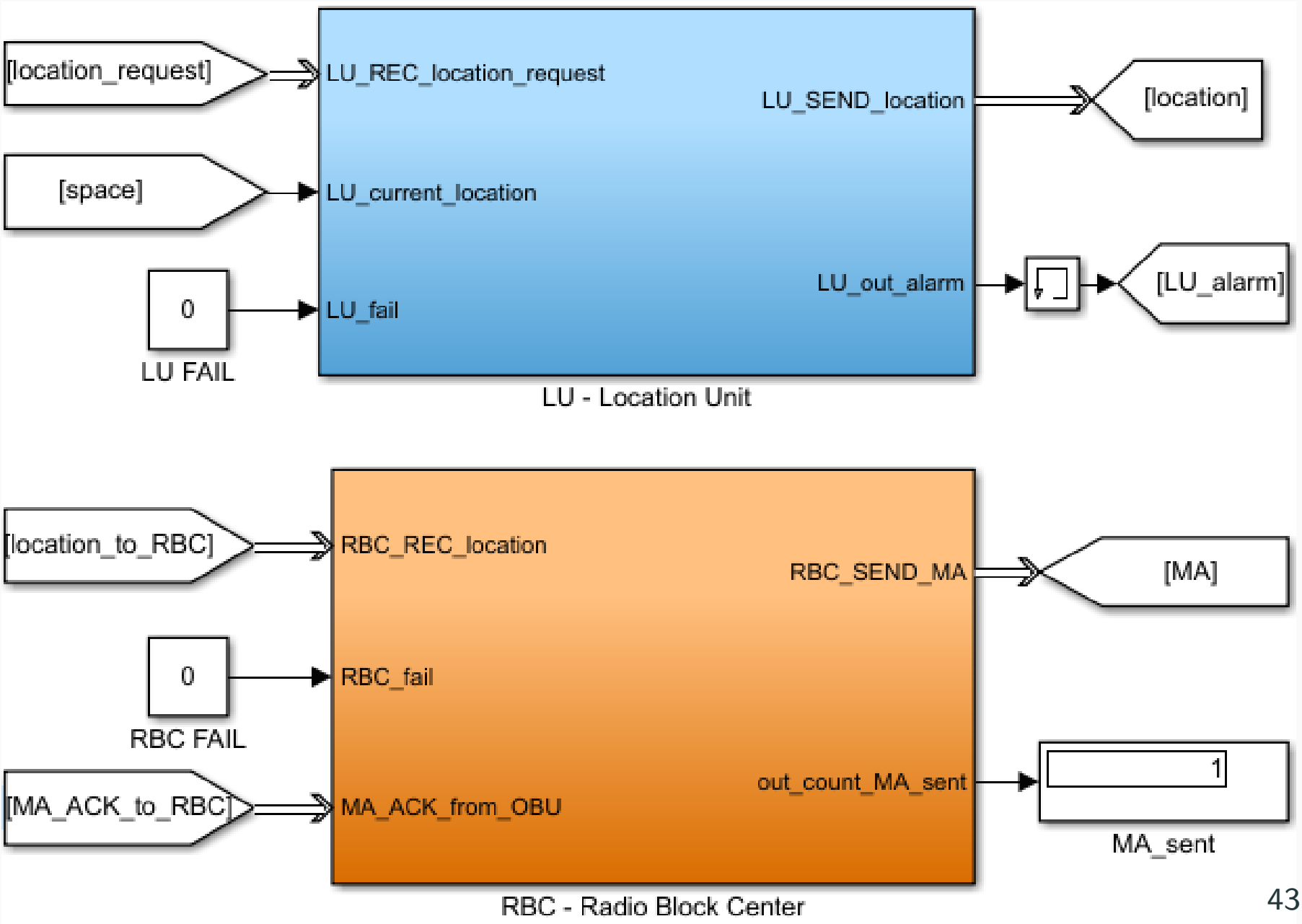


Recall: Simulink models obtained upon requirements elicitation and refinement with industrial partners

Architecture of Simulink model (1/2)

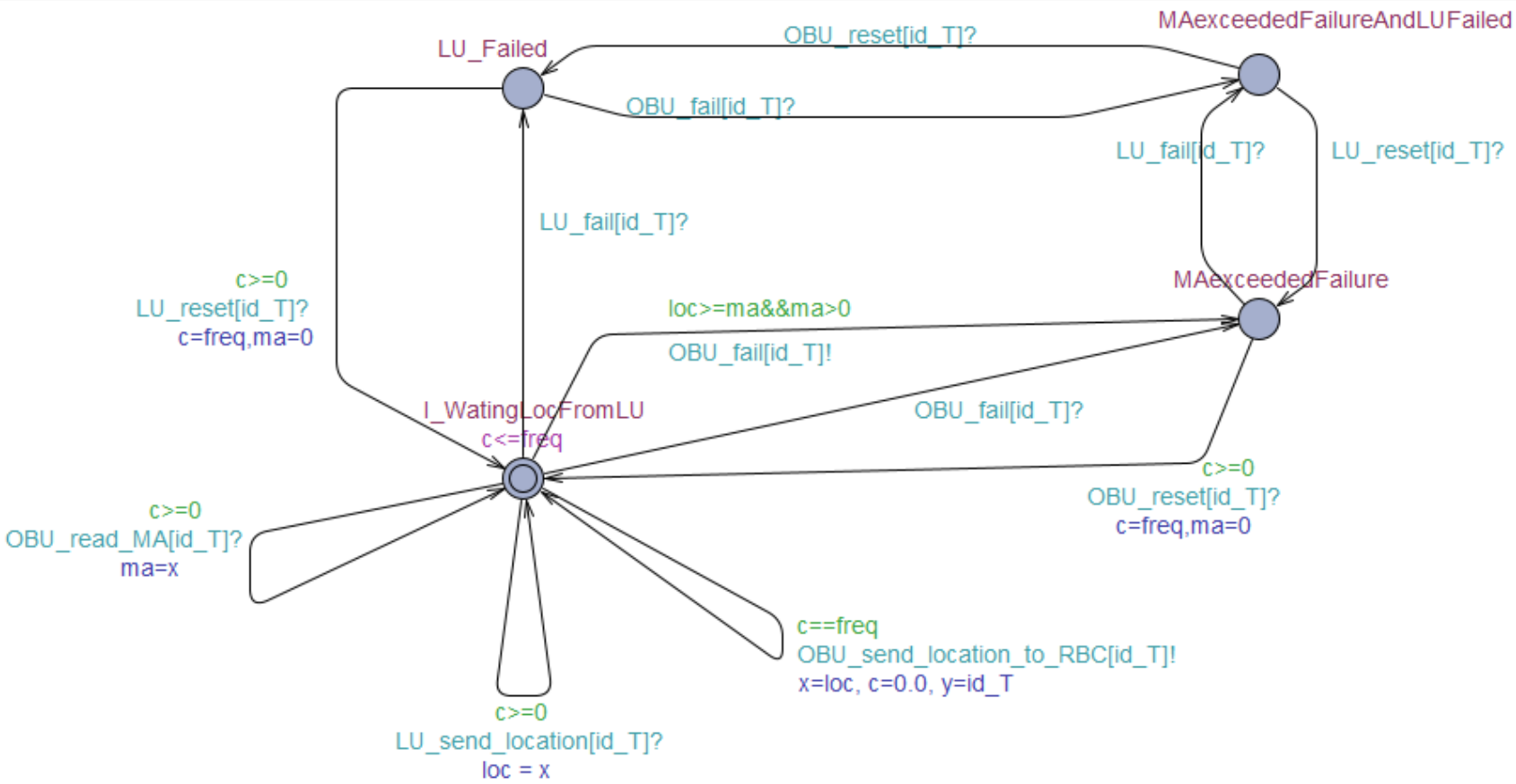


Architecture of Simulink model (2/2)



UPPAAL model of moving block signalling scenario

OBU_MAIN_SendLocationToRBC_T:



TRAIN_ATO_T: model train movement (speed, acceleration/deceleration triggered by approaching the limit of the MA, simulating braking curves when reaching failure states)

Analyses with UPPAAL SMC

1 Probability that the train's position exceeds the MA (with ma = 1000 m):

`Pr[<= 1000](<> OBU_MAIN_SendLocationToRBC.MAexceededFailure)`

UPPAAL SMC reports that this probability is in interval [0,0.00998576], with confidence 0.995 and obtained from 597 runs in ± 8 min.

Analyses with UPPAAL SMC

1 Probability that the train's position exceeds the MA (with ma = 1000 m):

`Pr[<= 1000](<> OBU_MAIN_SendLocationToRBC.MAexceededFailure)`

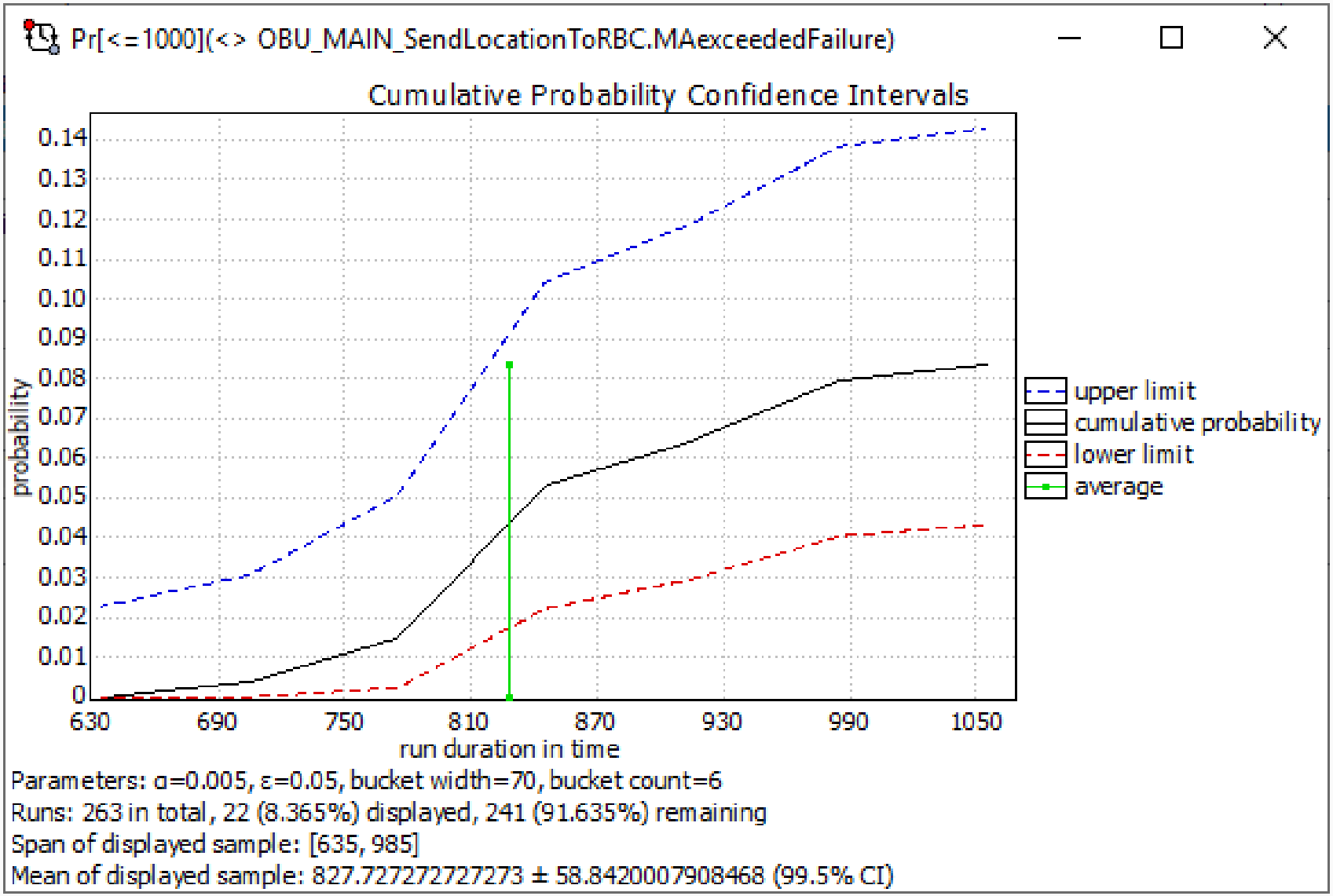
UPPAAL SMC reports that this probability is in interval [0,0.00998576], with confidence 0.995 and obtained from 597 runs in ± 8 min.

2 Probability that the train's position exceeds the MA (with ma = 500 m):

`Pr[<= 1000](<> OBU_MAIN_SendLocationToRBC.MAexceededFailure)`

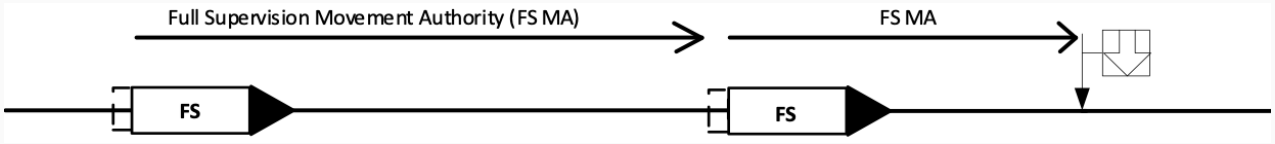
UPPAAL SMC reports that this probability is in interval [0.0430205,0.14268], with confidence 0.995 and obtained from 263 runs in ± 3 min.

Analyses with UPPAAL SMC



Full moving block without trackside train detection

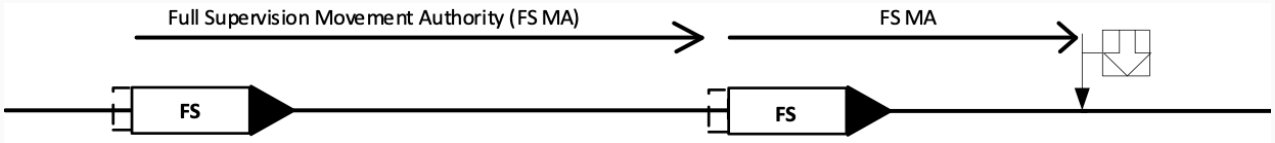
Model: 3 trains/line + train's MA based on location of train ahead



[STTT22] D. Basile, M.H. ter Beek, A. Ferrari & A. Legay, Exploring the ERTMS/ETCS full moving block specification: An experience with formal methods. *International Journal on Software Tools for Technology Transfer* (2022)

Full moving block without trackside train detection

Model: 3 trains/line + train's MA based on location of train ahead



1 Verifying MAs:

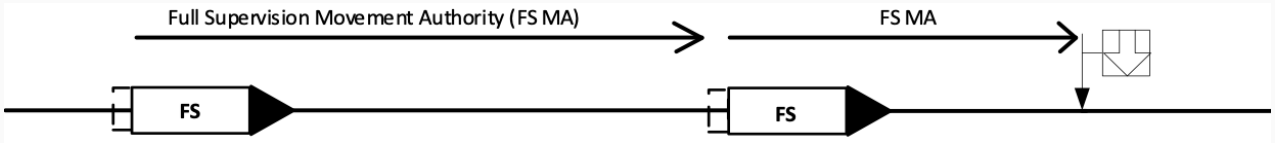
```
loc[id_Train]<=ma && forall(id:int[0,nTrain-1])
       ( loc[id]<=loc[id_Train] || (loc[id]>=ma
       && exists(id1:int[0,nTrain-1])ma==loc[id1]) )
```

State invariant: either the assigned train is ahead of all other trains, or there's no other train whose location is between `id_Train` and its MA, and the MA is equal to one such a train ahead

[STTT22] D. Basile, M.H. ter Beek, A. Ferrari & A. Legay, Exploring the ERTMS/ETCS full moving block specification: An experience with formal methods. *International Journal on Software Tools for Technology Transfer* (2022)

Full moving block without trackside train detection

Model: 3 trains/line + train's MA based on location of train ahead



1 Verifying MAs:

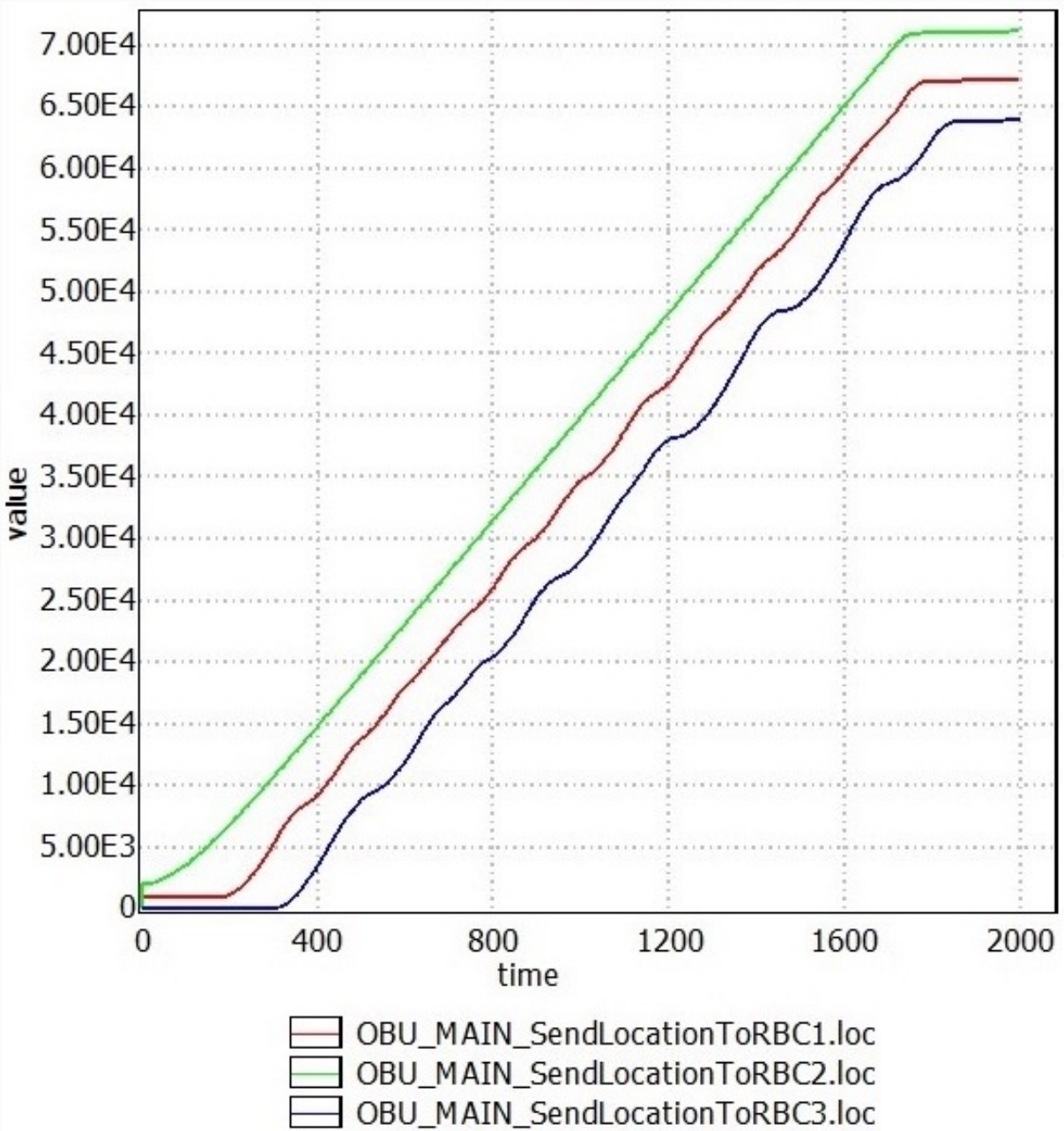
```
loc[id_Train]<=ma && forall(id:int[0,nTrain-1])
       ( loc[id]<=loc[id_Train] || (loc[id]>=ma
       && exists(id1:int[0,nTrain-1])ma==loc[id1]) )
```

State invariant: either the assigned train is ahead of all other trains, or there's no other train whose location is between `id_Train` and its MA, and the MA is equal to one such a train ahead

2 Probability to exceed MA: `Pr[<=1000](<>whofailed==SENDLOC)`

[STTT22] D. Basile, M.H. ter Beek, A. Ferrari & A. Legay, Exploring the ERTMS/ETCS full moving block specification: An experience with formal methods. *International Journal on Software Tools for Technology Transfer* (2022)

Full moving block without trackside train detection



Results

Analysis	Results	Lessons learned
Location freshness	The RBC shall always be ready to receive new locations from the OBU	Identified communication mismatch due to modelling errors between OBU and RBC, by comparing the last location received by RBC with the last location sent by OBU
Message loss	All unexpected messages from the RBC/LU/OBU can safely be discarded	Errors introduced by refactoring the model were identified, by analysing all possible message exchanges
Verifying MA	At each attempt, the RBC shall compute the MA the moment it is sent to the OBU	Formalising the requirements of 3 trains led to detect flaws in communication of MA, due to interplay between delays in communications and concurrent RBC threads
Exceeding MA	Under the given assumptions, the model confirms that trains can travel with a headway of 1 minute	Identified a model with a parameter setup that confirmed the values from the literature about headway in high-speed trains

Success story 3

Synthesis of autonomous driving strategies

Model transformation 3: UPPAAL SMC $\rightarrow$ UPPAAL Stratego

- From stochastic timed automata to stochastic priced timed games

[FORTE20] D. Basile, M.H. ter Beek & A. Legay, Strategy Synthesis for Autonomous Driving in a Moving Block Railway System with Uppaal Stratego @ FORTE'20

Synthesis of autonomous driving strategies

Model transformation 3: UPPAAL SMC $\rightarrow$ UPPAAL Stratego

- From stochastic timed automata to stochastic priced timed games

Formal modelling and analysis:

- UPPAAL Stratego: strategy synthesis for timed games (safety) and reinforcement learning of the optimal strategy (reliability)
- While changing the set-up of the parameters, the driving strategy is automatically tuned to retain safety (MA is never exceeded) as well as reliability (minimal expected arrival time)

[FORTE20] D. Basile, M.H. ter Beek & A. Legay, Strategy Synthesis for Autonomous Driving in a Moving Block Railway System with Uppaal Stratego @ FORTE'20

Synthesis of autonomous driving strategies

Model transformation 3: UPPAAL SMC $\rightarrow$ UPPAAL Stratego

- From stochastic timed automata to stochastic priced timed games

Formal modelling and analysis:

- UPPAAL Stratego: strategy synthesis for timed games (safety) and reinforcement learning of the optimal strategy (reliability)
- While changing the set-up of the parameters, the driving strategy is automatically tuned to retain safety (MA is never exceeded) as well as reliability (minimal expected arrival time)

Results:

- Experimentation required interacting with the tool developers, resulting in new releases, with patches fixing issues which were discovered through our model!

[FORTE20] D. Basile, M.H. ter Beek & A. Legay, Strategy Synthesis for Autonomous Driving in a Moving Block Railway System with Uppaal Stratego @ FORTE'20

Synthesis of autonomous driving strategies

Model transformation 3: UPPAAL SMC $\rightarrow$ UPPAAL Stratego

- From stochastic timed automata to stochastic priced timed games

Formal modelling and analysis:

- UPPAAL Stratego: strategy synthesis for timed games (safety) and reinforcement learning of the optimal strategy (reliability)
- While changing the set-up of the parameters, the driving strategy is automatically tuned to retain safety (MA is never exceeded) as well as reliability (minimal expected arrival time)

Results:

- Experimentation required interacting with the tool developers, resulting in new releases, with patches fixing issues which were discovered through our model!

Related work from Mälardalen (UPPAAL SMC / Stratego / Tiga):

[PhD22] R. Gu, *Formal Methods for Scalable Synthesis and Verification of Autonomous Systems: Mission Planning and Collision Avoidance* (2022)

Success story 4

Smart energy consumption: energy efficiency vs. dependability

Smart railway systems and stations of the future

Smart energy consumption: energy efficiency vs. dependability

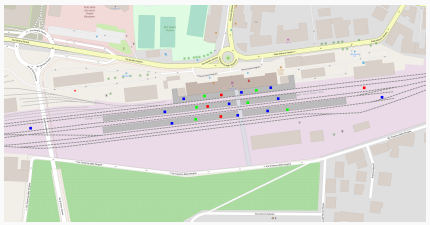
- Smart deicing system: railroad switch heaters for correct working of switches in case of ice / snow

Smart railway systems and stations of the future

Formal Methods and Tools Lab

Smart energy consumption: energy efficiency vs. dependability

- Smart deicing system: railroad switch heaters for correct working of switches in case of ice / snow
- Smart station lighting: reduce illumination whenever (time) and wherever (space) possible, guaranteeing legal minimum levels

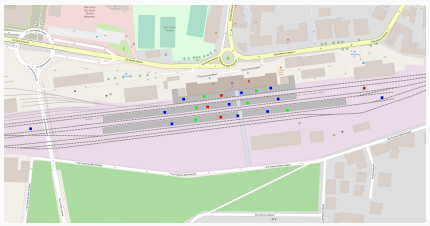



Smart railway systems and stations of the future

Smart energy consumption: energy efficiency vs. dependability

- Smart deicing system: railroad switch heaters for correct working of switches in case of ice / snow

- Smart station lighting: reduce illumination whenever (time) and wherever (space) possible, guaranteeing legal minimum levels



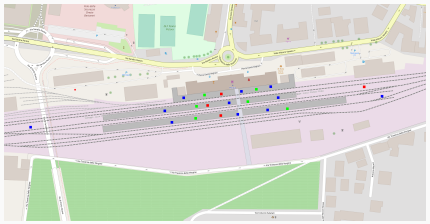

[ISoLA20] D. Basile, M.H. ter Beek, F. Di Giandomenico, A. Fantechi, S. Gnesi & G.O. Spagnolo, 30 Years of Simulation-Based Quantitative Analysis Tools: A Comparison Experiment Between Möbius and Uppaal SMC @ ISoLA'20

[FMICS21] M.H. ter Beek, V. Ciancia, D. Latella, M. Massink & G.O. Spagnolo, Spatial Model Checking for Smart Stations: Research Challenges @ FMICS'21

Smart railway systems and stations of the future

Smart energy consumption: energy efficiency vs. dependability

- Smart deicing system: railroad switch heaters for correct working of switches in case of ice / snow
- Smart station lighting: reduce illumination whenever (time) and wherever (space) possible, guaranteeing legal minimum levels




Smart maintenance: railway specific model learning techniques (with real-time aspects and a degree of uncertainty) to facilitate predictive maintenance by real-time monitoring and simulation

Case study: railroad switch heating system

Energy management: relevant political, societal & technological concern

Case study: railroad switch heating system

Energy management: relevant political, societal & technological concern

- Railroad switches: critical system

Case study: railroad switch heating system

Energy management: relevant political, societal & technological concern

- Railroad switches: critical system
- Heating system: tubular flat heaters, induction heating, possibility for energy management

Case study: railroad switch heating system

Energy management: relevant political, societal & technological concern

- Railroad switches: critical system
- Heating system: tubular flat heaters, induction heating, possibility for energy management
- Cyber-Physical System: sensors for the temperatures
  - weather forecast: *stochastic* (physical)
  - temperature evolution: *continuous* (physical)
  - on/off power management: *discrete* (cyber)

Case study: railroad switch heating system

Energy management: relevant political, societal & technological concern

- Railroad switches: critical system
- Heating system: tubular flat heaters, induction heating, possibility for energy management
- Cyber-Physical System: sensors for the temperatures
  - weather forecast: *stochastic* (physical)
  - temperature evolution: *continuous* (physical)
  - on/off power management: *discrete* (cyber)
- Quantitative properties: energy vs. reliability through policies of energy consumption

Case study: railroad switch heating system

Energy management: relevant political, societal & technological concern

- Railroad switches: critical system
- Heating system: tubular flat heaters, induction heating, possibility for energy management
- Cyber-Physical System: sensors for the temperatures
  - weather forecast: *stochastic* (physical)
  - temperature evolution: *continuous* (physical)
  - on/off power management: *discrete* (cyber)
- Quantitative properties: energy vs. reliability through policies of energy consumption

Contribution: comparison experiment on selected features between two formal modelling and analysis frameworks:

- Stochastic Activity Networks (SAN) and Möbius
- Stochastic Hybrid Automata (SHA) and UPPAAL SMC

Energy-saving policies

Dynamic power management, on/off policy based on 2 thresholds:

on warning threshold T_{wa} guarantees reliability

off working threshold T_{wo} guarantees energy saving

Energy-saving policies

Dynamic power management, on/off policy based on 2 thresholds:

on warning threshold T_{wa} guarantees reliability

off working threshold T_{wo} guarantees energy saving

3rd parameter NH_{max}: max. power supply (% of heaters turned on)

Energy-saving policies

Dynamic power management, on/off policy based on 2 thresholds:

on warning threshold T_{wa} guarantees reliability

off working threshold T_{wo} guarantees energy saving

3rd parameter NH_{max}: max. power supply (% of heaters turned on)

FIFO priorities for main railroad tracks

Energy-saving policies

Dynamic power management, on/off policy based on 2 thresholds:

on warning threshold T_{wa} guarantees reliability

off working threshold T_{wo} guarantees energy saving

3rd parameter NH_{max}: max. power supply (% of heaters turned on)

FIFO priorities for main railroad tracks

Continuous physical behaviour concerning temperature increment and decrement of the railroad track when heater is turned on or off, resp., modelled by an ODE (representing the energy balance)

Stochastic model-based analysis and Möbius

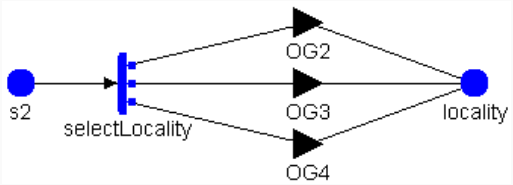
Möbius: distributed discrete-event simulator and explicit state-space generators and numerical solution algorithms for Markovian models; analysis of both transient and steady-state reward models

Stochastic model-based analysis and Möbius

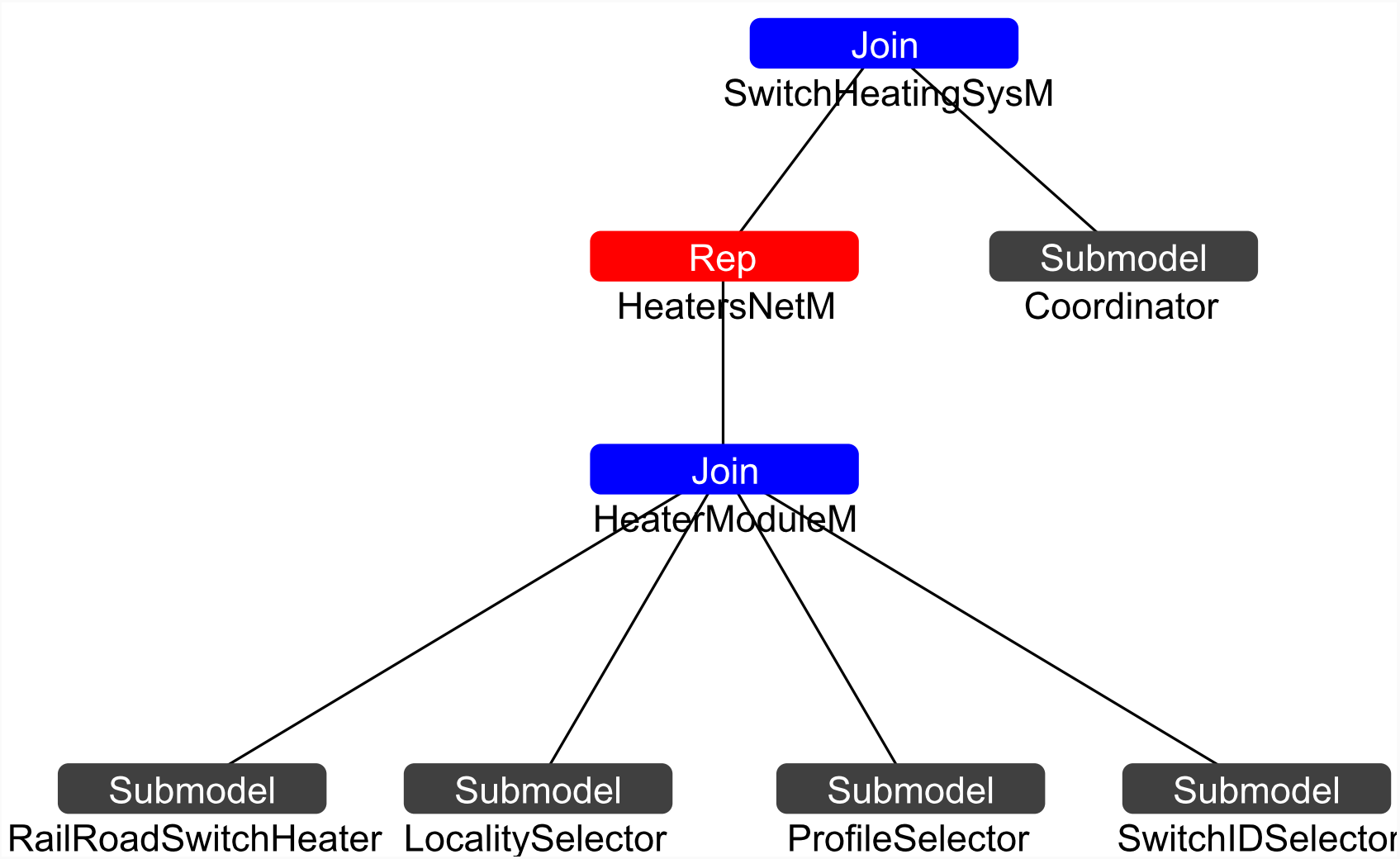
Möbius: distributed discrete-event simulator and explicit state-space generators and numerical solution algorithms for Markovian models; analysis of both transient and steady-state reward models

Stochastic Activity Networks: generalisation of stochastic Petri nets

- places & activities: same interpretation as places & transitions in PNs
- input gates control the enabling conditions of an activity and define the change of marking when an activity completes; output gates define the change of marking upon activity completion
- activities can be *instantaneous* or *timed*
  - instantaneous activities complete once enabling conditions are satisfied
  - timed activities take time to complete (stochastic distribution of time)
- cases are associated to activities, used to represent probabilistic uncertainty about the action taken upon completion of the activity
- policies of activation/reactivation of activities
- primitives are defined using C++ code

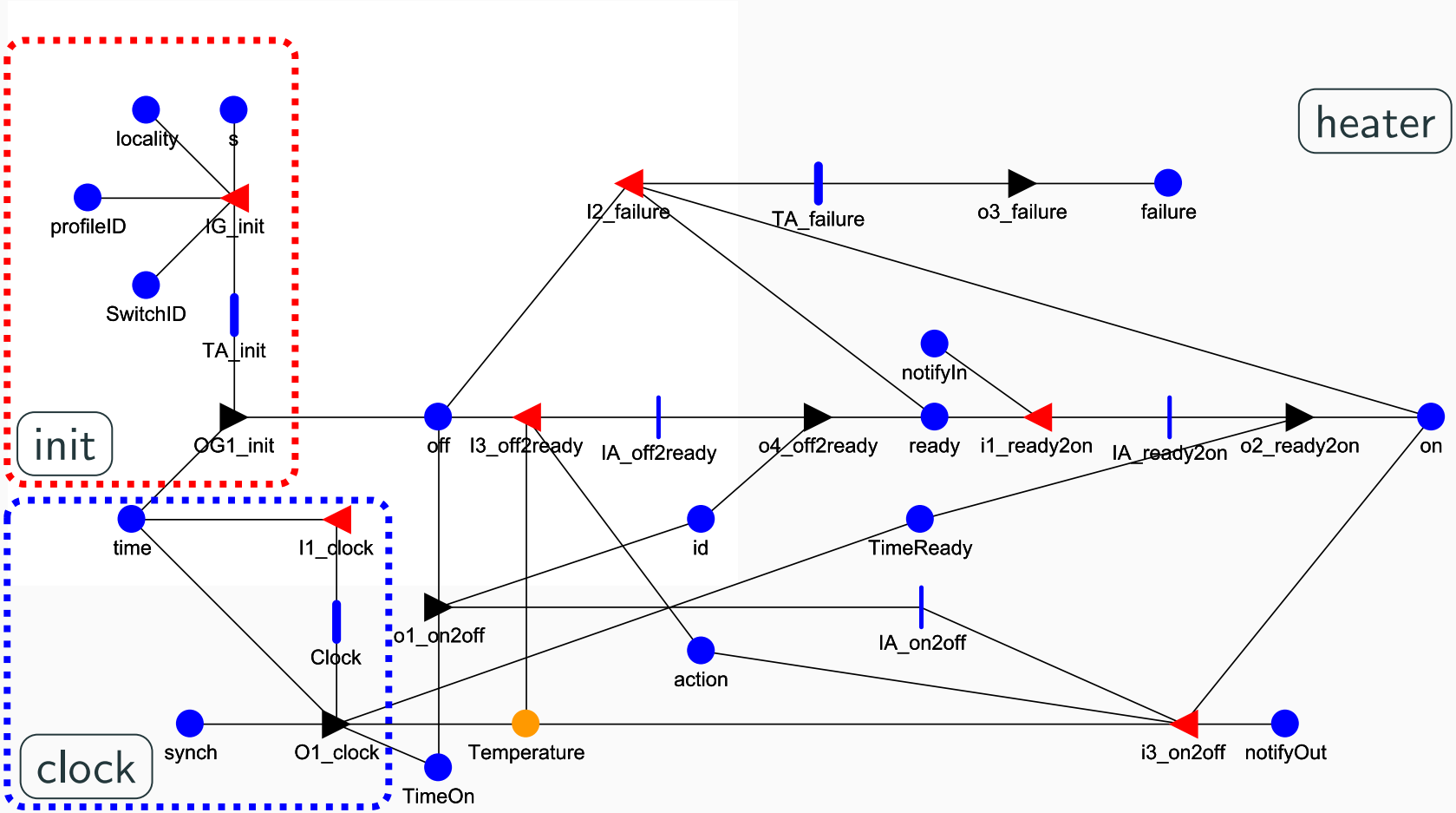


SAN Composed Model



Interactions through shared places

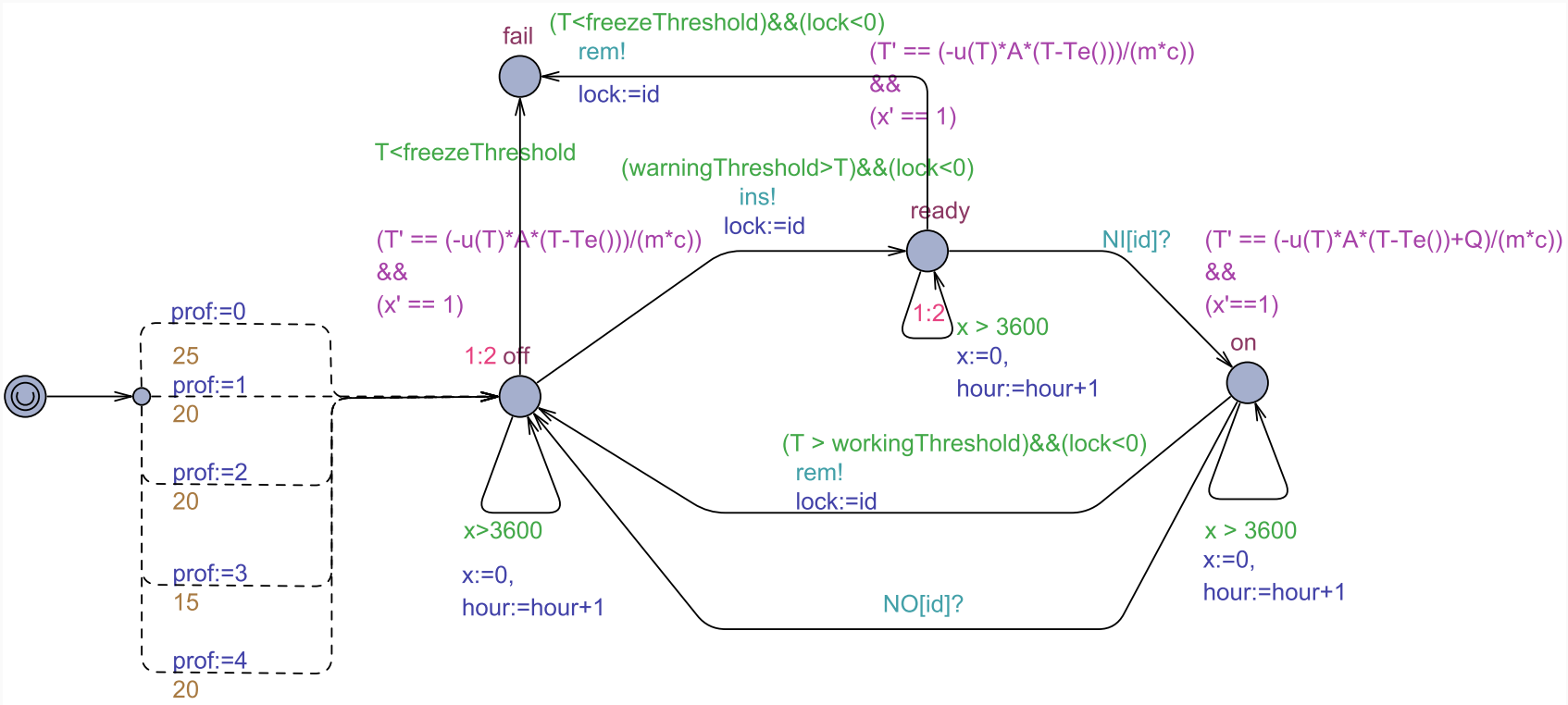
Möbius: RailRoadSwitchHeater SAN



$$mc\frac{\partial T}{\partial t} = -uA(T - T_{env}) + \dot{Q}$$

UPPAAL SMC: RailRoadSwitchHeater SHA

- A coordinator K to manage NH_{max} and priorities
- The composed system: $\mathtt{N} = (\bigotimes_{id \in 1,\ldots,n} \mathtt{H_{id}}) \otimes \mathtt{K}$
- Array of channels for one-to-one communication



$$mc\frac{\partial T}{\partial t} = -uA(T - T_{env}) + \dot{Q}$$

Quantitative analysis: measures of interest

Energy consumption: time (in hours) a generic heater is activated in a specific time interval; by multiplying such measurement for the power consumed (kW/h), one can derive the energy consumption

- Möbius: sum of time spent in specific markings (reward structure)
- UPPAAL: $\texttt{E}[<= 24; 10000]\ (\texttt{max} : \sum_{\texttt{i:id}_\texttt{t}} \texttt{H}_\texttt{i}.\texttt{energy})$

Quantitative analysis: measures of interest

Energy consumption: time (in hours) a generic heater is activated in a specific time interval; by multiplying such measurement for the power consumed (kW/h), one can derive the energy consumption

- Möbius: sum of time spent in specific markings (reward structure)
- UPPAAL: $\mathtt{E}[<= 24; 10000]\ (\mathtt{max} : \sum_{\mathtt{i}:\mathtt{id_t}} \mathtt{H_i.energy})$

Reliability: probability that a generic switch fails (i.e., it is frozen), computed as the probability that no failure occurs in the time interval

- Möbius: presence of one token at the end of the experiment in the place encoding a failure (reward structure)
- UPPAAL: $\mathbb{P}(\Diamond_{\mathtt{h}\leq 24} \exists(\mathtt{i} : \mathtt{id_t})(\mathtt{H_i.fail}))$

Quantitative analysis: measures of interest

Energy consumption: time (in hours) a generic heater is activated in a specific time interval; by multiplying such measurement for the power consumed (kW/h), one can derive the energy consumption

- Möbius: sum of time spent in specific markings (reward structure)
- UPPAAL: $\mathtt{E}[<= 24; 10000]\ (\mathtt{max} : \sum_{\mathtt{i}:\mathtt{id_t}} \mathtt{H_i.energy})$

Reliability: probability that a generic switch fails (i.e., it is frozen), computed as the probability that no failure occurs in the time interval

- Möbius: presence of one token at the end of the experiment in the place encoding a failure (reward structure)
- UPPAAL: $\mathbb{P}(\Diamond_{\mathtt{h}\leq 24} \exists(\mathtt{i} : \mathtt{id_t})(\mathtt{H_i.fail}))$

Cf. papers for results (aligned, find the right parameter setup as best compromise for tradeoff between energy consumption and reliability)

Comparison experiment

Only evaluated primitive support of features, usability & expressiveness

Comparison experiment

Only evaluated primitive support of features, usability & expressiveness

Modelling features: the composition of, and interactions between, different models (i.e., *heterogeneous formalisms, replicated models, dynamic process instantiation, communication primitives*) and the ability towards modelling hybrid and stochastic systems (i.e., *delay distributions, hybrid variables*)

Comparison experiment

Only evaluated primitive support of features, usability & expressiveness

Modelling features: the composition of, and interactions between, different models (i.e., *heterogeneous formalisms, replicated models, dynamic process instantiation, communication primitives*) and the ability towards modelling hybrid and stochastic systems (i.e., *delay distributions, hybrid variables*)

Properties specification: the definition of *measures of interest* and the ability to verify properties of the defined models (i.e., *property verification*)

Comparison experiment

Only evaluated primitive support of features, usability & expressiveness

Modelling features: the composition of, and interactions between, different models (i.e., *heterogeneous formalisms, replicated models, dynamic process instantiation, communication primitives*) and the ability towards modelling hybrid and stochastic systems (i.e., *delay distributions, hybrid variables*)

Properties specification: the definition of *measures of interest* and the ability to verify properties of the defined models (i.e., *property verification*)

Experiments and presentation of results: setup and execution of experiments, as well as data collection and plotting the results (i.e., *experiments parameter setup*)

Comparison

Features	SAN+Möbius	SHA+UPPAAL SMC
Measures of interest	Reward models	MITL formulae
Experiments parameter setup	Batches	Single
Replicated models	Anonymous	Distinguished
Dynamic process instantiation	Not available	Available
Heterogeneous formalisms	Available (SAN, PEPA, etc.)	Not available (SHA)
Communication primitives	Shared places	Channels
Delay distributions	Various distributions	Exponential, uniform
Hybrid variables	No primitive support	ODE solver available
Property verification	Not available	Temporal logics

Possible future improvements for usability

Possible future improvements for usability

Möbius could provide primitive support for:

- non-anonymous replicas and channel communication (already under development by tool's developers)
- graphical visualisation of data
- primitive support for ODEs

Results

Possible future improvements for usability

Möbius could provide primitive support for:

- non-anonymous replicas and channel communication (already under development by tool's developers)
- graphical visualisation of data
- primitive support for ODEs

UPPAAL SMC could provide primitive support for:

- batches of experiments with different parameters (well received by K. Larsen @ ISoLA's SMC track)
- other distribution delays (e.g., deterministic time)

Concluding remarks

Concluding remarks

Closing the gap:

semi-formal models popular and suitable to communicate with industry

formal models required to apply analysis tools in safety-critical domains (e.g., railways)

Concluding remarks

Closing the gap:

semi-formal models popular and suitable to communicate with industry

formal models required to apply analysis tools in safety-critical domains (e.g., railways)

Each formal tool:

- requires modelling in a different input language
- expert knowledge of different analysis technique

Concluding remarks

Closing the gap:

semi-formal models popular and suitable to communicate with industry

formal models required to apply analysis tools in safety-critical domains (e.g., railways)

Each formal tool:

- requires modelling in a different input language
- expert knowledge of different analysis technique

FMT: researchers with different and complementary expertise

Industry: in-house formal methods expertise rare (AWS, ASML, . . .)

Concluding remarks

Closing the gap:

semi-formal models popular and suitable to communicate with industry
formal models required to apply analysis tools in safety-critical domains (e.g., railways)

Each formal tool:

- requires modelling in a different input language
- expert knowledge of different analysis technique

FMT: researchers with different and complementary expertise
Industry: in-house formal methods expertise rare (AWS, ASML, . . .)

Road to success:

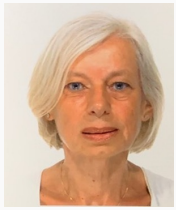
- capacity to abstract
- pick the right tool based on the industry input and requirements at hand

Acknowledgements

Thanks for your attention! And note that we're hiring!

Formal Methods and Tools Lab

And thanks to my co-authors of the work presented in this keynote:



Sirti
building the future

mermec
AN ANGEL COMPANY

PROVER®

. . .

References

References (1/3)

FB22 A. Ferrari & M.H. ter Beek, Formal Methods in Railways: A Systematic Mapping Study. *ACM Computing Surveys* 55, 4 (2022), 69:1–69:37. https://doi.org/10.1145/3520480

TSE22 A. Ferrari, F. Mazzanti, D. Basile & M.H. ter Beek, Systematic Evaluation and Usability Analysis of Formal Methods Tools for Railway Signaling System Design. *IEEE Transactions on Software Engineering* 48, 11 (2022), 4675–4691. https://doi.org/10.1109/TSE.2021.3124677

STTT22 D. Basile, M.H. ter Beek, A. Ferrari & A. Legay, Exploring the ERTMS/ETCS full moving block specification: An experience with formal methods. *International Journal on Software Tools for Technology Transfer* 24, 3 (2022), 351–370. https://doi.org/10.1007/s10009-022-00653-3

FMICS21 M.H. ter Beek, V. Ciancia, D. Latella, M. Massink & G.O. Spagnolo, Spatial Model Checking for Smart Stations: Research Challenges. In *Proceedings of the 26th International Conference on Formal Methods for Industrial Critical Systems (FMICS'21)*, *LNCS* 12863, Springer, 2021, 39–47. https://doi.org/10.1007/978-3-030-85248-1_3

ISoLA20 D. Basile, M.H. ter Beek, F. Di Giandomenico, A. Fantechi, S. Gnesi & G.O. Spagnolo, 30 Years of Simulation-Based Quantitative Analysis Tools: A Comparison Experiment Between Möbius and Uppaal SMC. In *Proceedings of the 9th International Symposium on Leveraging Applications of Formal Methods, Verification and Validation: Verification Principles (ISoLA'20)*, *LNCS* 12476, Springer, 2020, 368–384. https://doi.org/10.1007/978-3-030-61362-4_21

ICSE20 A. Ferrari, F. Mazzanti, D. Basile, M.H. ter Beek & A. Fantechi, Comparing Formal Tools for System Design: a Judgment Study. In *Proceedings of the 42nd International Conference on Software Engineering (ICSE'20)*, ACM, 2020, 62–74. https://doi.org/10.1145/3377811.3380373

References (2/3)

FORTE20 D. Basile, M.H. ter Beek & A. Legay, Strategy Synthesis for Autonomous Driving in a Moving Block Railway System with Uppaal Stratego. In *Proceedings of the 40th International Conference on FORmal TEchniques for Distributed Objects, Components, and Systems (FORTE'20)*, *LNCS* 12136, Springer, Berlin, 2020, 3–21. https://doi.org/10.1007/978-3-030-50086-3_1

FM19 M.H. ter Beek, A. Borälv, A. Fantechi, A. Ferrari, S. Gnesi, C. Löfving & F. Mazzanti, Adopting Formal Methods in an Industrial Setting: The Railways Case. In *Formal Methods – The Next 30 Years—Proceedings of the Third World Congress on Formal Methods (FM'19)*, *LNCS* 11800, Springer, 2019, 762–772. https://doi.org/10.1007/978-3-030-30942-8_46

FMICS19 D. Basile, M.H. ter Beek, A. Ferrari & A. Legay, Modelling and Analysing ERTMS L3 Moving Block Railway Signalling with Simulink and UPPAAL SMC. In *Proceedings of the 24th International Conference on Formal Methods for Industrial Critical Systems (FMICS'19)*, *LNCS* 11687, Springer, 2019, 1–21. https://doi.org/10.1007/978-3-030-27008-7_1

ISoLA18 D. Basile, M.H. ter Beek & V. Ciancia, Statistical Model Checking of a Moving Block Railway Signalling Scenario with Uppaal SMC. In *Proceedings of the 8th International Symposium on Leveraging Applications of Formal Methods, Verification and Validation: Verification (ISoLA'18)*, *LNCS* 11245, Springer, 2018, 372–391. https://doi.org/10.1007/978-3-030-03421-4_24

iFM18 D. Basile, M.H. ter Beek, A. Fantechi, S. Gnesi, F. Mazzanti, A. Piattino, D. Trentini & A. Ferrari, On the Industrial Uptake of Formal Methods in the Railway Domain: A Survey with Stakeholders. In *Proceedings of the 14th International Conference on Integrated Formal Methods (IFM'18)*, *LNCS* 11023, Springer, 2018, 20–29. https://doi.org/10.1007/978-3-319-98938-9_2

References (3/3)

MARS18 F. Mazzanti & A. Ferrari, Ten Diverse Formal Models for a CBTC Automatic Train Supervision System. In *Proceedings of the Third Workshop on Models for Formal Analysis of Real Systems (MARS'18)*, *EPTCS* 268, 2018, 104–149. https://doi.org/10.4204/EPTCS.268.4

SAC17 D. Basile, F. Di Giandomenico & S. Gnesi, Statistical Model Checking of an Energy-Saving Cyber-Physical System in the Railway Domain. In *Proceedings of the 32nd Symposium on Applied Computing (SAC'17)*, ACM, 2017, 1356–1363. https://doi.org/10.1145/3019612.3019824

JRTPM16 D. Basile, S. Chiaradonna, F. Di Giandomenico & S. Gnesi, A stochastic model-based approach to analyse reliable energy-saving rail road switch heating systems. *Journal of Rail Transport Planning & Management* 6, 2 (2016), 163–181. https://doi.org/10.1016/j.jrtpm.2016.03.003

NFM14 F. Mazzanti, G.O. Spagnolo & A. Ferrari, Designing a Deadlock-Free Train Scheduler: A Model Checking Approach. In *Proceedings of the 6th International NASA Formal Methods Symposium (NFM'14)*, *LNCS* 8430, Springer, 2014, 264–269. https://doi.org/10.1007/978-3-319-06200-6_22