

FINAL PROJECT REPORT

August 2007



Project no.: IST-2001-35304

Project Co-ordinator: Frits Vaandrager

Project Start Date: 1 April 02

Duration: 39 months

Project home page: <http://ametist.cs.utwente.nl/>

Status: Public

Contents

1 Project Overview	4
1.1 Consortium Description	4
1.2 Main Achievements	5
2 Project Objectives	6
3 Project Methodologies, Results and Achievements	7
3.1 Model-Based System Development	8
3.2 Project Results: Expressiveness, Performance & Tools	10
3.2.1 Priced Timed Automata: Theory, Algorithms and Applications . .	10
3.2.2 Stochastic Extensions of Timed Automata	11
3.2.3 Modelling Frameworks for Scheduling Under (Discrete) Uncertainty	11
3.2.4 Optimization and Constraint Satisfaction as a Tool for Timed Au- tomata Analysis	12
3.2.5 Improvements in Timed Automata Model Checking Technology .	12
3.2.6 Specialized Technology for Solving Scheduling Problems with Timed Automata	12
3.2.7 Tools	13
3.3 Project Results: Cybernetix Case Study	15
3.3.1 Activities in year one and two	16
3.3.2 Work in the third year	17
3.3.3 Conclusions	18
3.4 Project Results: Terma Case Study	19
3.4.1 Summary of the work	19
3.4.2 Conclusions	20
3.5 Project Results: Bosch Case Studies	20
3.5.1 The CPS Case Study	20
3.5.2 The Airbag ECU Case Study	22
3.6 Project Results: Axxom Case Study	25
3.6.1 Problem Statement	26
3.6.2 Modeling and Requirement Specification	27
3.6.3 Tool Application and Solution	29
3.6.4 Conclusions	42
3.7 Project Results: Other Case Studies	44

3.8 Towards a Unifying Framework	45
4 Outlook	47
5 Conclusions	49
A List of Deliverables	73

1 Project Overview

1.1 Consortium Description

The AMETIST consortium is composed of seven academic partners and four industrial partners as indicated in the table below. The coordinator is listed first, followed by the industrial partners on places 2-5, followed by the other academic partners on places 6-11.

No	Name	Contact person	Role
CO1	Radboud University Nijmegen	Frits Vaandrager	Technology Provider
AC2	Bosch	Marko Auerswald	User
AC3	Cybernetix	Patrice Gauthier	User
AC4	Axxom	Dagmar Ludewig	User
AC5	Terma	Thomas Hune	User
CR6	University of Aalborg	Kim Larsen	Technology Provider
CR7	University of Dortmund	Sebastian Engell	Technology Provider
CR8	VERIMAG	Oded Maler	Technology Provider
AC9	Weizmann Institute	Amir Pnueli	Technology Provider
CR10	LIF Marseille	Peter Niebert	Technology Provider
CR11	University of Twente	Ed Brinksma	Technology Provider

Already before the start of AMETIST , the academic partners have been cooperating successfully for many years in European projects. All academic partners share a common mission, which is to conduct research on the application of formal methods for the development of computer based systems with the long range objective of transforming the application of formal methods from an academic research topic into an engineering practice. Even though the academic partners have a common background knowledge which facilitates collaborative work, they all brought in complementary expertise:

Nijmegen	verification and analysis of distributed real-time algorithms and systems, model checking, application of formal methods
Aalborg	tool builder (UPPAAL), model checking
Dortmund	process control, hybrid systems, production planning and scheduling
Verimag	tool builder (IF), timed systems
Weizmann	synthesis, abstraction and composition techniques
Marseille	symbolic verification, constraint programming
Twente	validation tools, stochastic methods, verification of soft real-time systems

The industrial partners, which are all prominent players in the embedded systems area, contributed complementary case studies, and used and evaluated the project results. Each industrial partner had a privileged relation with one of the academic partners: Bosch and Axxom with Dortmund, Cybernetix with Marseille, and Terma with Aalborg.

1.2 Main Achievements

Scheduling and resource allocation problems occur in many different domains¹, for instance (1) scheduling of production lines in factories to optimize costs and delays, (2) scheduling of computer programs in (real-time) operating systems to meet deadline constraints, (3) scheduling of micro instructions inside a processor with a bounded number of registers and processing units, (4) scheduling of trains (or airplanes) over limited quantities of railway tracks and crossroads, and (5) mission planning for autonomous robots on spacecrafts. Typically, in each of these domain problems are solved using different approaches and mathematical tools. The AMETIST project has provided the foundations for a unifying framework for time-dependent behavior and dynamic resource allocation that crosses the boundaries of application domains.

In the AMETIST approach, components of a system are modeled as *dynamical systems* with a state space and a well-defined dynamics. All that can happen in a system is expressed in terms of *behaviors* that can be generated by the dynamical systems; these constitute the semantics of the problem. Verification, optimization, synthesis and other design activities explore and modify system structure so that the resulting behaviors are correct, optimal, etc. Preferably, the limitations of currently known computational solutions should not influence modeling too much: only after the semantics of a problem is properly understood, abstractions and specialization due to computational considerations can intervene. In such situations, the soundness of abstractions should ideally also be proved, either via deductive verification or model checking. AMETIST has shown that this approach, which underlies the successful domain of *formal verification*, can be extended to resource allocation, scheduling and other time-related problems.

AMETIST has made major advances in the area of (timed automata based) tools. Several (new versions) of tools were released, implementing algorithmic ideas that have been developed during the first two years of the project. Tight connections and interfaces between all of these tools exist or are currently being developed. If we compare the current capabilities of our tools with what existed at the start of AMETIST, then it is fair to say that indeed we have moved the state-of-the-art to a new level of maturity (one of the main objectives of the project). Also the convenience of using the tool has been greatly improved due to the enriched expressiveness of the input language and connections with other tools. Nowadays even high school students find it easy to modify and build timed automata models using the improved GUI. Through the website and new tutorial papers (such as [94]) the tool has become very easily accessible.

Profiting from the new tools, the AMETIST consortium has tackled more than 20 industrial sized case studies which were provided by our industrial partners (**Bosch**, **CYR**, **Axxom**, **Terma**) or obtained from other sources. The general conclusion is that using our new methods we can handle bigger problems faster and in a much more routine manner than at the start of the project. Using UPPAAL CORA, for instance, we succeeded to derive schedules that are competitive with those that were provided by industrial partner **Axxom**

¹This subsection has been taken from [56].

using its own tool Orion-Pi. Our experience with the AXXOM case study shows the application of model checking techniques for scheduling is promising. Still, timed automata are not yet a push-button technology to be applied without problem specific modeling and solution strategies. But the generation of libraries of templates for typical configurations seems promising and appears as a path towards more widespread and easier application for non-TA-specialist users. Considerable further work on modelling methods, reusability of modelling patterns, identification and evaluation of heuristics, comparison with alternative approaches, all in the context of case studies of greater orders of magnitude, is needed to develop it into a readily applicable standard technique for scheduling.

2 Project Objectives

The following description of the objectives of AMETIST has been taken from the Technical Annex.

AMETIST intends to contribute to solutions for the growing industrial need to design reliable and efficient time dependent systems. In particular, it intends to provide theory and tools for error-detection, control and optimisation of real-time distributed systems. Its approach will be based on translating state-of-the-art academic research into methods and tools that can be a basis for an industrial design practice of such systems.

In addition to its technological contributions, AMETIST invests actively in knowledge transfer to the European industry of computer-aided timing analysis and design. Moreover, it is expected that the academic dissemination of the AMETIST research results will influence and advance the field of timed systems research, and (indirectly) contribute to the education of future generations of system engineers.

Whereas timed automata and the tools for their analysis are widely accepted in academia and are being used at hundreds of universities and research laboratories all around the world, they have yet to find their way into industry. The aim of AMETIST is to advance and mature the related models, tools, and methods to allow this situation to change.

The need for automatic tools that allow reasoning about time is evident. Beyond manufacturing, telecommunication and hardware, it is of essential importance for the growing market of embedded systems (from car electronics to home automation). However, there are several obstacles that seem to hinder the use of timed automata technology in industry at this time:

- Scalability: Currently, tools based on timed automata do not allow to handle big examples. There are industrial scale examples that have been treated with these tools but only after tedious manual simplification involving a lot of work in each case.
- Convenience: Current timed automata tools are stand-alone programs and their input formalisms lack important features for convenient specification in an industrial setting.

- Accessibility: To make optimal use of the currently available tools requires quite some sophistication on the user’s part, which makes them practically inaccessible even to well-trained engineers.

AMETIST aims at the (at least partial) elimination of these obstacles. The project moves towards this goal along several tracks. One is the treatment of real-life case studies from some candidate application domains to see if, indeed, the proposed models, tools and methodology are suited for them. Indeed much of the project’s resources are being spent on case studies. A second direction aims to improve the situation regarding scalability, by introducing better algorithms and data-structures to model and manipulate large systems, in particular in the area of real-time controller synthesis, planning and scheduling. Moreover, the project aims at tool interaction to allow the interfacing of different tools, which can help to improve usability/convenience. The third track aims at synthesizing the accumulated results in order to assess the applicability of the project’s vision and modify it according to feedback from the field.

3 Project Methodologies, Results and Achievements

The purpose of this section² is to summarize the developments that took place within the AMETIST project and put them in a larger scientific and technological context. We start with a general overview of *model-based* design and analysis of systems from which the approach advocated by AMETIST has emerged. The principles underlying this approach are those underlying the verification of reactive computer systems. We then move to the more specific goal of the project, namely the exportation and adaptation of this approach to a wide class of systems where *quantitative timing information* plays a major role. It happens sometime that such systems are treated by techniques that do not follow these principles for one or more of the following reasons:

1. The communities in question follow their old ways and have not assimilated the computer science way of looking at such problems;
2. Being “semantically correct” is not the most urgent issue in these problems compared to being able to express many *real-life details* and treat hard computational problems. It is sometimes believed that taking the trouble to formalize cleanly one does not progress toward a solution and often adds another level of artificial complexity.

Finally we mention some of the major achievements of the project and assess their contribution toward fulfilling that vision, or perhaps adapting it to real life in a more realistic manner.

²The material in this section is based on [53, 52, 48, 45, 46].

3.1 Model-Based System Development

A large part of engineering and applied mathematics is concerned with building *mathematical models* of systems and using these models to validate the correct functioning of the system and to choose between design alternatives in order to optimize system performance. The nature of the system in question determines the types of mathematical models that are useful for its analysis. The internal operation of a combustion engine will be modeled by partial differential equations, the car dynamics by ordinary differential equations and an automatic driver by, say, a finite-state automaton.

The fact that a class of models is used in an application domain is not always correlated with its adequacy for solving the domain's problem. In many cases it is a combination of the latter with historical and cultural coincidences. Practitioners, in general, do not have the time to build new clean and rigorous models.³ They either use what was invented by theoreticians in the past or develop ad-hoc (and, sometimes, ingenious) models that allow them to solve the concrete problems they face within the time frame they have been allocated. It usually requires the intervention of theoreticians in order to clean and generalize these models.

Academics, on the other hand, tend to live in an imaginary world and have the privilege of being allowed to ignore the feed-back from reality about the short- and mid-term relevance of their models. Hence they can publish inertial papers that solve problems whose only significance is internal to the academic community to which they belong.⁴ This was not the main goal of AMETIST but rather the establishment of timed automata as the underlying model for a large class of problems and application domains in the same sense that differential equations underly a large part of physics and traditional engineering, or that transition systems are used in software and hardware engineering.

The class of models advocated in AMETIST has its origins in the domain often called *formal verification* whose goal is to prove that certain systems behave correctly for *all the external contexts* in which they can find themselves. Essentially these are models of discrete dynamical systems whose most notable features are:

1. The different *components* of the system in question are clearly identified and the model is given in terms of a *composition* of simpler models.
2. The *interaction* and mutual *influence* between the components is easily visible from the interconnection scheme of the components.
3. The *external environment* of the system is also modeled as one or more components of the same kind.
4. Each component is modeled by an *automaton*, the archetypical model of a *discrete dynamical system*. The state transitions are labeled (and enabled) by *interaction*

³In retrospect, one may say that the real world does not always admit clean and rigorous models.

⁴This is not meant to undermine fundamental purely-theoretical research. Certain (but not all) mathematical objects are worth being studied for their own internal sake.

conditions between the components and by *input* and *output events*.

5. Putting all components together yields a *global automaton* in which the origin of each transition can be traced back to the participating components.
6. The global system model is the basis of all design activities such as *simulation, validation, evaluation and optimization*. These activities are supported by tools that provide for automatic and semi-automatic analysis.

This approach has been extended in the last decade to treat time-dependent behaviors using the *timed automaton* model, proposed by Alur and Dill. The project aimed at establishing this framework as a unifying model for a large class of timing related problems and to *scheduling* in particular.

To contrast this approach we discuss the ways these problems are treated today in industry and academia. This is an intentional straw man, whose description is quite caricaturized but, nevertheless, it may represent the existing spirit in some areas. In many application domains the models are of a more ad-hoc nature without making a distinction between the essential and the accidental details of the problem. The latter may be the syntax of a given programming language, scheduling policies of a given operating systems, etc.⁵ This approach may lead to practical solutions but not always to solutions which are scalable to more general problems. The prime example in the success of Science, Newtonian mechanics, abstracted from the specific details of masses, small stones and planets and gave a general rule governing the behavior or both, although every practitioner will tell you that the “application” of throwing stones and the “application” of predicting where the stars will appear next evening, are quite different. A more down-to-earth example is the relative success of formal verification which can be partly attributed to the use of the transition system model (automaton) which abstracts away from such details. Of course, one needs at the end to treat such details in order to solve concrete problems, but much of the insights obtained on the more abstract model could not have been reached on models cluttered with details.

Another feature of commonly-used approaches is the modeling of the system in a form which is already geared toward a *particular solution technique*, although this is not always the most natural and suitable model for the problem at hand. To take an example, someone who is accustomed to linear programming will tend to model problems as linear optimization and if the phenomena persists and refuses to be modeled that way, there will be still an attempt to keep the models close to linear programming, e.g. by adding integer variables. This approach, summarized by the proverb “*When you have a hammer, everything looks like a nail*”, can be justified for practitioners who have to solve problems routinely and quickly⁶ and it may be useful in the short-term, but cannot be recommended at times of

⁵One may argue that these details, which the theoretician will call accidental are what makes real problems (and life in general) so complex, horrible and wonderful, but the role of science is to generate useful abstractions whenever possible.

⁶After all, scientists do not change their favorite tool too often.

“paradigm shift” where *new phenomena* are to be modeled, or when two phenomena which used to be modeled separately need to be modeled together.

We strongly believe that *phenomena come first* and that it is more useful to understand them and devise new formal models whose semantics corresponds faithfully to these phenomena rather than to rush and translate them into one’s favorite computational problem. An attack of the computational problems associated with the system analysis should follow only *after* the nature of the problem is understood. To be fair to industry, we should not forget that researchers and engineers work on different time scales and contractual constraints, and the re-use of what have been already done in the past might be the only way to meet deadlines.

It should be noted that when we say “semantics” or “timed automata” we do not refer to 90% of the work published in these domains. By semantics we do not mean fancy formalisms full of Greek letters and complex definitions. We pick from it the basic idea that a system description, *any system description*, denotes a *set of behaviors*, and that these behaviors are the objects of study, those according to which we *evaluate* system *correctness* and *performance*. Likewise, by timed automata, we do not necessarily refer to a particular definition, analysis method or a tool, but rather to the more essential mathematical model of a discrete dynamical system with clock variables.

3.2 Project Results: Expressiveness, Performance & Tools

At the end of the AMETIST project we can try to summarize the achievement along four major axes, *expressiveness*, *performance*, *tools* and *applications*. Expressiveness, is the big promise of TA technology, which is claimed to facilitate the rigorous formulation of complex planning and scheduling situation which cannot be expressed naturally within the standard academic models of operations research, and which are solved in practice using somewhat ad hoc methods. Performance is the other side of the coin, an ongoing fight against the complexity inherent in TA-based analysis and synthesis techniques, a complexity which grows as models become larger and richer. Expressiveness and performance should materialize in *tools*, and the effectiveness of these tools is demonstrated and assessed by applying them to serious (industrial) case studies.

Below we first present three main results of the projects related to expressiveness, then three main results on performance, followed by a discussion of our results in the areas of tools and applications.

3.2.1 Priced Timed Automata: Theory, Algorithms and Applications

In ordinary timed automata (TA) essentially only time can be optimized as it can be represented by an additional clock that measures its value. There are many situations in which the relevant cost functions are richer, involving cost variables that grow at *different rates* at different states or are associated with certain transitions. Some motivating examples are memory and power consumption in computers, setup and machine occupation costs

in manufacturing as well as non uniform penalties for missing deadlines in systems with *soft constraints*. Priced timed automata constitute a natural extension of timed automata with such cost variables. Although the dynamics of these variables renders the automata *hybrid* rather than timed, these variables do not really participate in the rest of the system dynamics (they do not appear in transition guards) and many problem are still decidable for this model. Some new significant decidability results and algorithms for priced timed automata have been obtained in the second and third years of the project and have been integrated into to UPPAAL CORA tool. We provided templates for modelling classical scheduling problems extended with costs, such as job-shop scheduling, task graph scheduling, aircraft landing, and vehicle routing with time windows. Also, we demonstrated the new modelling framework on some industrial scheduling problems, i.e. steel and lacquer production. The development of the priced timed automata framework is one of the major scientific results of the AMETIST project[92, 91, 86].

3.2.2 Stochastic Extensions of Timed Automata

Probabilities can be added to timed automata in two ways. The first is by adding probabilities to transitions, an extensions which transform reachable sets from unions of zones into probabilities on zones. The second and more challenging extension is to replace delay bounds by probabilities on delays, similar to the way this done in continuous time Markov chains. Numerous topics related to both extensions were investigated within the project leading to results that clarify some of the relation between timing and probability, in particular for continuous-time Markov chains and branching time [43]. A stochastic variant of the Axxom case study has been treated. [102].

3.2.3 Modelling Frameworks for Scheduling Under (Discrete) Uncertainty

At the beginning of the project, one class of problems of scheduling under uncertainty was explored, where the uncertainty is associated with task *durations*. For this problem interesting results that propose an optimality criterion more refined than simple worst-case performance. A scheduler synthesis algorithm was implemented for this class of problems and the schedules it generated on simple examples turned out to be much more *adaptive* to the *real duration* of tasks than other types of solutions [2, 1].

In the second and third year the complementary problem of scheduling under *discrete uncertainty* has been tackled. It covers the situation where the choice of tasks that need to be executed may depend on the *results* of other tasks, results that become known only after the termination of these tasks. Such situations are very common in scheduling of real time programs, where the results correspond to testing conditions inside *if* statements, but it can also be found in manufacturing, for example when certain production steps may terminate successfully or fail. A modeling framework for this problem using *conditional dependency graphs*, transformed into timed automata with discrete adversaries have been developed. Several exact and heuristic algorithms for synthesizing optimal and sub-optimal scheduling policies for this problem have been implemented. The most recent progress with

heuristic depth-first search allowed us to synthesize adaptive schedulers for problems with 400 tasks and up to 20 conditions [113].

3.2.4 Optimization and Constraint Satisfaction as a Tool for Timed Automata Analysis

During the second and third years the applicability of mixed integer linear programming (MILP) as a tool for TA analysis has been explored. An interesting observation is that relaxed models (with “integers” interpreted as reals) may sometimes give more interesting lower bounds on the costs that extend a partial solution and hence can be used for this purpose with reachability-based algorithms. All these results were integrated into the prototype tool TAOPT [244, 245, 246].

For bounded horizon problems, questions related to timed automata reachability are transformed into *satisfiability* problems for *difference logic*. Scheduling problem do translate naturally to this logic without going through automata. During the project we have developed a series of solvers for this logic, culminating in the current version of the performant solver DL-SAT [134] and jat [135]. The tool ELSE [283] can serve as a front end for such solvers by generating efficient difference logic formulae (exploiting partial-order ideas) that correspond to bounded model-checking formulations for timed automata.

3.2.5 Improvements in Timed Automata Model Checking Technology

The basic cycle of TA verification tools consists of taking a zone (a conjunction of difference constraints on clocks) and applying to it some operations in order to produce its successors on which the same process is iterated. The number of these zones and the size of their representations is a major bottleneck for TA verification. Zones are typically represented as difference bounds matrices (DBMs) of size quadratic in the number of clocks, and it has already been known that their dimensionality can be reduced in each state to the number of clocks active in that state. More recent work in AMETIST shows that performing a finer analysis of the structure of the TA, may yield for some states DBM representations which can be as good as linear in the number of clocks [83, 84]. Among the other important contributions to improving performance of TA tools we mentioned ideas inspired by partial order methods [5, 227], symmetry reduction [178, 180], and more clever memory management [90] during exploration. By implementing these new ideas, the project succeeded to improve the performance of UPPAAL by several orders of magnitude [88, 51].

3.2.6 Specialized Technology for Solving Scheduling Problems with Timed Automata

The zone based technology has its roots in verification where the temporal uncertainty is viewed as coming from the *external environment* and the system should be correct with respect to *all* environment choices. Around the beginning of the AMETIST project it was

observed that when uncertainty is associated with the scheduler decisions, for example in problems of scheduling under certainty, sometimes there is a unique successor among the uncountably many which gives the optimum (“non-lazy” schedules) [2]. Consequently the problem can be solved without using zones at all but rather using *vectors* of clock variables. This way certain problems can be formulated as shortest paths in *discrete weighted graphs* and solved much more efficiently. They can also benefit from existing search algorithms for on game graphs in order to find sub-optimal schedules for scheduling with discrete uncertainties.

Even in the case of dense uncertainty coming from the environment/adversary side, there might be some clever ways to avoid zones. When the adversary has a choice in some interval $I = [a, b]$ we may still relax the problem by assuming only a finite subset I' of the interval. Solving the problem of synthesizing an optimal scheduling strategy with respect to I' may have the following consequences:

1. The actual value of the chosen strategy with respect to I may be *worse* than the value computed based on I' . This is more problematic for *qualitative* criteria where the system may be correct with respect to I' but not with respect to I . For quantitative criteria this is less critical because we already accept sub optimal solutions when the problem is large.
2. During execution the adversary can make a choice in $I - I'$ and the system will find itself in a state for which the optimal action has not been computed. However in many problems some default rules can be used to determine the action at such a state based on the optimal computed action in its neighborhood.

3.2.7 Tools

AMETIST has made major advances in the area of (timed automata based) tools. Several (new versions) of tools were released, implementing algorithmic ideas that have been developed during the first two years of the project. Tight connections and interfaces between all of these tools exist or are currently being developed. Figure1 presents an overview of the tools that have been developed (or used) within AMETIST (we refer to Deliverable D2.5.b [51] for a more detailed overview of their functionality). We have classified these tools along two dimensions: (1) the total number of AMETIST PM's spent on their development during the whole lifetime of the project, and (2) the “maturity” of the tool on a scale of 0 (academic prototype, minimal user interface, for internal use only), to 1 (industrial, commercial, good support, widely used). All numbers are rough estimates. Assessing maturity, of course, is a difficult exercise anyhow.

As becomes clear from the figure, there is a spectrum ranging from tools such as UPPAAL, which have a very nice GUI, are easy to use, and although academic have almost industrial quality, to highly experimental tools such as ELSE which helped us to test the viability of some new verification approaches (partial order reduction, symmetry, model checking via constraint solving,...).

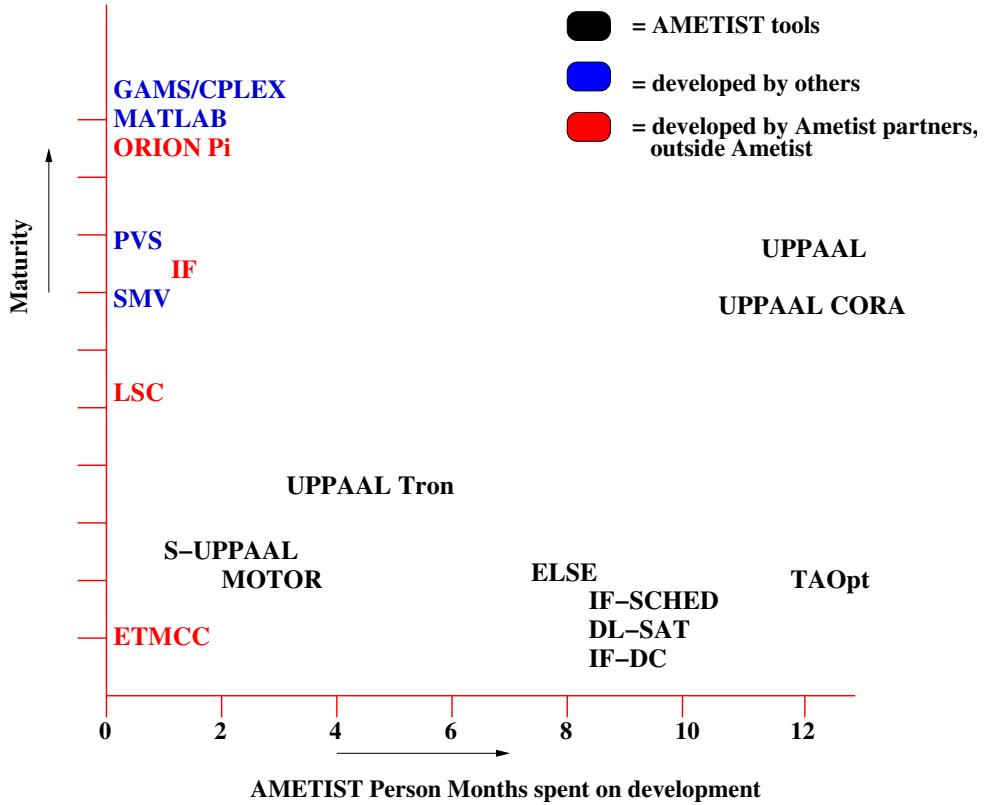


Figure 1: Overview of the tools developed/used by AMETIST.

If we compare the current capabilities of these tools with what existed at the start of AMETIST, then it is fair to say that indeed we have moved the state-of-the-art to a new level of maturity (one of the main objectives of the project). At the start of the project, only UPPAAL existed and some precursors of UPPAAL CORA and IF-SCHED. In terms of scalability, convenience and accessibility UPPAAL was much less mature (it was about as mature as UPPAAL Tron is right now). The performance of UPPAAL has improved several orders of magnitude. The following major changes (not easily quantifiable as a single number) in particular boosted the performance:

- New extrapolations: Several orders of magnitude for some models (e.g. Fischer).
- New internal representation of states: Approximately factor 3 memory reduction.
- Liveness checker and deadlock checker: Speedup of factor 2-4.
- Remodeling with extended subset of C: Depending on the model this may provide a speedup of factor 2.
- Optimizations for handling models with a huge number of concurrent components and optimizations for handling systems with a huge number of deadlocks: if applicable, one order of magnitude.

For CORA the change in how the infimum operation is computed gave about one order of magnitude, and other CORA specific optimizations have provided another speedup with a factor 2-4 (depending on the model). Also the convenience of using the tool has been greatly improved due to the enriched expressiveness of the input language and connections with other tools. Nowadays even high school students find it easy to modify and build timed automata models using the improved GUI. Through the website and new tutorial papers (such as [94]) the tool has become very easily accessible.

3.3 Project Results: Cybernetix Case Study

The Cyberntix case study treats coordination aspects of a smart card *personalization* machine. This machine is one element in chain of relatively compact machines for smart card productions.

The particularity of the machine HPX4000 is the hardware for transporting cards through the machine, a patented combination of a conveyor belt (which results in a linear design of the machine) with mechanical elements to lift cards to positions with programming interfaces (programming stations). This allows to perform the time consuming programming step on several cards in parallel.

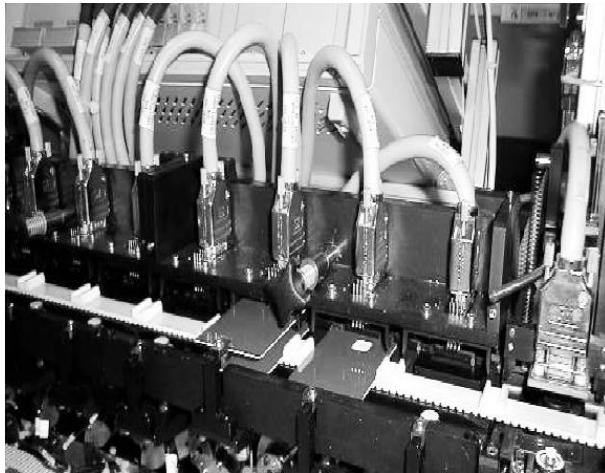


Figure 2: The conveyor belt and programming stations

The machine is designed to work on batches, i.e. it takes piles of *raw* cards and produces piles of programmed and printed cards. However, production need follow a rythm of one batch at a time. The principle design goal for this machine in a very competitive market was to build relatively small machines (fitting into normal rooms) with a high throughput. For personalization, one of the determining factors is the programming or personalization time. For an increased throughput, a parallel architecture with several programming stations is an obvious solution. Getting the cards from the input stations to the programming stations, from there to the printing stations and from there to the output, while removing cards failing the tests requires an efficient transport system. For this transport system,

Cyberntix invented and patented a dedicated conveyor belt for moving a sequence of cards and a mechanism for lifting cards from this conveyor to a programming station above. In particular, the mechanism allows to move the conveyor while a card is lifted and other cards may pass on the conveyor below. The second challenge of the design is the development of a scheduling algorithm that routes the cards through the system on this given architecture in an efficient manner. Cyberntix found a particular algorithm for this baptized *SuperSingle Mode*, which is equally part of the patent for the conveyor system.

A second concern for the scheduling focuses on faulty cards. A certain fraction of the cards contains electronic defects not detected before actually programming the cards. A defective card must be replaced by a new card without modifying the order of output and at the lowest possible cost. For the SuperSingle Mode, Cyberntix invented a recovery method that actually required to modify the machine design (the length of the conveyor), which has been thoroughly tested in various failure scenarios and never failed. But the method was not strictly understood to be correct under any circumstances nor known to be optimal.

3.3.1 Activities in year one and two

The case study was presented to the AMETIST consortium by a detailed description with timing parameters [12] and a first formal model [11] was provided to the consortium for reference.

Automatic synthesis. The consortium took up the challenge with a surprising amount of energy and several groups produced competing models for the machine and the related scheduling, of two of which are fully documented [169, 251]. The remarkable achievement of these works is that the patented “SuperSingleMode” was synthesised for small problem instances (few programming stations, few cards) by fully automatic means. However, it was quickly remarked that the fully automatic approaches do not scale up on the models, and little additional progress was made in this direction until year three.

Performance evaluation. One of the original questions of the case study concerns the optimality of the SuperSingle Mode: Is it also optimal or, if that can not be answered, how far is it from optimal throughput.

To approach this question, in [13], an actual performance evaluation of the throughput of two modes, the “BatchMode” and the “SuperSingle Mode” was done and parametric formulae (with various delay parameters) were derived. This allowed to visualize the actual performance of these modes. While trivial upper bounds are easily established, an enigma of the SuperSingle Mode is its use of *holes* on the conveyor. By a sophisticated modeling technique, we managed to prove that the Super Single mode is very close to optimal throughput, with a maximal error of 1% for typical parameter values.

While the latter evaluation was done manually, [229] proposed a modelling with timed automata using UPPAAL for automatic performance evaluation: Rather than synthesizing

a certain scheduler, it is possible to evaluate the throughput of a given scheduler with UPPAAL efficiently.

A very interesting aspect of this work is the proposition of an alternative architecture of the Smartcard machine: Instead of adjacent programming stations, Mader proposes to have such stations only every second position. This architecture allows for a completely different, significantly simpler, yet well performing operation mode and was a welcome surprise to Cyberntix engineers.

Thus, a high degree of understanding of the HPX machine was achieved.

Simulation and Play-in play-out In order to render the complex combinatorics of the HPX machine more accessible, an interactive simulator based on Java was developed [9] and continuously extended until year three. The simulator implements different configurations of the HPX machine, including error handling. Among other aspects, it allowed to discover certain aspects that are difficult to imagine theoretically, like a fundamental change in machine behaviour when passing from four to eight programming stations. The simulator was also very useful for debugging other models.

On the other hand [280] showed how to obtain a controller for the HPX machine with the Play-In/Play-Out approach for Live Sequence Charts (LSCs). LSCs are a formal graphical inter-object scenario-based language. In comparison to other models, LSCs models are easier to read and maintain due to its scenario based nature. The PlayEngin moreover allows to add visualisation plugins. Thus, in [280], a simplified representation of the HPX state is visible while “teaching” the machine by (test) “cases”.

3.3.2 Work in the third year

The third year of the case study turned out to be difficult, as the economic trouble hitting Cyberntix group resulted in the sell-off of the microelectronics branch that had developed the HPX machine. This had two consequences : On the one hand, the AMETIST consortium no longer had access to the engineers developing the machine, on the other hand, the management of the remainder of the company struggling for survival showed very little interest in pursuing the case study, which put the PhD student assigned to this task in a very difficult situation. Nevertheless, we decided to pursue the work on the basis of previous understanding.

A second problem arose when the thesis project was finally aborted: An important part of the results actually achieved in the third year go without documentation. We nevertheless try to summarize the work done in [241] and document what turned out to be the most important discovery in the last year, i.e. a very good abstraction for the smart card machine.

In [241], we report two contributions to the case study:

- A new abstraction technique allowing to break down the state space of the HPX model to a significantly reduced set of states : basically, the abstraction is based on *passing*

from absolute to relative card identities and it allows, different from previous models, the verification of unbounded production (no limit to the batch size). Nevertheless, combinatory explosion remains and the problem is exponential in the number of programming stations.

- A model of the error handling as used by Cyberntix and our verification efforts using Verimag's IF tool.

It is obvious, that the first contribution significantly improves the prospects of the second. With a model consisting of three parts, (1) the raw machine and cards, (2) the controller (a model of Cyberntix control program) and as a third part, (3) the error model, we managed to verify the error handling mode as used by Cyberntix. In simulation, this model behaved as expected and the IF-verifier gave the expected yes/no answers to verification questions depending on the error model for cases with a bounded number of defective cards that could show up nondeterministically in any order.

Unfortunately, the IF-verifier was not capable of extracting error traces, due to the structure of the μ -calculus queries (the required feature was not fully implemented at the time of our experiments).

The simulator [9] was extended by a switch to choose between absolute and relative card identities from the new abstraction. This extension thus represents an implementation of the abstraction reported above. Moreover, the error handling mode as implemented by Cyberntix was added to the simulator.

3.3.3 Conclusions

The Cyberntix case study was very inspiring to the project, in particular during the first half of the project. It exposed at the same time the strength – modeling and automatic resolution of small problem instances – and weaknesses – most notably concerning complexity issues when scaling to bigger problem sizes – of the consortium's technology. The results were considered with a lot of interest by the industrial partner.

With exception of the new abstraction reported here, one can safely state that a full comprehension of the case study was achieved after about 18 months and that, consequently, the interest of the consortium shifted to harder and unresolved challenges of other case studies. One factor that certainly played a role in this was also the restructuring of the company, so that the results of the consortium were no longer fed back to the department developing the machine.

The abstraction reported in [241] was communicated to the consortium early in the third year, but since the focus was mostly on other case studies, a reevaluation of the experimental results obtained in the first and second year was not considered. Nevertheless, it is obvious that with this abstraction, all experimental results achieved could be seriously strengthened.

Concerning the original goals of the case study, important parts were achieved already in the first half of the project. The impression that we did not have the right abstraction

technique for conquering the complexity of the machine, was relativized by the new abstraction found in the last year, which falls into the class of symmetry reductions. While symmetry reduction was introduced into UPPAAL, the kind of symmetry in question here would not be handled automatically.

In summary, the Cyberntix case study can be considered a success, a proof that the use of AMETIST technology can lead to new insights into a complex production problem.

3.4 Project Results: Terma Case Study

The Case Study on Memory Management in Radar Sensor Equipment has been provided by the partner Terma A/S who is developing and producing (amongst other things) radar sensor equipment typically used in areas where high performance is needed (monitoring of ground movement in airports, coastal surveillance, etc). The radar transceivers of Terma have a wide range of video processing facilities and in this case study the purpose is to model, analyze and synthesize the scheduling of memory access of two of these processing facilities. We refer to [82, 97] for a complete specification of the case study.

3.4.1 Summary of the work

During the first year of AMETIST all aspects of the case study was dealt with in substantial detail, including:

- An encoding of the existing scheduler in the BDD-based verification tool visual-STATE allowing correctness to be verified partially [82, 97].
- A sequence of encodings of the existing scheduler in UPPAAL leading to a complete verification of the correctness of the existing scheduler. Given the size of this model (21 timed automata and 20 integer variables) the verification time (less than 1 minute on a regular PC) is encouraging⁷. An initial UPPAAL model can be found in [256].
- Design of an optimal scheduler with respect to required size of buffers. This (general) design was obtained from experiments with an SMV model of a down-scaled scenario [279].

These results were presented to the hardware engineers at Terma in April 2003 with very positive response from the company and with suggested directions for further treatment. Thus, during the second year we have

- provided an extension of the UPPAAL model detailing the initialization and in particular the SDRAM structure⁸. This model was discussed with Terma A/S on the

⁷The model can be downloaded from <http://www.cs.auc.dk/~kg1/Terma/terma-tsh.xml> and <http://www.cs.auc.dk/~kg1/Terma/terma-tsh.q>.

⁸See <http://www.cs.auc.dk/~kg1/Terma/terma-banks.xml> and <http://www.cs.auc.dk/~kg1/Terma/terma-banks.q>.

AMETIST meeting in September 2003 and found to be quite compatible with the VHDL design⁹.

During the final year of AMETIST we have finally succeeded in

- automatically synthesizing the optimal schedule in terms of required buffer-size from [279] directly using the UPPAAL models [157, 156]. This work was carried out by Juhani Ernits, Tallin, during a Marie Curie Fellowship at BRICS autumn 2003 and finished during 2004. The method is based on an interesting (simple and efficient) synthesis method applying bit-state-hashing.

3.4.2 Conclusions

Concluding on the case study, we notice that the techniques and tools developed during AMETIST have been successfully used to *verify* the existing design provided by TERMA and *synthesize* optimal solutions to the scheduling problem, that would allow TERMA to completely eliminate the current large buffers used in the design and replace the current logic to dynamically arbitrate between many channels with a small precomputed fixed schedule. Although the system described in this case study is no longer under development, and thus direct application of our solutions are unlikely, we are confident that our approach could be applied to similar designs.

3.5 Project Results: Bosch Case Studies

The original case study that was proposed by Bosch as a challenge for the AMETIST project concerned the analysis of a Car Periphery Supervision system (CPS). At the beginning of Year 2 of the AMETIST project, Bosch decided to put the further development of the CPS system to a halt. Consequently, it was decided that also within AMETIST the work on the CPS system would be finished after completion of a paper summarizing the results on this case study [168]. An alternative case study was proposed by Bosch concerning the development of a SRS (Supplemental Restraint System) for an airbag ECU (Electronic Control Unit) to be tackled by the consortium during the remainder of the project. Below we will summarize the results obtained for both case studies.

3.5.1 The CPS Case Study

First, we briefly summarize the work carried out within the AMETIST project on the case study “Real-time service allocation for Car Periphery Supervision”. Part of the text below is taken from Deliverable 3.3.3 [31]. For a more extensive overview of the results obtained for this case study, we refer to [168].

⁹See http://ametist.cs.utwente.nl/RESEARCH/AALBORG/GERD/VP3_STHArbitrationindex.htm

The term *Car Periphery Supervision (CPS)* refers to technology for obtaining information about the environment of a car. Such technology can serve as the basis for many driver assistance services, e.g., parking assistance, pre-crash detection, blind spot supervision, lane change assistance, etc., most of which are still under development. There are different sensor technologies available for CPS realizations, e.g., ultrasonic, radar, lidar, infrared and video. In the case study we concentrate on Short Range Radar (SRR) technology. In Deliverable 3.3.1 [205], a preliminary description of the CPS system was provided by Bosch, as well as a list of timing related problems.

An initial study by partners UT and KUN made it clear that the initial description in [205] contained insufficient information to make formal analysis possible. In particular, the system requirements, assumptions about the timing behavior of the various system components, and a listing of the assumptions on the environment (e.g., number and relative speed of approaching objects) were unclear. Based on the discussions with the domain experts from Bosch, study of relevant literature, and use of MatLab to visualize the sensor visibility areas and to analyse possible trajectories of approaching objects, we were able to supplement the preliminary description of [205] with the necessary information to permit formal modelling and analysis [166, 167]. Our results show once more the importance of such an analysis: they allow us to make the right abstractions in the dynamical model of the system, and to interpret correctly the results obtained by analyzing this model. The next step we took was to actually construct a formal dynamical model using UPPAAL. In this model the environment, the sensors and the ECU are all modelled as timed automata. This is achieved by splitting the value domains of nonlinear continuous variables into a finite number of regions, and overapproximating (with the help of MatLab) the possible behavior of the continuous variables in the timed automata model. In Deliverable 3.3.2 [19], we reported on our modelling efforts.

The efforts during the second year of the project concentrated on verification. By encoding the various assumptions on the environment within our UPPAAL model, we were able to establish some key correctness properties of the system (for specific choices of the system parameters):

1. The ECU has a sufficiently accurate view on what is happening in the environment.
2. When an object reaches the pre-crash region, the ECU knows about this within a small (specific) number of time units.
3. The ECU avoids false alarms.
4. The system is deadlock free.

Using the convex-hull approximation feature of UPPAAL, the time required to verify these properties was less than a minute. A paper describing our model and results has been presented at WODES'04 [168].

Conclusions The original objective of the case study — as stated in the Technical Annex — was to use the timed automata framework to specify precisely the logical and timing requirements for the CPS system, and to design and verify (or alternatively, automatically synthesize) a dynamical resource allocation scheme on this basis. The original objective has been achieved, although the model of [168] is probably somewhat more abstract than what was originally intended, and our efforts were restricted to verification (rather than design and/or synthesis). Building more detailed models of the CPS system, and contributing to the design and synthesis of resource allocation schemes, made no sense given that the work on CPS had been abandoned by Bosch.

The basic approach that was taken in this case study — over-approximation of a hybrid system by a timed automaton — is well known and has been studied by several authors (see [168] for references). Nevertheless, there are not too many practical, realistic problems to which it has been applied. Our work demonstrates the practicality of the approach on the basis of a nontrivial, industrial case. The input language of UPPAAL turned out to be sufficiently expressive to allow for convenient modeling of the CPS system. The case study could have been carried out with the version of UPPAAL that existed before the start of AMETIST. So the technology and tools developed as part of AMETIST were not essential to solve this problem. However, in the model the feature of “broadcast channel” has been used, which helped to make the model more natural and compact. Use of the convex-hull approximation was essential for the successful verification with UPPAAL: without it the model becomes intractable due to the different time scales of the various components. An alternative verification approach (but is a topic for future research) is to apply a variation of the exact acceleration approach developed by Hendriks and Larsen [181] in Task 2.3 of the project.

3.5.2 The Airbag ECU Case Study

Much of the work on this case study was carried out by Marko Auerswald from **Bosch**, who provided an initial timed automata model of the airbag ECU with additional information about the behavior of each of the components [64]. Following the second review it was decided to limit the resources for this case study to a minimum. As a result, only Conrado Daws from **KUN** spent a couple of weeks on verification of the timed automata model. **KUN** intends to continue working on the case outside the scope of AMETIST.

The goal of the second **Bosch** case study is to explore — in the course of a pilot project — the suitability of using timed automata-based modeling for increasing product quality in a cost effective way. The pilot project is attached to a current platform development project within **Bosch** of an airbag ECU. Within the development life-cycle, **Bosch** identified four potential areas to be explored by the project:

1. **Specification of real-time behavior.** During ECU architecture definition, components and their behavior have to be specified. The benefit of using timed automata as the formal description of the behavior of such components is to serve as documentation, replacing natural language specifications, providing more clarity and

comprehensibility, thus simplifying the communication between development teams.

2. **Validation of the architecture specification.** The timed automata models of the components can be used to simulate system behavior for specific scenarios of the environment. A scenario-based analysis of the architecture is a common approach to validate architecture decisions.

3. **Verification of safety properties in a distributed real-time control system.**

Safety critical functionality usually is implemented in a redundant way with diverse channels. These channels have a logical as well as temporal control of their own. In the example of the airbag ECU, several logical or physical plausibilization channels exist to avoid inadvertent deployment of the airbag. Nevertheless, these shall not hinder correct deployment.

Modeling the real-time behavior of the plausibilization channels, and the relevant behavior of the environment, will allow to check whether there exist specific sequences in the environment which could lead to conditions under which plausibilization channels hinder the correct deployment of restraint devices.

4. **Validation of the implementation.** From formal specifications of the real-time behavior of components, test cases can be derived automatically. An appropriate coverage criteria based on the behavior model can be formulated. Testing the implementation with this generated test set ensures that the implementation really implements the specified behavior and thus also fulfills the verified safety properties.

In the course of our work, we contributed to the first three of the stated objectives.

Modelling The timed automata model of the airbag ECU provided by Bosch served as a clear and easy to understand documentation of the case study for the other partners of the project. The fact that the industrial partner came up with a good model using UPPAAL shows that the tool user-friendly graphical interface and its rather intuitive semantics makes UPPAAL accessible to people from the industry.

Scenario Analysis One of the questions Bosch had was about the possibility of using UPPAAL to analyse certain scenarios of the environment which had been identified as being able to exhibit an undesired behavior of the system. It was possible to verify with UPPAAL using a very abstract model of the ECU, that the undesired situation could emerge, but without providing a compact and valuable characterization of the type “when event A is followed by event B within between x and y time units, then the undesired behavior occurs”. UPPAAL provides a concrete counterexample, i.e., a step-by-step error scenario, but leaves it to the user to extract a characterization of the “essence” of the problem. Such a characterization would be valuable not only to give a human readable interpretation of the problem, but also to assess the likelihood of such a scenario. Clearly, this is an interesting topic for future research.

Verification of Properties We considered three properties that the airbag ECU model should satisfy:

- **Property 1:** it is possible to fire the airbag.
- **Property 2:** the system does not deadlock (this a property of the model rather than of the actual system itself).
- **Property 3:** when both the microcontroller and the approver enable the firing of the airbag, then the firing should not be hindered, that is, the firing stage should be unlocked.

The first property, for which it suffices to find an execution that leads to the firing of the airbag, was easily verified with UPPAAL. The two other properties require an *exhaustive exploration* of the state space of the system, which is highly memory- and time-consuming. The verification of these properties pushed UPPAAL to its limit, and it was first necessary to consider an untimed version of the model (i.e. the same model with the timing constraints removed) to get a result, and then showed that it was possible to carry out the verification on the constrained model on a machine with sufficient memory.

Moreover, by applying the convex-hull approximation, which computes an over-approximation of the set of reachable states, it was possible to verify both properties almost instantly and on machines with little memory. It is known that over-approximation preserves the validity of both invariant and absence of deadlock properties, so it can be concluded that the model satisfies properties 2 and 3. Table 1 shows the memory and time consumption of UPPAAL for the verification of each property using exhaustive analysis and over-approximation on a PC with a Pentium(R) 4 processor at 3.40GHz and 1518 MB of memory.

Table 1: Memory and time consumption

	exhaustive		over-approximation	
	mem (KB)	time (s)	mem (KB)	time (s)
P1	276	0.10	0	0.10
P2	33156	91.10	2772	0.92
P3	32984	77.82	2600	0.41

Future work Although the initial model of the airbag ECU was successfully verified with UPPAAL, several issues of interest to Bosch remain to be considered in order to assess the benefit of applying the timed automata modelling and analysis techniques to an industrial system.

The first issue concerns the suitability of timed automata to obtain a model which is as close as possible to the real system without recourse to artificial modelling tricks imposed by

the specification language, and how clear the modelling of such a complex system would be without a higher level description language with, for instance, hierarchical concepts. Another aspect of the airbag ECU that Bosch would like to address is whether the verified properties are preserved if non-deterministic processing delays and jitter are added to the system model. A third issue, which was already mentioned above, is to develop algorithms that extract the “essence” from a counterexample.

Finally, we have not considered yet the application of formal techniques for the generation and execution of test cases generated from a timed automata specification to validate an ECU implementation. However, formal black-box timed testing techniques were recently incorporated into the real-time model-checker UPPAAL[219] and the test tool TORX. Prototype implementations of the airbag ECU provide good targets on which to apply these techniques, where tests generated from the (verified) timed automata specification are executed on the ECU prototypes. As the models in the case study are not just models of software but also ASIC behaviour, such an approach could only be applied using hardware-in-the-loop tests with ECU prototypes. This means that the testing has to be carried out on site, and it remains to be seen whether it would be possible to do it in the business unit of Bosch.

Conclusions Also the second Bosch case study could have been carried out with the version of UPPAAL that existed before the start of AMETIST, and so the technology and tools developed as part of AMETIST were not essential for solving the problem. Only very limited resources (< 1.5PM) from AMETIST have been invested in this case study. Nevertheless, we believe the case study is a very interesting and appealing. When the confidentiality conditions are no longer applicable and the model can be published, this will definitely be considered to be a nice illustration of industrial use of formal methods, and potentially as a challenging benchmark for the application of model based test generation methods. A notable feature of the model is that verification becomes extremely hard because of the timing constraints, even though there are only two clocks in the system.

3.6 Project Results: Axxom Case Study

The industrial partner AXXOM provided a value chain optimization problem which represents a typical planning and scheduling task as it occurs in the lacquer producing industries. Throughout the duration of AMETIST, this case study served as a case study to (i) develop and compare different modeling formalisms for scheduling problems, (ii) to evaluate the solution performance of the various approaches to scheduling problems investigated within AMETIST, (iii) determine which type of constraints are difficult or relatively easy to handle, and (iv) to further develop the methods based on the bottlenecks found in the previous steps. This report first contains the problem description including different versions of the case study, and then summarizes the modeling formalisms, solution approaches, and results for six different tools applied to the problem.

3.6.1 Problem Statement

Original Formulation The benchmark is derived from an existing pipeless batch plant in which three different lacquers are produced according to the following scheme [224]: The materials required to obtain a specific lacquer are first prepared in a pre-dispersion and a dispersion unit, and are subsequently filled into mobile mixing vessels. After completing the filling procedure through a set of dose spinners, the mixing is carried out for a product-dependent duration. A quality check determines if the product meets given quality requirements. If so, the mixing vessels are emptied into filling stations and the product is delivered to the customers – if not the quality is improved by returning the lacquers to the dosing operation. The plant comprises 8 production resources and 6 mobile vessels, where some resources and one vessel are only used for one type of recipe while the others can be assigned freely. The operations cannot be interrupted (non-preemptive scheduling) and the material remains in the mixing vessels during some of the steps, as the quality check in the laboratory, or possible additional mixing. This leads to a situation where the operation times of the vessels are variable because they result from the scheduling of the operations for this batch. For some operations, more than one piece of equipment can be used. Each production order has a release date (earliest starting time) and a due date (deadline). The main interest is in meeting the deadlines.

The problem of assigning the jobs (consisting of the aforementioned sequence of operations to obtain a lacquer) to the 14 resources belongs to the class of job shop problems, with the additional complication that mixing vessels are required as a second resource to perform the operations on some machines. Another property of the problem, which is not standard in job-shop problems, but often found in scheduling problems of the chemical industries, is the presence of constraints on the storage times. These constraints are expressed as bounds on the differences between the starting and the ending times of two operations of a job. Three types of such constraints occur: start-start, end-start, and end-end constraints. The time horizon of the entire problem defined by the earliest release date and the latest deadline of all jobs comprises approximately 7 weeks, and the processing times on the machines range from few hours up to three days for laboratory testing. The objective function of the scheduling problem combines costs for the delayed finishing of jobs, operational costs per amount of product, and storage costs (i.e. early termination of jobs is penalized). When minimizing this cost function, it has to be considered that the three different lacquers incur different production cost.

Extended Version In an extended version of the case study (set up during the second year of the project), a set of new problem instances has been defined, which either involve new constraints or additional production orders. The first extension introduces two new types of constraints: sequence-dependent changeover procedures and the interruption of operations during weekends, holidays and at nights.

Changeover procedures are defined to prepare the filling stations for new operations, i.e. to change the configuration of the resource. Such procedures often represent cleaning of

the machines. The resource is allocated in an exclusive fashion for 5 to 20 hours and the desired operation is started immediately afterwards. Changeover procedures must be invoked when the filling station has to process a different type of lacquer, i.e. when it has to enter a new configuration. The possible configurations correspond to the three basic recipes for lacquers. The effect on the schedule is two-fold: the start of the operation is delayed by the changeover and it causes costs.

Since some machines are operated in a 2-shift mode on working days only, appropriate modeling of interruptions during night shifts and weekends is required. Interruptions mean that the operations have to be stopped at the beginning of the given interruption period, and to be resumed when the next working day starts. During the period, the resource remains allocated and the current operation cannot be preempted by other operations. Some operations may be interrupted for at most 12 hours. Both extensions reflect the industry's need for solutions to realistic and challenging problems.

In the other problem instances, the number of production orders has been increased to 73, 219, 500 and 1000. As a new optimization criterion, the minimization of total costs of the schedule is the goal. For this purpose, different costs have been assigned to the storage of the final products, to the delay of finishing jobs, and to changeover procedures. These extended settings involve fixed costs as well as cost rates to be integrated over the duration of operations or quantities of the orders.

3.6.2 Modeling and Requirement Specification

It was observed in the initial phase of the project, that the academic partners interpreted the various constraints and cost contributions contained in the problem formulation not identically. It was deemed necessary to produce a representation scheme that formulates the requirements in an intuitive and unique manner. A systematic modeling scheme comprising the following steps was developed [230, 87, 86]:

- (a) create a dictionary to explain the domain-specific vocabulary used by the industrial problem providers,
- (b) resolve semantic ambiguities in the specification of the production sequence and constraints,
- (c) define adequate levels of abstraction with an explicit representation of the design decisions,
- (d) supplement the so-called *product flow diagrams* (provided by AXXOM) with recipe-like representations, and
- (e) systematically transform the latter representations into timed automata.

With respect to step (d), the product flow diagrams provided by AXXOM, which represent the production steps in a graphical arrow-node representation, were first enriched by the

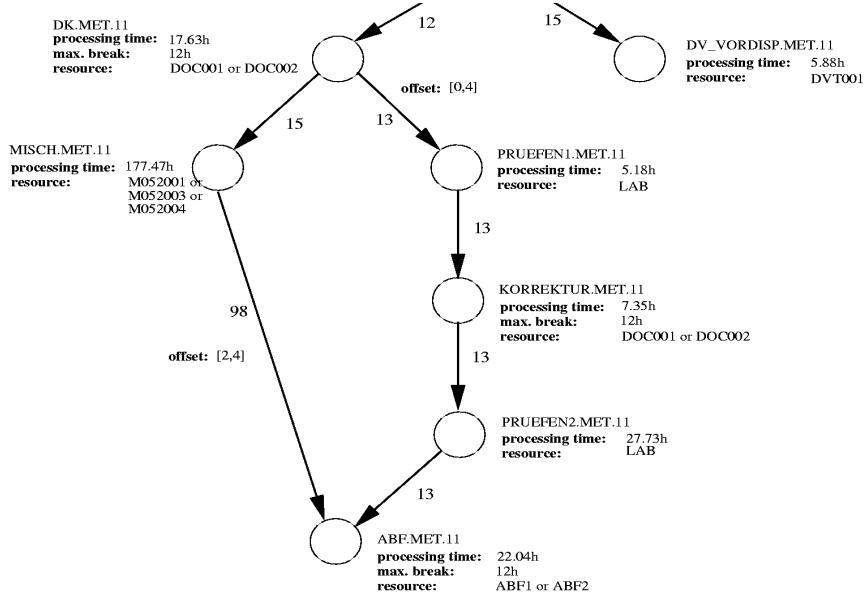


Figure 3: Example of a product flow diagram.

processing and offset times for the operations and by the information on the resources on which an operation can be carried out. (The latter data were originally provided in separate tables.) An example for such an extended product flow diagram is shown in Fig. 3 – a drawback of these diagrams is, however, that parallel and alternative operations cannot be distinguished in all cases, and that some timing restrictions have different meanings in different situations. It has thus been proposed in [87, 86] to supplement the product flow diagrams by the recipe-based representation shown exemplarily in Fig. 4. This representation illustrates in which sequence the different types of resources have to be allocated for a specific lacquer, and which timing requirements are relevant.

Based on these representations, the systematic transformation into Timed Automata is achieved relatively easily by:

1. translating each production step from the recipe representation into an automaton fragment (i.e. a sequence of states and transitions),
2. composing the fragments,
3. converting the timing constraints (e.g., the processing times) into corresponding guard and invariant conditions, and
4. modeling the resources by variables.

A different modeling approach, common in the industrial scheduling community, is the state-task-network (STN) formulation with subsequent solution as algebraic program by the use of mixed-integer optimization techniques. A standard representation of STN is one

with discrete time representation where changes of the states of the model can only occur at equally spaced time instants. As any possible event can occur at any of these instants (modeled by a binary variable), these models quickly become very large if the number of events is large, i.e. the scheduling horizon is large compared to the maximal resolution required to meet the constraints. For the given case study, this ratio becomes rather large and hence such models are not suitable here.

The alternative approach for algebraic modeling followed in this project is the use of continuous *event points* at which an event (e.g. the start or the end of an operation) can occur [22]. The number of such events can be computed from the production orders and the recipes (plus eventually a worst-case estimate for sequence dependent operations such as cleaning operations). In the example considered here, there are typically not more than a few hundred events whereas a discrete time STN formulation leads to several hundred thousand binary variables. On the other hand, though, the relative positions of the events must be represented by binary variables in order to express constraints of the type that two operations cannot be performed on the same machine at the same time (but either operation A must finish before B starts or B must finish before A starts). It turned out, that a standard formulation of a continuous time MILP model also gives rise to rather large models with thousands of binary variables for problems with more than 10 jobs. Therefore the model is reduced by additional heuristics, which exclude alternatives that cannot lead to an improvement of the schedule. Thus, the heuristics, which are described in more detail in the next section, eliminate a considerable number of binary variables, such that the total number of binary variables increases only linearly with the problem size.

3.6.3 Tool Application and Solution

Mixed-Integer Linear Programming with Cplex In order to solve the algebraic program sketched before, the following heuristics were employed [248]: (i) pre-ordering by earliest due dates (EDD), (ii) non-overtaking of operations belonging to jobs of the same type, and (iii) non-overtaking of operations whose start and finishing dates do not overlap. These heuristics were applied in a two-step-procedure as follows: First, the heuristics (i) and (iii) are applied by fixing some binary precedence variables, and the problem is solved. In the second step, the variables fixed before are relaxed, and the heuristics (ii) and (iii) are applied by fixing the corresponding binary variables. This problem is then solved by using the result of step one as initialization. This procedure was combined with a rolling-horizon solution procedure which decomposes the problem into a series of smaller sub-problems. These are first solved separately, and the solution of the complete problem is finally composed of the solutions of the sub-problems.

The optimization was performed with the commercial package GAMS/Cplex. Parameter studies of various Cplex parameters led to the result that the option $dpriind = 1$ and default values for all other parameters are the preferable option for the given problem. Table 2 shows results obtained for the original formulation of the case study, but with a varying number of jobs using GAMS/Cplex, version 21.7. The optimality gap as termination

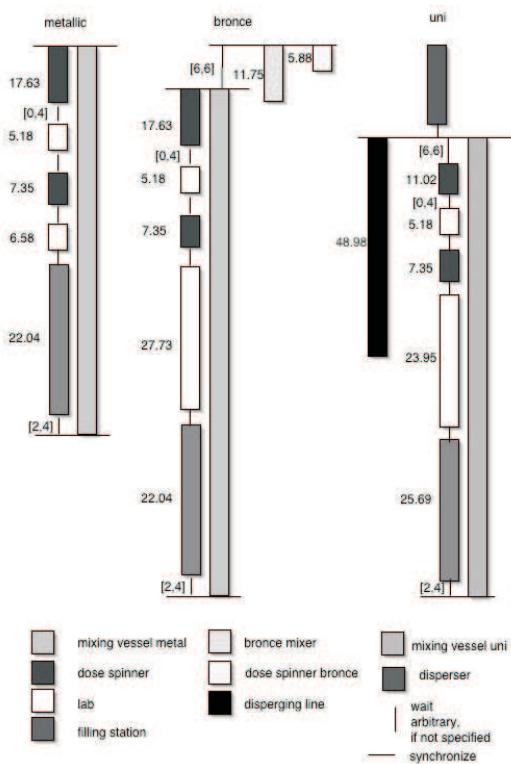


Figure 4: Recipe-based representation of the production sequence.

criterion was set to zero, meaning that the search is continued until the optimal solution is found. In all tests, an optimal solution with zero accumulated tardiness was found.

Further results are reported in [155, 248, 243]. In the last year of AMETIST, the MILP

Table 2: Optimization of the original problem formulation with a variable number of jobs. *time 1* and *time 2* refer to the two stages of the solution procedure. Times are in seconds.

number of jobs	time 1	time 2
10	3.95	2.87
14	15.86	0.61
16	23.83	0.92
18	46.72	1.20
20	58.71	1.53
22	100.23	1.85
29	412.93	2.61

model has been extended by constraints for changeover procedures. It involves additional real variables to determine direct predecessors of operations on the filling stations. If a changeover procedure is necessary for a pair of two successive operations, the duration of the second operation becomes extended by the duration of the changeover procedure. The results in Tab. 3 show the computational effort required to solve the extended model for a varying number of jobs. Two versions of GAMS/Cplex were used to document the impact of the software version on the results. The optimization objective here was to minimize the aggregated tardiness of all jobs. In all test configurations optimal schedules with zero accumulated tardiness were found and the optimality gap was set to zero. The experimental environment was a 2.4 GHz Pentium 4 machine with 1 GB of memory and the Linux operating system. The following conclusions can be drawn from the experiments:

1. The version of the software has no considerable influence on the solution performance and, thus, the results are comparable to those published in [248].
2. Despite the fact that new variables and inequalities have been defined for the modeling of changeover procedures, this extension only causes a moderate increase of the computational effort.

ORION-PI Axxom has applied its own software for specifying and solving scheduling problems, ORION-PI. It uses a particular modeling concept based on so-called *quants*, which represent the smallest logistic units required to model the problem. ORION-PI performs a quant-based combinatorial optimization algorithm that employs the principle of branch-and-bound. The exact solution algorithm is, however, not public domain.

Axxom has performed two series of tests, one for the original version, and one for the extended version that takes additional constraints for the operating hours, maximum break times, and production rates into account. The cost function is given as the sum of delay

Table 3: Results for a MILP model extended to changeover procedures. Solution times are given for problem instances with varying numbers of jobs and two versions of GAMS/Cplex. The pairs of numbers refer to the two stages of the solution procedure.

number of jobs	GAMS 21.3	GAMS 21.7
10	0.90/2.85	3.95/2.87
20	232.48/5.64	245.55/5.78
25	779.10/7.56	321.38/7.76
29	848.60/5.88	830.26/6.42

costs (delay time times input costs times the quantity of the order) and storage costs (storage time times input costs times quantity of the order). Different scenarios for 29, 500, and 1000 orders are considered, and for 29 orders, cases with and without requiring a delay-free solution are included.

Table 4 shows results for the original version of the case study and for four different scenarios. For the first scenario (29 orders, delay-free solution), results obtained with Uppaal and GAMS/Cplex are included for comparison.

Table 4: Results for the original version of the case study. The column *delay costs* answers the question whether a delay-free schedule could be found or not (this was the only optimization criterion required for this comparison). For the first scenario, 'O' refers to results generated with ORION-PI, 'U' to Uppaal, and 'G' to GAMS/Cplex – if not marked the results are obtained with ORION-PI. The tests with ORION-PI were carried out on a PC with 2.66 GHz processor and 1 GB RAM, and the last two on a PC with 3.06 GHz processor and 2 GB RAM.

Number of jobs	storage costs	delay costs	total costs	jobs in time	calc. time
29	O: 2627.32	O: 0	O: 2627.32	29	11 sec
	U: 2840.18	U: 0	U: 2840.18	29	-
	G: 2113.83	G: 0	G: 2113.83	29	-
29	1459.62	146.08	1605.71	22	13 sec
500	10648.16	34102.61	44750.77	200	32 min
1000	21750.51	66645.25	88395.77	403	97 min

Results for the extended version of the case study are shown in Tab. 5, and Fig. 5 contains exemplarily the Gantt chart obtained with ORION-PI for the first scenario listed in Tab. 5. The visualization shows the single quants assigned to the resources with the following meaning: red - delayed, yellow - just in time, green - too early. The generated schedules meet the given constraints, as e.g. the requirement that the resources can only be used within the given operating hours.

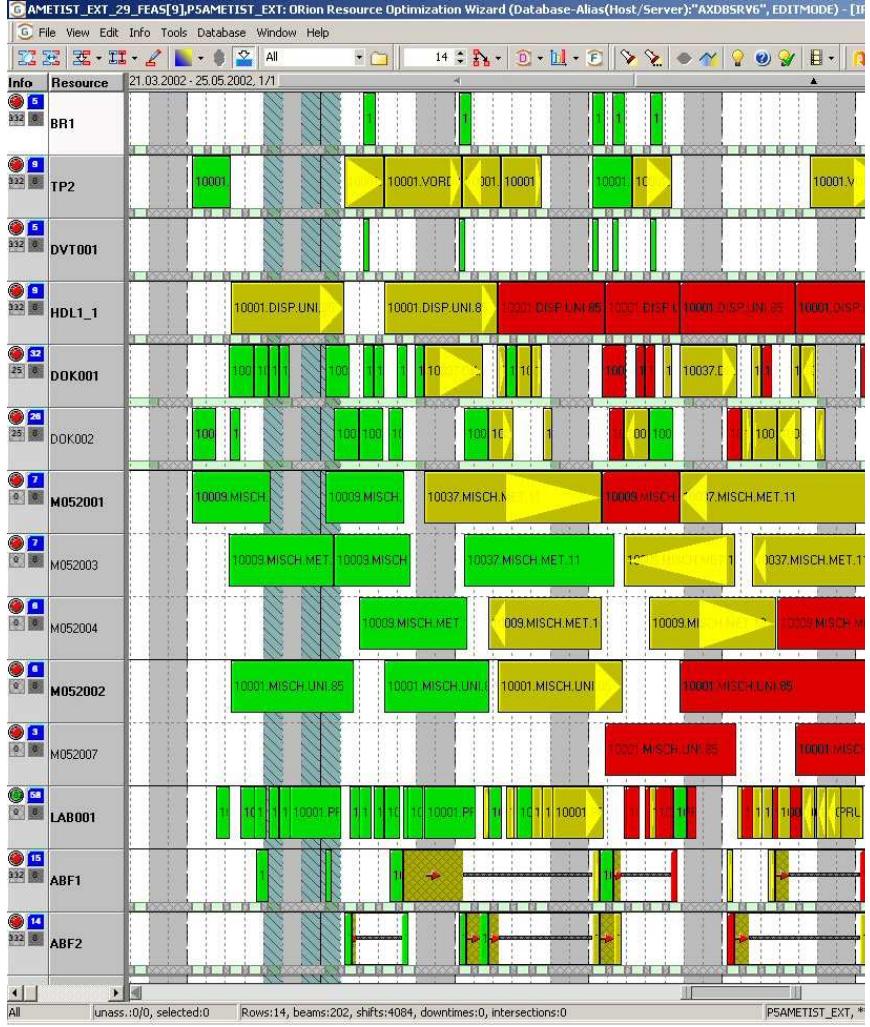


Figure 5: Gantt chart for the case: extended problem version, 29 orders, red (dark) operations are delayed.

Synthesis of Schedules with Uppaal Using the procedure to obtain timed automaton models of the cases study, sketched in Sec. 3.6.2, the tool Uppaal [93, 78] was used to derive feasible and optimal schedules[87, 86]. The scheduling problems were composed of the following components:

1. a set of recipe templates representing individual types of lacquers; these templates were instantiated with parameters (release and due dates, quantities, costs) to model concrete production orders as job automata;
2. additional automata to model resources which have their own clocks (for changeovers) as well as for interruptions and synchronization purposes;

Table 5: Results for the extended version of the case study. The accumulated delays of the schedules are shown in column *delay costs*. 'O' / 'U' refer to results obtained with ORION-PI, or Uppaal respectively. All ORION-PI test were performed a PC with 3.06MHz processor and 2 GB RAM.

Number of jobs	storage costs	delay costs	total costs	calc. time
29	O: 2662.68 U: 5706.78	O: 9070.04 U: 64.02	O: 11732.73 U: 5770.80	11 sec -
29	409.46	17496.61	17906.07	13 sec
500	33363.29	335754.56	369117.86	65 min
1000	138085.68	301544.73	439630.42	145 min

3. extensions of the automata to model various heuristics and cost criteria;
4. a property of the composed TA to be satisfied (minimizing a given cost criterion).

The following set of heuristics was considered: no overtaking of jobs (jobs started earlier receive a resource earlier), non-laziness (a required and available resource must not remain unused), greediness (a job allocates a resource as soon as possible), reduction of active orders (i.e. the number of jobs currently processed), increase of release dates (to reduce the possible storage costs). How these different heuristics and the different types of costs are established in Uppaal (or Uppaal-Cora for cost-optimal reachability) is described in detail in [87, 86]. Overall, 33 Uppaal models have been created for a varying number of jobs (29, 73, 219), the availability of resources (always available, or use of availability factors [as the fraction of time in which a resource can be used because operators are available], or available at explicit working times), optimization criteria (with and without costs), heuristics (non-laziness, greedy, non-overtaking, increment of release dates, limitation of active jobs).

The groups of problem instances are numbered from 1 to 7, and the corresponding problem properties can be summarized as follows:

1. 29 orders, hard release and due dates, feasibility problem (i.e, deadlines must be satisfied, no cost optimization);
2. 73 orders, hard release and due dates, feasibility problem;
3. $n \times 73$ orders as in 2), hard release and due dates, n vertical concatenations of the job table from 2) with proportional extensions of release and due dates, feasibility problem;
4. $n \times 73$ orders as in 2), hard release and due dates, n horizontal concatenations of the job table from 2) with the release and due dates, and the n replicates of all resources from case 2); feasibility problem;

5. 29 orders, no hard deadlines, cost minimization of storage and delay costs for final products and setup costs for filling lines;
6. 29 orders, no hard deadlines, working time constraints, cost minimization of storage and delay costs for final products and setup costs for filling lines;
7. 29 orders, no hard deadlines, working time constraints, cost minimization of storage costs for intermediate and final products, delay and setup costs for filling lines.

If the generation of delay-free schedules is the objective, the reachability algorithm of UPPAAL searches for a reachable state representing that all jobs are completed before their due dates are passed. The trace into such a state (produced by UPPAAL) represents directly the schedule. The analysis is performed by choosing depth-first or random depth-first search, where the latter was more successful. Table 6 shows results for the feasibility analysis as obtained with Uppaal 3.5.6 on a 2.6 GHz Intel-P4-Xeon processor and 2.5 GB of memory running Linux kernel 2.6.8. The model number refers to the problem instances listed above.

Initial experiments revealed scalability problems, partly caused by the large number of clocks contained in the models. The heuristics limiting the number of active jobs also provides a limit on the number of clocks needed (one per active job instead of one per job), and the non-overtaking heuristics provides an easy way of uniquely assigning shared clocks to jobs, since the starting order of jobs of a particular type is fixed. This change reduced the number of clocks to $3 \cdot A + 3$, where A is the maximum number of active jobs. The results show that, even for the case of 29 jobs, the use of the heuristics is essential. The non-overtaking heuristics does not make much difference for a case without availability factors, whereas in the case with availability factors the performance gets worse. For 73 jobs, however, non-overtaking decreases the computation time considerably. Limiting the number of active jobs increases the speed by several orders of magnitude (partly due the possible reuse of clocks).

In addition, experiments have been performed for the extended version of the case study using Uppaal-Cora, see Tab. 7. Although the constraints of explicit working hours (used in the models 6 and 7) makes the model much more complex, the results show that the schedules have much smaller cost as the schedules for model 5. A possible explanation is that the availability factors distribute the availability uniformly over time. In reality, however, a dose spinner is completely available during the weekdays and totally unavailable in the weekends. Therefore, the availability factors give in some cases a large over-approximation of the processing times, with the result that scheduling becomes much harder. The meaning of the costs is as follows: in model 6, a schedule with cost of approximately 2 million (the best schedule of the 10 runs) is a schedule in which 2 orders are a bit late (1 day and 45 minutes respectively), and the others are much too early (since intermediate products do not incur storage costs). In model 7, this effect is countered by storage costs for intermediate products, what makes the schedules more expensive. The investigation of these effects is still ongoing work. It should be noted that the non-laziness heuristics is not applicable to the extended case, since storage costs make it advantageous to be lazy.

model version	# jobs	heuristics	max. act. ord.	no av, no pf	av, no pf	av, pf
1	29	-	-	78.0	-	-
1	29	nl	-	0.3	1.5	3.5
1	29	nl no	-	0.3	2.5	3.6
1	29	g	-	0.2	2.8	-
1	29	g no	-	0.2	2.9	-
2	73	-	-	-	-	-
2	73	nl	-	-	-	-
2	73	nl no	-	24.6	-	-
2	73	nl no	3	0.4	-	-
2	73	nl no	4	1.1	0.4	0.3
2	73	nl no	5	4.4	0.6	0.9
2	73	g	-	15.0	-	-
2	73	g no	-	9.2	43.3	30.5
2	73	g no	3	0.3	165.1	-
2	73	g no	4	0.4	0.3	0.3
2	73	g no	5	2.2	0.4	6.8
3	146	g no	4	1.3	0.9	0.8
3	219	g no	4	3.5	2.1	1.9
3	292	g no	4	7.7	4.4	4.0
3	365	g no	4	13.3	7.5	6.7
3	438	g no	4	20.0	11.2	10.0
3	511	g no	4	28.2	15.8	14.0
3	584	g no	4	37.7	21.2	18.6
3	876	g no	4	89.7	49.7	43.8
3	1168	g no	4	166.4	92.1	80.7
3	1460	g no	4	270.9	149.4	131.0
3	1752	g no	4	401.3	222.3	194.0
3	2044	g no	4	565.4	311.6	271.9
4	146	g no	5	9.7	-	-
4	146	g no	6	167.8	-	-
4	146	g no	7	-	21.0	-
4	146	g no	8	-	18.2	12.6
4	219	g no	8	-	-	-
4	219	g no	9	-	-	347.9
4	219	g no	10	-	-	-
4	219	g no	11	-	-	-
4	219	g no	12	-	-	-

Table 6: Experiments for the generation of delay-free schedules. Abbreviations for the heuristics are: *g* - greedy, *nl* - not lazy, and *no* - no overtaking. Each experiment was repeated without working hours (*no av, no pf*), with availability factors and no performance factor (*av, no pf*) and with both availability factors and performance factors (*av & pf*). For each experiment the run time in seconds is provided. All measurements were done using depth-first search. A run was terminated after 10 minutes or when memory consumption reached 2GB (indicated by a '-').

model version	number of jobs	heuristics	pf / av / ex	max. active orders	termination rate	average costs
5	29	g	yes/yes/no	5	10/10	$21 \cdot 10^6$
6	29	-	yes/no/yes	5	4/10	$5.6 \cdot 10^6$
7	29	-	yes/no/yes	5	3/10	$81 \cdot 10^6$

Table 7: Table of experiments for the versions including costs with performance factors (pf) for all models, availability factors (av) in model 5, and explicit working hours (ex) in the models 6 and 7. A random-best-depth-first search of 10 seconds was used for all experiments. The number of successful runs (termination rate), and the average cost of each run is shown.

Also the non-overtaking heuristics cannot be adapted, since it can be advantageous for a job with low storage costs to overtake a job with higher storage costs. The greediness heuristics cannot be used either in the models 6 and 7, due a limitation in Uppaal. Current work includes to adapt all heuristics to the extended version.

The results show that feasible schedules can be derived with Uppaal such that the case with 29 orders is solved within 1 second, and the extension to 73 orders does not significantly increase the computation times (if suitable heuristics are used). To further scale up the model size, a vertical multiplication of two models with 73 orders was considered, and solutions could be obtained for up to 2044 orders. For horizontal composition of the case with 73 orders (multiplying also the resources) was successful up to 146 orders. To deal with the full set of constraints of the original problem (incurring setup-costs for filling stations, storage costs, and delay costs) the problem was transformed into a cost-optimization problem, and the latter was solved by Uppaal-Cora. A further extension was required to deal with the constraints for working-hours, which increased the size and complexity of the model significantly. But also for this case, feasible schedules could be derived with Uppaal-Cora.

Stochastic Analysis with Moebius For the feasible schedules obtained with Uppaal, our objective was to investigate the robustness with respect to breakdowns of the resources, and to rank alternative schedules accordingly. The problem formulation provided by Axxom contains performance factors (formulating the fraction of time in which a resource is not operational) and availability factors (as the fraction of time at which a resource can be used because operators are available). Axxom proposed to extend the durations of operations by considering these factors. In order to check whether this modification is

advantageous and to accomplish the schedule assessment, the following approach was taken [102]: The scheduling task was modeled in the language *MoDeST*, which combines modeling features from stochastic process algebra and from timed and probabilistic automata with light-weight notations such as exception handling. It is supported by the *Motor* tool, which facilitates the execution and evaluation of *MoDeST* specifications by means of the discrete event simulation engine of the *Moebius* tool. The performance and availability were modeled in detail by considering the mean time between failures (MTBF), the mean time to repair (MTTR), and a pace as the frequency of failure occurrence.

The performance analysis with *Moebius* included 80 experiments overall, where 20 different feasible schedules were investigated for two different paces and two different deadline policies (1: give up a job once it is sure it misses its deadline, 2: process all jobs to the end). One half of the schedules included the modeling of availability and performance, while the other half did not. Each experiment was executed 20,000 times and took around 4 to 5 hours. Figure 6 summarizes the results from this investigation (left part: availability and performance not considered; right part: both factors are considered). It can be concluded that: (a) a higher pace is advantageous to successfully complete a job in time, (b) the success rate of the investigated schedules does not differ significantly, and (c) it is not advantageous to simply extend the duration of operations by the availability and performance factors since jobs are started later as necessary if an equipment malfunction does not occur. Hence, the availability and performance factors should be used only for sequencing and the prediction of delivery dates but not for the timing of operations.

The investigation does furthermore allow to quantify the success rate for each individual job. The obtained rates confirm the expectation that jobs which are scheduled to start late finish less likely on time and that two jobs which are roughly started at the same time have a lower probability to be finished timely.

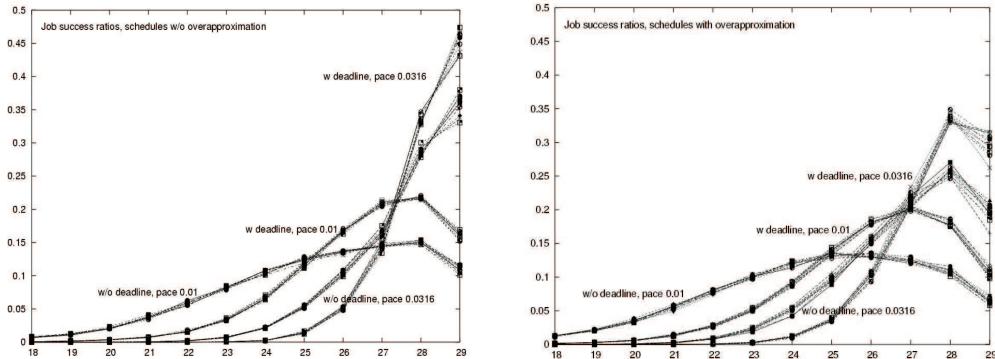


Figure 6: Assessment of schedules.

Generation of Schedules by IF Schedules for the original formulation of the case study were also derived with the tool IF [114]. Using the restrictions of fixed processing

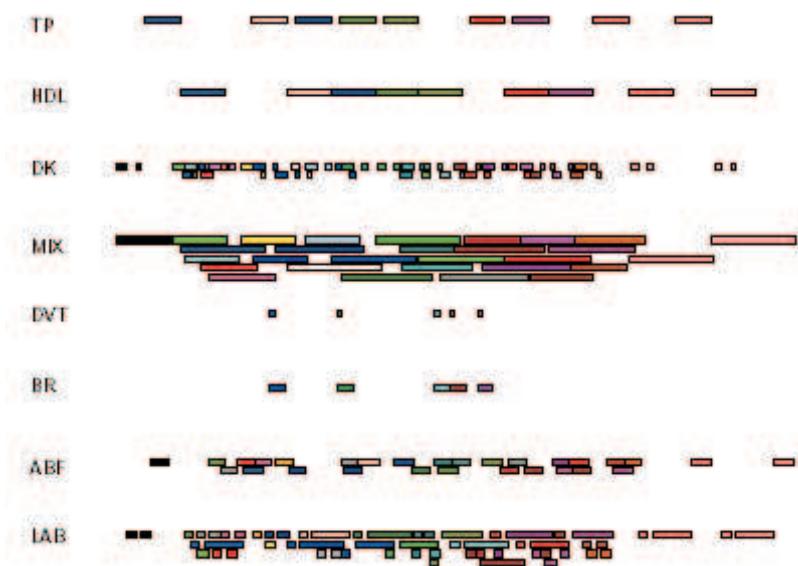


Figure 7: Schedule obtained with the tool IF.

times on resources and exactly two quality checks per job, a timed automaton model was built such that (i) the production sequence of each lacquer was modeled by an acyclic timed automaton, and (ii) the availability of resources was represented by shared variables. For the composition of all resulting automata, the scheduling task means to find a minimum cost path leading to a (global) final state in which all lacquers are finished. Since brute-force exploration of the state-space obtained for the problem with 29 jobs was impossible, three heuristics were employed: (a) only non-lazy runs are explored, (b) overtaking of jobs is forbidden, and (c) minimal separation times between the starting times of lacquers of the same type are chosen.

Modeling these heuristics in the IF language and solving the original version of the problem with the associated IF tools (in particular the module for computing paths with minimal costs), the following results were obtained: a feasible schedule without delay is obtained in 15 seconds, and the schedule can be extracted from the corresponding execution path of the model (with a depth of 750) almost instantaneously by random execution. The schedule is shown as Gantt chart in Fig. 7.

The Solution by TAopt TAopt combines principles known from mixed-integer programming with reachability analysis of timed automata [245, 244], in order to establish efficient pruning criteria for the graph search. The main idea is to use linear programming to solve tailored relaxed subproblems, and thus to compute lower cost bounds for pruning, and as a heuristics to steer the search. The embedded linear programs are updated iteratively and then solved by Cplex to compute the lower bound of cost-to-go for the current node of the reachability tree.

In order to model the case study in TAopt, the scheduling problems is first formulated as a resource task network (RTN) extended by additional information. RTN are a common and illustrative description method for recipes in the operations research community and are widely used to describe scheduling problems. A problem specification in TAopt consists of: (1) a description of the plant, including all resources with capacity constraints, resource configurations and transitions between configurations (which represent changeover procedures); (2) recipes formulated as RTN consisting of tasks, products, resource and configuration references, and arcs between the individual elements; additional timing and sequence constraints can be expressed using *places* labeled by a *marking* (similar to time Petri nets); (3) production orders, each of which reference a recipe and produce a new job as an instance of the recipe. The instances can be modified with respect to release dates, due dates, and production quantities according to requirements of the individual production orders. An example, of an RTN for one type of lacquer is shown in Fig. 8.

The meaning of the individual elements of this RTN is as follows: Tasks are represented by rectangles and decorated with durations. Places represent timing and sequencing constraints of tasks and are depicted by thin circles. Initial markings are shown as black bullets. The attached intervals contain the minimal and maximal time in which a marking may be in the place. The following constant values are used in the recipe to instantiate a job: the release date *rd*, the due date *dd* of the job, and the time horizon *H*, i.e. an

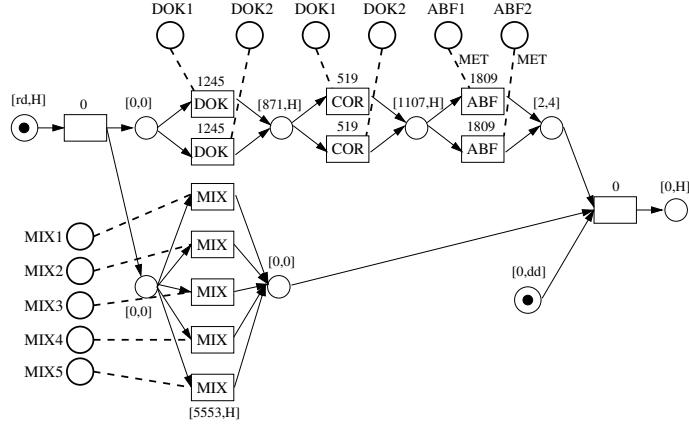


Figure 8: A RTN of the recipe for metallic lacquers.

upper bound of the makespan. Thick circles represent resources which are linked to tasks by dashed undirected lines. Some of the links are additionally decorated with *MET*, which is the name of the configuration required by the corresponding task. Tasks without resources are dummy tasks used for synchronization purposes only. The lab resource and the corresponding tasks have been removed here.

Given such an extended RTN as input, TAopt automatically generates a timed automaton (TA) model and a corresponding MILP model (with relaxed integrality constraints). Both models are then used in TAopt to perform the cost-optimal reachability analysis (of the TA model), and to compute lower bounds of cost-to-go using the LP model.

Table 8 shows results obtained with TAopt for the original version of the case study with reduced number of jobs. We considered the minimization of the makespan for a varying number of jobs, where the first five problems do not include hard deadlines for the jobs, while the last two do. The search was terminated when a specified number of nodes (*node limit*) was explored. The table lists the number of solutions found, the best makespan among all solutions required to process all jobs, and the computation time (in seconds) to obtain the best result. In all tests no heuristics were used, thus these results reflect the performance of a pure depth-first search with cost minimization. Both criteria, maximal depth and minimal costs, were used as decision rule for removing nodes from the waiting list. For the last instance, no feasible solution could be found, before the search was terminated after exploring $1 \cdot 10^6$ nodes.

As TAopt is still a prototype tool at this stage, embedded linear programs can so far be used only for job-shop problems. The solution of larger problem instances of the Axxom case study with embedded LPs is a subject of current work.

Table 8: Optimization of the original problem formulation with TAopt for a varying number of jobs using the makespan as the optimization criterion.

No. jobs	deadlines	node limit	No. of solutions	best makespan	calc. time (sec)
10	no	$2 \cdot 10^5$	9	30920	54.9
12	no	$2 \cdot 10^5$	3	53216	75.9
14	no	$1 \cdot 10^6$	2	55371	525.5
16	no	$1 \cdot 10^6$	5	53792	661.9
18	no	$1 \cdot 10^6$	5	53692	839.5
10	yes	$2 \cdot 10^5$	6	30920	65.9
12	yes	$1 \cdot 10^6$	0	-	-

3.6.4 Conclusions

Overall, different modeling techniques to suitably represent the value chain challenge problem were developed and tested within AMETIST, and the considered set of tools has been found to be able (after introduction of suitable heuristics in some cases) to solve different problem instances efficiently.

With respect to the approaches based on timed automata and (extended) model-checking techniques (Uppaal, IF), the experiences can be summarized as follows: A considerable effort had to be put into producing an appropriate TA representation of the problem specification provided by AXXOM. One part of this effort was to clarify and exactly interpret the constraints and relevant costs. In addition, specific schemes had to be developed to include the various constraints into the model. This shows, on the one hand, that the application of model checking techniques to this kind of production scheduling is not yet a push-button technology, since the models have to be constructed with care, and suitable heuristics have to be identified. On the other hand, one can realistically assume that many production scheduling problems have similar structures, such that the modeling patterns developed in AMETIST can be reused for similar scheduling problems. Further investigations have to identify a core collection of such patterns. Once the TA model is obtained, the application of model checking techniques for production scheduling is promising, as feasible schedules could be obtained in short computation times for not too large problems.

In direct comparison to the TA-based approaches, the effort for solving algebraic programs with the MILP approach was found to be higher in many instances. Even when heuristics were implemented, the solution procedure (which includes the time-consuming startup of GAMS/Cplex) often takes more time than the entire optimization performed by Uppaal. On the other hand, the solution quality is expected to be better because a true optimization problem is solved. Creating efficient MILP models is usually a laborious task and requires not only experience but also the investigation of suitable solver parameters to accelerate the solver performance. The advantage of the MILP approach is the possibility to assess the quality of the solution by evaluating lower bounds of the costs. Another advantage

arises from the fact that the optimization criterion can be an arbitrary linear function, and its choice hardly affects the rest of the model. Thus, it seems easier to combine different models, heuristics and objective functions. However, the MILP approach is limited to medium-sized problems.

First test results of the scheduling algorithm implemented in TAopt[246] have shown that embedding the solution of LPs into the reachability algorithm for timed automata can efficiently prune the reachability tree, and thus reduce the memory consumption considerably. This advantage dominates the effort to solve embedded LPs when large models are optimized. In comparison to GAMS/Cplex it was shown in [246] that feasible schedules for benchmark job-shop problems could be computed quickly and the quality was better than for GAMS/Cplex, when appropriate state space reduction methods and heuristics are activated. However, the applicability of the implementation of TAopt is currently limited to models that can be expressed as RTNs with timing constraints, i.e. only a simplified version of the case study could be solved.

The approach of the industrial partner Axxom is competitive to the other approaches with respect to the solution effort. Similarly to the MILP approach, it is based on a branching algorithm, extended by different heuristics to efficiently prune the state space and to provide feasible solutions quickly. Its advantage is the flexibility of using different (also non-linear) cost functions and constraints. The modeling is made comfortable by an appropriate GUI.

Based upon the results reported above, the performance and the range of application of the tools that were applied to the benchmark problem can be summarized as follows:

- For problem instances of moderate size, with possibly complex (but linear) cost functions and simple timing constraints (only conditions for the relative time-shift between operations), the MILP-approach is able to compute better solutions than a TA-based approach or ORION-PI in reasonable computing times. However, the modeling effort is comparatively large.
- For scheduling problems of moderate size but with complex timing constraints (e.g. limited working hours) and simple cost functions, the TA-based approach, e.g. Cora-Uppaal, is able to provide better solutions than ORION-PI if suitable reductions and heuristics are used. The modeling by MILPs in these cases gives rise to very large models in a continuous as well as a discrete time formulation that cannot be solved without further simplification.
- For large-scale problems with complex timing constraints (e.g. the consideration of night shifts) and cost minimization, ORION-PI at present is the only tool that can provide solutions with reasonable computational effort.
- Timed Automata are not yet a push-button technology to be applied without problem specific modeling and solution strategies. But the generation of libraries of templates for typical configurations seems promising and appears as a path towards to more widespread and easier application for non-TA-specialist users.

As for the stochastic assessment of schedules (with Moebius/Modest/Motor), the use of the performance and availability factors led to the question of proper interpretation. Extending the processing times by these factors can be used to analyze how many jobs can be handled on a longer time horizon. However, the stochastic analysis has shown that using performance and availability factors to obtain concrete schedules, increases the probability to miss deadlines if not only the sequencing but also the timing of such schedules is implemented. It is unclear at this stage what modeling assumptions are best suitable for the derivation of concrete short-term schedules, where storage costs have to be minimized, and where delay costs should to be avoided. A suitable idea may be to use a form of schedule refinement, taking rough long-term schedules as a basis for obtaining precise schedules for a shorter term – such a refinement approach was successfully used in the Cybernetix case study [229]. Another idea to be investigated in the future is that of searching for schedules in reverse time, starting from the due dates of orders – valid schedules obtained this way avoid storage and delay costs by construction.

3.7 Project Results: Other Case Studies

Profiting from its new tools, the AMETIST consortium has tackled 18 other industrial sized case studies involving the application of timed automata technology which were provided by its industrial partners, members from the industrial end use panel, or obtained from other sources. For a more detailed description/overview/discussion of our work on these case studies we refer to Deliverable 3.5.3 [54].

1. Verification and improvement of the sliding window protocol [130]
2. Modeling and verifying a Lego car using hybrid I/O automata [158]
3. Correctness of an intrusion-tolerant group communication protocol [223]
4. Testing conformance of real-time applications by automatic generation of observers [96]
5. A model of the Welch/Lynch Clock Synchronization Protocol [7]
6. Modelling and Analysis of a Leader Election Algorithm for Mobile Ad Hoc Networks [89]
7. Synthesis of Safe, QoS Extendible, Application Specific Schedulers for Heterogeneous Real-Time Systems [203]
8. Description and Schedulability Analysis of the Software Architecture of an Automated Vehicle Control System [272]
9. Verification of Asynchronous Circuits using Timed Automata [112]

10. Analysis of a Protocol for Dynamic Configuration of IPv4 Link Local Addresses using UPPAAL [172]
11. Model checking dependability attributes of wireless group communication [238]
12. Cost-Optimisation of the IPv4 Zeroconf Protocol [101]
13. Model Checker Aided Design of a Controller for a Wafer Scanner [182]
14. Analysis of a Biphase Mark Protocol with UPPAAL and PVS [277]
15. Automatic verification of the IEEE 1394 root contention protocol with KRONOS and PRISM [144]
16. Model Checking the Time to Reach Agreement [179]
17. Timed Automata Based Analysis of Embedded System Architectures [185]
18. From StoCharts to MoDeST: a comparative reliability analysis of train radio communications [187]

Our general conclusion is that using the new methods developed by AMETIST we can handle bigger problems faster and in a much more routine manner than at the start of the project. Applications of timed automata technology to analysis of scheduling problems, embedded system architectures and communication protocols have been very successful. Our case studies provide insight in the current state-of-the-art of timed automata technology, provide modelling patterns that can be used by future users, and benchmarks for verification and analysis tools.

3.8 Towards a Unifying Framework

Working with discrete and continuous systems, being exposed to control, verification, scheduling and other domains, one cannot but observe that many problems treated under different names within different disciplines, have more resemblance if we look at them through an appropriate abstraction that filters their domain-specific details. Among these problems and techniques we mention the algorithmic approach to discrete systems verification by forward or backward fixpoint computation, the derived reachability algorithms for continuous and hybrid systems, bounded model checking (using satisfiability solvers to verify correctness for a bounded horizon), computational techniques for optimal control such as dynamic programming and model-predictive control, simulation, search methods in AI and Markov decision processes. Much of our effort during project was concerned with building a general unifying game-theoretic scheme, for which various system design and validation problems are concrete instances, most notably the problem of scheduling under uncertainty.

The paper [234] lays down a *unified* and *domain-independent model* for control in the presence of adversaries. The model uses the metaphor of a two-player game between

the controller to be synthesized and the environment it is supposed to control. Control synthesis is viewed as finding an optimal (or satisfactory) strategy for the controller, where optimality is parameterized by two factors:

1. The cost function associated with *individual* trajectories
2. The way the costs of all possible adversary-induced trajectories under a given strategy are combined to compute the overall value of the strategy (worst case, average case).

Verification, and open-loop control (“ballistic” control, planning) are obtained as special cases where either one of the players is suppressed, i.e. is assumed to be deterministic with no choice. On this model we identify three generic approaches for solving controller synthesis problems:

- For *bounded time-horizon* the problem can be posed as a constrained optimization problem as is done in model-predictive control and bounded model-checking (in the latter, it is often the case that *feasibility* of the constraints is emphasized rather than cost optimality).
- The other class of methods, roughly characterized as *dynamic programming (DP)*, is based on iterative computation of a value function (cost-to-go), which determines the optimal cost and action at every state of the system. For discrete systems such techniques are used in backward verification and synthesis for automata and for Markov decision processes. In continuous systems the value function is often computed as a solutions of some partial differential equations due to Hamilton, Jacobi, Bellman and Isaacs (HJBI).
- The third approach is to perform a forward search in the space of strategies and trajectories, an approach used by game-playing programs. In verification, when the search is not exhaustive over all adversarial behaviors, this activity can be viewed as testing. For control this approach is not mainstream but it seems to be popular in some related domains such as reinforcement learning and robot motion planning. The advantage of this approach compared to DP is that the value function needs to be computed only for the reachable part of the state space, a fact that may moderate the curse of dimensionality.

The separation between the domain-independent abstract scheme and the specific computational aspects of each domain, may facilitate the development of a universal controller synthesis tool based on this model. The specificity of each domain will be manifested by the type of computational engine used, for example a SAT solver for discrete systems and an LP solver when the dynamics is linear. The study of solvers for hybrid constrained optimization problems, the basic computational tool for all verification and synthesis problems, is becoming an important research issue by itself, as described in previous sections.

4 Outlook

In this section, we present a short description on how the results and achievements of the project have benefited the project partners and how they intend to use and exploit these further.

- **Use of AMETIST results in education** (all academic partners).

Students very much enjoy the use of model checking technology. **KUN** started to use UPPAAL in a first year course on operating systems to teach students about mutual exclusion problems, semaphores, etc. Students like the graphical interface and simulator. This year they found several mistakes in a book on Semaphores by Allen Downey, using UPPAAL. At the Master level, we use the AMETIST case studies to teach students about modelling complex systems. In subsequent projects they tackle complex cases themselves, such as (critical parts of) Bluetooth and I2C. As a Master's Thesis project one student at **KUN** formally modelled a new internet protocol SHIM6 using UPPAAL and found several mistakes that were acknowledged by the protocol designers. The fixes proposed by the student have been incorporated in a new version of the internet draft.

- **Use of UPPAAL within LaQuSO (KUN).**

LaQuSo, <http://www.laquso.com>, the Laboratory for Software Quality is a joint organization of the Radboud University Nijmegen and the three technical universities within the Netherlands. LaQuSo aims to measure, quantify and predict the quality of the designed software in order to increase the predictability of the development process. Companies can ask LaQuSo to do quick assessments (usually ≤ 2 PMs) of software. UPPAAL is one of the tools used within LaQuSo, and the AMETIST case studies are very useful to demonstrate the capabilities of this technology

- **(Bosch).**

Bosch intends to use UPPAAL in futher verification tasks, following a problem-oriented approach, and to exploit the AMETIST concepts in in-house development of system engineering methods and tools to enable wide-range use.

- **Exploitation throug CISS and ESI (UT, AAU).**

Kim Larsen from partner **AAU** is the director of the Danish competence center for embedded software systems (**CISS**), and Ed Brinksma from parter **UT** is the director of the Dutch Embedded Systems Institute (**ESI**). A special role in the promotion of exploitation will be played by these two research institutes. Both **ESI** and **CISS** are national centers of competence focusing exclusively on embedded systems and committed to collaborative projects between research and industry. They play a pivotal role in both national (D-ARTEMIS DK and Point-One NL) and European innovation ecosystems (ARTEMIS/ARTEMISIA, ARTIST2 and HYCON Networks of Excellence) for embedded systems engineering. They have well-established courses and training programmes for industry in the area of embedded system engineering, and

specialised knowledge and technology transfer activities, both for OEM industries (e.g. ASML, Philips, NXP, Oce Technologies, Thales, TERMA, Danfoss, Grundfos) and many high-tech SME companies. CISS and ESI will make results of the AMETIST project available to these European infrastructures through these training and transfer programmes. Further activities include knowledge exchange circles and high-tech spin-off programmes.

- **Industrial exploitation of UPPAAL and UPPAAL CORA (AAU).**

We have a subcontract with Airbus related to the application of UPPAAL and a contract with the University of Ohio. We have close contacts with OWO, Honeywel and NASA about the possible application of UPPAAL CORA. We are currently setting up a spin-off company centered around the UPPAAL tool.

- **Spreading the word to other universities (UT).**

Holger Hermanns from UT has become professor at Saarland University, heading the Dependable Systems and Software group. AMETIST results are used within several projects in which the group participates, notably AVACS, a Research Center sponsored by the German Research Foundation DFG, which is devoted to foundational research on Automated Verification and Analysis of Complex Systems. Joost-Pieter Katoen from UT has taken up a chair on software modelling and verification at RWTH Aachen. The group participates in several projects founded by the German and Dutch research council that are all focussed on different aspects of the verification and modelling of stochastic systems.

- **QUASIMODO initiative (AAU, Terma, ex-UT, KUN).**

Several former AMETIST participants are involved in the FP7 application QUASI-MODO that aims at the further improvement and extension of AMETIST tools.

- **Participation in Other Projects (all).**

We firmly intend to work on the further exploitation of the results obtained within AMETIST through our extensive network of academic and industrial contacts, which encompasses the ARTISTS2 and HYCON networks of excellence.

As an illustration, we mention three projects in which partner KUN is involved. In the Octopus project of Oce technologies and the Embedded Systems Institute, which will start on July 1, 2007, KUN will address the problem of the high level design and scheduling of the data path in a copier machine, and in particular how we can make this data path more adaptive, i.e. able to change based upon changing conditions in the environment. We expect that timed priced automata will be a very useful technology to address these problems. Within two projects funded by NWO, the Netherlands Organisation for Scientific Research, KUN wil continue its work on further improving timed automata model checking technology. One project, FRAAI, is exploiting compositional reasoning and another project (that will start at the end of 2007) will study counterexample guided abstraction refinement.

5 Conclusions

With all the accumulated effort invested in AMETIST, there are reasons to believe that at the end of the project timed automata technology finds itself in a much better shape and quite closer to industrial acceptance than it was. We summarize the progress with respect to the main objectives of the project, namely, *scalability*, *convenience* and *accessibility*.

- *Scalability*: This is a major obstacle for acceptance of the technology which has been subject to numerous intensive attacks during the project lifetime. Specialized solutions tailored for scheduling problems and optimizations of existing tools have led to significant improvements that make TA-based methods not inferior to other commonly-used methods. As mentioned in Deliverable 1.1 [55], many real-life solution to scheduling problems do not try to solve the intractable optimization problems but rather use heuristics that constrain the solution space to the point that a program can find a solution.

The above observation should not be used as an excuse to neglect scalability altogether. The project has brought us closer to the long-awaited performance breakthrough which, like the “BDD revolution” in untimed verification, is a pre-requisite for the proliferation of timed automata to all the application domains that can benefit from their rich and useful modeling capabilities.

- *Convenience*: The pleasant user interface of UPPAAL clearly makes the working of timed automata much more intuitive for the layman. It has been demonstrated in the project that specialized descriptions of scheduling problems can be transformed automatically into timed automaton description, something that many potential users cannot do. The project did not develop a general language/formalism for scheduling for reasons related to the issues described in the next paragraph.
- *Accessibility*: By this term we understand the increased awareness in the client disciplines and application domains of the potential of timed automata and what they can do for the scheduling professional. Unlike domains such as formal verification where it is sufficient to convince someone at Intel or Microsoft in order to generate a huge impact, the scheduling arena is much more fragmented, among application domains, in-house vs.commercial tools, and geographic locations. The project made the partners in contact with Axxom, a small SME which had the opportunity to observe closely what timed automata can (and cannot) do for its problems. Like any encounter with reality, the collaboration with Axxom was humbling and will certainly help partners in their future efforts to make TA technology more widely accessible.

Let us remark that one of the apparent major obstacles for gaining acceptance for timed automata technology lies in timed automata being a quite non-standard model from the engineer’s point of view. Purely continuous systems (differential equations) as well as discrete time models (difference equations, synchronous automata) are very

well understood, while discrete systems working on “asynchronous” dense time are much less intuitive to grasp upon a first encounter. Perhaps some more pedagogical considerations should be employed while writing papers on the domain, rather than addressing (and impressing) a small community of experts.

References

- [1] Y. Abdeddaïm, E. Asarin, and O. Maler. On optimal scheduling under uncertainty. In H. Gargamé and J. Hatcliff, editors, *Proc. TACAS*, volume 2619 of *LNCS*. Springer, 2003. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/Papers/uncertain.ps>.
- [2] Y. Abdeddaïm, E. Asarin, and O. Maler. Scheduling with timed automata. *Theoretical Computer Science*, 2005. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/Papers/schedule-tcs.ps>.
- [3] Y. Abdeddaïm, A. Kerbaa, and O. Maler. Task graph scheduling using timed automata. In *FMPPTA*, 2003. Available from World Wide Web: <http://www-verimag.imag.fr/~kerbaa/task-graph.ps>.
- [4] Y. Abdeddaïm and O. Maler. Preemptive job-shop scheduling using stopwatchautomata. In J.-P. Katoen and P. Stevens, editors, *TACAS*, 2002. Available from World Wide Web: <http://www-verimag.imag.fr/PEOPLE/Oded.Maler/Papers/preemption.ps>.
- [5] Y. Abdeddaïm and P. Niebert. On the use of partial order methods in scheduling. In *Ninth International Conference on Project Management and Scheduling (PMS 04)*, 2004. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/pms04.pdf>. to be published as online abstract.
- [6] Yasmina Abdeddaïm. *Scheduling with Timed Automata*. PhD thesis, INPG Grenoble, November 2002. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/Papers/thesis-yasmina.ps>.
- [7] L. Aceto, G. Behrmann, J.F. Groote, and K. Larsen. Notes on a Uppaal model of the Welch/Lynch clock synchronization protocol, 2004. Available from World Wide Web: <http://www.cs.auc.dk/~kg1/AcetoBehrmannGrooteLarsen.pdf>. Unpublished note.
- [8] L. Aceto, P. Bouyer, A. Burgueo, and K. G. Larsen. The limit of testing for timed automata. *Theoretical Computer Science (TCS)*, 300(1-3):411–475, 2003. Available from World Wide Web: <http://www.lsv.ens-cachan.fr/Publications/PAPERS/Bou-AABL02.ps>.
- [9] M. Agopian. A simulation tool for the SuperSingle mode, 2003. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/SuperSingleSimulator.zip>. Not a paper, a tool.
- [10] M. Agopian. A model of the faulty card problem of the smart card personalization machine, 2005. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/faultycards.pdf>.
- [11] S. Albert. Design/CPN model of Cybernetix Case Study. Technical report, Cybern’etix - LIF, 2002. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/cybernetix-cpn.tgz>.
- [12] Sarah Albert. Cybernetix case study – informal description. Technical report, Cybern’etix - LIF, 2002. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/cyx.pdf>.
- [13] Sarah Albert and Peter Niebert. Cybern’etix case study – performance analysis – optimality of the supersingle mode. Technical report, Cybern’etix -LIF, 2002. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/cybernetix-optimality.pdf>.

- [14] K. Altisen, G. Gößler, and J. Sifakis. Scheduler modeling based on the controller synthesis paradigm. *Journal of Real-Time Systems, special issue on Control Approaches to Real-Time Computing*, 23:55–84, 2002. Available from World Wide Web: http://www-verimag.imag.fr/~sifakis/paper_final.pdf.
- [15] K. Altisen and S. Tripakis. Tools for controller synthesis of timed systems. In *RT-TOOLS*, 2002. Available from World Wide Web: <http://www-verimag.imag.fr/~tripakis/final-rttools02.pdf>.
- [16] AMETIST. Ametist workshop, October 2002. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del4.1.1.pdf>. Deliverable 4.1.1 from the IST project AMETIST.
- [17] AMETIST. Progress report — progress and evaluation: Reference period april 2002 - september 2002, November 2002. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del0.1.1.pdf>. Deliverable 0.1.1 from the IST project AMETIST.
- [18] AMETIST. Analysis and tools: State space representations, June 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del2.3.a.pdf>. Deliverable 2.3.a from the IST project AMETIST.
- [19] AMETIST. Bosch case study: First year report, June 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.3.2.pdf>. Deliverable 3.3.2 from the IST project AMETIST.
- [20] AMETIST. Cybern'etix case study: First year report, June 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.1.2.pdf>. Deliverable 3.1.2 from the IST project AMETIST.
- [21] AMETIST. First year report on case study 2: Memory management in radar sensor equipment, May 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.2.2.ps>. Deliverable 3.2.2 from the IST project AMETIST.
- [22] AMETIST. First year report on case study 4: Value chain optimization, May 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.4.2.pdf>. Deliverable 3.4.2 from the IST project AMETIST.
- [23] AMETIST. Framework report (v1), June 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del0.2.1.pdf>. Deliverable 0.2.1 from the IST project AMETIST.
- [24] AMETIST. Miscellaneous case studies: First year report, May 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.5.1.pdf>. Deliverable 3.5.1 from the IST project AMETIST.
- [25] AMETIST. Modelling: Control synthesis, April 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del1.5.pdf>. Deliverable 1.5 from the IST project AMETIST.
- [26] AMETIST. Progress report: Reference period april 2002 - march 2003, June 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del0.1.2.pdf>. Deliverable 0.1.2 from the IST project AMETIST.
- [27] AMETIST. Analysis and tools: Abstraction and compositionality, May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del2.1.1.pdf>. Deliverable 2.1.1 from the IST project AMETIST.

- [28] AMETIST. Analysis and tools: Control synthesis algorithms, May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del2.2.1.pdf>. Deliverable 2.2.1 from the IST project AMETIST.
- [29] AMETIST. Analysis and tools: Stochastic analysis (second year report), May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del2.4.a.pdf>. Deliverable 2.4.a from the IST project AMETIST.
- [30] AMETIST. Analysis and tools: Tools and tool interaction, May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del2.5.a.ps>. Deliverable 2.5.a from the IST project AMETIST.
- [31] AMETIST. Bosch case study: Second year report, May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.3.3.pdf>. Deliverable 3.3.3 from the IST project AMETIST.
- [32] AMETIST. Cybern'etix case study: Second year report, April 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.1.3.pdf>. Deliverable 3.1.3 from the IST project AMETIST.
- [33] AMETIST. Data structures (second year report), May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del2.3.b.pdf>. Deliverable 2.3.b from the IST project AMETIST.
- [34] AMETIST. Framework report (v2), May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/framework2.pdf>. Deliverable 0.2.2 from the IST project AMETIST.
- [35] AMETIST. Miscellaneous case studies: Second year report, April 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.5.2.pdf>. Deliverable 3.5.2 from the IST project AMETIST.
- [36] AMETIST. Modelling: Model composition, May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del1.2.pdf>. Deliverable 1.2 from the IST project AMETIST.
- [37] AMETIST. Modelling: Model composition, May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del1.4.pdf>. Deliverable 1.4 from the IST project AMETIST.
- [38] AMETIST. Modelling: Quantitative modelling (second year report), May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del1.3.2.pdf>. Deliverable 1.3 from the IST project AMETIST.
- [39] AMETIST. Periodic progress and management report for period from 1 april 2003 to 31 march 2004, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del0.1.4.pdf>. Deliverable 0.1.4 from the IST project AMETIST.
- [40] AMETIST. Progress report: Reference period april 2003 - september 2003, April 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del0.1.3.pdf>. Deliverable 0.1.3 from the IST project AMETIST.
- [41] AMETIST. Second year report on case study 4: Improvement in modelling analysis, and solving the value chain optimization problem, April 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.4.3.pdf>. Deliverable 3.4.3 from the IST project AMETIST.
- [42] AMETIST. Terma case study: Second year report, May 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.2.3.pdf>. Deliverable 3.2.3 from the IST project AMETIST.

- [43] AMETIST. Analysis and tools: Stochastic analysis, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del2.4.b.pdf>. Deliverable 2.4.b from the IST project AMETIST.
- [44] AMETIST. Analysis and tools: Structure exploitation, June 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del2.1.2.pdf>. Deliverable 2.1.2 from the IST project AMETIST.
- [45] AMETIST. Bosch case study: Final report, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.3.4.pdf>. Deliverable 3.3.4 from the IST project AMETIST.
- [46] AMETIST. Case study 4: Value chain optimization — final report, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.3.4.pdf>. Deliverable 3.4.4 from the IST project AMETIST.
- [47] AMETIST. Progress report — reference period from 1 april 2004 to 30 september 2004 (revised version), 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del0.1.5.pdf>. Deliverable 0.1.5 from the IST project AMETIST.
- [48] AMETIST. Terma case study: Final report, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.2.4.pdf>. Deliverable 3.2.4 from the IST project AMETIST.
- [49] AMETIST. Analysis and tools: Scheduling and planning algorithms, May 2006. Available from World Wide Web: <http://www.cs.ru.nl/F.Vaandrager/AMETIST/del2.2.2.pdf>. Deliverable 2.2.2 from the IST project AMETIST.
- [50] AMETIST. Analysis and tools: State space representations (v3), May 2006. Available from World Wide Web: <http://www.cs.ru.nl/F.Vaandrager/AMETIST/del2.3.c.pdf>. Deliverable 2.3.c from the IST project AMETIST.
- [51] AMETIST. Analysis and tools: Tools and tool interaction (v2), Oct 2006. Available from World Wide Web: <http://www.cs.ru.nl/F.Vaandrager/AMETIST/del2.5.b.pdf>. Deliverable 2.5.b from the IST project AMETIST.
- [52] AMETIST. Cybernetix case study: Final report, May 2006. Available from World Wide Web: <http://www.cs.ru.nl/F.Vaandrager/AMETIST/del3.1.4.pdf>. Deliverable 3.1.4 from the IST project AMETIST.
- [53] AMETIST. Framework report (v3b), May 2006. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/framework3-revised.pdf>. Deliverable 0.2.3 from the IST project AMETIST.
- [54] AMETIST. Miscellaneous case studies: Final report, May 2006. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.5.3.pdf>. Deliverable 3.5.3 from the IST project AMETIST. Revised version.
- [55] AMETIST. Modelling: Model classification, May 2006. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del1.1.pdf>. Deliverable 1.1 from the IST project AMETIST.
- [56] AMETIST. Progress report for the last period, Oct 2006. Available from World Wide Web: <http://www.cs.ru.nl/F.Vaandrager/AMETIST/del0.1.6.pdf>. Deliverable 0.1.6 from the IST project AMETIST.
- [57] Suzana Andova, H. Hermanns, and Joost-Pieter Katoen. Discrete-time rewards model-checked. In *Formal Modelling and Analysis of Timed Systems (FORMATS 2003)*, Marseille, France, 2003. Lecture Notes in Computer Science, Springer-Verlag. Available from World Wide Web: http://fmt.cs.utwente.nl/publications/files/417_AHK03.ps.

- [58] Suzana Andova and Tim Willemse. Equivalences for silent transitions in probabilistic systems, 2004. Submitted to QEST.
- [59] Zvi Artstein and Gera Weiss. State nullification by memoryless output feedback. *MCSS*, 2004. In print.
- [60] Zvi Artstein and Gera Weiss. State nullification by memoryless output feedback. *Math. Control Signals Systems*, 17(1):38–56, 2005. Available from World Wide Web: <http://www.wisdom.weizmann.ac.il/~gera/nullification.pdf>.
- [61] E. Asarin, P. Caspi, and O. Maler. Timed regular expressions. *Journal of the ACM*, 49(02):172–206, 2002. Available from World Wide Web: <http://www-verimag.imag.fr/~asarin/papers/revised.pdf>.
- [62] E. Asarin and C. Dima. Balanced timed regular expressions. In *MTCS'2002*, volume 68 of *ENTCS*, Brno, Czech Republic, August 2002. Available from World Wide Web: <http://www.elsevier.com/gej-ng/31/29/23/121/53/25/68.5.003.pdf>. issue 5.
- [63] Eugene Asarin and Thao Dang. Abstraction by projection and application to multi-affine systems. In Rajeev Alur and George Pappas, editors, *Hybrid Systems: Computation and Control Proceedings of 7th International Workshop*, volume 2993 of *LNCS*, Philadelphia, PA, USA, March 2004.
- [64] M. Auerswald. SRS ECU Behaviour Modelling, December 2004. Confidential.
- [65] C. Baier, P.R. D’Argenio, and M. Größer. Partial order reducction for probabilistic branching time. In *3rd Workshop of Quantitative Aspects of Programming Languages*, QAPL’05, 2005.
- [66] C. Baier, B. Haverkort, H. Hermanns, and J.-P. Katoen. Model-checking algorithms for continuous-time markov chains. *IEEE Transactions on Software Engineering*, 29(6):524–541, 2003. Available from World Wide Web: http://fmt.cs.utwente.nl/publications/files/399_116221.pdf.
- [67] C. Baier, B. Haverkort, H. Hermanns, and J.-P. Katoen. Efficient computation of time-bounded reachability probabilities in uniform continuous-time markov decision processes. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, Barcelona, Spain, 2004. Lecture Notes in Computer Science, Springer-Verlag. Available from World Wide Web: http://fmt.cs.utwente.nl/publications/files/420_ctit_tr_03_50.pdf.
- [68] C. Baier, B. Haverkort, H. Hermanns, J.-P. Katoen, and M. Siegle, editors. *Validation of Stochastic Systems: A Guide to Current Research*, volume 2925 of *LNCS*. Springer-Verlag, 2004. 467 p. Tutorial volume.
- [69] C. Baier, H. Hermanns, and J.-P. Katoen. Probabilistic weak simulation is polynomially decidable. *Information Processing Letters*, Vol. 89, Issue: 3, pages 123–252, 2004.
- [70] C. Baier, H. Hermanns, J.-P. Katoen, and Verena Wolf. Comparative branching-time semantics for markov chains. In *Concurrency Theory (CONCUR)*, pages 492–507, Marseille, France, 2003. Lecture Notes in Computer Science, Vol. 2761, Springer-Verlag. Available from World Wide Web: http://fmt.cs.utwente.nl/publications/files/404_BHKW03.ps.
- [71] C. Baier, J.-P. Katoen, H. Hermanns, and Verena Wolf. Comparative branching-time semantics for markov chains. Technical Report CTIT-TR-04-32, University of Twente, 2004. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/461_TR-CTIT-04-32.pdf. Submitted for publication. 64 pages.
- [72] C. Baier, J.-P. Katoen, H. Hermanns, and Verena Wolf. Comparative branching-time semantics for markov chains. *Information and Computation*, 2005. Accepted, publication date unknown.

- [73] Christel Baier, Joost-Pieter Katoen, Holger Hermanns, and Boudewijn Haverkort. Simulation for continuous-time Markov chains. In L. Brim, P. Jancar, M. Kretinsky, and A. Kucera, editors, *Concurrency Theory*, volume 2421 of *Lecture Notes in Computer Science*, pages 338–354. Springer-Verlag, 2002. Available from World Wide Web: <http://link.springer.de/link/service/series/0558/papers/2421/24210338.pdf>.
- [74] N. Baudru and R. Morin. Safe implementability of regular message sequence chart specifications. In *Proceedings of the ACIS Fourth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD'03)*, 2003.
- [75] N. Baudru and P. Niebert. Controllers from proofs: An alternative approach to control synthesis for mu-calculus specifications. draft, 2004. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/controllersynthesis.pdf>.
- [76] N. Bauer, S. Engell, R. Huuck, S. Lohmann, B. Lukoschus, M. Remelhe, and O. Stursberg. Verification of plc programs given as sequential function charts. In *Integration of Software Specification Techniques for Applications in Engineering*, volume 3147 of *LNCS*, pages 517–540. Springer, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/BEH+04.pdf>.
- [77] N. Bauer, R. Huuck, B. Lukoschus, and S. Engell. A unifying semantics for sequential function charts. In *Integration of Software Specification Techniques for Applications in Engineering*, volume 3147 of *LNCS*, pages 400–418. Springer, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/BHLE04.pdf>.
- [78] G. Behrmann. Guiding and cost optimizing uppaal. Web-page, 2002. Available from World Wide Web: http://www.cs.auc.dk/~behrmann/_guiding/.
- [79] G. Behrmann. A performance study of distributed timed automata reachability analysis. In *Electronic Notes in Theoretical Computer Science*, 68(4), 2002. Presented at PDMC 2002, Brno.
- [80] G. Behrmann. *Data-structure Analysis for Formal Verification*. PhD thesis, Aalborg University, 2003.
- [81] G. Behrmann. Distributed reachability analysis in timed automata. *Software Tools for Technology Transfer*, 7(1):19–30, 2005. Available from World Wide Web: <http://www.cs.aau.dk/~kg1/AMETIST/Year3/behrmann05.ps>.
- [82] G. Behrmann, S. Bernicot, T. Hune, K.G. Larsen, S. Lecamp, and A. Skou. Case study 2: Memory interface for radar system, 2002. Available from World Wide Web: <http://www.cs.auc.dk/~kg1/AMETIST/bbh11s.ps>. Deliverable 3.2.1 from the IST project AMETIST.
- [83] G. Behrmann, P. Bouyer, E. Fleury, and K. G. Larsen. Static guard analysis in timed automata verification. In *Proc. 9th Int. Conf. Tools and Algorithms for the Construction and Analysis of Systems (TACAS'2003)*, volume 2619 of *Lecture Notes in Computer Science*, pages 254–277. Springer-Verlag, 2003. Available from World Wide Web: <http://www.lsv.ens-cachan.fr/Publis/PAPERS/BBFL-tacas-2003.ps>.
- [84] G. Behrmann, P. Bouyer, K. G. Larsen, and R. Pelánek. Lower and upper bounds in zone based abstractions of timed automata. In *Proc. 10th Int. Conf. Tools and Algorithms for the Construction and Analysis of Systems (TACAS'2004)*, volume 2988 of *Lecture Notes in Computer Science*, pages 312–326. Springer-Verlag, 2004. Available from World Wide Web: <http://www.lsv.ens-cachan.fr/Publis/PAPERS/BBLP-tacas04.ps>.
- [85] G. Behrmann, P. Bouyer, K.G. Larsen, and R. Pelanek. Zone based abstractions for timed automata exploiting lower and upper bounds. *Software Tools for Technology Transfer*, 2005. Available from World Wide Web: <http://www.cs.aau.dk/~kg1/AMETIST/Year3/bblp05.pdf>. Selected papers from TACAS'04.

- [86] G. Behrmann, E. Brinksma, M. Hendriks, and A. Mader. Scheduling lacquer production by reachability analysis – a case study. In *Workshop on Parallel and Distributed Real-Time Systems*. IEEE Computer Society Press, 2005. Available from World Wide Web: <http://www.cs.ru.nl/ita/publications/papers/martijnh/AXXOM/WPDRTS05.ps.gz>.
- [87] G. Behrmann, E. Brinksma, M. Hendriks, and A. Mader. Scheduling lacquer production by reachability analysis – a case study. In *Proceedings of the 16-th IFAC World Congress*, 2005. Available from World Wide Web: <http://www.cs.ru.nl/ita/publications/papers/martijnh/AXXOM/IFAC2005.pdf>. Extended abstract.
- [88] G. Behrmann, A. David, K. G. Larsen, J. Håkansson, P. Pettersson, W. Yi, and M. Hendriks. Uppaal 4.0. In *Third International Conference on the Quantitative Evaluation of SysTems (QEST 2006), 11-14 September 2006, Riverside, CA, USA*, pages 125–126. IEEE Computer Society, 2006.
- [89] G. Behrmann, K. Larsen, and A. Skou. Modelling and analysis of a leader election algorithm for mobile ad hoc networks. Modelling carried for on request by Leslie Lamport, links <http://www.cs.auc.dk/~kgl/Lamport1.pdf>, <http://www.cs.auc.dk/~kgl/Lamport2.pdf>, <http://www.cs.auc.dk/~kgl/leader.xml>, <http://www.cs.auc.dk/~kgl/leader.q>, 2003.
- [90] G. Behrmann, K. G. Larsen, and R. Pelánek. To store or not to store. In *Proc. of 15th Int. Conf. Computer Aided Verification (CAV'2003)*, volume 2725 of *Lecture Notes in Computer Science*, pages 433–445. Springer-Verlag, 2003. Available from World Wide Web: <http://www.fi.muni.cz/~xpelanek/publications/cav56.ps>.
- [91] G. Behrmann, K.G. Larsen, and J.I. Rasmussen. Optimal scheduling using priced timed automata. *Performance Evaluation Review, ACM Sigmetric*, 2005. Available from World Wide Web: <http://www.cs.aau.dk/~kgl/AMETIST/Year3/blr05.ps>.
- [92] G. Behrmann, K.G. Larsen, and J.I. Rasmussen. Priced timed automata: Algorithms and applications. In *Proceedings of FMCO'04*, Lecture Notes in Computer Science. Springer Verlag, 2005. Available from World Wide Web: <http://www.cs.aau.dk/~kgl/AMETIST/Year3/blr05b.pdf>.
- [93] Gerd Behrmann, Johan Bengtsson, Alexandre David, Kim G. Larsen, Paul Pettersson, and Wang Yi. UPPAAL implementation secrets. In *Proc. of 7em th International Symposium on Formal Techniques in Rea-Time and Fault Tolerant Systems*, 2002. Available from World Wide Web: <http://www.docs.uu.se/docs/rtmv/papers/bbdlpw-ftrtf02.ps.gz>.
- [94] Gerd Behrmann, Alexandre David, and Kim G. Larsen. A tutorial on uppaal. In *Formal Methods for the Design of Real-Time Systems*, number 3185 in Lecture Notes in Computer Science, pages 200–236. Springer Verlag, 2004. Available from World Wide Web: <http://www.cs.aau.dk/~kgl/AMETIST/Year2.5/tutorial.ps>.
- [95] Gerd Behrmann, Kim G. Larsen, and Jacob I. Rasmussen. Beyond liveness: Efficient parameter synthesis for time bounded liveness. 2004.
- [96] S. Bensalem, M. Bozga, M. Krichen, and S. Tripakis. Testing conformance of real-time applications by automatic generation of observers. In *Runtime Verification (RV'04)*, 2004.
- [97] S. Bernicot and S. Lecamp. *Modelling and analysis a memory interface*. Master's thesis, University of Aalborg, 2002. Available from World Wide Web: <http://www.cs.auc.dk/~kgl/AMETIST/bl.ps>. Internal document from the IST project AMETIST.
- [98] H. Bohnenkamp, P.R. D'Argenio, H. Hermanns, and J.-P. Katoen. MoDeST: A compositional modeling formalism for real-time and stochastic systems. Technical Report TR 04-46, CTIT, University of Twente, 2004. Submitted for publication.
- [99] H. Bohnenkamp, J. Gorter, J. Guidi, and J.-P. Katoen. Are you still there? a lightweight algorithm to monitor node absence in self-configuring networks. *Dependable Systems and Networks (DSN)*, June 2005. 6 pages.

- [100] H. Bohnenkamp, H. Hermanns, J.-P. Katoen, and R. Klaren. The modest modeling tool and its implementation. In P. Kemper and W.H. Sanders, editors, *Computer Performance Evaluation: Modeling Techniques and Tools*, Lecture Notes in Computer Science, 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/tools03.ps.gz>.
- [101] H. Bohnenkamp, P. van der Stok, H. Hermanns, and F.W. Vaandrager. Cost-optimisation of the IPv4 zeroconf protocol. In *Proceedings of the International Conference on Dependable Systems and Networks (DSN2003)*, em San Fransisco, pages 531–540, Los Alamitos, California, 2003. IEEE Computer Society. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/IPDS.html>.
- [102] H.C. Bohnenkamp, H. Hermanns, R. Klaren, A. Mader, and Y.S. Usenko. Synthesis and stochastic assessment of schedules for lacquer production. In *Proc. QEST'04*, LNCS, September 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/qest04.ps.gz>.
- [103] S. Bornot, R. Morin, P. Niebert, and S. Zennou. Black box unfolding with local first search. In *TACAS'2002*, volume 2280 of *LNCS*, page 386 ff., 2002. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/tacas02.pdf>.
- [104] P. Bouyer. Untameable timed automata. In *Proc. of 20th Ann. Symp. on Theoretical Aspects of Computer Science (STACS'2003)*, volume 2607 of *Lecture Notes in Computer Science*, pages 620–631. Springer-Verlag, 2003. Available from World Wide Web: <http://www.lsv.ens-cachan.fr/Publicis/PAPERS/Bou-stacs2003.ps>.
- [105] P. Bouyer, F. Cassez, E. Fleury, and K. G. Larsen. Optimal strategies in priced timed game automata. BRICS Report Series RS-04-4, Basic Research In Computer Science, 2004. Available from World Wide Web: <http://www.brics.dk/RS/04/4/BRICS-RS-04-4.ps.gz>.
- [106] P. Bouyer, D. D'Souza, P. Madhusudan, and A. Petit. Timed control with partial observability. In *Proc. of 15th Int. Conf. Computer Aided Verification (CAV'2003)*, volume 2725 of *Lecture Notes in Computer Science*, pages 180–192. Springer-Verlag, 2003. Available from World Wide Web: <http://www.lsv.ens-cachan.fr/Publicis/PAPERS/BDMP-cav-2003.ps>.
- [107] Patricia Bouyer, Ed Brinksma, and Kim G. Larsen. Staying alive as cheaply as possible. In Rajeev Alur and George J. Pappas, editors, *Proceedings of the 7th International Conference on Hybrid Systems: Computation and Control (HSCC'04)*, volume 2993 of *Lecture Notes in Computer Science*, pages 203–218, Philadelphia, Pennsylvania, USA, March 2004. Springer-Verlag. Available from World Wide Web: http://www.lsv.ens-cachan.fr/Publicis/RAPPORTS_LSV/rr-lsv-2004-2.rr.ps.
- [108] Patricia Bouyer, Franck Cassez, Emmanuel Fleury, and Kim G. Larsen. Optimal strategies in priced timed game automata. In Kamal Lodaya and Meena Mahajan, editors, *Proceedings of the 24th Conference on Fundations of Software Technology and Theoretical Computer Science (FSTTCS'04)*, volume 3328 of *Lecture Notes in Computer Science*, pages 148–160, Chennai, India, December 2004. Springer. Available from World Wide Web: <http://www.lsv.ens-cachan.fr/Publicis/PAPERS/PDF/>.
- [109] Patricia Bouyer, Franck Cassez, Emmanuel Fleury, and Kim G. Larsen. Synthesis of optimal strategies using HyTech. In Luca De Alfaro, editor, *Proceedings of the Workshop on Games in Design and Verification (GDV'04)*, volume 119 of *Electronic Notes in Theoretical Computer Science*, pages 11–31, Boston, Massachusetts, USA, February 2005. Elsevier Science Publishers. Available from World Wide Web: <http://www.lsv.ens-cachan.fr/Publicis/PAPERS/PDF/BCFL-gdv04.pdf>.
- [110] M. Bozga, S. Graf, and L. Mounier. If-2.0: A validation environment for component-based real-time systems. In K.G. Larsen Ed Brinksma, editor, *Proceedings of CAV'02*, volume 2404 of *LNCS*, pages 343–348, Copenhagen, Denmark, July 2002. Springer. Available from World Wide Web: <http://www-verimag.imag.fr/~async/BIBLIO/papers/Bozga-Graf-Mounier-02.ps.gz>.

- [111] M. Bozga, S. Graf, L. Mounier, and I. Ober. IF validation environment tutorial. In *SPIN'04 Workshop on Model-Checking of Software*, Barcelona, Spain, April 1-3, volume 2989 of *Lecture Notes in Computer Science*, pages 306–307. Springer, 2004.
- [112] M. Bozga, H. Jianmin, O. Maler, and S. Yovine. Verification of asynchronous circuits using timed automata. In *Proceedings of TPTS'02 Workshop*. Elsevier, April 2002. Available from World Wide Web: <http://www-verimag.imag.fr/PEOPLE/Oded.Maler/Papers/async.ps>.
- [113] M. Bozga, A. Kerbaa, and O. Maler. Optimal scheduling of acyclic branching programs on parallel machines. In *Real-Time Systems Symposium (RTSS)*, pages 208–217. IEEE Press, 2004. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/Papers/andor.pdf>.
- [114] M. Bozga and O. Maler. Timed automata approach for the axxom case study. Technical report, Verimag, 2003. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/AMETIST/axxom-report.pdf>.
- [115] M. Bravetti and P.R. D'Argenio. Tutte le algebre insieme: Concepts, discussions and relations of stochastic process algebras with general distributions. In C. Baier et al., editor, *Validation of Stochastic Systems: A Guide to Current Research*, volume 2925 of *LNCS*, pages 44–88, 2004.
- [116] E. Brinksma. Testing times: On model-driven test generation for non-deterministic real-time systems. In *In Proceedings Fourth International Conference on Application of Concurrency to System Design (ACSD 2004)*, pages 3–4. IEEE CS Press, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/ACSD04.ps>.
- [117] E. Brinksma, T. Krilavičius, and Y. S. Usenko. A process algebraic approach to hybrid systems, 2005. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/ifac_wc_2005.pdf. Submitted to the IFAC World Congress 2005. 6 pages.
- [118] E. Brinksma and K.G. Larsen, editors. *Computer Aided Verification, 14th International Conference*, volume 2404 of *Lecture Notes in Computer Science*. Springer Verlag, Copenhagen, Denmark, July 2002. Available from World Wide Web: <http://www.informatik.uni-trier.de/~ley/db/conf/cav/cav2002.html>.
- [119] E. Brinksma and A. Mader. Model checking embedded system designs (invited). In *6th Int. Workshop on Discrete Event Systems (WODES)*, pages 151–158, Zaragoza, Spain, October 2002. IEEE Computer Society Press, Los Alamitos, California. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/WODESexabs.pdf>.
- [120] Ed Brinksma. Compositional theories of qualitative and quantitative behaviour. In W. van der Aalst and Eike Best, editors, *Applications and Theory of Petri Nets 2003*, volume 2679 of *Lecture Notes in Computer Science*, pages 37–42. Springer-Verlag, 2003. Available from World Wide Web: <http://springerlink.metapress.com/openurl.asp?genre=article&issn=0302-9743&volume=2679&spage=37>.
- [121] Ed Brinksma and Angelika Mader. On verification modelling of embedded systems. In *SEES 2003: Software Engineering for Embedded Systems: from Requirements to Implementation*, Monterey Workshop Proceedings, pages 101–105, 2004.
- [122] L. Brandán Briones and E. Brinksma. A test generation framework for quiescent real-time systems. In *In Proceedings FATES 2004*, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/FATES04.pdf>. 18 pages, to appear in Springer LNCS.
- [123] Laura Brandán Briones and Mathias Roehl. Testing timed automata. In M. Broy, B. Jonsson, J.P. Katoen, M. Leucker, and A. Pretschner, editors, *Model-based Testing of Reactive Systems - A Seminar Volume*, number 3472 in LNCS. Springer Verlag, 2005.

- [124] P. Caspi, A. Curic, A. Maignan, C. Sofronis, and S. Tripakis. Translating discrete-time Simulink to Lustre. In *Embedded Software (EMSOFT'03)*, volume 2855 of *LNCS*. Springer, 2003.
- [125] P. Caspi, A. Curic, A. Maignan, C. Sofronis, S. Tripakis, and P. Niebert. From Simulink to SCADE/Lustre to TTA: a layered approach for distributed embedded applications. In *Languages, Compilers, and Tools for Embedded Systems (LCTES'03)*. ACM, 2003.
- [126] F. Cassez, A. David, E. Fleury, K.G. Larsen, and D. Lime. Efficient on-the-fly algorithms for the analysis of timed games. In Luca de Alfaro Martin Abadi, editor, *In Proceedings of CONCUR 2005*, Lecture Notes in Computer Science. Springer Verlag, 2005. Available from World Wide Web: <http://www.cs.aau.dk/~kg1/AMETIST/Year3/cdf1105.ps>.
- [127] A. Chakabarti, L. de Alfaro, T. A. Henzinger, and M. I. A. Stoelinga. Resource interfaces. In R. Alur and I. Lee, editors, *EMSOFT 03: 3rd Intl. Workshop on Embedded Software*, Lecture Notes in Computer Science. Springer, 2003. Available from World Wide Web: <http://wwwhome.cs.utwente.nl/~marielle/papers/resources.pdf>.
- [128] L. Cheung and M. Hendriks. Causal dependencies in parallel composition of stochastic processes, 2005. Available from World Wide Web: <http://www.nii.ru.nl/~lcheung/comp.pdf>.
- [129] L. Cheung, N.A. Lynch, R. Segala, and F.W. Vaandrager. Switched probabilistic I/O automata. In Z. Liu and K. Araki, editors, *Proceedings First International Colloquium on Theoretical Aspects of Computing (ICTAC2004), in Guiyang, China, 20-24 September 2004*, volume 3407 of *LNCS*, pages 494–510. Springer, 2005. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/switched.html>. Full version available as Technical Report NIII-R0427, NIII, Radboud University Nijmegen.
- [130] D. Chklaev, J. Hooman, and E. de Vink. Verification and improvement of the sliding window protocol. In *Proceedings TACAS'03*, pages 113–127. Lecture Notes in Computer Science 2619, Springer-Verlag, 2003. Available from World Wide Web: <http://www.cs.kun.nl/~hooman/SWP.html>.
- [131] E. Clarke, A. Fehnker, Z. Han, B. H. Krogh, O. Stursberg, and M. Theobald. Verification of hybrid systems based on counterexample-guided abstraction refinement. In *Tools and Algorithms for the Construction and Analysis of Systems, LNCS 2619*, pages 192–207. Springer, 2003. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/TACAS_03.pdf.
- [132] E. Clarke, A. Fehnker, Z. Han, B.H. Krogh, J. Ouaknine, O. Stursberg, and M. Theobald. Abstraction and counterexample-guided refinement in model checking of hybrid systems. *Int. Journal Foundations of Computer Science*, 14(4):583–604, 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/IJFCS03.pdf>.
- [133] L. Cloth, J.-P. Katoen, M. Khattri, and R. Pulungan. Model checking Markov reward models with impulse rewards. *Dependable Systems and Networks (DSN)*, June 2005. 10 pages.
- [134] S. Cotton, E. Asarin, O. Maler, and P. Niebert. Some progress in satisfiability checking for difference logic. In *FORMATS/FTRTFT'04*, number 3253 in *LNCS*, pages 263–276. Springer, 2004. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/Papers/dlsat.ps>.
- [135] Scott Cotton. DPPL and difference constraints. Master’s thesis, Max-Planck Doctoral School Saarbrucken, June 2005. Verimag.
- [136] Pedro R. D’Argenio. From stochastic automata to timed automata: Abstracting probability in a compositional manner. In *Proc. of the Argentinian Workshop on Theoretical Computer Science, WAIT 2003, Held as Part of the 32nd JAIIo*, Buenos Aires, September 2003. SADIO, 2003. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~dargenio/papers/wait2003.ps.gz>.

- [137] P.R. D'Argenio and B. Gebremichael. The coarsest congruence for timed automata with deadlines contained in bisimulation. Technical Report ICIS-R05015, Institute for Computing and Information Sciences, Radboud University Nijmegen, 2005. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/biniam/dag.html>. In Proc. of 16th International Conference on Concurrency Theory (CONCUR05).
- [138] P.R. D'Argenio, B. Jeannet, H.E. Jensen, and K.G. Larsen. Reduction and refinement strategies for probabilistic analysis. In H. Hermanns and R. Segala, editors, *Proceedings of Process Algebra and Probabilistic Methods. Performance Modeling and Verification. Joint International Workshop, PAPM-PROBMIV 2001*, Copenhagen, Denmark, Lecture Notes in Computer Science. Springer-Verlag, 2002. Available from World Wide Web: <http://www.cs.famaf.unc.edu.ar/dargenio/papers/papm-probmiv2002.ps.gz>.
- [139] P.R. D'Argenio and P. Niebert. Partial order reduction for concurrent probabilistic programs. In B.R. Haverkort et al., editor, *Proc. of the 1st International Conference on Quantitative Evaluation of Systems*, QEST 2004, pages 240–249. IEEE Computer Society Press, 2004.
- [140] A. David, G. Behrmann, K. G. Larsen, and W. Yi. A tool architecture for the next generation of extscUppaal. In *Proc. of 10th Ann. Colloquium of UNU/IIST*, volume 2757 of *Lecture Notes in Computer Science*, page ??? Springer-Verlag, 2003. Available from World Wide Web: <http://www.docs.uu.se/docs/rtmv/papers/2003-011.pdf>.
- [141] A. David, G. Behrmann, K.G. Larsen, and W. Yi. New uppaal architecture. In *Proceedings of RTTOOLS 2002*, 2002.
- [142] A. David, G. Behrmann, K.G. Larsen, and W. Yi. Unification and sharing in timed automata verification. In *in Proceedings of SPIN 2003 Workshop*, 2003.
- [143] C. Daws. Symbolic and parametric model checking of discrete time Markov chains. In Zhiming Liu and Keiji Araki, editors, *Proc. International Colloquium on Theoretical Aspects of Computing, ICTAC'04*, volume 3407 of *Lecture Notes in Computer Science*, pages 280–294, Guiyang, China, 2005. Springer. Available from World Wide Web: <http://springerlink.metapress.com/openurl.asp?genre=article&issn=0302-9743&volume=3407&spage=280>.
- [144] C. Daws, M.Z. Kwiatkowska, and G. Norman. Automatic verification of the IEEE 1394 root contention protocol with KRONOS and PRISM. *STTT*, 5(2-3):221–236, 2004. Available from World Wide Web: <http://www.springerlink.com/index/10.1007/s10009-003-0118-5>.
- [145] L. de Alfaro, M. Faella, T. A. Henzinger, R. Majumdar, and M.I.A. Stoelinga. Model checking discounted temporal properties. *Theoretical Computer Science*, 2005. Available from World Wide Web: <http://wwwhome.cs.utwente.nl/~marielle/papers/dct1.pdf>. Accepted for publication.
- [146] L. de Alfaro, M. Faella, and M.I.A. Stoelinga. Linear and branching metrics for quantitative transition systems. In *Proceedings 31st International Colloquium on Automata, Languages and Programming (ICALP'04)*, volume 3142 of *Lecture Notes in Computer Science*. Springer-Verlag, 2004. Available from World Wide Web: <http://wwwhome.cs.utwente.nl/~marielle/papers/qdist.pdf>.
- [147] L. de Alfaro and M. I. A. Stoelinga. Interfaces: a game-theoretic framework to reason about component-based systems. In *EMSOFT 03: 2nd Intl Workshop on Foundations of Coordination Languages and Software Architectures*, Electronic Notes in Computer Science. Elsevier, 2003. Available from World Wide Web: <http://wwwhome.cs.utwente.nl/~marielle/papers/tutorial.pdf>.
- [148] Luca de Alfaro, Marco Faella, Thomas A. Henzinger, Rupak Majumdar, and Marielle Stoelinga. The element of surprise in timed games. In *Proceedings CONCUR*, Lecture Notes in Computer Science. Springer-Verlag, 2003. Available from World Wide Web: <http://wwwhome.cs.utwente.nl/~marielle/papers/surprise.ps>.

- [149] G. Diaz, K. Larsen, J. Pardo, F. Cuartero, and V. Valero. An approach to handle real time and probabilistic behaviors in e-commerce: validating the set protocol. In *SAC '05: Proceedings of the 2005 ACM symposium on Applied computing*, pages 815–820, New York, NY, USA, 2005. ACM Press.
- [150] H. Dierks, G. Behrmann, and K.G. Larsen. Solving planning problems using real-time model checking (translating pddl3 into timed automata). In *In Proceddings of AIPS 2002 Workshop on Planning via Model Checking*, 2003. Available from World Wide Web: <http://semantik.informatik.uni-oldenburg.de/~dierks/Berichte/AIPS.ps.gz>.
- [151] Catalin Dima. Computing reachability relations in timed automata. In *LICS*, 2002.
- [152] S. Engell, S. Lohmann, and O. Stursberg. Verification of embedded supervisory controllers considering hybrid dynamics. *Int. Journal of Software Engineering and Knowledge Engineering*, 15(2):307–312, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/ELS05.pdf>.
- [153] S. Engell, A. Maerkert, G. Sand, and R. Schultz. Aggregated scheduling of a multiproduct batch plant by two-stage stochastic integer programming. *Optimization and Engineering*, 5:335–359, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/EMSS04.pdf>.
- [154] S. Engell and G. Sand. A two-stage stochastic integer programming approach to real-time scheduling. In I. E. Grossmann and C. M. McDonald, editors, *4th Int. Conf. on Foundations of Computer-Aided Process Operations*, pages 347–350, Austin, 2003. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/cp2_focapo2003sandengell.pdf.
- [155] Sebastian Engell and Sebastian Panek. Mathematical model formulation for the axxon case study. Technical report, University of Dortmund, May 2003. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/ametist_report_dortmund.pdf.
- [156] Juhan Ernits. Abstraction based analysis and arbiter synthesis: Radar memory interface card case study revised. In *Proceedings of Nordic Workshop on Programming Theory 2004*, 2004. Available from World Wide Web: http://www.cs.aau.dk/~kg1/AMETIST/Year2.5/nwpt04_juhan_ernits.pdf.
- [157] Juhan Ernits. Memory arbiter synthesis and verification for a radar memory interface card. *Nordic Journal of Computing*, 2005. Available from World Wide Web: <http://www.cs.aau.dk/~kg1/AMETIST/Year3/ernits.pdf>.
- [158] A. Fehnker, F.W. Vaandrager, and M. Zhang. Modeling and verifying a Lego car using hybrid I/O automata. In *Third International Conference on Quality Software (QSIC 2003)*, Dallas, Texas, USA, November 6 - 7, pages 280–289. IEEE Computer Society Press, 2003. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/LEGO.html>.
- [159] G. Franceschinis, B.R. Haverkort, J.-P. Katoen, and M. Woodside, editors. *Quantitive Evaluation of Systems Proceedings*. IEEE CS Press, 2004.
- [160] H. Garavel and H. Hermanns. On combining functional verification and performance evaluation using CADP. In L. Eriksson and P. Lindsay, editors, *FME 2002: International Symposium of Formal Methods Europe*, volume 2391 of *LNCS*, pages 410–429. Springer, 2002. Available from World Wide Web: <http://www.inrialpes.fr/vasy/Publications/Garavel-Hermanns-02.html>.
- [161] Frederic Gardi. An efficient algorithm for maximum disjoint matchings in a set of intervals and related problems. Research report 07-2002, LIF, Marseille, France, May 2002. Available from World Wide Web: <http://www.lim.univ-mrs.fr/Rapports/07-2002-Gardi.html>.

- [162] Frederic Gardi. On the Partition of an Interval Graph into Proper Interval Subgraphs. Research report 01-2002, LIF, Marseille, France, April 2002. Available from World Wide Web: <http://www.lim.univ-mrs.fr/Rapports/01-2002-Gardi.html>.
- [163] Frederic Gardi. The Mutual Exclusion Scheduling Problem for Proper Interval Graphs. Research report 02-2002, LIF, Marseille, France, April 2002. Available from World Wide Web: <http://www.lim.univ-mrs.fr/Rapports/02-2002-Gardi.html>.
- [164] Frederic Gardi. A note on the Roberts characterization of proper and unit interval graphs. Research report 11-2003, LIF, Marseille, France, January 2003. Available from World Wide Web: <http://www.lim.univ-mrs.fr/Rapports/11-2003-Gardi.html>.
- [165] Frederic Gardi. Mutual exclusion scheduling with interval graphs and related classes. Research report 12-2003, LIF, Marseille, France, May 2003. Available from World Wide Web: <http://www.lim.univ-mrs.fr/Rapports/12-2003-Gardi.html>.
- [166] B. Gebremichael, H. Hermanns, T. Krilavičius, and Y.S. Usenko. Hybrid modeling of a vehicle surveillance system with real-time data processing. In *Proc. Int. Conf. on Dynamical Systems Modeling and Stability Investigation*, page 419, Kyiv, Ukraine, May 2003. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/biniam/gkhu.html>.
- [167] B. Gebremichael, T. Krilavičius, and Y.S. Usenko. Real-time service allocation for car periphery supervision: Requirements and environment analysis, 2003. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/biniam/gku2.html>. Internal Ametist document.
- [168] B. Gebremichael, T. Krilavičius, and Y.S. Usenko. A formal analysis of a car periphery supervision system. In J. Zaytoon, V. Carre-Mennetrier, X. Cao, and C. Cassandras, editors, *Seventh International Workshop on Discrete Event Systems WODES'04, Reims, France*, pages 433–439, September 2004. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/biniam/gku.html>. Also available as Technical Report NIII-R0418, NIII, University of Nijmegen.
- [169] B. Gebremichael and F.W. Vaandrager. Control synthesis for a smart card personalization system using symbolic model checking. Report NIII-R0312, Nijmegen Institute for Computing and Information Sciences, University of Nijmegen, May 2003. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/smart.html>.
- [170] B. Gebremichael and F.W. Vaandrager. Control synthesis for a smart card personalization system using symbolic model checking. In *Proceedings First International Workshop on Formal Modeling and Analysis of Timed Systems (FORMATS 2003), m September 6-7 2003, Marseille, France*, volume 2791 of *LNCS*. Springer Verlag, 2003. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/smart.html>.
- [171] B. Gebremichael and F.W. Vaandrager. Specifying urgency in timed I/O automata. In *In proc. of the 3rd IEEE International Conference on Software Engineering and Formal Methods (SEFM05)*, December 2004. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/urgency.html>.
- [172] B. Gebremichael, F.W. Vaandrager, and M. Zhang. Analysis of a protocol for dynamic configuration of IPv4 link local addresses using Uppaal. Report ICIS-R06016, Institute for Computing and Information Sciences, Radboud University Nijmegen, 2006. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/zeroconf.html>.
- [173] D. Harel, H. Kugler, R. Marely, and A. Pnueli. Smart play-out of behavioral requirements. In *Proc. 4th Intl. Conference on Formal Methods in Computer-Aided Design (FMCAD'02), Portland, Oregon*, volume 2517 of *Lect. Notes in Comp. Sci.*, pages 378–398, 2002. Available from World Wide Web: <http://www.wisdom.weizmann.ac.il/~dharel/papers/FMCAD02.pdf>. Also available as Tech. Report MCS02-08, The Weizmann Institute of Science.

- [174] D. Harel and R. Marelly. Playing with time: On the specification and execution of time-enriched LSCs. In *Proc. 10th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'02)*, Fort Worth, Texas, 2002. Available from World Wide Web: <http://www.wisdom.weizmann.ac.il/~dharel/papers/TimedLSCs.pdf>.
- [175] D. Harel and R. Marelly. *Come, Let's Play: Scenario-Based Programming Using LSCs and the Play-Engine*. Springer-Verlag, 2003. Available from World Wide Web: <http://www.wisdom.weizmann.ac.il/~dharel/comeplay.html>.
- [176] David Harel, Hillel Kugler, and Gera Weiss. Some methodological observations resulting from experience using lscs and the play-in/play-out approach. In *Proc. Scenarios: Models, Algorithms and Tools*, volume 3466 of *Lect. Notes in Comp. Sci.*, pages 26–42. Springer-Verlag, 2005. Available from World Wide Web: <http://www.wisdom.weizmann.ac.il/~gera/methodology.pdf>.
- [177] B. Haverkort, L. Cloth, H. Hermanns, J.-P. Katoen, and C. Baier. Model-checking performability properties. In *International Conference on Dependable Systems and Networks (DSN)*, pages 103–112. IEEE CS Press, 2002. Available from World Wide Web: <http://computer.org/proceedings/dsn/1597/15970103abs.htm>.
- [178] M. Hendriks. Enhancing Uppaal by exploiting symmetry. Report NIII-R0208, Nijmegen Institute for Computing and Information Sciences, University of Nijmegen, October 2002. Available from World Wide Web: <http://www.cs.kun.nl/research/reports/info/NIII-R0208.html>.
- [179] M. Hendriks. Model checking the time to reach agreement. Technical Report ICIS-R05014, Institute for Computing and Information Sciences, Radboud University Nijmegen, 2005. Available from World Wide Web: <http://www.cs.ru.nl/research/reports/info/ICIS-R05014.html>. Submitted to the International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS'05).
- [180] M. Hendriks, G. Behrmann, K.G. Larsen, P. Niebert, and F.W. Vaandrager. Adding symmetry reduction to Uppaal. In *Proceedings First International Workshop on Formal Modeling and Analysis of Timed Systems (FORMATS 2003)*, September 6–7 2003, Marseille, France, volume 2791 of *LNCS*. Springer Verlag, 2004. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/symmetry.html>. Full version available as Technical Report NIII-R0407, NIII, University of Nijmegen, February 2004.
- [181] M. Hendriks and K.G. Larsen. Exact acceleration of real-time model checking. *Electronic Notes in Theoretical Computer Science*, 65(6), April 2002. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/martijnh/TPTS02.pdf>.
- [182] M. Hendriks, N.J.M. van den Nieuwelaar, and F.W. Vaandrager. Model checker aided design of a controller for a wafer scanner. Report NIII-R0430, Nijmegen Institute for Computing and Information Sciences, University of Nijmegen, 2004. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/HNV04.html>.
- [183] M. Hendriks, N.J.M. van den Nieuwelaar, and F.W. Vaandrager. Model checker aided design of a controller for a wafer scanner. In T. Margaria, B. Steffen, A. Philippou, and M. Reitenspiess, editors, *Preliminary Proceedings International Symposium on Leveraging Applications of Formal Methods (ISoLA 2004)*, in October/November 2004, Paphos, Cyprus, pages 201–209. Department of Computer Science, University of Cyprus, 2004. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/HNV04.html>. Technical Report TR-2004-6. Full version of paper submitted to STTT.
- [184] M. Hendriks, N.J.M. van den Nieuwelaar, and F.W. Vaandrager. Recognizing finite repetitive scheduling patterns in manufacturing systems. In G. Kendall, E. Burke, and S. Petrovic, editors, *Proceedings of the 1st Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA 2003)*, Nottingham, UK, Volume I, pages 291–319. The University of Nottingham, August 2003. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/HNV03.html>. ISBN 0-9545821-0-1.

- [185] M. Hendriks and M. Verhoef. Timed automata based analysis of embedded system architectures, 2005. Available from World Wide Web: <http://www.cs.ru.nl/ita/publications/papers/martijnh/CarRadio/CarRadio.pdf>.
- [186] H. Hermanns, H.C. Bohnenkamp, D.N. Jansen, J.-P. Katoen, and Y.S. Usenko. An industrial-strength formal method: A modest survey. In *In 1st International Conference on International Symposium on Leveraging Applications of Formal Methods, (ISOLA)*, LNCS, Paphos, Cyprus, 2004. Springer. Invited contribution.
- [187] H. Hermanns, D.N. Jansen, and Y.S. Usenko. From stocharts to modest: a comparative reliability analysis of train radio communications. In *Proc. 5th International Workshop on Software and Performance (WOSP'05)*, Palma de Mallorca, Spain, July 2005.
- [188] H. Hermanns, J.-P. Katoen, J. Meyer-Kayser, and M. Siegle. A tool for model-checking markov chains. *Int. Journal on Software Tools for Technology Transfer*, 4(2):153–172, 2003. Available from World Wide Web: <http://link.springer.de/link/service/journals/10009/bibs/3004002/30040153.pdf>.
- [189] Holger Hermanns, Ulrich Herzog, and Joost-Pieter Katoen. Process algebra for performance evaluation. *Theoretical Computer Science*, 274(1-2):43–87, 2002. Available from World Wide Web: [http://dx.doi.org/10.1016/S0304-3975\(01\)00042-1](http://dx.doi.org/10.1016/S0304-3975(01)00042-1).
- [190] Holger Hermanns and Christophe Joubert. A set of performance and dependability analysis components for cadp. In *Proceedings TACAS 2003*, 2003. Available from World Wide Web: <http://www.inrialpes.fr/vasy/Publications/Hermanns-Joubert-03.html>.
- [191] A. Hessel, K. G. Larsen, B. Nielsen, P. Pettersson, and A. Skou. Time-optimal real-time test case generation using UPPAAL. In Alexandre Petrenko and Andreas Ulrich, editors, *Proc. of 3rd Int. Workshop on Formal Approaches to Testing of Software (FATES'2003)*, number 2931 in Lecture Notes in Computer Science, pages 136–151. Springer-Verlag, 2003. Available from World Wide Web: <http://www.docs.uu.se/docs/rtmv/papers/hlnps-fates03.pdf>.
- [192] A. Hessel, K. G. Larsen, B. Nielsen, P. Pettersson, and A. Skou. Time-optimal test cases for real-time systems. In *Proc. of 1st Int. Workshop on Formal Modeling and Analysis of Timed Systems (FORMATS'2003)*, Lecture Notes in Computer Science. Springer-Verlag, 2003. Available from World Wide Web: <http://www.docs.uu.se/docs/rtmv/papers/hlnps-formats03.pdf>.
- [193] Anders Hessel, Kim G. Larsen, Brian Nielsen, Paul Pettersson, and Arne Skou. Time-Optimal Test Cases for Real-Time Systems. In *3rd International Workshop on FORMAL APPROACHES TO TESTING OF SOFTWARE (FATES 2003)*, Montral, Qubec, Canada, October 2003. Available from World Wide Web: <http://www.cs.auc.dk/~bnielsen/Published/timeoptimal.pdf>. In affiliation with the 18th IEEE International Conference on AUTOMATED SOFTWARE ENGINEERING (ASE 2003).
- [194] Anders Hessel, Kim G. Larsen, Brian Nielsen, Paul Pettersson, and Arne Skou. Time-Optimal Test Cases for Real-Time Systems–extended abstract. In *1st International Workshop on Formal Modeling and Analysis of Timed Systems (FORMATS)*, September 2003. Available from World Wide Web: <http://www.cs.auc.dk/~bnielsen/Published/format03.pdf>. Invited Talk by Paul Pettersson.
- [195] J. Hooman and M.B. van der Zwaag. A semantics of communicating reactive objects with timing. In *Proceedings SVERTS Workshop of the Sixth International Conference on the Unified Modeling Language, UML 2003*, 2003. Available from World Wide Web: <http://www-verimag.imag.fr/EVENTS/2003/SVERTS/PAPERS-WEB/13-HoomanZwaag.pdf>.
- [196] David N. Jansen, H. Hermanns, and Joost-Pieter Katoen. A QoS-oriented extension of UML statecharts. In *UML 2003 — The Unified Modeling Language*, pages 76–91, San Fransisco, USA, 2003. Lecture Notes in Computer Science, Vol. 2863, Springer-Verlag. Available from World Wide Web: http://fmt.cs.utwente.nl/publications/files/400_jhk03.pdf.

- [197] D.N. Jansen and H. Hermanns. Dependability checking with stocharts: Is train radio reliable enough for trains? In *In 1st International Conference on Quantitative Evaluation of Systems (QEST)*, pages 250–259. IEEE CS Press, 2004.
- [198] B. Jeannet, P.R. D’Argenio, and K.G. Larsen. RAPTURE: A tool for verifying Markov Decision Processes. In I. Cerna, editor, *Tools Day’02*, Brno, Czech Republic, Technical Report. Faculty of Informatics, Masaryk University Brno, 2002. Available from World Wide Web: <http://www.cs.famaf.unc.edu.ar/dargenio/papers/tools-day-concur2002.ps.gz>.
- [199] J. Kapinski, O. Maler, O. Stursberg, and B. H. Krogh. On systematic simulation of open continuous systems. In *Hybrid Systems: Computation and Control, LNCS 2623*, pages 283–297. Springer, 2003. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/hsc03_simu.ps.
- [200] J.-P. Katoen, Henrik Bohnenkamp, H. Hermanns, and J. Klaren. *Embedded software analysis with MOTOR*, pages 268–294. LNCS. Springer, Bertinoro, Italy, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/KBHK04.ps.gz>.
- [201] D.K. Kaynar, N.A. Lynch, R. Segala, and F.W. Vaandrager. A framework for modelling timed systems with restricted hybrid automata. In *Proceedings of the 24th International IEEE Real-Time Systems Symposium (RTSS03)*, December 3–5, 2003, Cancun, Mexico, pages 166–177. IEEE Computer Society, 2003. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/RTSS03.html>.
- [202] D.K. Kaynar, N.A. Lynch, R. Segala, and F.W. Vaandrager. The theory of timed I/O automata. Technical Report MIT-LCS-TR-917, MIT Laboratory for Computer Science, Cambridge, MA, 2003. Available from World Wide Web: <http://theory.lcs.mit.edu/tds/papers/Kirli/TIOA-TR.ps>.
- [203] Ch. Kloukinas and S. Yovine. Synthesis of safe, qos extendible, application specific schedulers for heterogeneous real-time systems. In *Proceedings of ’5th Euromicro Conference on Real-Time Systems (ECRTS’03)’, Porto, Portugal, July 2003*. Available from World Wide Web: http://www-verimag.imag.fr/PEOPLE/Christos.Kloukinas/IN2P3/kloukinas_yovine.pdf.
- [204] Christos Kloukinas, Chaker Nakhli, and Sergio Yovine. A methodology and tool support for generating scheduled native code for real-time Java applications. In Rajeev Alur and Insup Lee, editors, *EMSOFT 2003*, volume 2855 of *Lecture Notes in Computer Science*, pages 274–289, Philadelphia, Pennsylvania, USA, October 2003. Springer-Verlag.
- [205] S. Kowalewski and M. Rittel. Real-time service allocation for car periphery supervision, 2002. Available from World Wide Web: http://ametist.cs.utwente.nl/RESEARCH/AMETIST_CPSPrelimDescription_1_0.pdf. Deliverable 3.3.1 from the IST project AMETIST.
- [206] M. Krichen and S. Tripakis. Black-box conformance testing for real-time systems. In *11th International SPIN Workshop on Model Checking of Software (SPIN’04)*, volume 2989 of *LNCS*. Springer, 2004.
- [207] M. Krichen and S. Tripakis. An expressive and implementable formal framework for testing real-time systems. In *The 17th IFIP Intl. Conf. on Testing of Communicating Systems TESTCOM’05*, LNCS. Springer, 2005.
- [208] M. Krichen and S. Tripakis. State identification problems for timed automata. In *The 17th IFIP Intl. Conf. on Testing of Communicating Systems (TESTCOM’05)*, LNCS. Springer, 2005.
- [209] Hillel Kugler and Gera Weiss. Planning a production line with LSCs. Research report, Weizmann, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/WISPublications/cybernetix.zip>.
- [210] M. Kurban, P. Niebert, and W. Vogler. Stronger reduction criteria for Local First Search. draft, 2004.

- [211] R. Langerak, J.W. Polderman, and T. Krilavičius. Stability analysis for hybrid automata using conservative gains. In *In proc. IFAC Conference on Analysis and Design of Hybrid Systems (ADHS03)*, St. Malo, France, 2003. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/LangerakPoldermanKrilavicius_Stability.ps.
- [212] R. Langerak, J.W. Polderman, and T. Krilavičius. Stability analysis for hybrid automata using optimal lyapunov functions. In *Proc. International Conference on Dynamical Systems Modeling and Stability Investigation*, May 27-30, 2003, Kyiv, Ukraine, 2003. 1 page abstract, to appear.
- [213] K. G. Larsen. Resource-efficient scheduling for real time systems. In *Proc. of Third Int. Conf. On Embedded Software (EMSOFT'2003)*, volume 2855 of *Lecture Notes in Computer Science*, pages 16–19. Springer-Verlag, 2003.
- [214] K. G. Larsen, F. Larsson, P. Pettersson, and W. Yi. Compact data structure and state-space reduction. *The International Journal of Time-Critical Computing Systems*, 25(2):255–275, September 2003. Available from World Wide Web: <http://www.docs.uu.se/docs/rtmv/papers/11pw-rtss97.ps.gz>.
- [215] K. G. Larsen, M. Mikucionis, and B. Nielsen. Real-time system testing on-the-fly. Brics report series, Basic Research In Computer Science, 2003. Available from World Wide Web: <http://www.brics.dk/RS/03/49/BRICS-RS-03-49.pdf>.
- [216] K. G. Larsen and P. Niebert, editors. *Formal Modeling and Analysis of Timed Systems (FORMATS)*, volume 279 of *Lecture Notes in Computer Science*. Springer-Verlag, 2004. Available from World Wide Web: <http://www.springeronline.com/sgw/cda/frontpage/0,10735,5-156-22-29559443-0,00.html>.
- [217] K.G. Larsen, F. Larsson, P. Pettersson, and W. Yi. Compact data structure and state-space reduction for model-checking real-time systems. *In Real-Time Systems - The International Journal of Time-Critical Computing System*, 25(1), 2003. Available from World Wide Web: <http://www.docs.uu.se/docs/rtmv/papers/11pw-rts02.ps.gz>.
- [218] K.G. Larsen, M. Mikucionis, B. Nielsen, and A. Skou. Testing real-time embedded software using UPPAAL-Tron: An inudstrial case study. In *to appear in Proceedings of EMSOFT 2005*, Lecture Notes in Computer Science. Springer Verlag, 2005. Available from World Wide Web: <http://www.cs.aau.dk/~kg1/AMETIST/Year3/1mns05.ps>.
- [219] Kim Larsen, Marius Mikucionis, and Brian Nielsen. Online Testing of Real-time Systems using Uppaal. In Jens Grabowski and Brian Nielsen, editors, *International workshop on Formal Approaches to Testing of Software*, Co-located with IEEE Conference on Automates Software Engineering 2004, Linz, Austria., September 2004. Available from World Wide Web: <http://www.cs.auc.dk/~bnielsen/Published/fates04.ps>.
- [220] Kim G. Larsen, Alexandre David, John Haakansson, and Paul Pettersson. Minimal dbm subtraction. In Paul Pettersson and Wang Yi, editors, *Proceedings of 16th Nordic Workshop on Programming Theory*, number 2004-041 in Uppsala technical report, pages 17–21. Uppsala University, October 2004. Available from World Wide Web: <http://www.cs.aau.dk/~kg1/AMETIST/Year3/DBM.ps>.
- [221] Kim G. Larsen and Brian Nielsen. ROAD MAP on Hard Real-Time Development Environments. Chapter 7: Tools for Verification and Validation. Year 1 deliverables of Project IST-2001-34820 ARTIST:Advanced Real-Time Systems ARTIST IST-2001-34820, Project IST-2001-34820 ARTIST:Advanced Real-Time Systems, May 2003. Available from World Wide Web: <http://www.artist-embedded.org/>. Road map is to be publised as a book/volume by Springer Verlag.
- [222] Kim G. Larsen and Jacob I. Rasmussen. Optimal conditional reachability for multi-priced timed automata. In *In Proceedings of FoSSACS 2005*, number 3441 in Lecture Notes in Computer Science, pages 234–249, 2005. Available from World Wide Web: <http://www.cs.aau.dk/~kg1/AMETIST/Year2.5/optimalconditional.ps>.

- [223] M. Layouni, J. Hooman, and S. Tahar. On the correctness of an intrusion-tolerant group communication protocol. In *Proceedings 12th Conference on Correct Hardware Design and Verification Methods (CHARME 2003)*, pages 231–246. Lecture Notes in Computer Science 2860, Springer-Verlag, 2003. Available from World Wide Web: <http://www.niii.kun.nl/~hooman/CHARME03.html>.
- [224] S. Loeschmann and D. Ludewig. Case study 4: Detailed description of the model of a lacquer production, 2002. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del3.4.1.pdf>. Deliverable 3.4.1 from the IST project AMETIST.
- [225] D. Lugiez, P. Niebert, and S. Zennou. Dynamic bounds and transition merging for local first search. In *Model Checking Software*, volume 2318 of *LNCS*, pages 221–229, 2002. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/sw02.pdf>.
- [226] D. Lugiez, P. Niebert, and S. Zennou. Clocked mazurkiewicz traces for partial order reductions of timed automata, 2003. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/clockedmazu.pdf>.
- [227] Denis Lugiez, Peter Niebert, and Sarah Zennou. A partial order semantics approach to the clock explosion problem of timed automata. In Kurt Jensen and Andreas Podelski, editors, *Tools and Algorithms for the Construction and Analysis of Systems: 10th International Conference, TACAS 2004*, volume 2988 of *LNCS*, pages 296–311. Springer-Verlag, 2004. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/tacas04.pdf>.
- [228] Denis Lugiez, Peter Niebert, and Sarah Zennou. A partial order semantics approach to the clock explosion problem of timed automata. In Kurt Jensen and Andreas Podelski, editors, *accepted for Theoretical Computer Science - special issue on selected papers of TACAS 2004*, volume 2988 of *LNCS*, page 40 pages. Springer-Verlag, 2005. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/specialissue.pdf>.
- [229] A. Mader. Deriving schedules for the cybernetix case study, 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/Docs/INTERNAL/PUBLICATIONS/UTPublications/mader-cybernetix2003.ps>.
- [230] A. Mader. Towards modelling a value chain management example with uppaal - ametist case study 4, 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/mader-axxon2003.ps.gz>.
- [231] M. Mahfoudh, P. Niebert, E. Asarin, and O. Maler. A satisfiability checker for difference logic. In *SAT*, 2002. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/Papers/solver.ps>.
- [232] Moez Mahfoudh. *On Satisfiability Checking for Difference Logic*. PhD thesis, UJF Grenoble, May 2003. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/Papers/thesis-moez.ps>.
- [233] O. Maler. Dissemination and use plan, October 2002. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DELIVERABLES/del4.ps>. Deliverable 4 from the IST project AMETIST.
- [234] O. Maler. On optimal and sub-optimal control in the presence of adversaries. In *Workshop on Discrete Event Systems (WODES)*, pages 1–12. IFAC, 2004. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/Papers/wodes.ps>. Invited talk.
- [235] O. Maler, B. Krogh, and M. Mahfoudh. On control with bounded computational resources. In W. Damm and E-R Olderdög, editors, *FTRTFT'02*, volume 2469 of *LNCS*, pages 147–164. Springer, 2002. Available from World Wide Web: <http://www-verimag.imag.fr/PEOPLE/Oded.Maler/Papers/resources.ps>.

- [236] O. Maler and D. Nickovic. Monitoring temporal properties of continuous signals. In *FORMATS/FTRTFT'04*, number 3523 in LNCS, pages 152–166. Springer, 2004. Available from World Wide Web: <http://www-verimag.imag.fr/~maler/Papers/monitor.ps>.
- [237] Oded Maler and Amir Pnueli. On Recognizable Timed Languages. In Igor Walukiewicz, editor, *Proceedings FOSSACS 2004, Barcelona, Spain, March 29 - April 2, 2004*, volume 2987 of *Lecture Notes in Computer Science*, pages 348–362. Springer, 2004. Available from World Wide Web: <http://www.informatik.uni-trier.de/~ley/db/conf/fossacs/fossacs2004.html#MalerP04>.
- [238] M. Massink, J.-P. Katoen, and D. Latella. Model checking dependability attributes of wireless group communication. In *Dependable Systems and Networks - Performance and Dependability Symposium (DSN 2004)*, pages 711–720, Firenze, Italy, 2004. IEEE CS Press. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/MKL04.ps>.
- [239] P. Niebert, M. Mahfoudh, E. Asarin, M. Bozga, N. Jain, and O. Maler. Verification of timed automata via satisfiability checking. In W. Damm and E-R Olderog, editors, *FTRTFT*, volume 2469 of *LNCS*, pages 225–244. Springer, 2002. Available from World Wide Web: <http://www-verimag.imag.fr/PEOPLE/Oded.Maler/Papers/timedbmc.ps>.
- [240] Peter Niebert. Petri nets an intuitive formalism for concurrency, 2 parts (in german). *Automatisierungs Technik*, March 2003. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/at0303304.pdf>.
- [241] Peter Niebert and Mathieu Agopian. Notes on the verification of the hpx error handling mode, May 2005. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/errorhandling.pdf>. manuscript.
- [242] Julian Ober, Susanne Graf, and Ileana Ober. Model checking of UML models via a mapping to communicating extended timed automata. In *SPIN'04 Workshop on Model Checking of Software, 2004*, volume LNCS 2989, pages 127–145, 2004. Available from World Wide Web: <http://springerlink.metapress.com/content/2u0c0n4a5r10kkqd/fulltext.pdf>.
- [243] S. Panek, S. Engell, and C. Lessner. Scheduling of a pipeless multi-product batch plant using mixed-integer programming combined with heuristics. In *Proc. European Symposium on Computer Aided Process Engineering, ESCAPE 15*, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/PEL05.pdf>. accepted.
- [244] S. Panek, O. Stursberg, and S. Engell. Job-shop scheduling by combining reachability analysis with linear programming. In *Proc. 7th Int. Workshop on Discrete Event Systems*, pages 199–204, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/PSE04b.pdf>.
- [245] S. Panek, O. Stursberg, and S. Engell. Optimization of timed automata models using mixed-integer programming. In *Formal Modeling And Analysis of Timed Systems*, volume 2791 of *LNCS*, pages 73–87. Springer, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/PSE04a.pdf>.
- [246] S. Panek, O. Stursberg, and S. Engell. Efficient synthesis of production schedules by optimization of timed automata. *Control Engineering Practice*, 2005. submitted.
- [247] Sebastian Panek and Sebastian Engell. Scheduling of a pipeless multi-product batch plant using mixed-integer programming combined with heuristics. In *Proc. ESCAPE 15*. ESCAPE 15, 2004. PO-151.
- [248] Sebastian Panek and Sebastian Engell. Value chain optimisation / improvements in the solution by mathematical programming. internal report ametist, University of Dortmund, May 2004. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/milp_approach.pdf. Case study 4, deliverable 3.4.3.

- [249] J. Rasmussen, K. G. Larsen, and K. Subramani. Resource-optimal scheduling using priced timed automata. In *Proc. 10th Int. Conf. of Tools and Algorithms for the Construction and Analysis of Systems (TACAS'2004)*, volume 2988 of *Lecture Notes in Computer Science*, pages 220–235. Springer-Verlag, 2004. Available from World Wide Web: <http://www.springerlink.com/app/home/contribution.asp?wasp=9a0qbvyyrjdjt6rvmewp&referrer=parent&back>.
- [250] M.P. Remelhe, S. Lohmann, O. Stursberg, S. Engell, and N. Bauer. Algorithmic verification of logic controllers given as sequential function charts. In *Proc. IEEE Conf. on Computer-Aided Control System Design*, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/RLSEB04.pdf>.
- [251] Theo Ruys. Optimal Scheduling Using Branch and Bound with SPIN 4.0. In Thomas Ball and Sriram K. Rajamani, editors, *Model Checking Software – Proceedings of the 10th International SPIN Workshop (SPIN 2003)*, volume 2648 of *Lecture Notes in Computer Science*, pages 1–17, Portland, OR, USA, May 2003. Springer-Verlag, Berlin. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/UTPublications/ruys-spin2003.pdf>.
- [252] Theo Ruys and Ed Brinksma. Managing the verification trajectory. *STTT*, 4(2):246–259, 2003. Available from World Wide Web: <http://springerlink.metapress.com/openurl.asp?genre=article&id=doi:10.1007/s10009-002-0078-1>.
- [253] G. Sand and S. Engell. Aggregated batch scheduling in a feedback structure. In J. v. Schijndel and J. Grievink, editors, *European Symp. on Computer Aided Process Engineering-12*, volume 10 of *Computer-Aided Chemical Engineering*, pages 775–780. Elsevier Science, 2002. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/cp2_sand+_02.ps.
- [254] G. Sand and S. Engell. Modelling and solving real-timescheduling problems by stochastic integer programming. *Computers and Chemical Engineering*, 28:1087–1103, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/SaEn04b.pdf>.
- [255] G. Sand and S. Engell. Risk conscious scheduling of batch processes. In *Proc. Computer-Aided Chemical Engineering*, pages 588–593, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/CACE04.pdf>.
- [256] E. Sasnauskaite and M. Mikucionis. *Memory Interface Analysis using the Real-Time Model Checker UPPAAL*. Master's thesis, University of Aalborg, 2002. Available from World Wide Web: <http://www.cs.auc.dk/~kg1/AMETIST/sm.ps>. Internal document from the IST project AMETIST.
- [257] J. Sifakis, S. Tripakis, and S. Yovine. Building models of real-time systems from application software. In *Proceedings of the IEEE Special issue on modeling and design of embedded*, pages 91(1):100–111, January 2003. Available from World Wide Web: http://ieeexplore.ieee.org/xpls/abs_all.jsp?isNumber=263697=JNL&arnumber=1173199&arSt=+100&ared=+111&arAuthor=Sifakis%2C+J.%3B+Tripakis%2C+S.%3B+Yovine%2C+S.&arNumber=1173199&a_id0=1173177&a_id1=1173180&a_id2=1173184&a_id3=1173187&a_id4=1173191&a_id5=1173196&a_id6=1173199&a_id7=1173202&a_id8=1173203&a_id9=1173205&a_id10=1173207&a_id11=1173210&a_id12=1173213&a_id13=1173215&a_id14=1173229&count=15.
- [258] M.I.A. Stoelinga and F.W. Vaandrager. A testing scenario for probabilistic automata. In *Proceedings of the 30th International colloquium on automata, languages and programming (ICALP'03)*, volume 2719 of *Lecture Notes in Computer Science*, pages 407–418. Springer-Verlag, 2003. Available from World Wide Web: <http://wwwhome.cs.utwente.nl/~marielle/papers/testing-short.ps>.
- [259] O. Stursberg. Dynamic optimization of processing systems with mixed degrees of freedom. In *Proc. 7th Int. Symposium on Dynamics and Control of Process Systems*, page ID: 164, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/DYCOPS04.pdf>.

- [260] O. Stursberg. A graph search algorithm for optimal control of hybrid systems. In *Proc. 43rd IEEE Conf. on Decision and Control*, pages 1412–1417, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/Stu04b.pdf>.
- [261] O. Stursberg. Synthesis of supervisory controllers for hybrid systems using abstraction refinement. In *16th IFAC World Congress Prague*, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/Stu05.pdf>. accepted.
- [262] O. Stursberg, A. Fehnker, Z. Han, and B. H. Krogh. Specification-guided analysis of hybrid systems using a hierarchy of validation methods. In *IFAC Conf. on Analysis and Design of Hybrid Systems*, pages 289–295, 2003. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/adhs_cgv.pdf.
- [263] O. Stursberg, A. Fehnker, Z. Han, and B.H. Krogh. Verification of a cruise control system using counterexample-guided search. *Control Engineering Practice*, 12(10):1291–1303, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/SFHK04.pdf>.
- [264] O. Stursberg and B. H. Krogh. Efficient representation and computation of reachable sets for hybrid systems. In *Hybrid Systems: Computation and Control, LNCS 2623*, pages 482–497. Springer, 2003. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/hbcc03_hull.pdf.
- [265] O. Stursberg, S. Lohmann, and S. Engell. Improving dependability of logic controllers by algorithmic verification. In *16th IFAC World Congress Prague*, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/SLE05.pdf>. accepted.
- [266] Olaf Stursberg and Sebastian Engell. Optimal control of switched continuous systems using mixed-integer programming. In *15th IFAC World Congress of Automatic Control*, Barcelona, Spain, 2002. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/IFAC_WC_02.pdf.
- [267] Olaf Stursberg and Sebastian Panek. Control of switched hybrid systems based on disjunctive formulations. In *Hybrid Systems: Computation and Control, LNCS 2289*, pages 421–435. Springer, 2002. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/78hscc02.pdf>.
- [268] J. Till, S. Engell, S. Panek, and O. Stursberg. Applied hybrid system optimization - an empirical investigation of complexity. *Control Engineering Practice*, 12(10):1269–1278, 2004. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/TEPS04.pdf>.
- [269] J. Till, S. Engell, G. Sand, and E. Clostermann. Rigorous vs. stochastic algorithms for two-stage stochastic integer programming applications. In *Proc. IEEE International Conference on Intelligent Computing*, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/TESC05.pdf>. accepted.
- [270] J. Till, G. Sand, S. Engell, M. Emmerich, and L. Schoenemann. A hybrid algorithm for solving two-stage stochastic integer problems by combining evolutionary algorithms and mathematical programming methods. In *Proc. European Symposium on Computer Aided Process Engineering, ESCAPE 15*, 2005. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/TSEES05.pdf>. accepted.
- [271] Jochen Till, Sebastian Engell, Sebastian Panek, and Olaf Stursberg. Empirical complexity analysis of a milp-approach for optimization of hybrid systems. In *IFAC Conference on Analysis and Design of Hybrid Systems*, pages 129–134, Saint-Malo, France, 2003. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/DORTMUNDPublications/adhs03_paper88.pdf.

- [272] S. Tripakis. Description and schedulability analysis of the software architecture of an automated vehicle control system. In *EMSOFT*, 2002. Available from World Wide Web: <http://www-verimag.imag.fr/~tripakis/emsoft02-full.ps.gz>.
- [273] S. Tripakis. Fault diagnosis for timed automata. In *FTRTFT*, 2002. Available from World Wide Web: http://www-verimag.imag.fr/~tripakis/ta_diag.pdf.
- [274] S. Tripakis. Automated module composition. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS'03)*, volume 2619 of *LNCS*. Springer, 2003.
- [275] S. Tripakis. Folk theorems on the determinization and minimization of timed automata. In *Formal Modeling and Analysis of Timed Systems (FORMATS'03)*, LNCS. Springer, 2003.
- [276] S. Tripakis. Undecidable problems of decentralized observation and control. *Information Processing Letters*, 90(1):21–28, 2004.
- [277] F.W. Vaandrager and A.L. de Groot. Analysis of a biphasic mark protocol with Uppaal and PVS. Technical Report NIII-R0445, NIII, Radboud University Nijmegen, 2004. Available from World Wide Web: <http://www.cs.kun.nl/ita/publications/papers/fvaan/BMP.html>. Published in Formal Aspects of Computing Journal 18(4):433-458, December 2006.
- [278] W. Vogler and K.G. Larsen, editors. *Proceedings of the 3rd International Workshop on Models for Time-Critical Systems, MTCS'02*. Electronic Notes in Theoretical Computer Science, 2002.
- [279] Gera Weiss. Optimal Scheduler for a Memory Card. Research report, Weizmann, 2002. Available from World Wide Web: http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/WISPublications/Optimal_Schedule_for_a_Memory_Card.
- [280] Gera Weiss. Modeling smart-card personalization machine with LSCs. Research report, Weizmann, 2003. Available from World Wide Web: <http://ametist.cs.utwente.nl/INTERNAL/PUBLICATIONS/WISPublications/cybernetix.zip>.
- [281] Gera Weiss. Memoryless output feedback nullification and canonical forms, for time varying systems. 2005. Available from World Wide Web: <http://www.wisdom.weizmann.ac.il/~gera/TimeVarying.pdf>. Submitted to the international journal of control.
- [282] Sarah Zennou. *Methodes d'ordre partiel pour la verification de systemes concurrents et temps reel*. PhD thesis, Universite de Provence, Laboratoire d'Informatique Fondamentale de Marseille, 2004. Available from World Wide Web: <http://www.cmi.univ-mrs.fr/~niebert/docs/theseszennou.pdf>.
- [283] Sarah Zennou, Manuel Yguel, and Peter Niebert. *ELSE*: A new symbolic state generator for timed automata. In Kim G. Larsen and Peter Niebert, editors, *Proceedings of the 1st International Workshop on Formal Modelling and Analysis of Timed Systems, FORMATS 2003*, volume 2791 of *LNCS*, pages 263–270. Springer-Verlag, 2003. Available from World Wide Web: http://www.cmi.univ-mrs.fr/~niebert/docs/else_update.ps.

A List of Deliverables

No	Description	Due Date	Delivery	Status	Resp Partner
4.4	AMETIST Website	May 02	May 02	accepted	UT, all
0.1.1	Project Rep. - Progress & Evaluation	Oct 02	Nov 02	accepted	KUN, all
3.1.1	Case Study 1: Prel. Description	Oct 02	May 02	accepted	LIF, CYR
3.2.1	Case Study 2: Prel. Description	Oct 02	Apr 02	accepted	AAU, Terma
3.3.1	Case Study 3: Prel. Description	Oct 02	Sep 02	accepted	Uni DO, Bosch
3.4.1	Case Study 4: Prel. Description	Oct 02	Oct 02	accepted	Uni DO, Axxom
4	Dissemination and Use Plan	Oct 02	Oct 02	accepted (qual.)	VERIMAG, all
4.1.1	AMETIST Workshop	Oct 02	Apr 02	accepted	VERIMAG
0.1.2	Project Rep. - Progress & Evaluation	Apr 03	Jun 03	accepted	KUN, all
0.2.1	Framework Report (v1)	Apr 03	Jun 03	accepted	VERIMAG, all
0.3.1	Financial Review	Apr 03	Jun 03	accepted	KUN, all
1.5	Modeling: Controller Synthesis	Apr 03	Apr 03	accepted	VERIMAG
2.3.a	A & T: State Space Representations	Apr 03	Jun 03	accepted	LIF
3.1.2	Case Study 1: Model	Apr 03	Jun 03	accepted	LIF, CYR
3.2.2	Case Study 2: Model	Apr 03	May 03	accepted	AAU, Terma
3.3.2	Case Study 3: Model	Apr 03	Jun 03	accepted	Uni DO, Bosch
3.4.2	Case Study 4: Model	Apr 03	Jun 03	accepted	Uni DO, Axxom
3.5.1	Misc. Case Studies: First Year Report	Apr 03	May 03	accepted	UT, all CRs
0.1.3	Project Rep. - Progress & Evaluation	Oct 03	Apr 04	accepted	KUN, all
0.1.4	Mid Term Assessment Report	Apr 04	May 04	accepted	KUN, all
0.2.2	Framework Report (v2)	Apr 04	May 04	accepted	VERIMAG, all
0.3.2	Financial Review	Apr 04	June 04	accepted	KUN, all
1.2	Modelling: Model Composition	Apr 04	May 04	accepted (qual.)	KUN
1.3	Modelling: Quantitative Modelling	Apr 04	May 04	accepted	UT
1.4	Modelling: Scheduling and Planning	Apr 04	May 04	accepted	Uni DO
2.1.1	A & T: Abstraction and Compositionality	Apr 04	May 04	accepted	KUN
2.2.1	A & T: Control Synthesis Algorithms	Apr 04	May 04	accepted	VERIMAG
2.3.b	A & T: State Space Representations (v2)	Apr 04	May 04	accepted (qual.)	LIF
2.4.a	A & T: Stochastic Analysis (v1)	Apr 04	May 04	accepted	UT
2.5.a	A & T: Tool Interaction (v1)	Apr 04	May 04	accepted	AAU
3.1.3	Case Study 1: Optimisation	Apr 04	May 04	accepted	LIF, CYR
3.2.3	Case Study 2: Optimisation	Apr 04	May 04	accepted	AAU, Terma
3.3.3	Case Study 3: Optimisation	Apr 04	May 04	accepted	Uni DO, Bosch
3.4.3	Case Study 4: Optimisation	Apr 04	May 04	accepted	Uni DO, Axxom
3.5.2	Misc. Case Studies: Second Year Report	Apr 04	Apr 04	accepted	UT, all CRs
0.1.5	Project Rep. - Progress & Evaluation	Oct 04	May 05	accepted	KUN, all
0.1.6	Progress report for the last period	Apr 05	Oct 06	submitted	KUN, all
0.2.3	Framework Report (v3b)	Apr 05	May 06	submitted	VERIMAG, all
0.3.3	Financial Review	Aug 05	Oct 05	submitted	KUN, all
1.1	Modelling: Model Classification	Apr 05	May 06	submitted	VERIMAG
2.1.2	A & T: Structure Exploitation	Apr 05	Jun 05	accepted	KUN
2.2.2	A & T: Scheduling and Planning Algorithms	Apr 05	May 06	submitted	VERIMAG
2.3.c	A & T: State Space Representations (v3)	Apr 05	May 06	submitted	LIF
2.4.b	A & T: Stochastic Analysis (v2)	Apr 05	May 05	accepted	UT
2.5.b	A & T: Tools and Tool Interaction (v2)	Apr 05	Oct 06	accepted	AAU
3.1.4	Case Study 1: Final Report	Apr 05	May 06	submitted	LIF
3.2.4	Case Study 2: Final Report	Apr 05	May 05	accepted	AAU, Terma
3.3.4	Case Study 3: Final Report	Apr 05	May 05	accepted	KUN, Bosch
3.4.4	Case Study 4: Final Report	Apr 05	Jun 05	accepted	Uni DO, Axxom
3.5.3	Misc. Case Studies: Final Report	Apr 05	May 06	submitted	UT, all CRs
4.1.2	AMETIST Conference	Apr 05	Jun 05	accepted	VERIMAG
eTIP	Technological Implementation Plan	73	Jun 05	Jun 07	accepted