# Using Rigorous Simulation to Support ISO 26262 Hazard Analysis and Risk Assessment

Adam Duracz*, Henrik Eriksson†, Ferenc A. Bartha‡, Fei Xu*, Yingfu Zeng‡ and Walid Taha*‡
*School of Information Technology, Halmstad University, Halmstad, Sweden, Email: {adam.duracz, fei.xu, walid.taha}@hh.se
†Dependable Systems, SP Technical Research Institute of Sweden, Borås, Sweden, Email: henrik.eriksson@sp.se
‡Dept. of Computer Science, Rice University, Houston TX, USA, Email: {ferenc.a.bartha, yingfu.zeng, walid.taha}@rice.edu

***Abstract*—Rigorous simulation is a new technology that can play a key role in managing uncertainty in the design of safety-critical cyber-physical systems. One of its important applications is the analysis and evaluation of functional safety for road vehicles according to international standards such as ISO 26262. Previous work presented preliminary evidence to support the feasibility of using rigorous simulation for this purpose. Here we report on advances in our implementation of rigorous simulation and show how they enable the rigorous simulation of more refined and more complete models. A larger case study highlights the benefits of these advances and helps us identify new challenges that should be addressed by future work.**

## I. Introduction

As the level of automation and autonomous functions in cyber-physical systems increase, physical testing becomes less effective as a single method for evaluating the safety of a system. To manage the highly variable and hard-to-predict failure modes that result from automation, a wide range of analytic and computational methods are needed to facilitate the analysis and quality control of such systems. We are interested in the use of rigorous simulation techniques to virtualize functional safety tests to evaluate the severity of failures.

In previous work [1] we showed how this can be achieved using an abstract model. It is a simplistic model of a braking system in that it is based on one-dimensional representation of space. It also performs no collision detection. As a result, severity calculations must be performed manually based on simulation results. This simple model sufficed to illustrate relevance of rigorous simulation tools [2] to this problem domain. However, the simplicity of the model also limited its practical utility. At the time of this work, this model pushed the limit of what we could express with the tool, both in terms of domain expertise as well as the maturity of the tool. For example, the tool required all models to consist of a single match statement (representing a single hybrid automaton), which made non-trivial models difficult to express.

This paper reports on a significantly improved rigorous simulator for the Acumen language [3] that supports: non-linear models; multiple conditional statements such as `if` and `match` that may be combined arbitrarily; nested conditionals that can be used to form parallel and nested automata; local bindings; separate models of sub-systems; string and Boolean-valued variables. After reviewing the ISO 26262 standard and its associated Hazard Analysis and Risk Assessment guidelines (Sec. II), we present a case study (Sec. III), an improved model (Sec. III) enabled by the simulator improvements, and how it can be used to support the risk assessment (Sec. III). We conclude with a summary of a work flow that evolved while developing this model (Sec. IV), and a discussion of challenges that arise when simulating models rigorously (Sec. IV and IV).

This work was supported by US National Science Foundation award CPS-1136099, the Swedish Knowledge Foundation (KK), The Center for Research on Embedded Systems (CERES), and VINNOVA (Dnr. 2011-01819).

## II. ISO 26262 and HARA

In 2011 the International Organization for Standardization (ISO) released a standard for functional safety of electrical and/or electronic systems installed in road vehicles (the ISO 26262 [4]). The current edition of the standard addresses passenger vehicles (cars), but upcoming editions are expected to address commercial vehicles (trucks). The standard recognizes three stages of the safety life cycle: concept development, product development and after start-of-production. Naturally, it is more economical to discover and address issues earlier on in the process.

The ISO 26262 standard prescribes that the concept development phase should include a Hazard Analysis and Risk Assessment (HARA). A hazard is a *possible* source of harm caused by the malfunctioning behavior of an item. A risk is the *probability* of a harm. To manage risk, auxiliary quantitative measures of severity, exposure, and likelihood are introduced. Risk is then taken as the product of these three quantities. Severity is a measure of potential injury that follows from a given hazard. Exposure is the expected frequency of the conditions under which the injury can occur. Likelihood is the probability that an accident will occur.

In a variety of other safety-critical domains, such as the nuclear industry and avionics, rather than using arbitrary conventions for quantifying risk, the concept of *safety integrity level* is well-established. To conform with conventions of these domains, the standard introduces a risk classification scheme called Automotive Safety Integrity Level (ASIL). ASIL breaks down risks intro three dimensions: severity (S), exposure (E) and controllability (C). Compared to simple likelihood, controllability is the likelihood that the driver can act to prevent an injury. The ASIL can be used signal to the system developers the level of attention or investment that is needed to mitigate the risk associated with a particular hazard.

A four-level classification is used for each of these quantities, with 0 for lowest level and 3 for the highest. As part of HARA, every hazard is given a classification in terms of each

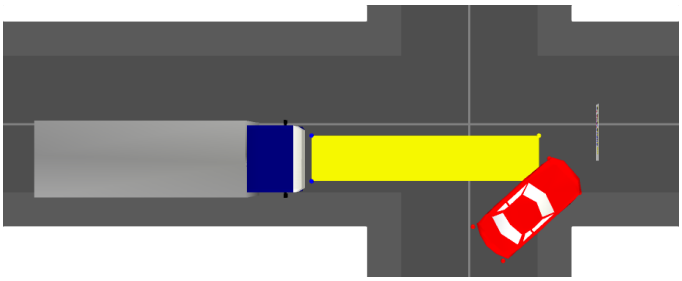

Fig. 1: Overview of Test Scenario.

of these three dimensions. If there is doubt, a conservative (upper bound) classification is made. The lowest severity level, S0, is dedicated to consequences having only material damages, and in that case no ASIL assignment is required. The lowest exposure level, E0, is dedicated to extremely unusual or incredible situations and requires no ASIL assignment. Between each exposure class there is one order of magnitude in probability. Controllability is not exclusive to the driver of the item-equipped vehicle, all people at risk should be considered. It is assumed that the driver is in condition to drive, has a driving license, and complies to legal regulations. Between each controllability class there is one order of magnitude in probability.

## III. Case Study: A More Refined Model

This section presents the case study that we us to evaluate the improvements to our rigorous simulation tool (outlined in the introduction). The key features of the model are: a two-dimensional scenario, braking criteria based on estimated time-to-collision (TTC), two-dimensional collision detection for rectangular vehicles, and explicit calculation of severity class bounds.

Our case study models a test scenario for an advanced emergency brake system (AEBS) for commercial vehicles. It is based on an EU regulation [5] that describes requirements and a type-approval test procedure. The test procedure defines a rear-end collision scenario where the distance between the test and target vehicles is at least 120 m when testing begins. The test vehicle (truck) travels at 80±2 km/h and the target vehicle (saloon car) at 12±2 km/h. To be approved, the truck shall not collide with the car. In our model, the car enters the path of the truck by making a right turn in an intersection. This is illustrated in Fig. 1, where the car (red) is detected by the truck (gray/blue) while making a turn. Detection occurs when the rectangle that bounds the car intersects the yellow rectangle that models the sensor area (shortened to 10 m in the illustration). Thus, the truck AEBS sensor can detect the car already during the turn. The truck is equipped with an idealized sensor with a field-of-view modeled by a rectangle (length = 50 m, width = 2 m). An obstacle is detected as soon as the rectangular bounding box surrounding the obstacle intersects the sensor field-of-view. A haptic warning (pre-brake = 2 m/s$^2$) is issued when the TTC is less than 3.5 s. At TTC = 2.5 s, full braking (5 m/s$^2$) is performed. The mass of the truck and car are set to 55 t and 1.5 t, respectively. An inelastic collision is assumed with a coefficient of restitution of 0.5.

Fig. 2 gives an overview of the model as a nested hybrid automaton. Collision detection and handling is controlled by

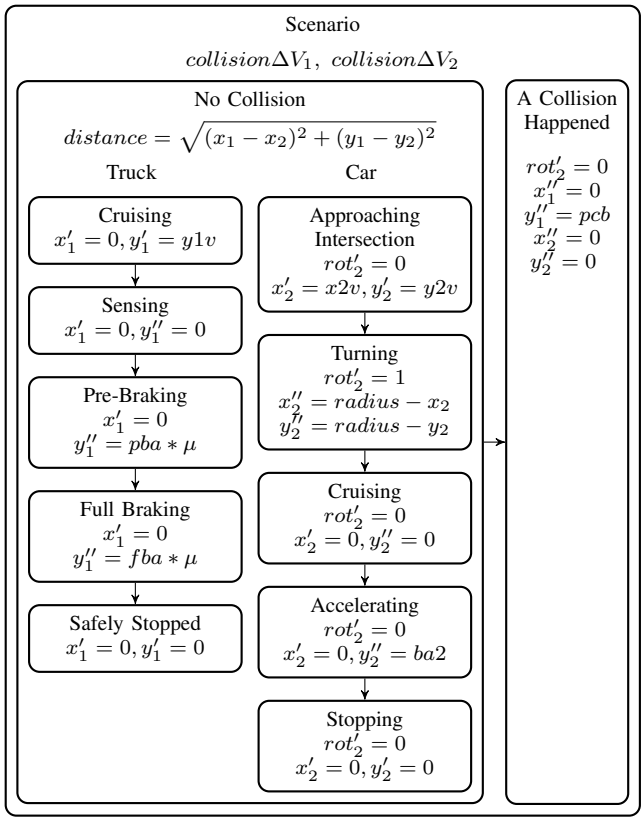


Fig. 2: Nested hybrid automaton representation of model.

the top-level automaton. In the "No Collision" mode, the dynamics of each vehicle are controlled by a separate automaton. Initial parameters passed to the `Scenario` submodel determine the initial positions and velocities for the two vehicles, as well as the braking/acceleration applied when the car enters its "Accelerating" mode. The modes of the truck automaton correspond to different levels of engagement of the truck's sensor and AEBS. The car automaton controls the car's behaviour through and after the turn.

Two key events can occur in the model. The first is detecting when the car enters the truck's sensor area. In the model, this is represented by the transition from "Cruising" to "Sensing" in Fig. 2). The second is when the car collides with the truck. This is represented by the transition from "No Collision" to "A Collision Happened". The conditions for both events are similar, in that they are triggered by conditions that correspond to the intersection between two rectangles. Computing the coordinates of rectangle corners based on the position and orientation of each vehicle requires using non-linear (trigonometric) functions. The expected TTC is used to trigger transitions between the "Sensing/Pre-Braking" and "Pre-Braking/Full Braking" modes of the Truck automaton. This quantity can be estimated based on the velocities and distance between the vehicles. This is another instance where the model requires a non-linear function (square root).

For our test scenario, there are different sources of hazards that must be addressed one at a time. However, we can start with some general remarks that apply to all hazards that we will consider. First, the exposure level classification should be E3. This determination is according to the ISO 26262

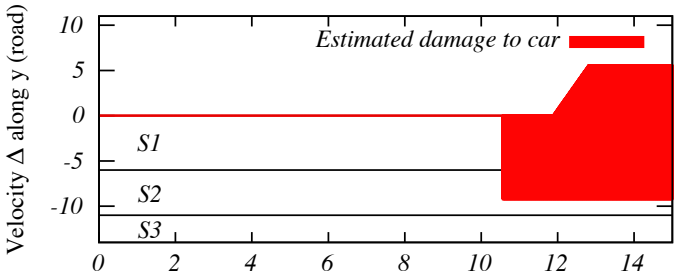


Fig. 3: Enclosure for car velocity change from collision.

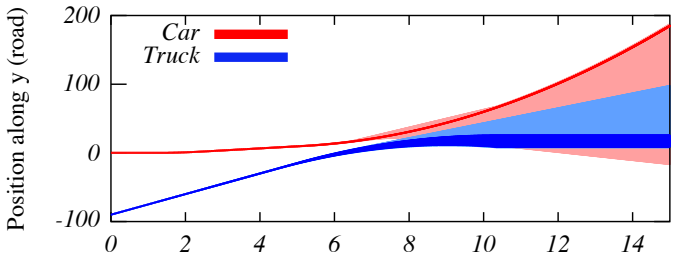


Fig. 4: Enclosures computed with two different step sizes.

standard, and is based on the fact that this scenario can occur in a traffic congestion. Controllability is C3 because in an emergency breaking scenario we can assume that driver take-over is unrealistic. What remains to calculate, therefore, is the severity class for each hazard.

Failing to sense or break would lead to severity class S3 since the speed is 80 km/h. The interesting failure mode that can benefit from simulation is if the AEBS brakes but not with full effect. We will assume (based on domain expertise) that the different severity classes correspond to the following collision speed changes: S3 ($\Delta V \geq 11$ m/s), S2 ($11 > \Delta V > 6$ m/s), and S1 (6 m/s$\geq \Delta V$). An example simulation is shown in Fig. 3. In this example, the full brake deceleration is less than it should be. The figure shows the effect of the resulting collision at time 10.5 on the velocity of the car. The horizontal lines indicate the bounds for the severity classes S1, S2 and S3. The plot shows that severity does not go beyond level S2.

## IV. Analysis and Discussion

Our key observations from this experience are as follows: First, after some work it was possible to rigorous simulate the new model with the intended results. However, getting the model to work with the rigorous simulator still requires some manual intervention. Care is needed to avoid formulations that can have unnecessarily undefined operations (Sec. IV). Selecting the most appropriate time step can be challenging (Sec. IV). We also find that it is very useful to use non-rigorous simulation to develop the model in the first place, before switching to rigorous simulation. The latter currently takes significantly more time to complete, and the run-time properties of rigorous simulation are more sensitive to the choice of simulation step. Finally, the integration of the rigorous simulation engine with the 3D visualization facility of Acumen is particularly important for enabling seamless alternation between non-rigorous and rigorous simulation as the model is being developed. In the rest of this section we describe in more detail the issues relating to partial functions and step size selection.

A pervasive problem when working with rigorous numerical computation is that over-approximation is inevitable. Over-approximation can give rise to operations being evaluated outside their domain. An example we encountered with this model is division by zero. In standard interval arithmetic, division is only defined when the denominator interval does not contain zero. Occasionally, over-approximation can be avoided by increasing the precision of the simulation. This can be achieved, for example, by decreasing the simulation time step. The improved precision (small resulting interval) can eliminate the problem of getting an interval that contains zero. However, this approach may lead to longer simulation times and can simply fail to solve the problem when the result is an open interval adjacent to (but not including) zero. In such cases it is important to consider whether the model can be reformulated to avoid the use of a partial operation. A practical example that arose in this model is a condition $a < b/c$, which was replaced with $a * c < b$ to avoid the problem under the assumption that $c$ is positive, and $a * c > b$ when it is non-positive. The most obvious lesson that can be drawn from this experience is that it is better to avoid the use of partial functions when total functions would suffice. A deeper insight is that rigorous simulation tools *nudge the user* in the direction of such better modeling practices sooner than traditional tools may, since rigorous simulation tools work with sets of values, which provide more extensive testing of models in one run than do traditional simulation tools.

In traditional numerical simulation, reducing simulation time steps generally leads to longer simulations but yields more precise results (until we get to very small time steps). Similar issues happen with rigorous simulation, but with the added complication that both increasing and decreasing simulation steps can increase computational cost. In particular, choosing a smaller step can help us obtain a conclusive outcome from a simulation. For example, one of the scenarios supported by the model has the vehicles come close to collision but still avert it. As shown in Fig. 4, selecting a step that is too large ($2^{-8}$, light colors) yields enclosures that include trajectories corresponding to a collision, which is insufficient to rule out the possibility. Decreasing the step by a factor of four (to $2^{-10}$, dark colors) shows conclusively that the collision does not occur. However, this does increase computational cost. Increasing simulation time steps can also increase runtime. This happens because it can lead to greater uncertainty in the values of variables, which in turn can lead to more branching. Branching can happen when multiple time steps are needed to determine conclusively that an enclosure does cross an event guard. During those steps, the simulator must branch the simulation to take into account both cases (of the event happening or not happening in that step). It is possible that this problem is compounded if, before the branches can be recombined, they lead to more branching. In such situations, decreasing the step can lead to a faster simulation, by reducing the time spent on crossing event boundaries.

## V. Related Work

Commercial tools exist that specifically support ISO 26262 HARA, for example: RiskCAT by CATS Software Tools [6], medini analyser by KPIT medini Technologies [7], SOX2

(SafetyOffice X2) by ENCO Software [8] and Polarium ALM by Polarium Software [9]. The goal of these tools is to support the work by collecting and suggesting hazards and operational situations, as well as combining and checking for completeness and consistency of hazardous events. The tools automatically determine ASILs based on the selected S, E, and C values for each hazardous event. However, they do not explicitly support design decisions and analyses of severity and exposure classes through modeling, simulation, and visualization.

There are also tools for formal analysis of models. Two such examples are Simulink Design Verifier [10] from The MathWorks and SCADE Suite Design Verifier [11] from Esterel Technologies. Both tools use the Prover Plug-In [12] to perform model checking and/or equivalence checking. However, this is limited to discrete-time control systems and does not handle hybrid systems.

Rigorous numerical methods have already been identified as useful in model analysis for the automotive domain. The ability of these methods to process models with uncertain parameters has been used [13] to more efficiently simulate an articulated vehicle control model. Several reachability analysis tools for hybrid systems are designed to produce the reach set-based over-approximation. These tools are often highly tuned for a particular class of problems. Notably, SpaceEx [14] is capable of simulating large systems (comprising hundreds of variables) provided that dynamics, resets and guards are affine. Tools such as Flow* [15], based on Taylor Models or those based on Taylor Integrators [16], are capable of simulating models with non-linear dynamics, guards and resets with good precision. Common to these tools is that they process models expressed in the hybrid automaton [17] formalism. It is simple and well understood, but its lack of support for common programming language features, such as nesting, can make the resulting models verbose.

Tools based on symbolic methods are also capable of rigorous simulation, and have the benefit of producing error-free results. For example, the hybrid theorem prover KeyMaera can be used both for simulating and verifying models of hybrid systems, and has been used for verification of highway traffic control systems [18]. Current implementations of these tools rely on symbolic algebra and theorem proving techniques that require the existence of analytical tools, which may limit their applicability compared to tools based on rigorous numerical methods.

## VI. Conclusions and Future Work

This paper reports on key improvements to the rigorous simulation semantics of Acumen that are intended to enable more precise and complete modeling of vehicle safety testing scenarios. It presents a case study of a substantially more refined model than was previously reported for Acumen. The model explicitly computes bounds on the ISO 26262 severity class of simulated events. Our experience using rigorous simulation to analyze this model is described, including some practical challenges and the workarounds that we have used to circumvent them. The simulation results presented in this paper are reproducible using a recent snapshot release of Acumen[1], which is freely available, open source software. The model is included in the distribution, and can be found under `examples/05_Language_Research/24_ISO26262`.

Important next steps include exploring the possibility of automating some model transformations to avoid problems such as unnecessarily undefined operations, and improving the performance of the enclosure simulation methods. Performance can be improved by reducing the amount of branching. Performance improvements can reduce the need to switch between rigorous and non-rigorous simulators while developing and debugging the model. In addition, extending 3D visualization to represent uncertainties can provide more intuitive visual queues about the propagation of uncertainties. Finally, we would like to explore the possibility of generalizing our case study model into a library of components that can facilitate rapid capture of new test scenarios for this domain.

## Acknowledgment

The authors would like to thank Julie Bonneau for help in the course of developing the model, Eugenio Moggi for feedback on a draft of this paper, and Christian Grante of AB Volvo for his support of this work.

## References

[1] J. Masood, R. Philippsen, J. Duracz, W. Taha, H. Eriksson, and C. Grante, "Domain analysis for standardised functional safety: a case study on design-time verification of automatic emergency braking," in *World Automotive Congress*. FISITA, 2014.

[2] M. Konecny, W. Taha, J. Duracz, A. Duracz, and A. Ames, "Enclosing the behavior of a hybrid system up to and beyond a zeno point," in *Cyber-Physical Systems, Networks, and Applications*, 2013.

[3] W. Taha and R. Philippsen, "Modeling basic aspects of cyber-physical systems," *arXiv preprint arXiv:1303.2792*, 2013.

[4] ISO26262, "Road vehicles – functional safety," 2011.

[5] EU Regulation No 347/2012, "Type-approval requirements for certain categories of motor vehicles with regard to advanced emergency braking systems," 2012.

[6] RiskCAT, http://cats-tools.de, [Acc. 2015-07-07].

[7] Medini Analyser, http://ikv.de, [Acc. 2015-07-07].

[8] SOX2, http://enco-software.com, [Acc. 2015-07-07].

[9] Polarium ALM, http://polarion.com, [Acc. 2015-07-07].

[10] Simulink Design Verifier, http://mathworks.com, [Acc. 2015-07-07].

[11] SCADE Design Verifier, http://esterel-technologies.com, [Acc. 2015-07-07].

[12] Prover, http://prover.com, [Acc. 2015-07-07].

[13] M. Althoff, C. Le Guernic, and B. H. Krogh, "Reachable set computation for uncertain time-varying linear systems," in *Hybrid Systems: Computation and Control*. ACM, 2011.

[14] G. Frehse, C. Le Guernic, A. Donzé, S. Cotton, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, and O. Maler, "SpaceEx: Scalable verification of hybrid systems," in *Computer Aided Verification*. Springer, 2011.

[15] X. Chen, E. Ábrahám, and S. Sankaranarayanan, "Flow*: An analyzer for non-linear hybrid systems," in *Computer Aided Verification*. Springer, 2013.

[16] N. Ramdani and N. S. Nedialkov, "Computing reachable sets for uncertain nonlinear hybrid systems using interval constraint-propagation techniques," *Nonlinear Analysis: Hybrid Systems*, vol. 5, no. 2, 2011.

[17] T. A. Henzinger, "The theory of hybrid automata," in *Logic in Computer Science*. IEEE Computer Society, 1996.

[18] S. Mitsch, S. M. Loos, and A. Platzer, "Towards formal verification of freeway traffic control," in *ICCPS*, 2012.

[1] http://bit.ly/iccps2015-acumen-bf06cac4. To simulate the model rigorously, "2015 Enclosure" must be chosen from the Semantics menu.