SAFETY OPTIMIZATION OF A RADIO-BASED RAILROAD CROSSING

F. Ortmeier, W. Reif, G.Schellhorn

University of Augsburg,
Address: Universitätsstr. 14, 86135 Augsburg, Germany
Phone: (+49) 821 598 2176, Fax: (+49) 821 598 2175, e-mail: {ortmeier,reif,schellhorn}@informatik.uni-augsburg.de

Abstract: In this paper we report on the safety analysis of a distributed and decentralized control of a railroad crossing: the radio-based level crossing. In particular we show how mathematical models and minimization techniques may be used to get good quantitative approximations for risk as well as to give advice to the system engineer how to choose free parameters like safety margins etc.

Keywords: formal safety analysis, fault tree analysis, formal methods, safety–critical systems

1. INTRODUCTION

Many modern books about safety [5][8][9] do not cover quantitative methods or only cover them on the fringes. This is, because quantitative methods usually rely heavily on statistics. So they are often seen as a problem of mathematics. However, it can improve safety analysis a lot, if good interfaces between mathematics and statistics are provided.

Safety optimization is an enhancement to the well-known fault tree analysis (FTA) [2][10] which makes it easy to integrate statistics and optimization techniques into safety analysis. The new aspect - compared to traditional FTA - is that a statistical model of the environment in addition to the components' failure probabilities – which are used for traditional FTA - is built. This allows finding optimal configurations of free parameters as the solution of a minimization problem. This problem can be solved automatically using numerics or in some cases even graphically.

In general technical applications have free parameters, which influence safety requirements: the tolerance of a speed indicator, the accepted time delay between requests and answers of a server or the average maintenance interval are all free parameters of different systems. Such parameters are normally chosen on a basis of previous experience and fine tuned once the system starts working. However, bad choices only become obvious when some hazards occur. So it would be very helpful, if these parameters could be

estimated in advance. This is what safety optimization does.

In this paper we report on the experiences we made when we applied safety optimization to the case study "radio-based level crossing". This work has been developed within the ForMoSA project which is part of the German research foundations priority program ``Integrating software specification techniques for engineering applications''. In Sect. 2 we give a brief introduction of FTA including quantitative FTA. The theoretical foundations of safety optimization are part of Sect. 3. The case study is presented in

Sect. 4. In Sect. 5 the results of safety optimization for the example system are described. Section 6 concludes the paper.

2. FAULT TREE ANALYSIS

In this section we start with a brief introduction into fault tree analysis (FTA). FTA is a top down technique to determine the possible basic component failures (primary failures) of a bad or catastrophic situation which must be avoided. This situation is called hazard. The hazard or top event is always the root of the fault tree and primary failures are its leaves. All inner nodes of the tree are called intermediate events. Starting with the top event the tree is generated by determining the immediate causes that lead to the top event. They are connected to their consequence through a gate. The gate indicates if all (AND-gate) or any (OR-gate) of the causes are necessary to make the consequence happen. The INHIBT-gate states,

that the cause is only critical if some environmental condition holds. Unlike all other nodes of the fault tree, this condition must not be a failure or undesired event. The leaves of a fault tree are primary failures which are not investigated further. Figure 1 shows the symbols for fault tree gates.
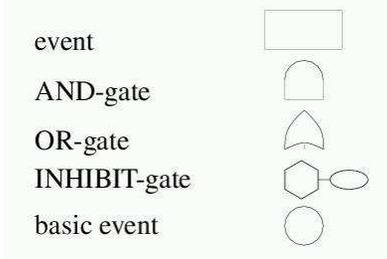

Fig. 1. Fault Tree Symbols

In the remainder of this section we will describe the most important terms of FTA in such detail, that the concept of safety optimization can be explained later.

2.1 Primary Failures and Hazards

For quantitative analysis roots and leaves - i.e. hazards and primary failures - are important. Let $F := \{PF_1,..., PF_n\}$ be the set of all primary failures PF_i and $H := \{H_1,..., H_n\}$ the set of all hazards H_i under consideration.

For each hazard H_i a separate fault tree must be constructed, that describes which combination of basic causes (= primary failures) may be the reason for the hazard.

2.2 Minimal cut sets

Interesting results of fault tree analysis are the minimal cut sets. A cut set CS_H for a hazard H is a set of primary failures, which together form a threat. This means if all primary failures of the cut set take place, then the hazard may occur.

Minimal cut sets MCS_H for a hazard H are cut sets, such that no real subset of MCS_H is a cut set. $MCSS_H$ is the set of all minimal cut sets for a hazard H. This set can be automatically generated from the fault tree[10]. The minimal cut sets describe qualitatively the dependency between hazards and primary failures.

Even more interesting for real world applications is the quantitative one i.e. the dependency between the probability of occurrence of the cut sets and the hazard. This question can be answered with quantitative FTA.

2.3 Calculating probabilities

For calculating probabilities we use a standard formula for calculating hazard probabilities from fault trees [10]. It calculates the probability of a cut set as the product of the probabilities of all its elements. The hazard's probability is calculated as the sum of all its minimal cut sets probabilities. So the probability of a hazard H is calculated as:

$$P_H := \sum_{MCS \in MCSS_H} P(MCS) \qquad (1)$$

$$where \ P(MCS) := \prod_{PF \in MCS} P(PF)$$

This formula is widely used in engineering and broadly accepted, but uses some assumptions about statistical independence. All primary failures are assumed to be pair wise independent. This holds for many applications. If statistical correlation has to be examined, FTA is not a good choice and another approach like common cause analysis or---on the formal side---stochastic model checking [1] has to be used and the probability of the minimal cut sets and hazards have to be calculated separately. This formula also neglects second and higher-order terms in the sum. This is in practice no problem as failure probabilities are usually very small.

In this paper we stick with the assumption of statistical independence and use the standard formula as starting point for our extension of FTA.

2.4 Generalizations

So far this is the standard method of applying quantitative FTA. But for our purposes this is not enough. As (i) for this point of view the worst case is always assumed. This means, all other environmental inputs are as ``bad'' as possible. Another deficiency is (ii) the use of fixed probabilities for failures. In reality these probabilities are usually not constant, but rather depend on some parameters. To overcome these problems, we generalize quantitative FTA by introducing two new types of probabilities: constraint probabilities and parameterized failure probabilities.

Constraint probabilities

Most of the cut sets cause the hazard only if one or more constraints are fulfilled. Sometimes a constraint must hold for all cut sets and sometimes only for some of them. For example the failure of a critical cooling unit is only dangerous if the system which has to be cooled is working. While it is turned off, the failure of such a cooling unit will not have any effect onto the super system. The qualitative dependence between such constraints, cut sets and hazards is often integrated in the fault tree with so called INHIBIT-gates (see Fig. 1). An INHIBIT-gate states a condition or constraint which has to be fulfilled such that the failure cause makes the consequence happen. This condition need not necessarily be a failure, but can also be some environmental influence. Although some types of FTA respect such dependencies in a qualitative manner, they are in general neglected for quantitative FTA.

We introduce constraint probabilities for quantitative analysis which reflect how probable it is that the inputs from the environment are ``bad'' enough to make the hazard happen. So we refine definition of a cut sets probability to get a better approximation:

$$P(MCS) := P(\text{Constraint } s_{MCS}) \prod_{PF \in MCS} P(PF) \quad (2)$$

If one chooses P(Constraints)=1, it means the environment always behaves as bad as possible and one gets the same formula as before. However, if one can estimate P(Constraints) a priori, then the results will be much more precise. This estimation can be approximated by calculating the probabilities of all conditions in INHIBIT-gates along the paths through the tree from the hazard to the elements of the cut sets. An upper bound for the constraint probability is then the product of all conditions' probabilities if statistical independence holds; if not then the maximum is an upper bound for it.

In practice these numbers are really hard to calculate exactly. So most of the time they are only approximated. But even if they can not be approximated very well, they still may be a great help for safety analysis. This is, because variation of the constraints allows examining the behaviour of the system in different working environments. This can give good advice, when trying to estimate how the system will scale in future. The benefit of this last methodology becomes obvious in the case study of Sect. 4.

Parameterized probabilities

The second important generalization we made is that we not only use constant failure probabilities for primary failures, but allow parameterized probabilities. This means, if the probability of a primary failure PF (e.g. a relay fails to close) depends on some parameter X (e.g. the spring tension of the relay), we use a functional mapping between X and P(PF)

$$P(PF): Domain \ (X) \to [0,1] \quad (3)$$

and write P(PF)(X). In principle, there is no restriction on the domain of X, as they only affect which methods are applicable for the solution of the resulting optimization problem. But finite and discrete domains are in general less interesting and rare. In practice P(PF) is usually a (continuous) probabilistic distribution. If the probabilities depend on more than one parameter, then take $X := [x_{1,...,} x_n]$ as the vector of all involved parameters.

All instances of failure probabilities are substituted with the according function. The algorithm for calculating the probability of a cut set is not changed. So the probability of a cut set is then also function of one or more variables. The same is true for the hazards' probabilities and formula (1) rewrites to:

$$P_H(X) := \sum_{MCS \in MCSS_H} P(MCS)(X) \quad (4)$$

$$where \ \ P(MCS)(X) := \prod_{PF \in MCS} P(PF)(X)$$

Note, that the probabilities of cut sets and hazards are no longer fixed numbers, but rather functions of the free parameters of the system. We call these functions parameterized probabilities.

3. SAFETY OPTIMIZATION

In this section we describe the combination of quantitative fault tree analysis and optimization techniques, which leads to safety optimization. The basic idea is as simple as effective. In practice for most systems safety is a trade-off between different undesired events and costs. For example in aviations the main goal of a pre-flight safety check on an airplane before start is to make sure the aircraft is working correctly and will not crash. However, another important goal of the safety check is that an aircraft, which allows safe flight, must not fail the check. Assume that one part of the check is aberration of the air speed indicator. Then it is obvious that the smaller the

allowed tolerance is, the safer the airplane operation will be. On the other hand too small acceptable tolerances will result in many safe aircraft failing the pre-flight check and thus in delay or cancelled flights. So what is the solution? It's of course some middle value between zero tolerance and arbitrary tolerance. Note, that we are not arguing for safety leaks which originated in design flaws, to be left open because of the costs. This approach only addresses hardware failure which can not ultimately be avoided.

This is exactly the point where safety optimization works. It uses mathematical optimization to find the best value for this tolerance parameter. To do this we only need one more information: the cost function.

3.1 Cost function

To do mathematical analysis, a cost function is needed. A cost function describes the total costs that all hazards together cause in average to the operator. This is done by risk assessment. The cost of each hazard will be defined. It is common practice - even as it may seem un-ethical - to do this in cash (e.g. US railways organizations calculate each dead person a standard amount of dollars).

More important for the operators of the system are the mean costs. These are the costs, which have to be expected. These costs depend on the probability of occurrence and absolute cost of the hazard.

In general the cost function is a mapping from the free parameters of the system into the domain of real numbers. In many cases the cost function consists of two terms: one term for hazard costs (this term reflects the mean costs associated with the effects of the hazards) and a second term which directly depends on the parameters themselves (this term reflects the cost the parameter causes – like more reliable sensors are more expensive).

$$\text{costs}(X) := \text{costs}_{hazards}(X) + \text{costs}_{params}(X) \quad (5)$$

$$\text{costs}_{hazards}(X) := \sum_{hazards\ H} P_H(X)\text{costs}_H$$

The costs associated with the hazards (costs$_{hazards}$(X)) may be approximated by the weighted sum of hazard probabilities $P_H(X)$ and mean costs associated with the hazard costs$_H$.

In our approach, the probabilities of the hazards are not, as for standard quantitative FTA,

necessarily constants, but rather functions of free parameters $x_1,...,x_n$. Therefore, we write $P_H(X)$ as described in Sect. 2. The cost function is then a real value function of the free parameters $x_1,...,x_n$. This allows us to use optimization techniques.

3.2 Mathematical optimization

Once the cost function is defined the problem is not any more a safety analysis problem but a mathematical one. The goal is to choose the free parameters $x_1,...,x_n$ such that the cost function is minimized. So the problem is:

Find $x_1,...,x_n$ such that: costs($x_1,...,x_n$) is minimal

To guarantee the existence of the minimum we restrict the domains to be compact intervals. This problem can then be solved with different methods. In simple cases analytical solutions may be found. If the problem is more complex and the cost function still smooth enough (e.g. twice-continuously differentiable) then there exist a lot of algorithms from the domain of nonlinear programming to solve the problem. The simplest one is the gradient method which finds local minima by calculating gradients iteratively by always following the steepest descent. But there exists a wide variety of more elaborate and efficient algorithms. A good introduction to optimization of nonlinear problems may be found in [11] and [12].

If there are only two free variables (as in the following example) and the functions are smooth, then the solutions may be found by using a 3D plot of the cost function and zooming into it. Even if a specific optimization problem is neither analytically nor numerically solvable, can this method yield some good results by testing possible combinations. It is possible to test large number of combinations in very short time. So this technique gives a good impression about the quantitative dependencies between mean costs and free parameters.

4. EXAMPLE: RADIO-BASED LEVEL CROSSING

The German railway organization, Deutsche Bahn, prepares a novel technique to control level crossings: the decentralized, radio-based level crossing control. This technique aims at medium

speed routes, i.e. routes with maximum speed of 160 km/h. An overview is given in [13].
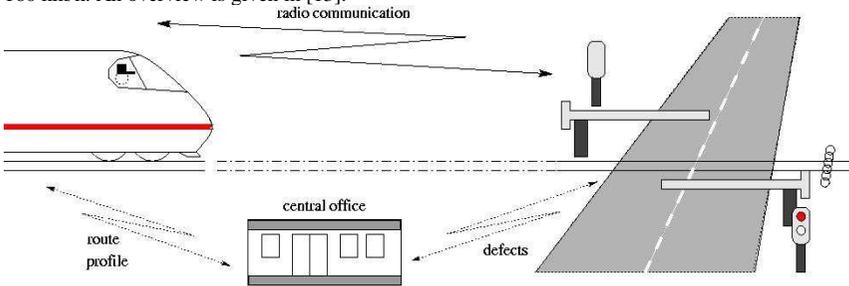

Fig. 2. Radio-based level crossing

The main difference between this technology and the traditional control of level crossings is that signals and sensors on the route are replaced by radio communication and software computations in the train and level crossing. This offers cheaper and more flexible solutions, but also shifts safety critical functionality from hardware to software.

Instead of detecting an approaching train by a sensor, the train computes the position where it has to send a signal to secure the level crossing. To calculate the activation point the train uses data about its position, maximum deceleration and the position of the crossing. Therefore the train has to know the position of the level crossing, the time needed to secure the level crossing, and its current speed and position. The first two items are memorized in a data store and the last two items are measured by an odometer. For safety reasons a safety margin is added to the activation distance. This allows compensating some deviations in the odometer.

When the level crossing receives the command `secure', it switches on the traffic lights, first the `yellow' light, then the `red' light, and finally closes the barriers. When they are closed, the level crossing is `safe' for a certain period of time. The `stop' signal on the train route, indicating an insecure crossing, is also substituted by computation and communication. Shortly before the train arrives the `latest braking point' (latest point, where it is possible for the train to stop before the crossing), it requests the status of the level crossing. When the crossing is safe, it responds with a `release' signal which indicates, that the train may pass the crossing. Otherwise

the train has to brake and stop before the crossing.

The level crossing periodically performs self-diagnosis and automatically informs the central office about defects and problems. The central office is responsible for repair and provides route descriptions for trains. These descriptions indicate the positions of level crossings and maximum speed on the route.

The safety goal of the system is clear: it must never happen, that the train is on the crossing and a car is passing the crossing at the same time. A well designed control system must assure this property at least as long as no component failures occur.

The corresponding hazard H is "whenever a train passes the crossing, the crossing must be safe". This is the only hazard which we will consider in this case study

4.1 Fault Tree Analysis

Component failures may be defects in the brakes of the train, unwanted opening of the level crossing or even drivers, who disregard the red light and the closed barriers (for many crossings the barriers only block on driving lane in each direction so it is possible to drive around the closed barriers without leaving the road and as a matter of fact this really seems to be a problem in some regions).

Fault tree analysis shows which combination of component failures are necessary to make the hazard happen.
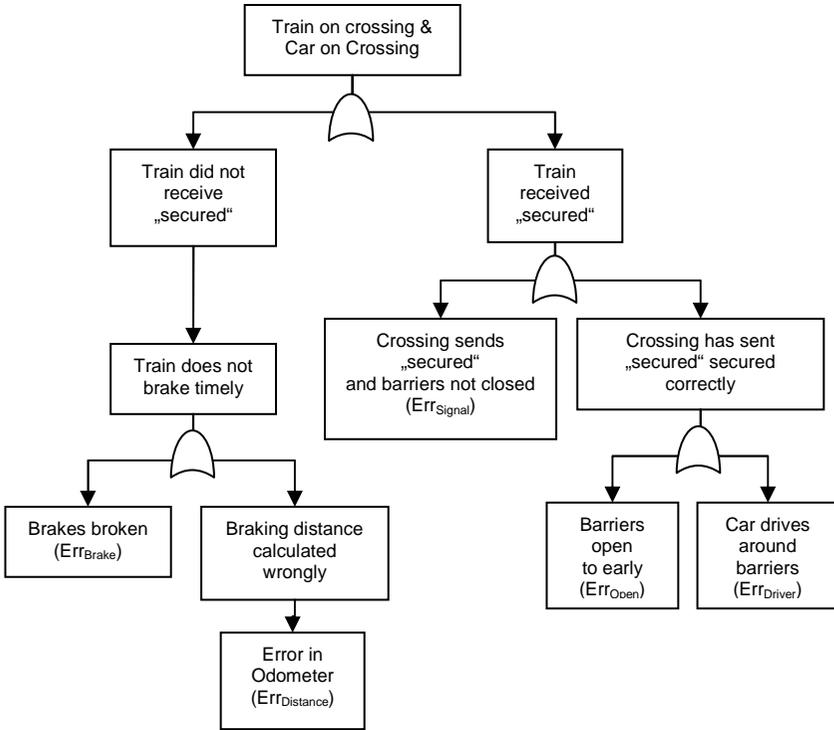
Fig. 3. Fault tree for radio-based level crossing

In Fig. 3 the fault tree for the example is presented. The direct cause of a collision are that either the train does not receive a "crossing secured" message and fails to brake in time (fault of the train) or that the train receives the signal, although the crossing is not secured (fault in the crossing control). These causes are analyzed further until only basic component failures are left. In the example system all cut sets consist only of one element. So in the notation of Sect. 3 we get:

$MCSS_{Collision}$ = { {Err_{Brake}}, {$Err_{Distance}$}, {Err_{Signal}}, {Err_{Open}}, {Err_{Driver}} }

In this example redundant subsystems like emergency brakes etc. have been abstracted. So for example the failure mode "Err_{Brake}" is to be read as "primary and secondary braking system fail

simultaneously. This is also reflected in the following quantitative analysis. Note, that failure of radio-communication does not directly influence safety. This is, because if the communication fails, then the train will not receive "crossing secured" signal and thus it will brake.

This allows an approximation of the probability of the hazard „collision" by using the formula of Sect. 3.

$$P(collision) = P(Err_{Brake}) + P(Err_{Distance}) \quad (6)$$
$$+ P(Err_{Signal}) + P(Err_{Open}) + P(Err_{Driver})$$

This yields an value of about $1.5 * 10^{-6}$. This means in one of 600000 crossings a collision may occur. To get this number a lot of assumptions about the environment have been made (e.g.

allowed speed on crossing 20m/s, safety margin 750m, only in 10% of all cases are cars waiting at the crossing,…).

4.2 Parameterized probalilities

Some of the probability assumptions depend inherently on the choice of free parameters. In this example system the two obvious free parameters are allowed speed for the train and the safety margin. Both values can be chosen freely. However it is clear, that they influence component failure probabilities. For example the probability of the failure mode "braking distance to short" (P(Err$_{Distance}$)) depends on both the safety margin and the allowed speed. Instead of using a fixed value we will introduce a parameterized probability for the failure mode. This can be easily compute by using the standard formula to calculate braking distance ($x_{brake}(v) = v^2/2a$). This allows calculating the braking distance, if the speed measurement v is incorrect by a deviation of Δv. Err$_{Distance}$ means then, that the braking distance of $v + \Delta v$ is greater than the calculate breaking distance plus the safety margin ($x_{brake}(v) + x_{safe}$). The probability of this event can be expressed as follows:

$$P\left(Err_{Distance}\right)(x_{safe}, v_{allowed}) = \tag{7}$$
$$P\left(\Delta v < \sqrt{v_{allowed}^2 + 2 A_{Brake} x_{safe}} - v_{allowed}\right)$$

Figure. 4 shows the parameterized probability of failure mode "braking distance to short" P(Err$_{Distance}$)(safety_margin, allowed_speed).
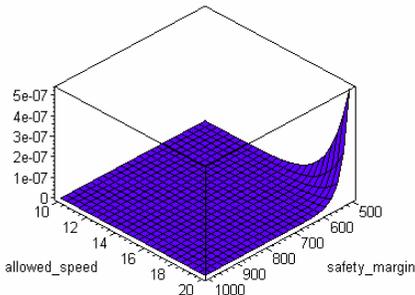

Fig. 4. P(Err$_{Distance}$)

The probability of this event decreases with large safety margins and slow allowed speeds. This corresponds to the intuitive understanding. The slower the speed and the larger the safety margin the less likely will exceedance of the braking distance be. An antagonistic failure mode is Err$_{Driver}$. This failure becomes more probable if the barriers are closed for a very long time. If we assume the average driver to wait 5 minutes before he passes the crossing even if the lights are red. We can also parameterize this probability to allowed speed and safety margin (see Fig: 5).
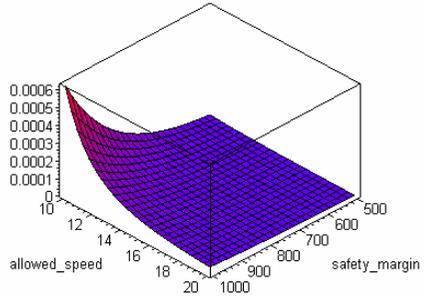

Fig. 5. P(Err$_{Driver}$)

This failure mode has antagonistic growth behaviour. Larger safety margin and lower allowed speeds result in additional waiting time for car drivers. This results in increased risk of drivers neglecting traffic rules and passing the crossing. It is intuitively clear, that an optimum configuration lies somewhere in between. This optimum can be calculated using optimization techniques. In the following figure the parameterized probability is shown around its minimum.
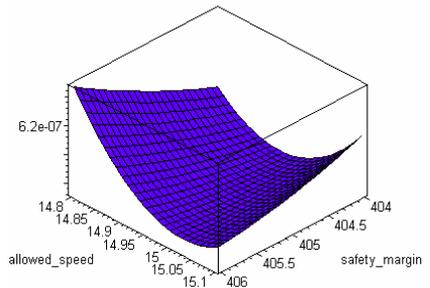

Fig. 6. P(Collision)

Closer examination yields an optimum at safety_margin = 406m and allowed_speed = 15m/s. These values result in a probability for collision of 6.1*10^{-7}. So we gained almost 100% without additional costs.

4.3 Safety Optimization

However, this optimum might not be the best solution. If we take an economical point of view, then one might argue, that allowed speed of 15m/s is by far to slow. The additional travel time for trains will have a serious impact on economic efficiency of the route.

So a compromise must be found. Assume that the average travel speed of the train on this route is 25m/s. Then we could calculate the time delay the railroad crossing causes by:

$$delay = \frac{x_{safe} + x_{brake}\left(v_{allowed\_speed}\right)}{v_{average}} - \frac{x_{safe} + x_{brake}\left(v_{allowed\_speed}\right)}{v_{allowed\_speed}} \quad (8)$$

Also assume that economics tell us that one collision costs us 100.000Euro and each second of time delay costs 0.1 Cent. Then we can use these values to compile a cost function and use formula of Sect. 3 to calculate new optimum, which respects these economical arguments.
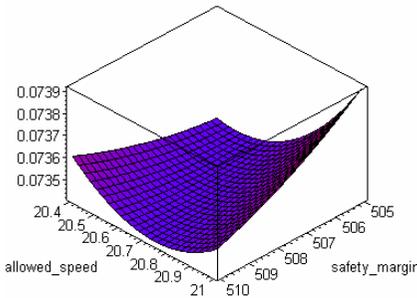

Fig. 7. Cost function

Figure 7 shows the cost function. The minimum lies at 508m safety margin and 20.6m/s. The probability of collision evaluates to $6.7*10^{-7}$. Note, that this choice of parameters not only allows faster travel than the initial guess, but is also safer. The reason for this is, that for optimum safety the choices of allowed speed and safety margin must match. This was not the case in the starting parameters, the safety margin was chosen to large for the allowed speed.

5. CONCLUSION

Our experience is that it is important to combine different techniques for safety analysis. This is because different methods not only examine different aspects of the system, but also give contrary views [4].

Safety optimization is one such technique. Safety optimization is an extension of fault tree analysis. It extends the quantitative aspects of FTA. Together with formal FTA [7][6] which extends the qualitative aspects and allows to prove that the cause-consequence relationship between primary failures and hazards is correct, this analysis is of very high significance.

The idea of safety optimization is as simple as promising: do a fault tree analysis of the systems hazards, use statistical distribution for failure probabilities, estimate the costs of each hazard with a cost function and do mathematical optimization. The result will be an optimal configuration of the system with respect to the examined hazards.

Safety optimization is not restricted to formal fault tree analysis. The concepts can easily transfered to other metrics of risk; for e.g. MEM or GAMAB (see [14, 15]) may be used as well. All that changes, is how probability distributions are combined into a cost function.

We illustrated the benefits of the new method with a real world case-study: the radio-based level crossing. The starting point for this type of analysis are minimal cut sets, which may be determined for example by fault tree analysis. The use of statistical distributions and parameterized probability required only a small extra effort. This helped firstly in quickly comparing different choices of free parameters. Secondly it allowed finding optimum configurations of free parameters with respect to safety.

While traditional safety analysis does not assess the problem of usability and trustworthiness, such issue may be considered with parameterized probabilities as well. A cost function allowed bringing in economical aspects and analyzing them.

In conclusion, we find that examining only hazards and estimating their probabilities is not enough. It is rather important to examine all hazards of a system in parallel with its intention, economical background and the planned working environment. A combined approach of traditional safety analysis, formal methods and mathematics can accomplish this. Such an integrated approach is being developed within the ForMoSA research project [3][6].

REFERENCES

[1] Christel Baier, Edmund M. Clarke, Vassili Hartonas-Garmhausen, Marta~Z. Kwiat-

kowska, and Mark Ryan: Symbolic model checking for probabilistic processes. In "Automata, Languages and Programming", pages 430--440, 1997.

[2] J. Fragole J.,Minarik II J. and Railsback Dr. W. Vesley, Dr. Joanne~Dugan: Fault Tree Handbook with Aerospace Applications. NASA Office of Safety and Mission Assurance, NASA Headquarters, Washington DC 20546, August 2002.

[3] FORMOSA - formal models and safety analysis, 2001. http://www.informatik.uni-augsburg.de/swt/formosa/.

[4] E.G. van den Blieck andJ.L. Rouvroye: Comparing safety analysis techniques. In "Reliability Engineering & System Safety", 2002.

[5] N. Leveson: Safeware: System Safety and Computers. Addison Wesley, 1995.

[6] F.Ortmeier and A.Thums: Formale Methoden und Sicherheitsanalyse. Technical Report 15, Universität Augsburg, 2002.

[7] G. Schellhorn, A. Thums, and W. Reif. Formal fault tree semantics. In "Proceedings of The Sixth World Conference on Integrated Design & Process Technology", Pasadena, CA, 2002.

[8] N. Storey: Safety-Critical Computer Systems. Addison-Wesley, 1996.

[9] Nancy G. Leveson: A new approach to system safety engineering. Aeronautics and Astronautics Massachusetts Institute of Technology.

[10] W. E. Vesely, F. F. Goldberg, N. H. Roberts, and D. F. Haasl: Fault Tree Handbook. Washington, D.C., 1981. NUREG-0492.

[11] A. H. G. Rinnooy Kan and G. L. Nemhauser, editor: Optimization, volume Vol 1. Elsevier Science Publishers B.V, 1989.

[12] David G. Luenberger: Linear and nonlinear programming. Addison-Wesley Publishing Company, 1989.

[13] J. Klose and A. Thums: The STATEMATE Reference Model of the Reference Case Study "Verkehrsleittechnik", Universität Augsburg, http://www.informatik.uni-augsburg.de/swt/fmg/papers/, number = 2002-01[13] J. Klose, A. Thums: The STATEMATE Reference Model of the Reference Case Study "Verkehrsleittechnik", Universität Augsburg, http://www.informatik.uni-augsburg.de/swt/fmg/papers/, number = 2002-01

[14] J. Braband: Risikoakzeptanzkriterien und -Bewertungsmethoden - Ein systematischer Vergleich, *Signal+Draht*, 4, 2004, 6-9J.

[15] Schnieder, E.; Slovak, R. and Wegele, S.: Neue und Herkömmliche Maße zur quantifizierung des Risikos im Eisenbahnverkehr, Erschienen in: Tagungsband der Zel 2004 S. 160-171, 2004. 11th International Symposium Zel 2004