Seeking Rare Events in a Simulated System using Risk Distances

Volker Gollücke
Institute for Information Technology
OFFIS e.V.
Oldenburg, Germany
email: volker.golluecke@offis.de

Axel Hahn
Department of Computing Science
Carl von Ossietzky Universität Oldenburg
Oldenburg, Germany
email: hahn@wi-ol.de

Abstract— **The identification and avoidance of potential risks is an important task in any system design process. The probability of underestimating a risk depends strongly on the complexity of the system under consideration. It is helpful to know the ways and means, which help reviewing the suspected risks. Current safety regulations, for example in the domain of offshore operations, require a description of all the risks involved. In order to create this description, a system including a behavioral specification modeled by a system expert is analyzed. Within this system, risks have to be searched and reasons for their occurrence have to be analyzed. The approach presented in this work describes a method to calculate the distance to a risk using a simulative analysis. The approach uses concepts from the field of Rare Event Simulations, which are used to accelerate the simulative reach of a risky situation. During the simulation runs data is collected about the way, which was taken to reach the avoidable situation. These data can be further used in a manual risk analysis and for matching rules and processes in terms of safety. The presented work describes how distance functions are created and used to assess system states in terms of their proximity to risky situations. In addition, the question of how the results of the distance functions can be used to guide co-simulations to the examined risky situations is answered.**

Keywords-System Analysis; Risk Analysis; Simulation, Distance Metrics.

I. INTRODUCTION

Many products and technologies for risk assessment are only suitable for specific application areas. An important need for action can be seen in the time consuming process of describing functions to assess a risk or measure the proximity to a risk situation. It was noticeable that in the examined programs [1]–[5], either only certain predetermined risks could be assessed or general functions could be specified. In addition, the consideration of distributed simulators (especially external simulators) was not the focus of any software under consideration. A structured guide with regard to the creation of risk-distance functions was not available in the investigated methods and tools. In addition, the consideration of the support of black-box simulators and the use of simulation run overlapping information were little or not considered at all.

When using classical methods, such as fault tree analysis, there are questions about the completeness of the information, e.g. whether all the causes for the identified errors have been identified. In this context, the use of simulation as a method of investigation is suitable, which allows experiments to be carried out under various configurations of the modeled system and to collect information on the occurrence of risks.

The approach presented in this work describes a methodology that can be used to define and calculate the distance to a risk in a simulative analysis using a so-called risk-distance function. The distance does not describe a spatial distance, but the proximity of a system state to an avoidable system state - the risk situation.

In addition, the use of concepts from the field of rare event simulation, which are used to accelerate the simulative reach of a risk situation, is implemented. The approach supports a part of the Importance Splitting technique [6]–[8], which is used to guide black box simulations to risky situations. It is assumed that a simulated state closer to a danger reaches it faster than a more distant one. In this case, the risk distance function is used as an importance function, which is used to evaluate the simulation states in the importance splitting technique.

In order to comprehensively collect information about the course of the simulation run, simulation states achieved in the simulation runs can additionally be persisted in a database and enriched with meta information. These include the evaluation of the state using the risk distance function, how often the persistent state was reached, how often this state was explored and, in addition, the minimum risk distance, the states reached from this state from previous simulation runs. All this information can also be integrated into the control of the simulation with regard to a faster reaching of risky states In order to implement this approach, description methods are required for situations and similarity measures with which different situations can be described and compared with each other. In order to establish and apply these techniques, techniques from the domain of information retrieval are used, which can compare the two system states with regard to their similarity in an appropriate time [9]. This paper is structured as follows: in Section 1 an introduction to the topic is given, in Section 2 the methodology to create a risk distance function and the use of the risk distance functions in a co-simulation are presented, we move on with an evaluation experiment to show the applicability of risk distance functions in Section 3, and conclude in Section 4.

 ISBN: 978-1-61208-594-4

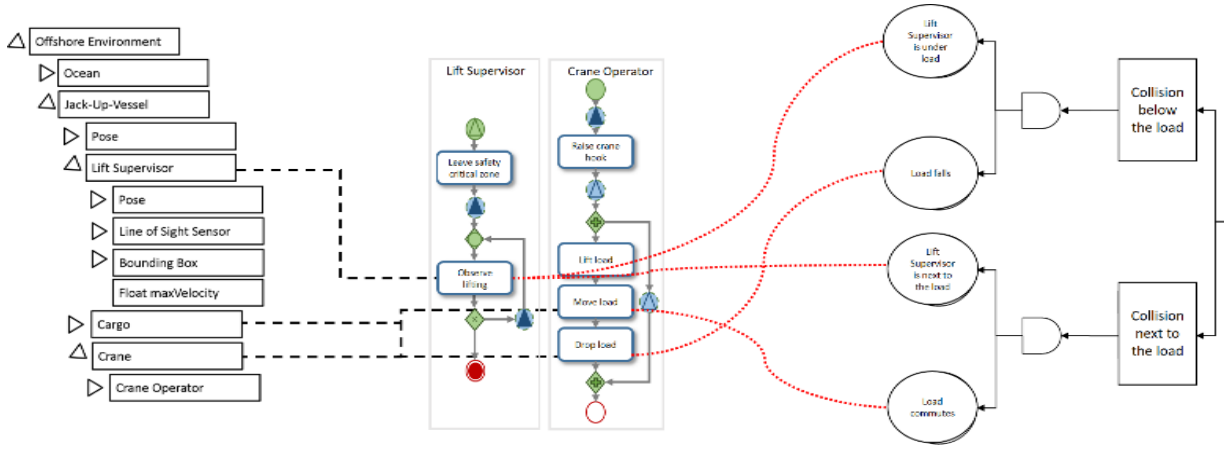

Figure 1. Representation of the linked elements of the system, behavioral and fault tree description.

II. METHODOLOGY

The methodology developed consists of three aspects and their predetermined order. The first aspect is the modeling of the system to be analyzed, which reveals a system, behavior and hazard description. The first aspect is achieved by the use of Droste's (cf. [10], [11]) and Pinkowski's (cf. [12], [13]) work and developed tools. The second aspect is the risk-distance description used to create the risk-distance functions. The last aspect is the simulative analysis, in which the previously ascertained descriptions enter in order to describe the simulation process.

A. *Modelling the system under test*

A model of the system to be analyzed is needed as starting point for the investigation. This consists of a system description including a behavior specification. A process model based on concepts of business process models and notation languages is used for the behavioral specification. The process describes actors, tasks and related interactions. This model type has been developed for use in the maritime domain, more specifically for offshore operations but can also be applied to other domains [10]. The graphical elements of the model describe task sequences and interactions. It is also possible to annotate these with associated hazards and their causes. A hazard describes a potential risk, such as the injury of a person or a machine damage. A cause describes a possible trigger for a hazard. When annotating the process model, a formal specification for hazards and causes is used [14]. Using a system model, the described physical objects and environmental conditions can be described. These are also useful to map actors from the process model to specific avatars (physical objects of the system) in the simulated environment. For example, a ship can be assigned to a ship resource of the simulated environment that has a position, maximum speed, and other attributes relevant to the simulation.

A well-known way to carry out a characterization of the risks involved are fault trees [15]. Fault trees provide an overview of the potential risks and the connection of their possible causes. In addition, fault trees split the risk in different events that presumably lead in their combination to the undesirable risk. For each annotated hazard to the behavior specification, a fault tree is created to logically structure the respective failures that may cause the hazard. Since each cause is related to an element of the process, the fault tree is also associated with this element. Conversely, each top event of a fault tree (root node) is associated with a hazard. For example, causes associated with a task item may be associated with multiple fault trees representing the hazards and associated causes graphically. In contrast to other methods for fault tree generation, which use UML diagrams as sources for generation (cf. [16], [17]), all necessary information is available in the presented process model. In order to avoid errors, the specific hazards and causes associated with the process have to be modeled by an experienced system expert [12]. Figure 1 shows an example of the three used models and their connection between each other. Basic Events are connected to specific tasks from the behavior description while tasks are linked to elements from the system model. The *Cargo* element from the system model is connected to the task *Observe lifting* from the behavioral specification and the basic event *Lift Supervisor is next to load.*

B. *Description of the risk distance*

This section presents the methodology used to determine the distance description to a risk. Figure 2 shows the construction of the risk-distance function with the new approach. Using the System Description, a Fault Tree is generated from which the structure of the risk distance function is derived. Hazards and faults, which are annotated at the behaviour specification by a system expert describe the sub-distance functions. The combination of the structure of the risk distance function and the sub-distance functions lead to the complete risk distance function. This process is described in more detail in the following sections.

 ISBN: 978-1-61208-594-4

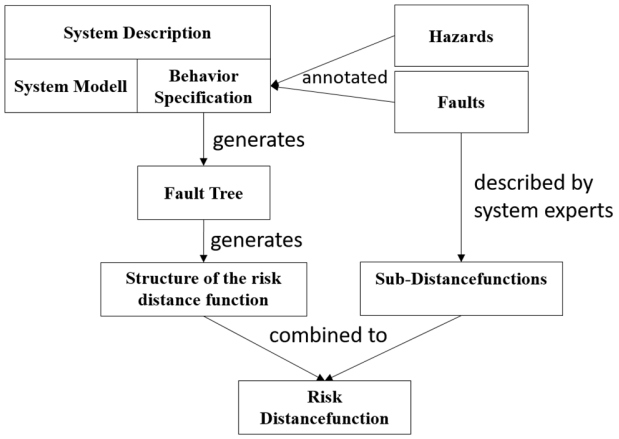

Figure 2. Overview of the creation of the risk distance function.

The process is based on the basis presented in the previous section for describing the system including the risks and causes expected by the system expert. The result of the distance description is a risk distance function, which represents a mathematical function that evaluates the proximity to an abstract risk situation to be investigated. The lower the distance to the risk, the lower the result of the risk distance function. The result of a risk distance function is always a value between 0 and 1, where 0 indicates that the risk situation has occurred, while 1 means that the currently checked situation has no relevant proximity to the risk situation under investigation.

1) Deriving the structure of the risk distance function

In order to obtain the structure of a risk-distance function, the logical links ("and" / "or" relationships) of the basic and intermediate Events (sub-elements of the top event, which are not leaf elements) of the fault tree are used. In Figure 3, the mapping between fault tree and the structure of the risk distance function is shown by an example. The top-level element sets the name for the created risk distance function (D_A). By "and" linked elements are calculated by multiplication. Since the calculation is not based on probabilities but on distances, the incoming, previously calculated distances are inverted, multiplied and inverted again to obtain a distance as a result. The equation, for elements linked by "and", is $D_A = 1 - ((1 - D_B) * (1 - D_C))$. Multiple elements associated with "or" are determined by the calculation of the minimum distance ($D_B = \min(D_D, D_E)$). The calculation of the leaf elements is done by the calculation of the products of the distances to the various properties of the causes, which are described in this work as sub-distance functions ($D_C = 1 - \prod_{i \in 1..n} (1 - d_{c_i})$).

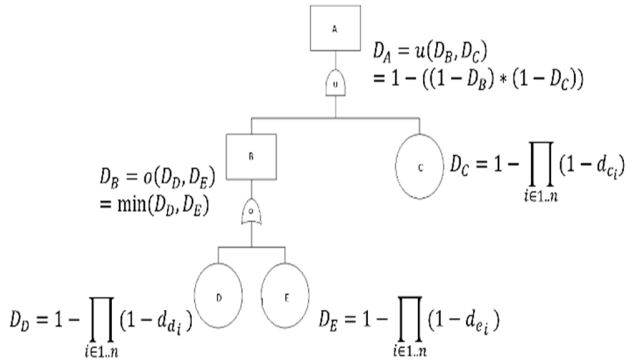

Figure 3. Relationship between fault tree and structure of the risk distance function.

The resulting structure of the risk distance function for the fault tree from Figure 3 can be seen in (1).
The sub distance functions, which are needed to complete the risk distance function, are derived in a subsequent step.

2) Deriving the sub distance function

Figure 4 shows the link between a fault tree basic event and the path to create the required sub distance functions.

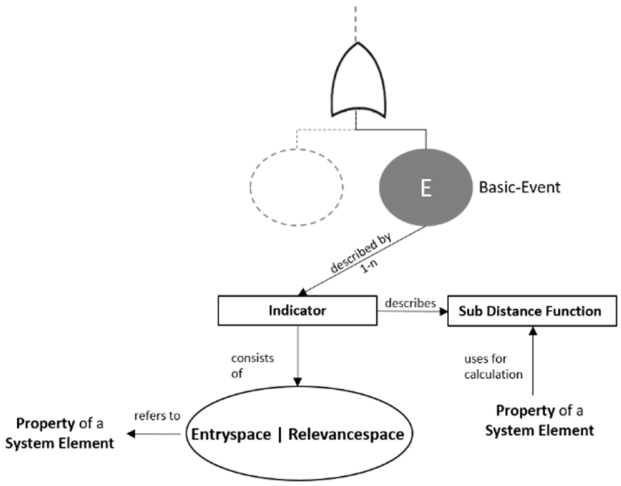

Figure 4. Description of the basic events for the risk distance function.

In the first step, the system experts determine so-called indicators, describing the space in which the cause described in the Basic Event is considered to have occurred.
For the basic event E of the fault tree in Figure 4 the equation 1-$\prod_{i \in 1..n}(1 - d_{E_i})$ was created to calculate the cause distance. The product is calculated because the different indicators are specified as joint events that must have occurred so that the cause specified in the basic event may be considered to have occurred. The combination of the indicators by mathematical operators is

therefore done in the same way as the elements in the fault tree linked by the logical "and".

d_{E_i} represents the sub distance function for the i-th indicator of the leaf node E. Each indicator refers to exactly one property of an element of the system with continuous values and is defined by an entry space (E_E) and relevance space (R_E). Starting from the considered indicator, the entry space and relevance space can be described by a lower $E_{E_{min}}/R_{E_{min}}$ and/or upper limit $E_{E_{max}}$ respective $R_{E_{max}}$.

The entry space describes the value range of the property that favors the cause described in the basic event.

$$D^A = 1 - \left(\left(1 - \min\left(1 - \prod_{i\in 1..n} 1 - d_{d_i}, 1 - \prod_{i\in 1..n} 1 - d_{e_i}\right)\right) * \left(1 - \left(1 - \prod_{i\in 1..n} 1 - d_{c_i}\right)\right)\right) \quad (1)$$

The relevance space describes the value range of the property, which is relevant for an approach to the risk under investigation, and from which value of the property there is no relevance for the risk under investigation.
A very simple example is the risk of a broken mobile device. Let us assume that the device is only prone to a too low (-20°C) or too high (40°C) temperature. Then, the entry space is defined by those two values. The relevance space is determined by two more temperature values, which define when the temperature of the device gets close to the risky temperatures.
Figure 5 shows an illustration for the entry and relevance spaces for this example.

Figure 5. Illustration for the use of the entry and relevance spaces.

The entry and relevance spaces are given by the following equations (2), (3), (4) and (5), where A, B, C and D are constants to be determined by the system expert, with which an indicator is described. Y can represent a predefined value of the considered property of the system (e.g., the centre point of the bounding box of a container).

$$E_{min}(Y) = Y - A \quad (2)$$

$$E_{max}(Y) = Y + B \quad (3)$$

$$R_{min}(Y) = Y - C \quad (4)$$

$$R_{max}(Y) = Y + D \quad (5)$$

When the indicator is described by the entry and relevance space, the sub distance function is created for the respective indicator (6).
This consists of the determined limits of the entry and relevance space as well as the considered property X of the system.

$$d_{D_i} = \begin{cases} 0, if\ X \geq E_{D_{min}}(Y) \wedge X \leq E_{D_{max}}(Y) \\ 1, if\ X \leq R_{D_{min}}(Y) \vee X \geq R_{D_{max}}(Y) \\ \min\left(\frac{\left(X - E_{D_{min}}(Y)\right)}{(R_{D_{min}}(Y) - E_{D_{min}}(Y))}, \frac{\left(X + E_{D_{max}}(Y)\right)}{(R_{D_{max}}(Y) - E_{D_{max}}(Y))}\right), else \end{cases} \quad (6)$$

At first, it is checked whether the property under consideration is already in the entrance space. If this is the case, the result of the sub distance function is *0*. The next check refers to the relevance space. If the property is outside the relevance space, *1* is output as the result of the sub distance function. If neither assumption is true, the result of the sub distance function is determined by the distance of the system element property to the minimum and maximum entry space. These distances are then normalized over the distance between minimum or maximum event space and relevance space. The normalized values of the sub distance functions, which all describe a partial distance irrespective of the properties used, allow a comparability. The minimum is then selected from the normalized distances and output as the result of the sub distance function. If there is only a minimum or maximum limit of the entry space, the distance to it is the result.

C. Simulative Analysis

This section describes how the simulative analysis is carried out using the described system and the determined risk distance functions. In this context, the co-simulation environment used is discussed and the use of the distance functions for controlling the simulation with regard to a reduction of simulation runs is explained. To be able to use the importance splitting technique, the used simulators, must ensure a state control. This means that they must be able to store and load their current situation at the request of a co-simulation controller.

1) Integration of the risk distance functions

Figure 6 shows the integration of the values determined in the methodology, in the context of co-simulation. A system model instance is linked to the connected simulators and

receives the values updated by the simulators. Through the system model instance, the components "Risk Distance Calculation" and "Situation Comparison" necessary for the integration of a distance description are informed about the values of a new simulation state. In this context, the risk distance measurement uses the risk distance function determined in the methodology in order to calculate the risk distance to an abstract risk (i.e., "collision with cargo") using the values of the system model instance.

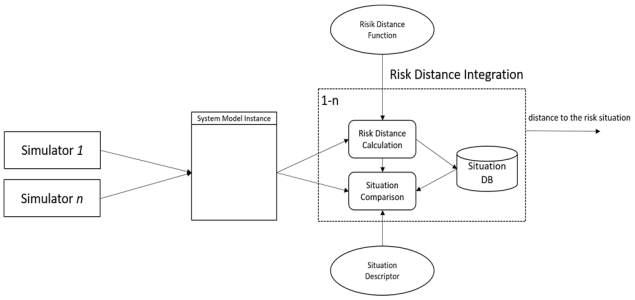

Figure 6 Use of the risk-distance evaluation in a co-simulation environment

The situation database is based on the information retrieval technique used to classify information. A descriptor is automatically defined by the underlying system model and its characteristic, the system model instance. It allows the request for the similarity to already reached situations. For this purpose, the comparative component uses a situation descriptor based on the elements of the system model and the data of the system model instance. The situation descriptor is a vector whose elements reflect the characteristics of the system elements. The order of the element properties in the vector is fixed, so that the comparison is not falsified. The size or the number of elements of the descriptor for a simulation scenario is defined by the structure of the system model. The comparison is made based on a distance measure. Static properties can be ignored for the calculation of the similarity. Properties with numeric values can be used directly while Boolean values are used as false (value 0) and true (value 1) and textual values in comparison over their equality or inequality. Within the developed framework, the use of the Euclidean distance (L2), the known universally applicable distance measure, was chosen.

Through the use of databases developed for the information retrieval technique (e.g., Apache Lucene), the similarity calculation can be performed within a reasonable time by the predetermined distance measure, despite invoking mathematical operations to calculate the distance.

If there is no similar situation in the database, the database is supplemented by this data set and enriched after a simulation run by the minimally achieved risk proximity. If a similar situation is persisted in the database, it is checked and updated how many simulation runs have already been carried out and how the maximum approach to the abstract risk situation was.

If not enough simulation runs explored to the same situation or the maximum approximation to the risk situation is promising, the results of the situation evaluation are output and can be used, e.g., to save the current situation or to abort the current simulation run.

2) Exploration to the risk situation

As described in the previous section, the inclusion of the risk distance measurement within the simulation can be used to react to the distance of the respective situation to the examined situation. An advantage that is derived from this is the description of a controlled co-simulation based on the calculated distances. As known from the previous section, the functionality of the risk distance calculation is achieved via a link to the system model instance. The current values stored in the system model instance are forwarded to the distance calculation, the calculation of the distance takes place and can be used for further processing in a simulation sequence logic. In this, all decisions regarding the simulation process are defined. It thus allows the distance calculations to be tested with respect to a threshold value and on this basis, the further course of the simulation can be determined.

In order to define the course of the simulation the control functions are used.

This means that the situation of the co-simulation, after the risk situation has been approached according to the calculation of the distance function, can be stored and further simulation runs can be started from this.

III. VALIDATION

In this section, the simulation guidance by risk distance functions is evaluated. For this purpose, a scenario was chosen with two vessels involved, which are in the opposite direction on an inland waterway. A collision between these two ships has to be observed, which, however, can only be achieved very rarely by the conditions set out in the following sections. In order to obtain a more frequent observation of a collision, the approaches and concepts presented in this work were used to control the simulation.

A. Structure and procedure of the experiment

In the experiment under consideration, two simulators were used, which control the two participating ships. Both ships have a high probability of reaching their destination and are only very unlikely to deviate from their direct course to their destination (Figure 7).

 ISBN: 978-1-61208-594-4

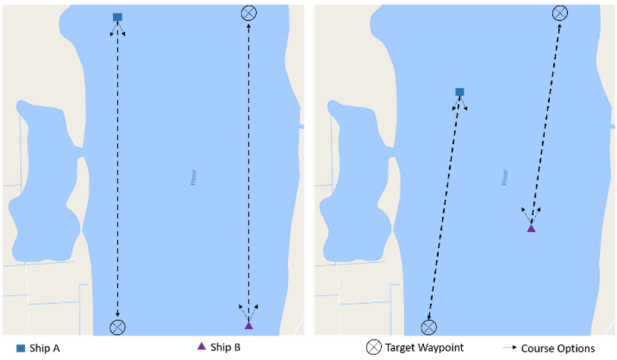

Figure 7. Evaluation scenario: The two ships involved try to reach their destination. With a very low probability, the ships depart to the left or right of their course.

As a termination condition for a simulation run, the arrival of the destination waypoints by both ships was determined. The event that was investigated in this evaluation was a collision between the two ships, which should be observed within this evaluation 100 times under the respective simulation settings. The simulation settings consisted of a naive and a guided simulation, in which there were three variants of how to determine the next splitting point (the next lower risk distance level). On the one hand, an adaptive method, which in each simulation step checks whether a lower risk distance evaluation has been achieved than before and, secondly, a method, which has applied the splitting points at predetermined intervals (0.01 and 0.05).

In all methods, except for the naive simulation, a new simulation run was started when a state with a new lower risk distance was reached in the case of the adaptive method or a new risk level in the case of the predefined intervals.

All simulation runs in which no collision occurred were counted as well as the real time measured up to the time of the occurrence of a collision. This operation was performed 100 times, with no parallel execution. The generated simple fault tree for the scenario consists of three basic events (close range, dangerous speed and a specific heading angle of the two ships) which have to occur together so that a high risk of a frontal ship collision exists (Figure 8).

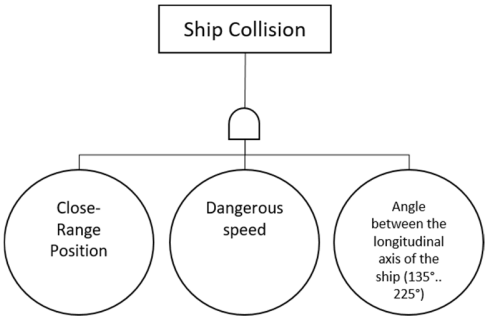

Figure 8. Simple Fault Tree for a frontal collision between two ships.

The following risk distance function was derived (s. equation (7)), which uses the proximity, speed and orientation of the two ships to determine the risk distance.

As a basis for the sub-distance functions values "Collisions and their Causes" [18] and "Managing Collision Avoidance at Sea: A Practical Guide" [19] where used.

$$D_K = 1 - \left(\left(1 - \begin{cases} 0, if\ distance = 0m \\ 1, if\ distance \geq 4980m \\ \dfrac{distance}{4980m}, else \end{cases} \right) \times \left(1 - \begin{cases} 0, if\ speed \geq 12{,}5kn \\ 1, if\ speed \leq 10kn \\ \dfrac{(speed - 12{,}5kn)}{(10kn - 12{,}5kn)}, else \end{cases} \right) \times \left(1 - \begin{cases} 0, if\ angle \geq 135° \wedge angle \leq 225° \\ 1, if\ angle \leq 100° \vee angle \geq 250° \\ \min\left(\dfrac{(angle - 135°)}{(100° - 135°)}, \dfrac{(angle + 225°)}{(250° - 225°)} \right), else \end{cases} \right) \right) \quad (7)$$

During the experiment, the two ships had a 99.9% probability to drive towards their target waypoint, while they changed their course to the left or right with a probability of 0.1%. The decision to change the course was recalculated by the two ships in each simulation step.

B. *Evaluation of the experiments*

The results of the experiment showed that the naive simulation took about 152 minutes to observe 100 collisions, while the simulation runs, which were guided by means of a risk distance function, produced 100 collisions after approx. 3 seconds (distance 0.05) ,95 milliseconds (distance 0.01) and 92 milliseconds (adaptive). While in the naive simulation 44,076,619 simulation runs were required before 100 collisions could be observed, the number of simulation runs for the distance of the splitting points from 0.05 was 5.374.930 runs, for the distance of the splitting points from 0.01 59.551 runs and for the adjusted distance 53.150 runs.

In the case of this experiment, the method with the adaptive distance was the best, with regard to the sum of the simulation runs as well as the measured real time, until 100 independent collisions were observed.

IV. CONCLUSION

The approach developed in this work, consisting of a methodology for describing risk assessment and the use of this in co-simulations, differs in some aspects from current programs and concepts in science and technology. In particular, the approach to assess a risk is revisited in using a system-, process and fault tree model, which gives the possibility for non-information technicians to create risk distance functions. Additionally, it is described how the specific risk distance functions are used to reach risky situations faster by using approaches from the rare event simulation and the information retrieval field also when using black-box simulators. A situation database helps to store

 ISBN: 978-1-61208-594-4

simulation run spanning information about the achieved simulation states and to use them for the further control of the co-simulation.

ACKNOWLEDGMENT

The work presented in this paper is supported by the Center for Critical System Engineering for Sociotechnical Systems at the University of Oldenburg, OFFIS and DLR. The center is funded by the Federal State of Lower Saxony, Germany under grant numbers VWZN3237 and VWZN3270.

REFERENCES

[1] EMSO, "EMSO Environment for Modelling, Simulation, and Optimization," *EMSO Environment for Modelling, Simulation, and Optimization*, 2004. [Online]. Available: http://www.vrtech.com.br/rps/emso.html. [Accessed: 14-Jun-2015].

[2] Plant Simulation, "Plant Simulation," *Simulation mit Plant Simulation*, 2015. [Online]. Available: http://www.plant-simulation.de/?gclid=Cj0KEQjwzPSrBRC_oOXfxPWP6t0BEiQARqav2OQQo8_Xb5B-iBS-zLZQ0TyqNfXTt7NH0KGFPEggQI8aAm-58P8HAQ. [Accessed: 14-Jun-2015].

[3] K. W. Ross and J. Wang, "Implementation of Monte Carlo integration for the analysis of product-form queueing networks," *Perform. Eval.*, vol. 29, no. 4, pp. 273–292, 1997.

[4] S. Shyam and V. Bertacco, "Distance-guided hybrid verification with GUIDO," in *Proceedings of the conference on Design, automation and test in Europe: Proceedings*, 2006, pp. 1211–1216.

[5] The OptQuest Engine, "The OptQuest Engine," *OptQuest | OptTek Systems, Inc.*, 2011. [Online]. Available: http://www.opttek.com/OptQuest. [Accessed: 14-Jun-2015].

[6] C. Jegourel, A. Legay, and S. Sedwards, "Importance splitting for statistical model checking rare properties," in *Computer Aided Verification*, 2013, pp. 576–591.

[7] S. Juneja and P. Shahabuddin, "Rare-event simulation techniques: an introduction and recent advances," *Handb. Oper. Res. Manag. Sci.*, vol. 13, pp. 291–350, 2006.

[8] J. Morio, R. Pastel, and F. Le Gland, "An overview of importance splitting for rare event simulation," *Eur. J. Phys.*, vol. 31, no. 5, pp. 1295–1303, Sep. 2010.

[9] S. Büttcher, C. Clarke, and G. V. Cormack, *Information Retrieval: Implementing and Evaluating Search Engines*. The MIT Press, 2010.

[10] R. Droste, "Modellbasierte Planung und Analyse von Offshore-Operationen (unveröffentlicht)," Dissertation, Carl-von-Ossietzky Universität Oldenburg, Oldenburg, 2016.

[11] C. Lässche, J. Pinkowski, S. Gerwinn, R. Droste, and A. Hahn, "Model-Based Risk Assessment of Offshore Operations," in *ASME 2014 33rd International Conference on Ocean, Offshore and Arctic Engineering*, 2014, p. V01BT01A010–V01BT01A010.

[12] J. Pinkowski, "Prozessgetriebene Risikoanalyse zur Bewertung maritimer Operationen," Dissertation, Carl-von-Ossietzky Universität Oldenburg, Oldenburg, 2015.

[13] V. Gollücke, J. Pinkowski, C. Lässche, S. Gerwinn, and A. Hahn, "Simulation-based Completeness Analysis and Adaption of Fault Trees," in *SIMUL 2014, The Sixth International Conference on Advances in System Simulation*, Nice, France, 2014, p. 228 to 235.

[14] C. Lässche, E. Böde, and T. Peikenkamp, "A method for guided hazard identification and risk mitigation for offshore operations," in *Computer Safety, Reliability, and Security*, Springer, 2012, pp. 37–48.

[15] W. Vesely, J. Dugan, J. Fragola, Minarick, and J. Railsback, "Fault Tree Handbook with Aerospace Applications," National Aeronautics and Space Administration, Washington, DC, Handbook, 2002.

[16] C. Lauer, R. German, and J. Pollmer, "Fault tree synthesis from UML models for reliability analysis at early design stages," *ACM SIGSOFT Softw. Eng. Notes*, vol. 36, no. 1, pp. 1–8, 2011.

[17] G. J. Pai and J. B. Dugan, "Automatic synthesis of dynamic fault trees from UML system models," in *Software Reliability Engineering, 2002. ISSRE 2003. Proceedings. 13th International Symposium on*, 2002, pp. 243–254.

[18] R. A. Cahill and N. I. (Great Britain), *Collisions and Their Causes*. Nautical Institute, 2002.

[19] G. W. U. Lee and J. Parker, *Managing Collision Avoidance at Sea: A Practical Guide*. Nautical Institute, 2007.

 ISBN: 978-1-61208-594-4