Failure Behaviour Analysis of Automated Pond Oxygen Management System: a Qualitative Study

Sohag Kabir[1], Youcef Gheraibia[2], Martin Walker[1], and Tanzima Azad[3]

[1] Department of Computer Science, University of Hull, Hull, UK
[2] Department of Computer Science and Mathematics, Mohamed-Cherif Messaadia University, Souk Ahras, Algeria
[3] Department of Computer Science and Engineering, Bangladesh University of Enegineerinag & Technology, Dhaka, Bangladesh
s.kabir@hull.ac.uk, youcef.gheraibia@univ-soukahras.dz, Martin.Walker@hull.ac.uk, azad.tanzima@dbbl.com.bd

Abstract. Dependability of safety-critical systems are a prime concern of the modern society due to the growing dependence on those systems. Safety of systems is a fundamental requirement for system reliability. Fault tree analysis (FTA) is powerful, well-established, and widely used tool for evaluating system safety. Fault trees provide a graphical and logical framework for analysing the dependability of systems. They have been successfully employed in studying the failure behaviour of a variety of real-world systems. In aquaculture, the volume of oxygen contained in water is considered as a critical parameter for the health and well-being of fishes. If oxygen levels drop below 4 mg/L then fishes may stop feeding, stressed and begin to die. Automated Pond Oxygen Management System (APOMS) is a critical component in aquaculture to maintain the proper oxygen level in water. In summer months, when oxygen level starts dropping due to increased temperature, then the APOMS can balance the oxygen level in water by generating oxygen artificially. Therefore, the failure of this system may result in a disastrous outcome. In this paper, we have used fault tree analysis to determine the failure behaviour of an automated fault tolerant pond oxygen management system.

1 Introduction

The increasing complexity of systems has considerable significance for safety analysis as a part of dependability assessment. Dependability assessment of safety-critical systems should begin early in the design phase so that potential problems can be identified and rectified as soon as possible to avoid expensive changes later in the system lifecycle. If problems are not rectified, system failures can result in unacceptable costs in terms of loss of life, environmental damage, and loss of resources [4]. Most of today's systems have multiple modes of operation and many offer some level of robustness built into the design. Many systems have achieved the capability to operate in a degraded mode without failing completely by adopting some sort of fault-tolerant strategy like using redundant components or using parallel architectures. However, such complexity also poses new challenges for systems analysts, who need to understand how such systems behave and estimate how reliable and safe they really are. System analysis can help reliability engineers understand how systems work and how they can fail, thus allowing them to determine necessary actions to prevent that failure [21].

Fault tree analysis (FTA) is well-established and very popular technique for evaluating the safety and reliability of complex safety critical systems [13]. Fault trees utilise graphical representations based on Boolean logic to show logical connections between different faults and their causes. They are a deductive analysis method, which means analysis starts with a system failure known as the 'top event' and works backwards to determine its root causes [17]. From a fault tree, it is possible to understand how combinations of failures of different components or certain environmental circumstances can lead to system failure. Qualitative analysis of fault tree is performed using Boolean logic by reducing it to minimal cut sets (MCSs) which are the smallest combinations of failure of system components that are necessary and sufficient to cause the system failure. Quantitative analysis of a fault tree, which follows qualitative analysis, can help to estimate the probability of the top event occurring from the given failure rates of basic failure modes of the system [18].

Commercial aquaculture is growing worldwide and according to the Food and Agriculture Organization (FAO), in 2012 aquaculture supplied the world with about 154 million tonnes of fish, of which 131 tonnes were consumed as human food [10]. Ponds are the most common production systems used globally for fish production. Managing water quality in the pond is one of the most challenging tasks due to their exposure to factors of climate and topography [9]. Growth rates and survival of the organisms within the pond environment is largely depending on the amount of available oxygen, dissolved oxygen levels and temperatures of pond water [5, 6]. Specially, in the summer time, when the water temperature goes up, then dissolved oxygen level in the pond water goes down. Due to this reduced oxygen level, fish may stop feeding and begin to die. Fish farmers could prevent oxygen depletion through monitoring and controlling oxygen level by using dissolved oxygen management systems. Therefore the success of oxygen monitoring and controlling process is primarily depends on the successful operation of the oxygen management systems. The failure of this system could lead to a catastrophic outcome. Considering the importance of such a system in fish farming, in this paper, we have performed the failure behaviour analysis of an automated pond oxygen management system.

The paper is organised as follows: Section 2 presents the fundamentals of fault tree analysis. A description of the automated oxygen management system is provided in Section 3. In Section 4, the result of the analysis is presented. Finally, concluding remarks is presented in Section 5.

2 Fault Tree Analysis

FTA is the most widely used deductive, top down analysis method which allows analysts to perform both qualitative and quantitative analysis [17, 19]. Fault trees (FTs) utilise graphical model to show how low-level component failure combines together to cause a top level system failure. FTA starts with an undesired event, such as system failure, known as the 'top event', and then works backwards to deduce the causes of the top event in terms of logical combination of basic events [21]. Symbols used in classical FTs to represent different events are shown in Fig.1.

A basic event is an initiating or basic fault that does not require any further development or expansion. It is represented as a leaf node in the fault tree. An intermediate event is a fault that is caused by the logical combinations of other events occurring further down the tree. As intermediate events are caused by other events therefore they are almost always a type of logical gate. An undeveloped event is an event whose contributions are not considered in the analysis, either because it is considered as unnecessary, or because insufficient information is available. Normal event does not represent any fault and it is part of the nominal behaviour of the system. A conditioning event does not necessarily represent a fault, it serves as a special condition or constraint for certain types of gates. Symbols used in classical FTs to represent different logic gates are shown in Fig.2. The outcome of OR gate is true if any of the input events are true, whereas an AND gate is true if all of its input events are true. The XOR gate is true if one and only one of its input events is true. The outcome of PAND gate is true if all of its events are true and they occur in a specific sequence. The INHIBIT gate produce an output when the input event is true in the presence of a conditioning event. Example of a fault tree is shown in Fig.3.

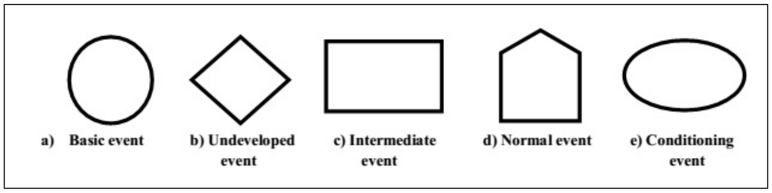

Fig. 1. Fault Tree Event Symbols

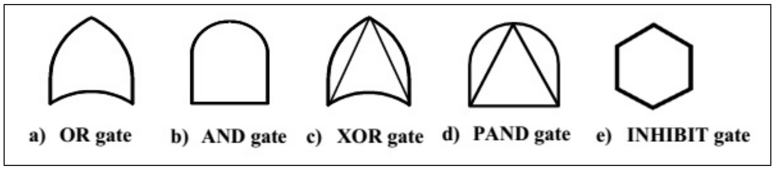

Fig. 2. Fault Tree Logic Gate Symbols

The roles of FTA in decision making are [17]:

- To understand the logic leading to the top event.
- To prioritise the contributors leading to the top event.
- As a proactive tool to prevent the top event.
- To monitor the performance of the system.

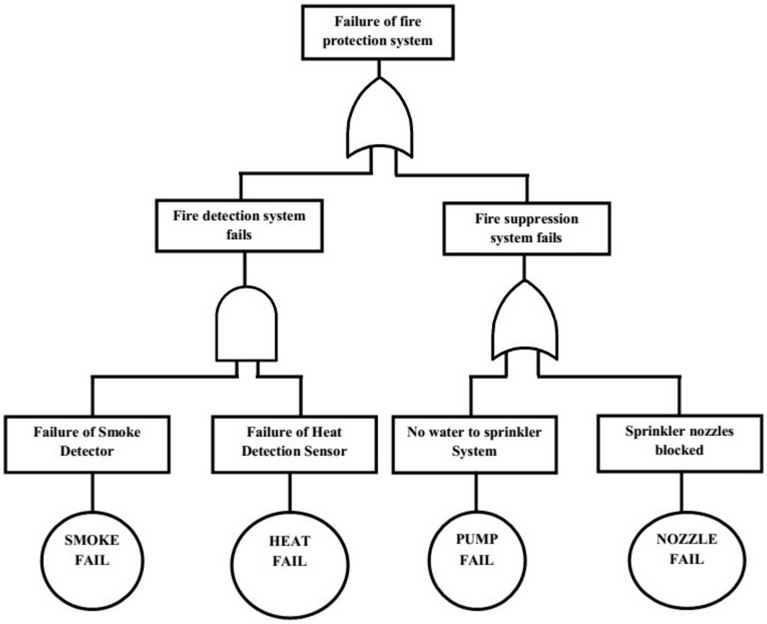
Failure of fire protection system
Fire detection system fails
Fire suppression system fails
Failure of Smoke Detector
Failure of Heat Detection Sensor
No water to sprinkler System
Sprinkler nozzles blocked
SMOKE FAIL
HEAT FAIL
PUMP FAIL
NOZZLE FAIL

Fig. 3. Example of a Fault Tree [3]

- To minimise and optimise resources.
- To assist the designing of the system.
- As a diagnostic tool to identify and correct causes of the top event

Steps required for successful FTA and the interrelationship between the steps are shown in Fig.4. The first step is to define the objective of the FTA, i.e., defining the objectives in terms of a failure of the system to be analysed. The top event of the FT is defined in step 2 just after defining the objective of FTA. After that the scope of the FTA is defined by indicating which of the component failures and contributions will be considered and which will not be considered. The level of detail to which the failure causes for the top event will be developed is defined is step 4 as the resolution of FTA. In step 5, ground rules for the FTA are defined by declaring the procedure and nomenclature by which events and gates are named. The ground rules are very important in creating an understandable FT. The actual construction of FT is performed in step 6 and the evaluation is done in Step 7. Both qualitative and quantitative evaluations of system failure behaviour could be performed using FTA. The qualitative evaluation provides information about the minimal cut sets which are necessary and sufficient to cause the top event. The quantitative evaluation usually provides the probability of the top event and importance measure of events based on their contribution to the top event. Quantitative analysis is out of scope of this paper. Finally, at the last step the results are interpreted and presented.

FTA has been widely used in different areas to perform both qualitative and quantitative dependability analysis of the systems. Systems that have been analysed using

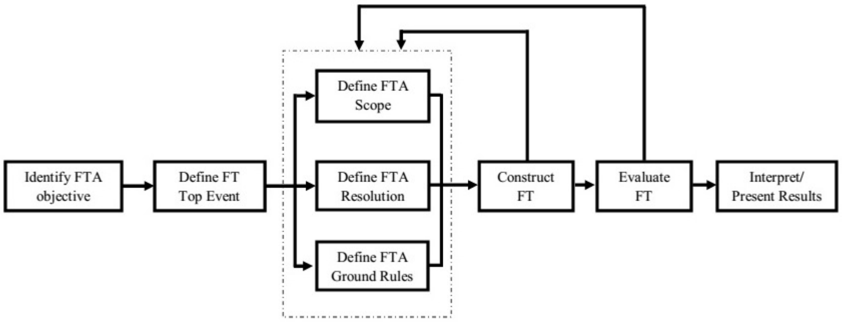

Fig. 4. Fault Tree Analysis Steps [17]

FTA include but not limited to propulsion systems on spacecraft, braking systems in cars, nuclear power plants, and so on. Different tools like HiP-HOPS [14], Reliability Workbench [12] are available to perform fault tree analysis. FTA has been used to analyse the reliability of the power system in [7, 20], power supply system to railways in [8], and cogeneration power plant in textile mill in [15]. Recently, fault tree analysis has been applied on clinical workflows to ensure the safety of these high-risk workflows [2] and it is also used to prevent wrong-side surgery to patients [1]. Authors in [16] uses FTA to analyse the potential failure scenarios in the assembly process of the printed circuit board industries. In the biological domain, FTA is used to analyse the failure of biomedical operation such as analysis for biological invasions [11].

3 Case Study: Automatic Pond Oxygen Management System

An automated, fault-tolerant pond oxygen management system is shown in Fig.5. The role of this system is to automatically generate oxygen whenever the oxygen level of pond water starts falling. Broadly, this system has a power generation unit and two subsystems (subsystem 1 and 2). The power generation unit consists of an electricity generator and a main electricity supply line. In normal condition, the system is powered by the electricity from the main line. If the main line fails (disconnected or load-shedding) then the generator is enabled to provide electricity to the system. Subsystem 1 and 2 are identical because subsystem 2 is introduced to make the system fault-tolerant. Subsystem 1 has an oxygen level sensing block OSB_1. OSB_1 block has 2 oxygen dissolver sensor $OS_i \, (i = 1, 2)$. Each oxygen dissolver sensor OS_i consists of a battery unit and a sensor. For simplicity, the battery unit and the sensor of the oxygen dissolver sensors are not shown in the figure. The oxygen level sensing block OSB_1 senses the oxygen level of the pond by using oxygen dissolver sensor OS_1 and sends the reading to the decision making block DM_1. If OS_1 fails, then OS_2 performs the task of sensing. Therefore, OSB_1 can still operate if one of its sensors fails. Characteristics and functionality of OSB_2 are similar to that of OSB_1. Decision making block can decide whether to turn on an oxygen generator after getting the oxygen level reading from the OSB_1. If required DM_1 can turn on an oxygen generator from the oxygen generator block OGB_1. Failure of DM_1 would cause the subsystem 1 to fail because there is no spare component to take over its tasks. OGB_1 block has a dedicated (private) oxygen generator (OG_1) and a shared oxygen generator (OG_3). In normal condition, OG_1 is used to generate oxygen. If OG_1fails then OGB_1 can use OG_3 to generate oxygen. Similarly, if OG_2 of OGB_2 fails, then OGB_2 can use OG_3 to generate oxygen.

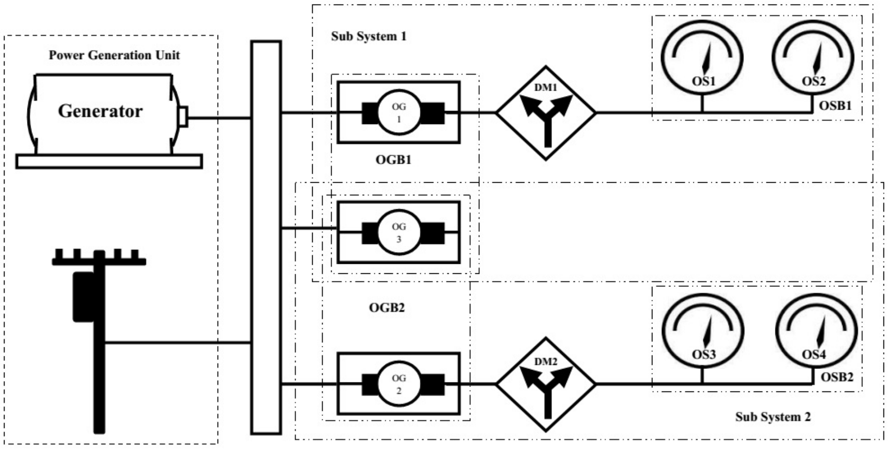

Fig. 5. Automated Fault Tolerant Pond Oxygen Management System

4 Fault Tree Generation and Evaluation

Before constructing the FT for the system in Fig.5, we have defined the objective of the FTA for this system. The objective of the analysis is to find the possible causes that could prohibit the system from generating oxygen when required. To do this we have identified the inability of the system to generate oxygen as the top event (system failure). In this paper, events are assumed to be statistically independent, and to be non-repairable—all common assumptions in FTA. At the same time, we also considered that a component could be either in failed or fully functional state, no other state is possible. The fault tree creation process starts with the top event, represented as the disjunction of the causes of the top event. In each iteration, the process propagates downwards to find the causes of system failure from the system level to the subsystem level, and it continues till the propagation reaches to the basic component level. In other words, we can say that in each iteration the fault tree expands and grows downwards. For example, at the higher level, the system failure can be caused by the failure of the power supply block (PS) or the failure of both the subsystems (SSF). The power supply block itself will be failed if and only if the main supply line and the electricity generator fails. The fault tree of the system of Fig.5 is shown in Fig.6. The ID and names of the basic events, intermediate events, and top event are shown in Table 1 and Table 2 respectively.

Table 1. ID and Name of Basic events

Event ID	Event Name
MS	Failure of main electricity line
EG	Failure of electricity Generator
DM1	Failure of Decision Maker in subsystem 1
DM2	Failure of Decision Maker in subsystem 2
OG1	Failure of Oxygen Generator 1 in OGB_1
OG2	Failure of Oxygen Generator 2 in OGB_2
OG3	Failure of Oxygen Generator 3 (shared)
S1	Failure of sensor in OS_1
S2	Failure of sensor in OS_2
S3	Failure of sensor in OS_3
S4	Failure of sensor in OS_4
B1	Failure of battery unit in OS_1
B2	Failure of battery unit in OS_2
B3	Failure of battery unit in OS_3
B4	Failure of battery unit in OS_4

Table 2. ID and Name of Intermediate and top events

Event ID	Event Name
TE	System Failure—no oxygen supply
SSF	Failure of Subsystem 1 and 2
PS	Failure of Power Supply Block
SS1	Failure of Subsystem 1
SS2	Failure of Subsystem 2
OSB1	Failure of OSB_1 block
OSB2	Failure of OSB_2 block
OGB1	Failure of OGB_1 block
OGB2	Failure of OGB_2 block
OS1	Failure of oxygen dissolver OS_1
OS2	Failure of oxygen dissolver OS_2
OS3	Failure of oxygen dissolver OS_3
OS4	Failure of oxygen dissolver OS_4

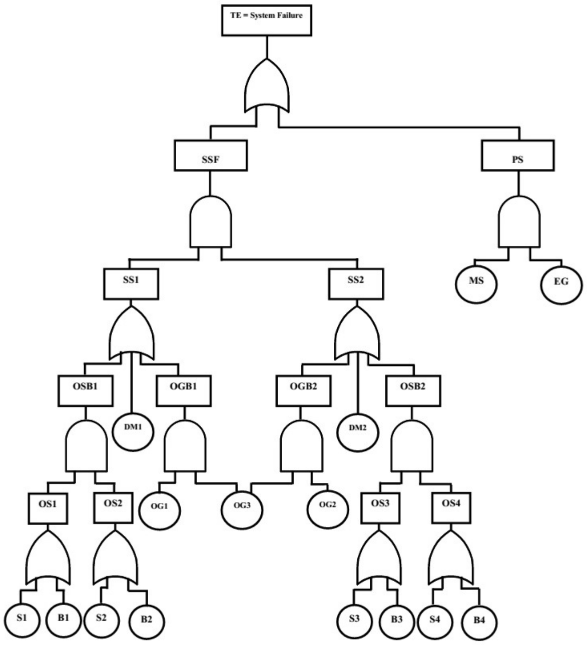

Fig. 6. Fault Tree of the System in Fig.5

The logical expression of the top event as the disjoint sum of minimal cut sets can be given by following expression:

$$\begin{aligned}System\ Failure = {} & MS \cdot EG + S_1S_2S_3S_4 + S_1S_2S_3B_4 + S_1S_2B_3S_4 + S_1S_2B_3B_4 \\ & + S_1S_2DM_2 + S_1S_2OG_2OG_3 + S_1B_2S_3S_4 + S_1B_2S_3B_4 + S_1B_2B_3S_4 \\ & + S_1B_2B_3B_4 + S_1B_2DM_2 + S_1B_2OG_2OG_3 + B_1S_2S_3S_4 + B_1S_2S_3B_4 \\ & + B_1S_2B_3S_4 + B_1S_2B_3B_4 + B_1S_2DM_2 + B_1S_2OG_2OG_3 + B_1B_2S_3S_4 \\ & + B_1B_2S_3B_4 + B_1B_2B_3S_4 + B_1B_2B_3B_4 + B_1B_2DM_2 + B_1B_2OG_2OG_3 \\ & + DM_1S_3S_4 + DM_1S_3B_4 + DM_1B_3S_4 + DM_1B_3B_4 + DM_1DM_2 \\ & + DM_1OG_2OG_3 + OG_1OG_3S_3S_4 + OG_1OG_3S_3B_4 + OG_1OG_3B_3S_4 \\ & + OG_1OG_3B_3B_4 + OG_1OG_3DM_2 + OG_1OG_2OG_3\end{aligned}$$

From the above expression, we can see that there are 37 minimal cut sets which can cause the system failure. For example, the first cut set $MS \cdot EG$ corresponds to the power generation unit, if both main electricity line and electricity generator fail, then it will cause the power generation unit to fail which will eventually cause the system failure. Because without electricity the system will not be able to operate even though all other components are fully functional. Similarly, the last cut set $OG_1 \cdot OG_2 \cdot OG_3$ represents the scenario when all the oxygen generators fail. In this case, even all other components of the system work properly to detect the oxygen level of the water still the system cannot produce oxygen to balance the oxygen level because none of the oxygen generator is operational to generate oxygen. All other cut sets can be interpreted in the similar way. The results produced by the FTA provide useful information about the root causes of the system failure, which help the analysts to understand the failure behaviour of the system. Now, if required, the analysts can decide to change the system model based on the failure behaviour of the system to satisfy their safety requirements. More information about the reliability of the system can be generated through quantitative analysis. At present, the quantitative analysis of the system is not performed due to the unavailability of sufficient numerical data for the components of the system.

5 Conclusion

Fault tree analysis is an effective method to evaluate the safety and reliability of safety-critical systems. It has been widely used in the dependability analysis for many complex systems involved in, nuclear reactor, aerospace industry, and chemical plants. This paper demonstrates the use of the FTA for performing failure behaviour analysis of automated pond oxygen management system. We have performed the qualitative analysis to determine the minimal cut sets that are necessary and sufficient to cause the system failure. Our analysis determines 37 different combinations of system component failures that can cause the system failure. The analysis results can be combined to improve the system design, which may lead to a system with higher quality and reliability. At present, as we do not have sufficient quantitative information (e.g., failure rates) about the system components therefore quantitative analysis is not performed. In the future, we have a plan to extend this work by collecting quantitative information about the components and performing quantitative analysis of this system.

References

1. Abecassis, Z.A., McElroy, L.M., Patel, R.M., Khorzad, R., Carroll, C., Mehrotra, S.: Applying fault tree analysis to the prevention of wrong-site surgery. Journal of Surgical Research 193(1), 88–94 (2015)
2. Al-Qoran, L., Gordon, N., Walker, M., Sharvia, S., Kabir, S.: A safety analysis approach to clinical workflows: Application and evaluation. International Journal of Advanced Computer Science and Applications(IJACSA) 4(3), 82–91 (2014)
3. Andrews, J.: Tutorial: Fault Tree Analysis. In: Proceedings of the 16th International System Safety Conference (1998), `http://www.fault-tree.net/papers/andrews-fta-tutor.pdf`
4. Bernardi, S., Merseguer, J.: A uml profile for dependability analysis of real-time embedded systems. In: Proceedings of the 6th International Workshop on Software and Performance. pp. 115–124. ACM, New York, USA (2007)
5. Boyd, C.E., Romaire, R.P., Johnston, E.: Predicting early morning dissolved oxygen concentrations in channel catfish ponds. Transactions of the American Fisheries Society 107(3), 484–492 (1978)
6. Boyd, C.E., Wood, C.W., Thunjai, T.: Aquaculture pond bottom soil quality management. Pond Dynamics/Aquaculture Collaborative Research Support Program, Oregon State University (2002)

7. Cepin, M.: Assessment of power system reliability. springer (2011)
8. Chen, S., Ho, T., Mao, B.: Reliability evaluations of railway power supplies by fault-tree analysis. Electric Power Applications, IET 1(2), 161–172 (March 2007)
9. Culberson, S.D., Piedrahita, R.H.: Aquaculture pond ecosystem model: temperature and dissolved oxygen prediction mechanism and application. Ecological modelling 89(1), 231–258 (1996)
10. FAO: The state of world fisheries and aquaculture 2012. Selected issues in fisheries and aquaculture pp. 1–148 (2012), `http://www.fao.org/docrep/016/i2727e/i2727e01.pdf`
11. Hayes, K.R.: Identifying hazards in complex ecological systems. part 1: fault-tree analysis for biological invasions. Biological Invasions 4(3), 235–249 (2002)
12. Isograph: Reliability workbench. `http://www.isograph.com/software/reliability-workbench/` (2014), accessed: 2014-09-01
13. Leveson, N.G.: Safeware: System Safety and Computers. ACM, New York, NY, USA (1995)
14. Papadopoulos, Y.: HiP-HOPS. `http://hip-hops.eu/` (2012), accessed: 2014-09-01
15. Ramesh, V., Saravannan, R.: Reliability assessment of cogeneration power plant in textile mill using fault tree analysis. Journal of Failure Analysis and Prevention 11(1), 56–70 (2011)
16. Shu, M.H., Cheng, C.H., Chang, J.R.: Using intuitionistic fuzzy sets for fault-tree analysis on printed circuit board assembly. Microelectronics Reliability 46(12), 2139–2148 (2006)
17. Vesely, W., Dugan, J., Fragola, J., Minarick, Railsback, J.: Fault Tree Handbook with Aerospace Applications. Tech. rep., NASA office of safety and mission assurance, Washington DC (2002)
18. Vesely, W., Goldberg, F., Roberts, N., Haasl, D.: Fault Tree Handbook: NUREG-0492. US Nuclear Regulatory Commission, Washington DC (1981)
19. Villemeur, A.: Reliability, Availability, Maintainability and Safety Assessment: Methods and Techniques, vol. 1. John Wiley & Sons, Chichester (1991)
20. Volkanovski, A., Čepin, M., Mavko, B.: Application of the fault tree analysis for assessment of power system reliability. Reliability Engineering & System Safety 94(6), 1116–1127 (2009)
21. Walker, M.D.: Pandora: A Logic for the Qualitative Analysis of Temporal Fault Trees. Ph.D. thesis, University of Hull (2009)