Advanced Fault Tree Analysis for Improved Quality and Risk Assessment

Jussi-Pekka Penttinen[1], Timo Lehtinen[2]

Abstract Traditional fault tree analysis (FTA) has formed a common language for the modelling and analysis of risks of complex systems. The approach can be used for a large variety of cases, and several tools exists for the method. Advanced FTA is needed to apply the FTA approach also for dynamic systems with time dependent activities. The traditional FTA is extended by including for example modelling and simulation of operation or phase changes of the systems, stochasticity and delays of relations, and maintenance actions for the components. The rules between the failures and other simulated properties are defined by including Java-based scripts to the model. This allows a great freedom to model any type of dynamic situations. A tool named ELMAS has been created to make the use of the advanced FTA approach as straightforward as possible. It allows to model the static and simple relations by using the traditional FTA method. Complex and dynamic relations are included by using modules that can be freely defined based on the needs of the case.

1 Introduction

It is very beneficial, or even mandatory, to be able to verify the quality and to assess (identify, analyse, and evaluate) the risks of the system. This requires explicit understanding of risk related system quality attributes, such as reliability, availability, maintainability and safety. Processing of data into information and gradually to knowledge about these essential factors is also needed for making rational decisions on how to improve the system most efficiently.

[1] J-P. Penttinen (✉)
Ramentor Oy, Tampere, Finland
e-mail: jussi-pekka.penttinen@ramentor.com

[2] T. Lehtinen
Ramentor Oy, Tampere, Finland
e-mail: timo.lehtinen@ramentor.com

At design phase the alternatives can be compared by creating and analysing a model that describes the features of the system under design. For already existing systems it is possible through modelling to predict the overall effect of proposed modifications. Without any major modifications and investments the maintenance and spare part policy optimization can be used to improve the performance of the current system. In each of these three situations all the knowledge must be collected, combined and analysed to obtain the ideal solution.

It is most effective to have a single method for a variety of systems and cases. Consistent method forms a common language for separate systems engineering projects. This method must be expressive enough to handle complex cases but still as easy to adopt as possible. The method should be never a reason to make any compromise in modelling or analysis. Also it must be systematic and explicit so that the results obtained from it are clear and unambiguous. A need of a tool is obvious to simplify the use of the method so that the efforts can be directed solely to the features of the studied system.

Fault tree analysis (FTA) has been traditionally applied for complex systems engineering projects. (Roberts et al., 1981) Improvements are needed for this over 50 year old method to make it more suitable for various needs related to comprehensive risk assessment. (Virtanen et al., 2006) With traditional fault tree it is not straightforward to model for example systems with several operating modes or phases, exclusive alternative consequences with stochasticity, relations that contain delays, cold redundancy with start-up time, other dynamic structures, nor effects of maintenance actions. Also for quantitative FTA there is more feasible and detailed input data for the events than a probability.

Advanced FTA uses improved fault tree model to collect efficiently experts' knowledge about the system operation profile, structure and cause-consequence-relations. This skeleton model is combined with the information related to failure, restoration, maintenance and other events. The information can be read and processed for example directly from available big data. The advanced analytics tool ELMAS (Event Logic Modelling and Analysis Software) offers a graphical user interface for the sophisticated modelling, efficient data import and stochastic discrete event simulation based analysis that is used for the overall model. The explicit analysis results can be improved to be more concrete and versatile by including for example data related to repair and maintenance costs, other expenses, production profiles or any other information available for the model.

2 Risk Assessment and RAMS

Risk assessment is the overall process of risk identification, risk analysis and risk evaluation. (ISO GUIDE 73, 2009) The first step risk identification includes finding, recognizing and describing risks. In advanced FTA approach this means

collecting the available information to a comprehensive model. The second step, risk analysis, aims to comprehend the nature and determine the level of risk. In advanced FTA approach this is done with stochastic discrete event simulation of the model. The last step, risk evaluation, compares the results with risk criteria to determine whether the risk and its magnitude is acceptable or tolerable. In advanced FTA approach this is done with the help of ELMAS tool by creating reports of explicit results and comparing various scenarios.

The risk assessment is a part of risk management, which means the coordinated activities to direct and control an organization with regard to risk. (ISO 31000, 2009) The risk management process is included in the systems engineering approach. Systems engineering is a methodical, disciplined approach for the design, realization, technical management, operations, and retirement of a system. (NASA SE, 2007) The three steps of the risk assessment and the general idea of the risk management process are illustrated in Figure 1.

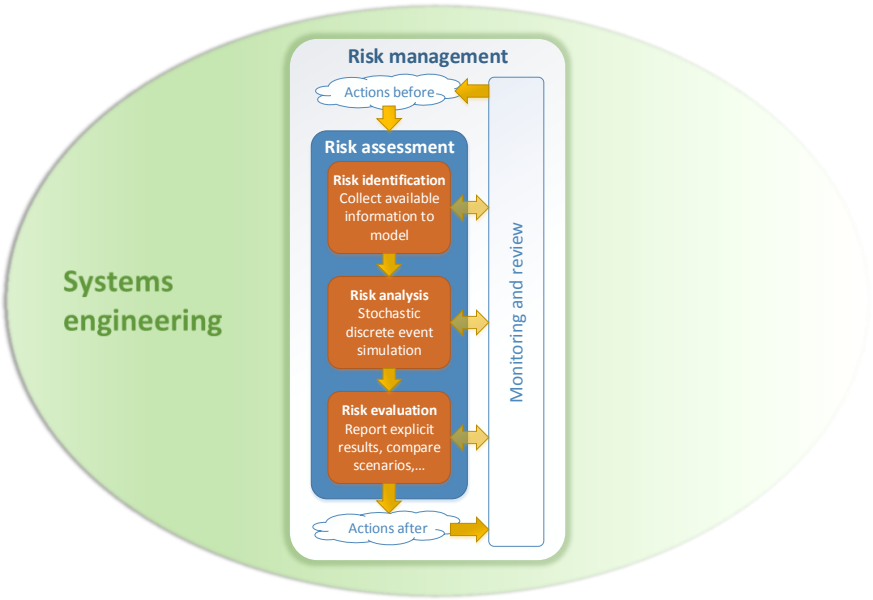

Figure 1: Risk assessment process as a part of risk management and systems engineering

An acronym RAMS is used for terms reliability, availability, maintainability and safety. These essential risk related system quality attributes are generic. They can be used for all types of risk management irrespective of the type of item considered. An item is defined as part, component, device, subsystem, functional unit, equipment or system that can be individually described and considered. (EN 13306, 2010)

Risk is defined as an effect of uncertainty on objectives. (ISO GUIDE 73, 2009) The risks of a system can be divided to availability and safety risks. The combination of likelihoods and consequences of dependability related risk sources form availability risks of the system. Similarly, the combination of likelihoods and consequences of hazards form safety risks of the system. RAMS includes dependability (RAM) and safety (S). The terms of the risks and RAMS are illustrated in Figure 2.

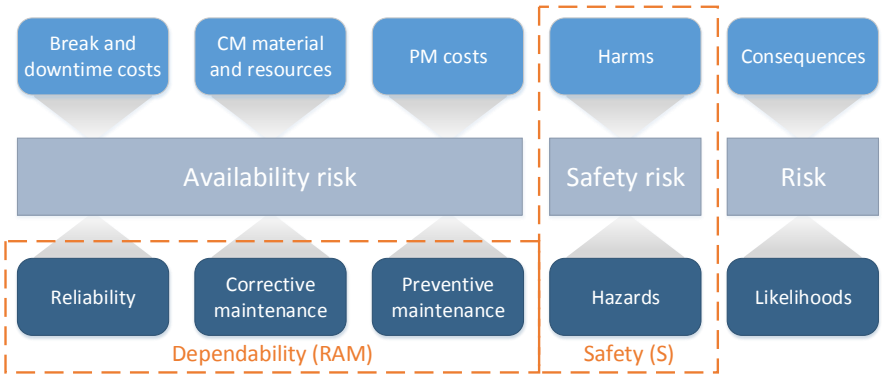

Figure 2: The terms of risk and RAMS

3 Advanced Fault Tree Model

The advanced fault tree model is created on based on a general analysis of things (AoT) base model that consists of nodes and relations. The base model allows to define variables and parameters needed for each node to model the properties of an item of the studied entity. Also relations between nodes are freely defined so that for example logic and stochastic rules, delays and operation profile connections can be modelled as needed.

3.1 Analysis of Things (AoT) Base Model

The fundamental principle of the AoT base model is that the models of the standard quantitative RAMS analysis methods should be included easily. Such analysis methods are for example Failure modes and effects and criticality analysis (FMECA), Fault tree analysis (FTA), Event tree analysis (ETA), Cause-consequence analysis and Markov analysis. (EN 31010, 2010)

The same AoT base model data structure also able to store data related to qualitative RAMS analysis methods. Such analysis methods are for example Failure modes and effects analysis (FMEA), Reliability centred maintenance (RCM), Root cause analysis (RCA, “5 whys”), Hazard and operability studies (HAZOP) and Check-lists. (EN 31010, 2010) It is beneficial to have common data model

that collects all the data required for different analyses. The similar data used by separate analyses needs to be updated only once and connections between results of different analyses can be easily pointed out.

3.2 Nodes of the Advanced Fault Tree Model

In advanced fault tree model each node has a state variable that defines whether an item of the studied entity can perform a required function. There can be also parameters related to the state variable, for example a cost of a spare part spent each time the restoration is made after the fault state. A node can have also for example a parameter related to its production capability that can be used in process diagrams to calculate total production speed of the system.

In addition to a state parameter related to ability to perform a required function, there can be for example a phase parameter for each node that defines whether the item should operate or not. This parameter is needed in simulation of batch production. The approach is that there can be more or less parameters and variables depending of the needs of the studied case. For example the preventive maintenance actions are included to the model by adding a variable that defines whether the maintenance action is currently made for the item.

3.3 Relations of the Advanced Fault Tree Model

Relations are defined from a group of input nodes to a group of output nodes. The output node of a relation is for example the node that models the system and the input nodes of a relation are all its sub systems. In addition to these lists, parameters are used to define the nature of the relation. For example min and max parameters can be used to model the logic rule. (Virtanen et al., 2006) In addition a probability parameter can be used for stochastic relation or some parameters can be used to define the delay distribution between the input and output nodes.

A special relation, called root relation, of a node contains only the node itself in both input and output nodes lists. A root relation can contain for example mean time to failure (MTTF) and mean time to restoration (MTTR) parameters that are used to define the cumulative distribution function (CDF) for the lengths of states of the node. After each state change new delay time is defined, the new state change is handled after the delay and this procedure is continued independently.

Phase relations are other special relations that define when the item is needed and when it can be idle. These can be defined similarly with state relations by using parameters that define delay between phase changes. For more complex phase changes, freely defined scripts can be defined to model the various phase change

situations. Also maintenance actions are defined similarly with phase and state relations.

3.4 Stochastic Discrete Event Simulation

The created model is analysed through stochastic discrete event simulation. Events in the simulation mean a change of a state parameter value. Similarly phase parameter values are changed by events. At the beginning of each simulation round first events are defined for root relations. If needed a freely defined script can be given to define the initial phase of a round.

When an event is handled the needed variable value changes are made for the node. After the changes all the relations that has the node as an input node are also handled. The relations will define for example whether a state change of a system is needed after a state change of its sub-system. Also the root relation is handled again after a state change of previous root relation to define the timing of the next state change.

Data is collected during the simulation process for example by counting the number of state changes made for each node or the cumulative time the node spends in some state. The data of each simulation round is combined so that distributions can be drawn and mean and quantile values can be calculated.

4 Improvements to the Traditional FTA

Following chapters define simple example cases in which different properties of the advanced fault tree model are needed.

4.1 Operation Phases

The traditional fault tree model is a static structure. Advanced FTA approach is needed for systems with separate operation phases or other dynamic properties. For example each phase of a patch process can have different failure modes or the same failure modes can have different failure distributions. Also the failures in each phase can have different consequences, for example the time to restart after a failure can be different for each phase. In addition to this system level dynamic properties there can be component level dynamic properties, for example cold redundancy.

A phase variable added to nodes of a model can be used to simulate dynamic changes. The variable tells whether the system or component is currently used or not. The changes of these phase variables can be simulated independently similarly with the simulation of failures. For example by using parameters the distribu-

tion of the duration of each phase can be defined. After previous phase ends the next phase simulation is started.

Rules are defined to model the relations between states and phases. For example the normal path of phases can be interrupted when failure occurs and restarted after some delay. Also with cold redundancy the redundant component is started after a failure of the main component with some delay and possibly also failure probability. The relation can be also to opposite direction, for example at some phase some subsystems are not needed and their failures can be ignored or the operation is paused during the phase and failures do not even occur. In advanced FTA approach the rules of these relations are defined by writing Java based script.

4.2 Stochastic and Delay Relations

In traditional fault tree the relations are immediate, but sometimes delays between events need to be modelled. In advanced FTA approach the delay between events is modelled similarly with the definition root relations. In root relation the node itself triggers the new delay time but in delay relation the trigger event is a state change of any other node.

There can be delay after a failure for example related to costs. For example the downtime cost can get larger after some failed time. There can be also with some probability some cost if the fault is long enough. Also after restoration there can be delay before the system can be started again. Delays between phases and possible probabilities related to the phase changes are modelled and simulated similarly with the stochasticity and delays between state changes.

4.3 Maintenance Plan of Inspection Actions

Preventive maintenance actions are not included in the traditional fault tree. In advanced FTA approach these are modelled and simulated similarly with the state and phase changes by defining a maintenance variable for the nodes. At some predefined, stochastic or condition based time intervals a maintenance action is made for a node that may affects to its state or the phase of the system. The optimal maintenance plan is tried to be reached by analysing how the changes in maintenance schedule of a component affects to the whole system.

The moment of the action is defined similarly with root relations by using cumulative distribution functions (CDF) or static time intervals. Other possibility is to define rules when the action is made by using Java based scripts. Also the effect of an action can be defined by using a script. There are also predefined actions for the most common maintenance actions. For example inspection action can be modelled just by defining the symptom time and the probability that the symptom is recognized and the starting failure can be prevented before it occurs.

5 ELMAS

ELMAS is a tool that helps to use the advanced FTA approach by producing graphical user interface to create hierarchies, flow diagrams and other structures of nodes and their relations. From the options the used parameters and variables are selected for nodes and relations and the corresponding fields are shown in the user interface. Also an interface to include freely defined Java-based scripts is included to model more complex relations.

ELMAS produces also interfaces to import parameter data from the databases. For example failure history data can be used to create distribution that models the delay before next failure. The simulation results can be exported for example to data tables or various visual presentations of the results can be shown directly from the ELMAS.

References

N. H. Roberts, W. E. Vesely, D. F. Haasl & F. F. Goldberg, 1981. *Fault Tree Handbook*, U.S. Nuclear Regulatory Commission, NUREG-0492

S. Virtanen, P-E. Hagmark, & J-P. Penttinen, 2006. *Modelling and Analysis of Causes and Consequences of Failures,* Annual Reliability and Maintainability Symposium (RAMS). January 23 – 26, 2006. Newport Beach, CA, USA

ISO GUIDE 73, 2009. *Risk management. Vocabulary*, International Organization for Standardization*,* ISO GUIDE 73:2009

ISO 31000, 2009. Risk management. Principles and guidelines, International Organization for Standardization*,* ISO 31000:2009

NASA SE, 2007. *NASA Systems Engineering Handbook*, National Aeronautics and Space Administration, NASA/SP-2007-6105 Rev1

EN 13306, 2010. *Maintenance. Maintenance terminology*, European Committee for Standardization, EN 13306:2010

EN 31010, 2010. *Risk management. Risk assessment techniques*, European Committee for Electrotechnical Standardization, EN 31010:2010