IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 12, December 2017

Algorithms for Reducing Cut Sets in Fault Tree Analysis

Akinode, John Lekan[1], Oloruntoba S.A[2]

Department of Computer Science, The Federal Polytechnic Ilaro, Nigeria [1,2]

Abstract: Fault Tree Analysis is a graphical and analytical model for identifying and accessing the relevant causes of a system risk or fault or undesired event. it is used in process system for identifying basic events or causes of an event, identifying common-cause(Minimal cut sets) failures, displaying causes and consequence of undesired event, evaluating design and investigating process accidents/incidents. The main aim of any fault-tree algorithm is to compute the minimal cut sets as quickly as possible. A cut set is a collection of component failure modes that could lead to a system failure. Cut Set Analysis (CSA) is applied to critical systems to identify and rank system vulnerabilities at design time. This paper presents a critical review of the various quantitative fault tree analysis techniques and also provide a formal procedure of each of the techniques.

Keywords: Fault Trees, Cut sets, Algorithm, Fault Tree Analysis, Minimal Cut set.

I. INTRODUCTION

The complexity in the design and variation in operating conditions in critical system necessitate the uncertainty and randomness of fault associated with them [33]. [32] reiterate that the failure of a Safety-critical systems can lead to environment harm, fatal injury to people and economical loss. Analysis of safety-critical system boast of several techniques, this include Fault Tree Analysis (FTA), Failure Mode And Effects (FMEA), Hazard And Operability Analysis (HAZOP) and State Machine Hazard Analysis (SMHA) [30]. Fault Tree Analysis (FTA) has been extensively used to determine the reliabilities of complex systems [31].It has a reputation of being an effective technique to predict probability of hazard caused by a sequence and integration of faults and failure events. It is the most important process for system reliability assessment [29].it has become a benchmark for safety and reliability of systems. Fault Tree Analysis (FTA) is an essential method for the evaluation of complex system's safety and reliability. It describe a top-down approach to failure analysis. It is basically an analytical method used for identifying and classifying hazards and computing reliability of systems [1].The reliability engineer designates a top event (failure or accident) and then builds the series of faults leading to the top event. It is basically a deductive process for obtaining combinations of component failures and human error that could lead to a specified undesired system failure mode [29]. The process starts with an undesirable event known as top-level event and it determine the scenarios which the event may occur in the system. The basic concept in fault tree analysis is the conversion of a physical system into a structured logic diagram (fault tree) in which certain causes lead to one specified TOP event of interest [24]. A Fault Tree depicts a logic diagram that represents certain events that must occur in order for other events to take place. Finding Minimal Cut Sets (MCSs) is one of the fundamental focus of fault tree analysis [3].However, a fundamental question in fault tree analysis is how many levels of events should be considered. [4] emphasized that obtaining the Minimal cut sets is a fundamental step for analysing a fault tree. The main challenge in fault tree analysis is the computation of the minimal cut sets of the fault tree [5] and FTA [6]. [6] claimed that the computation of Minimal cut set is NP-hard. [3] described the Minimal Cut Set (MCS) as a significant concept in fault tree assessment and submit that they represent minimal scenario of failure. According to the Fault Tree Handbook , Minimal cut set, described the smallest combination of basic events that causes a top event. Similarly, Fault Tree Handbook opined that successful computation of Minimal Cut Set (MCS) simplified the process of quantifying the process of Fault trees. The process involve in computing the Minimal cut sets for smaller Fault tree is less tedious. However, the task becomes complex when a larger Fault tree is involved as algorithms and codes are required to determine the minimal cut sets. Making effective use of fault trees that resolve causes of system failure is dependent on the fault tree algorithms that provides optimal cut sets. This paper gives an insight into how some of the popular fault tree algorithms carry out the computation of minimal cut sets of complex fault trees.

II. RELATED WORK

Research in the context of Fault tree analysis has a long history. There is a huge literature on the concept of fault tree analysis. It has been of great interest to safety and dependability domain started from 1960s and has continued to gain vast recognition. [7] adopted the concept of primary failure technique to construct a fault tree. [8] developed a computer program –ELRAFT basically for automation of fault tree construction. [2] published a comprehensive review

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 12, December 2017

and classification of fault tree analysis methods. The review was divided into: fault tree introduction, fault tree construction and minimal cut sets

A new algorithm for reducing repeated events in the fault tree was proposed by [4].The algorithm enhance the MOCUS- top down algorithm to obtain all minimal cut sets in a faster way. The improvement is based on reducing the number of comparisons needed to obtain the minimal cut sets.

[6] introduced an algorithm based on Binary decision diagram (BDD) tree that compute and memorize the cuts with high degree of efficiency.

[5] in his work developed an algorithm that store a transformed structure of fault tree into dynamical structure from which Minimal cut sets computation is determined with ease. The fault tree structure is dependent on the analysed fault tree and is limited by the amount of free space in the computer's memory.

[9] highlights the works of Fussel and Vesely –MOCUS (Method of obtaining Cut Sets Upward) and [10] on MISCSUP (Minimal Cut Sets upward).He also emphasise on some of the applications of Fault tree analysis

III. FAULT TREES

The fault tree is simply a graphical representation of Boolean logic related to the design of a specific system failure, known as the top event, to basic failure also known as the primary events [7]. [11] describe a fault tree as a graphical description that shows how several combinations of basic events, associated to the components of the system, result to the TOP EVENT. It is a Boolean logic model that describe the relationships between events in a system that result to an outcome known as the top event [13]. A typical structure of a fault tree is shown in figure 1. It is made up of a top event which is normally a system failure and connected to one or more basic events through a system of logical gates [12].

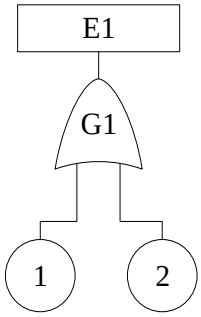

Figure 1.Sample Tree.

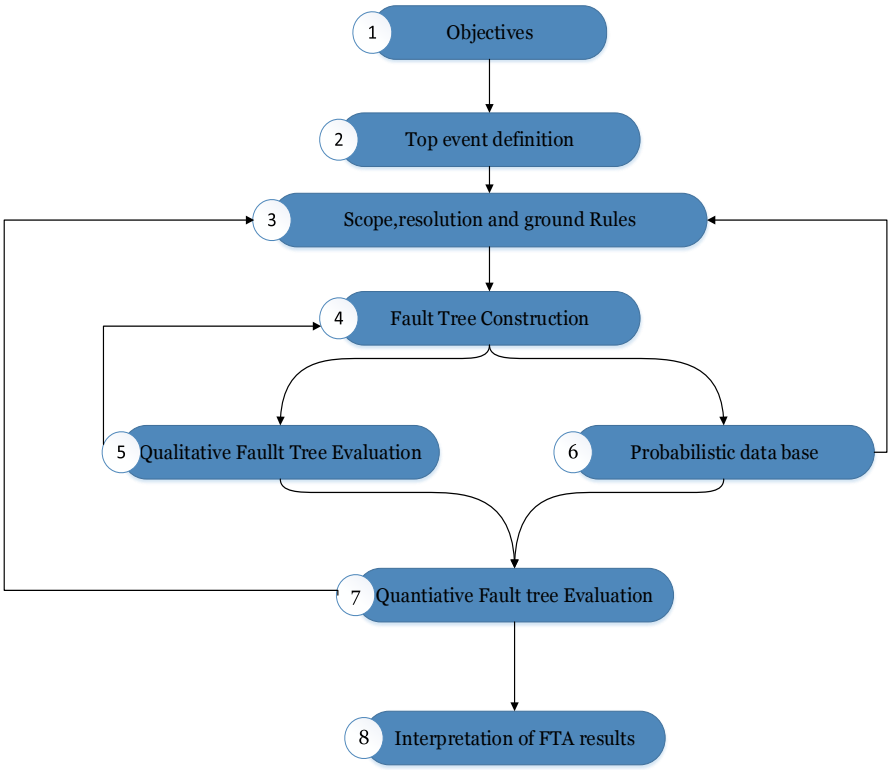

Figure 2: Basic procedure for Fault tree analysis [28]

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 12, December 2017

IV. MINIMAL CUT SETS

The concept of minimal cut sets of fault tree is of immense significance in reliability industry. System failure mode describes the specific way that system failure can occur. This entails the failure of a singular component or combination of components. Boolean equations describe how component failures can result to failure in the entire system.A cut set of a fault tree is a collection of combination of component failure (input events) which association can cause system failure. A minimal cut set is the smallest combination of component failures that lead to a failure in the entire system.

V. SIGNIFICANT OF MINIMAL CUT SETS

Minimal cut sets are used to compute the probability, unavailability or unreliability of top event of certain codes [27].A typical example of the code is KITT and SUPER-POCUS [27].Similarly, some codes also employ the concept of minimal cut sets for sensitivity or uncertainty analysis. Minimal cut sets also proffer helpful information concerning design weaknesses of the system. The resulting minimal cut sets can be placed side by side with the prototype tree to discover errors in the fault tree logic.The knowledge of minimal Cut Sets permit the analyst to evaluate the criticality of component failures to enhance the reliability of the modeled system [14].

VI. MINIMAL CUT SET ALGORITHMS

Overtime, several quantitative techniques have been proposed to determine the minimal cut sets of a fault trees. [15] reiterates that the minimal cut sets in some of the army systems can be determined by mere inspection. However, this approach fails when it encountered a large fault tree. Therefore, several techniques have been proposed to compute the minimal cut sets of a fault tree. Most of the techniques for computing minimal cut sets of fault tree hinged on the concept of Boolean reduction methods [16].

A. Direct Boolean Reduction Method

This technique manipulates the fault tree based on the set of Boolean Algebraic laws. In this approach, the reduction of fault tree to its equivalent minimal cut set can be carried out by directly applying Boolean reduction rules.

B. MOCUS Method

MOCUS Method

MOCUS (Method of Obtaining Cut Sets) was initially suggested by [17]. It is regarded as one of the traditional fault tree reduction algorithms and one which other solutions were derived [18].MOCUS is an efficient top-down gate substitution technique for generating cut sets. It adopts a top-down approach that start its operation from the root(top event level) and advance to decompose each gate at higher level until only basic events are determined.

The essence of the method according to [18] is summarized as follows:

1. Label or number all gates and events.
2. Create a two dimensional Array (table) B with row and column.
3. Initialise the first element of the matrix B (1, 1) with the top event operator array. The top event is placed in the first column of the first row
4. Search through the table find an OR/AND gate:

a) If the gate is an AND gate, place each of the gate's children in a new column
b) If the gate is an OR gate, place each of its children in a new row

5. Step 4 is repeated continuously until there are no gates in the array.
6. Generate only minimal cut sets by removing non-minimal basic events. This is achieved by removing all redundancies within the array based on the Boolean laws (Absorption, Distributive, Associative, Idempotent etc.).The final list contains a minimal cut set in each row .MOCUS is faster and ideal for the analysis of smaller trees. However, size of the array is dependent on the number and position of OR gates in the fault tree.

C. SETS Method

The SETS (Equation Transformation System) technique, designed by Sandia National Laboratories, is a fault tree evaluation method that adopt simplification of Boolean expression to determine minimal cut set [19] .It manipulates the Boolean expression created by a sequence of events operated on by certain set of intersection, union and complement operators [20].The Boolean equation is similar to those explained above.

SETS is operated by the user's program developed by the user. Generally, two main algorithms are applied to accomplish the SETS code.

[20] highlighted the basic steps in the algorithm as follow:

1. Firstly, replace the Boolean equation of each from the top event to the branches at the lowest level of the tree.
2. Establish independent subtrees, substitute them by a module, and then carry out a simple substitution of the Boolean expression from the top event to the Basic events at the lowest level.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 12, December 2017

3. By handling the user's program, the two algorithms can be used first to intermediate gates and subsequently to gates in higher level, which trigger a bottom-up resolution of the tree.

The SETS fault tree evaluation method has a provision for logical merging of fault tree. However, its main problem hinged on the premise that the efficiency of the algorithm is dependent on the perfect setup of user's program.

D. SIFTA Method

The SIFTA algorithm for fault tree analysis adopts pattern-recognition approach restructuring the fault tree by employing the basic Boolean algebra law and reorganizing certain patterns of events [20].This process is a deviation from the conventional generation of cut sets. For instance, if repeated event E is an input to two different OR gates B and D,and the gates are input to AND gate F,then event E can be derived from B and C and placed directly into primary event.Hence,diverse complex patterns are adopted to obtain a reduced form. The numerical estimation begins with the branches rendered independent as a result of the reduction of common events. The residual tree is processed by simulation in a case where the tree does not permit further tree.

SIFTA method is very easy to use and can manage trees with several top events

E. MICSUP Method

The concept of MICSUP (Minimal Cut Sets Upward) algorithm for fault tree analysis was conceived by [21] in 1974.The algorithm adopts a bottom up approach technique. It starts the processing from the bottom of the tree, it gets G_j, the list of all cut sets inputted to gate "j".Hence, the minimal cut sets are computed by a basic search to obtain G^*_j.However, this reduction approach is not ideal for a fault tree whose basic events inputted are mutually disjoint (Gordon, 1975).In the scenario of an OR gate, Gj is the union of all inputted G^*_k .Similarly, in an AND gate, Gj is the Cartesian product.

(Mark, 2011) gave a summary of the operation of MICSUP algorithm as follows:

Step 0: Initialize a sequence of primary gates:G=setofallprimarygates.
Step 1: While G still have primary gates,
Step 2: Adopt Boolean reductions.
Step 2: Substitute the product terms in G
Step 4: Apply Boolean reduction to primary gate of top event
Step 5: G ensue in only basic events.

F. WAM-CUT Method

WAMCUT fault tree evaluation method comprises of two parts: WAM and CUT. The WAM is basically a pre-rocessor that check for logic and syntax errors in the fault tree description. CUT is the cut-set finder module that take the reassembled input fault tree from WAM unit and determine all the cut sets, of each gate, traversing from the bottom to the top of the tree

G. ALLCUT Method

ALLCUTS algorithm determine minimal cut sets from fault trees with AND and OR gates [27].it adopts a top down Boolean substitution similar to the MOCUS technique. In this technique, BRANCH, an additional program module is used to inspect the input and thoroughly check the gates and input primary events.

The application of ALLCUTS technique in fault tree evaluation is hampered by the limited number of cut sets it generate [27].

H. FATRAM Method

The FATRAM algorithm process is the same with the MOCUS algorithm. However, the computation time in this method is minimal. Thus, larger fault trees can be handled efficiently with this technique.

[27] gave the main reduction process of FATRAM method as follow:

1. Simplifying the OR gates with gates inputs and AND gates from the start.
2. Managing repeated primary events.
3. Suspending until end the resolution of OR gates with the only primary-events.
4. Listing out the cut sets without expanding the core. Removing supersets at initial stages

I. ELFRAT- Efficient Logic Reduction Analysis of Fault Trees

ELFRAT- Efficient Logic Reduction Analysis of Fault Trees

ELFRAT, a mathematical simplification algorithm was conceived by [23] in 1971.The method is based on the notion of prime number description of basic events for reduction of fault trees .ELFRAT adopts the Boolean algebra law of Idempotence and Absorption to simplify the fault trees to its minimal cut sets ,but check the whole cut sets once to ascertain whether it can be reduced, an approach different from MOCUS algorithm that check each event against each other[18].it is useful in keeping the cut sets and removing the supersets.

J. BDD Method -Binary Decision Diagram

Fault tree analysis for complex system can be intensive in terms of computation and may entail the use of approximations [25].This leads to incorrectness, in the evaluation of system reliability.

The Binary decision diagram was developed to overcome such difficulties. This techniques enhance the exactness and efficiency since the exact solution can be computed without the need to calculate the minimal cut sets. Presently, BDD has proved to be invaluable to system reliability [26] .However, the main problem with this approach is with the translation of the fault tree to the BDDs.

In this FTA method, the tree is initially converted to a binary decision diagram, which describes the Boolean equation for the top event. However, the main challenge is the translation of the fault tree to the Binary Decision Diagram[26].

VII. CONCLUSION

Fault Tree Analysis is a graphical and analytical model for identifying and accessing the relevant causes of a system risk or fault. Finding minimal cut sets (MCSs) is the fundamental aim of fault tree analysis. In this paper, we have presented an extensive overview of different fault trees analysis techniques for computation of cut sets. The paper focuses on the quantitative methods of fault tree analysis. The paper also highlights the strength and weaknesses of fault tree analysis algorithms. This will provide adequate information on the formal procedure of different fault tree analysis algorithms to both experienced and newbie in the field of reliability engineering.

REFERENCES

1. Dan M. S & Joseph T, 2007, Condition-based fault tree analysis (CBTA): A new method for improved fault tree analysis (FTA), reliability and safety calculations, Reliability Engineering and System Safety
2. Lee, W S, Grosz, D L., Tillman F A, Lie, C H, 1985, Fault Tree Analysis, Methods, and Applications –A review: IEEE Transactions on Reliability.
3. Jianwen X, Kazuo Y, Yoshiharu M, Kumiko T, Fumio M, Atsushi K and Takao O, 2011, Efficient Analysis of Fault Trees with Voting Gates.IEEE International Symposium on software Reliability Engineering.
4. Limnios N. & Ziani R, 1986, Algorithm for Reducing Cut Sets in Fault Tree Analysis, IEEE Transactions on Reliability, Vol. 5.
5. Ladislav R, 1996, Algorithm for finding minimal cut sets in a fault tree, Reliability Engineering and System Safety: Elsevier Science Limited.
6. Antoine R, 1993, New Algorithms for fault trees analysis, Reliability Engineering and System Safety.
7. Hassl D.F.,"Advanced, 1965, Concept in Fault tree Analysis" System Safety Symposium
8. Fussel J B, 1972, Synthetic Tree Model, A Formal Methodology for Fault Tree Construction, PhD dissertation, Georgia Institute of Technology.
9. Ericsson C A, 1999, Fault Tree Analysis-History, Proceedings of the Twelfth International System on Safety Conference.
10. Pande,Pradip K. ; Spector,Michael E. ; Chatterjee,Purnendu ,1975 Computerized Fault Tree Analysis: TREEL and MICSUP.1975, CALIFORNIA UNIV BERKELEY OPERATIONS RESEARCH CENTER
11. Chatterjee P, 1974, Fault Tree Analysis: Reliability Theory and Systems Safety, PhD Thesis, University of California, Berkeley.Microfim.
12. Papadopoulos Y,Walker M.,Parker D.,Erich R.,Hamman R.,Uhlig A.,Gratz U. and Lien R,2011,Engineering Failure Analysis and design Optimisation with HiP-HOPS, International Conference on Engineering Failure Analysis,Vol.18,Issue 2,pp. 590-608.
13. Jong Soo CHOI & Nam Zin CHO 2005.Truncation Error Evaluation Method for Minimal Cut Set-Based Fault Tree Analysis, Journal of Nuclear Science and Technology, 42:10, 854-860
14. Carrasco J A and Sune V, 1999, An algorithm to find Minimal cuts of coherent Fault-Trees with Event-classes, using a Decision Tree, IEEE Transaction, Vol.48, NO 1.
15. Gordon Lee Rankin, 1975, An Application of Fault Tree Analysis to Operational Testing, Master's Thesis, Georgia Institute of Technology.
16. Mark L,M,2011,A Methodology and Tool Support for the Design and Evaluation of Fault Tolerant, Distributed Embedded Systems, PhD Thesis, University of California, Berkeley.
17. Fussell, J. B. and W. E. Vesely, "A New Methodology for Obtaining Cut Sets for Fault Trees," American Nuclear Society Transactions, Vol. 15, No. 1, pp. 262-263, (1972)
18. Walker M.D, 2009, Pandora: A logic for the Quantitative Analysis of Temporal Fault Trees, PhD Thesis, University of Hull
19. Worrell R.B and Burdick G.R, 1976, Qualitative Analysis in Reliability and Safety, IEEE Transactions on Reliability, Vol.R-25, NO.3.
20. Karimi R, 1980, Qualitative and Quantitative Reliability Analysis of Safety Systems, PhD Thesis, Department of Nuclear Engineering, Massachusetts Institute of Technology, Cambridge.
21. Chatterjee P, 1974, Fault Tree Analysis: Reliability Theory and Systems Safety, PhD Thesis, University of California, Berkeley.Microfim.
22. Gordon Lee Rankin, 1975, An Application of Fault Tree Analysis to Operational Testing, Master's Thesis, Georgia Institute of Technology.
23. Semanderes S.N,1971,"EFRAT" A Computer Program for the efficient logic Reduction Analysis of Fault Trees,IEEE Transaction on Nuclear Science Vol.Ns-18,pp. 481-487.
24. Lee, W S, Grosz, D L., Tillman F A, Lie, C H, 1985, Fault Tree Analysis, Methods, and Applications –A review: IEEE Transactions on Reliability.
25. http://www.nottingham.ac.uk/research/groups/ntec/centre-for-risk-and-reliability-engineering/research-projects/system-reliability-modelling-using-binary-decision-diagrams.aspx ,retrieved on 7th sept ,2016.
26. Reay K,A and Andrews J,D,2006, A fault tree analysis strategy using binary decision disagrams,Reliability Engineering and System Safety ,78(1),pp.45-56.
27. Hickman J, W, et al, 1983, PRA Procedures Guide: A Guide to the Performance of Probabilistic Risk Assessments for Nuclear Power Plants, The American Nuclear Society and The Institute of Electrical and Electronics Engineers.
28. Eugen Petrosan,2013,fault tree analysis for the EnergyGrid,Universitat Augsburg University
29. Sinnamon R M, 1996, Binary Decision Diagram, PhD Thesis, Loughborough University, UK.
30. Jianwen X, Kazuhiro O ,Weiqiang K and Kokichi F,2006,From Fault Tree Analysis to Formal System Specification and Verification with OTS/CafeOBJ, Japan Society for Software Science and Technology, Vol.2, No.2, pp.448-460.
31. Han Suk Pan &WonYoungYun,1997, Fault tree analysis with fuzzy gate Computers & Industrial Engineering Volume 33, Issues 3–4, December 1997, Pages 569-572

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 12, December 2017

32. Marco Bozzano , Adolfo Villafiorita,2003, Integrating Fault Tree Analysis with Event Ordering Information,ITC - IRST, Via Sommarive 18, Povo, 38050 Trento, Italy
33. Sanjay Kumar Tyagi , Diwakar Pandey , Vinesh Kumar,2011,Fuzzy Fault Tree Analysis for Fault Diagnosis of Cannula Fault in Power Transformer.Scientific Research Corporation.
34. Antoine R, 1993, New Algorithms for fault trees analysis, Reliability Engineering and System Safety.

BIOGRAPHIES

Akinode John Lekan obtained his M.SC. in Computer Science(Distributed Systems) from the University of Hull, United Kingdom in 2014. He has been working as a Lecturer in the Department of Computer Science,FederalPolytechnic,Ilaro,Nigeria.His research interest includes Safety Critical System, Distributed Systems, Creative Education, and Data Mining .

Oloruntoba Samson Abiodun, had a Bachelor of Technology(1992) and Master of Technology(2015) in Computer Science from Federal University of Technology Akure. He was formerly a head of Computer Science department at Federal Polytechnic, Ilaro Ogun State, Nigeria. He had published papers in International journal. His area of interest includes Neural network, Data Communication , E- commerce and Artificial Intelligent.

DOI 10.17148/IJARCCE.2017.61207