Research and Application of Fault Diagnosis Based on BDD Algorithm: Postprint

Authors: Li Shuying, Wang Peizhen, Yang Chun

Date: 2019-03-05T00:00:00+00:00

Abstract

Fault Tree Analysis (FTA) is susceptible to issues such as "dimension explosion" during its application. This paper focuses on BDD-based Fault Tree Analysis methods, examining the methodology for transforming fault trees into BDDs, and utilizes the BDD algorithm to calculate the probability of top event occurrence and the structural importance of basic events. Finally, this algorithm is employed for case study analysis and application.

Full Text

Research and Application of Fault Diagnosis Based on BDD Algorithm

Li Shuying[1], Wang Peizhen[2], Yang Chun[2]
[1] Nanning Rail Transit Group Co., Ltd., Nanning 530029, China
[2] Beijing Jiaotong University, Beijing 100044, China

Abstract

Fault tree analysis (FTA) often suffers from the "dimension explosion" problem during application. This paper focuses on BDD-based fault tree analysis, examining the transformation methodology from fault trees to Binary Decision Diagrams (BDD). The algorithm solves for the probability of top events and the structural importance of basic events. Finally, the method is applied to a case study for analysis and validation.

Keywords: Fault tree analysis, BDD algorithm, fault diagnosis, structural importance

Li Shuying, female, born in 1989, master' s degree, main research direction: vehicle communication and fault diagnosis technology.

Wang Peizhen, female, born in 1993, master' s degree, main research direction: power electronics and electrical drives.

Supported by the Joint Fund Project "Safety Prediction and Control of High-Speed Railway Power Traction Systems (U1134204)"

---

1 Introduction

Fault Tree Analysis (FTA) was first proposed in 1961 by researchers at Bell Labs in the United States and was initially applied to rocket launch safety assessment systems, achieving excellent results. Subsequently, Boeing Company further developed FTA in both qualitative and quantitative aspects [1]. FTA employs graphical deduction to clearly express the internal relationships of systems, offering strong intuitiveness and high flexibility.

The qualitative analysis of fault trees aims to identify minimal cut sets, which represent the smallest combinations of basic causes leading to system failure and depict the weakest links in the system. Traditional qualitative analysis methods for minimal cut sets primarily employ the downward or upward method. However, as the complexity of system fault trees increases, these methods are prone to the "combination explosion" problem. To address this issue, the BDD algorithm can be used to convert the fault tree into a disjoint form for further analysis [2].

2 BDD Algorithm

Binary Decision Diagram (BDD) [3] was first proposed by Sheldon B. Akers in 1978 and has been widely applied in fault diagnosis and reliability fields. BDD is a directed acyclic graph where all nodes are connected through 0 or 1 labels, obtained by simplifying the Shannon decomposition tree of Boolean functions. Its essence is the sum of minimal cut sets [4].

The Shannon decomposition theorem can be summarized as follows: Assume $f(x_1, x_2, \dots, x_n)$ is any Boolean function, and $x_i (i = 1, 2, \dots, n)$ is any variable of $f(x)$. Let f_0 denote the expression when x_i takes the value 0, and f_1 denote the expression when x_i takes the value 1. Then $f(x)$ can be expressed as:

$$f(x_1, x_2, \dots, x_n) = x_i \cdot f_1 + \overline{x_i} \cdot f_0$$

where $\overline{x_i}$ represents the negation of x_i.

From this equation, both f_1 and f_0 are Boolean functions. The ite (if-then-else) rule is used during the transformation process from fault tree to BDD, which is essentially a transformation of the Shannon decomposition tree expression. The mathematical expression of the ite rule is:

$$\mathrm{ite}(x_i, f_0, f_1) = x_i \cdot f_1 + \overline{x_i} \cdot f_0$$

The transformation from fault tree to BDD uses the ite rule recursively from the bottom events upward. Each step of substitution employs the ite rule for encoding until all logic gates are replaced by basic events, ultimately obtaining the ite structure of the top event, which is the BDD of the top event [5].

Assume $M = \mathrm{ite}(x_i, F_1, F_0)$ and $N = \mathrm{ite}(x_j, G_1, G_0)$ are two BDD substructures, with the fault tree basic events sorted as $x_1 < x_2 < \dots < x_n$ $(x_i, x_j \in \{x_1, x_2, \dots, x_n\})$. After standardization, M and N consist of "AND", "OR" logic gates and basic events, which can be connected through ite structures following these rules:

1. When $x_i = x_j$: $M\langle\mathrm{OP}\rangle N = \mathrm{ite}(x_i, F_1\langle\mathrm{OP}\rangle G_1, F_0\langle\mathrm{OP}\rangle G_0)$
2. When $x_i < x_j$: $M\langle\mathrm{OP}\rangle N = \mathrm{ite}(x_i, F_1\langle\mathrm{OP}\rangle N, F_0\langle\mathrm{OP}\rangle N)$

3 Analysis of Top Event Probability and Structural Importance Based on BDD Algorithm

3.1 Top Event Probability

The sum-of-products form of the structure function after BDD decomposition is disjoint, eliminating the need for inclusion-exclusion calculations when solving for top event probability. The probability calculation begins at the root node and traces back through all "1" terminal nodes. The corresponding disjoint expression of the fault tree is written, and the probability of the top event is calculated using the probability sum algorithm for mutually exclusive events.

3.2 Structural Importance Analysis

The structural importance of basic events refers to the significance of a component's position in the system, independent of its failure probability [5]. Through structural importance analysis of fault trees, components with high structural importance can be prioritized for inspection and testing, enabling preventive decision-making and significantly improving system reliability and safety performance [6].

If a fault tree contains n basic events, and the i-th component changes from normal state (denoted as $x_i = 0$) to failure state (denoted as $x_i = 1$) while other basic events remain unchanged, let $\Phi(0_i, X)$ represent the logical value of the top event when basic event i is in normal state $(x_i = 0)$ and other basic events are in arbitrary states, and $\Phi(1_i, X)$ represent the logical value when basic event i is in failure state $(x_i = 1)$.

The system top event may have four possible cases: 1. $\Phi(0_i, X) = 0 \rightarrow \Phi(1_i, X) = 1$ 2. $\Phi(0_i, X) = 0 \rightarrow \Phi(1_i, X) = 0$ 3. $\Phi(0_i, X) = 1 \rightarrow \Phi(1_i, X) = 1$ 4. $\Phi(0_i, X) = 1 \rightarrow \Phi(1_i, X) = 0$

Among these four cases, case (1) indicates that the change of basic event i from 0 to 1 causes the top event state to change from $\Phi(0_i, X) = 0$ to $\Phi(1_i, X) = 1$. The remaining $n-1$ basic events may have 2^{n-1} states. Structural importance accumulates all occurrences of case (1) and divides by 2^{n-1}, with the resulting value representing the contribution degree of basic event i to system failure. The structural importance calculation formula is:

$$I_i = \frac{n_i}{2^{n-1}}$$

where $n_i = \sum[\Phi(1_i, X) - \Phi(0_i, X)]$ for $i = 1, 2, \dots, n$.

4 Fault Diagnosis Logic Reasoning Flow Based on BDD Algorithm

The transformation from fault tree to BDD adopts the ite (if-then-else) rule. To implement this function, variables `ITE_str` and `BDD_str` are defined. The variable `ITE_str` stores the ite structure, including the ite structure name and its three elements, while `BDD_str` stores the BDD in binary tree form. The algorithm flow for converting a fault tree to BDD is shown in [Figure 1: see original paper].

The fault diagnosis process using FTA involves first converting the fault tree into a disjoint form using the BDD algorithm, followed by qualitative and quantitative analysis. The diagnostic flow is illustrated in [Figure 2: see original paper].

5 Traction Converter Fault Diagnosis Based on BDD Algorithm

The traction converter integrated logic judgment implements train traction and braking functions. Analyzing its critical equipment for fault analysis, improving system reliability, and quickly locating fault causes after failure are particularly important. When module faults occur, rapid diagnosis is essential.

The normal operating temperature of the traction converter module ranges between 20°C and 85°C. [Figure 3: see original paper] shows the structure diagram of the traction converter module temperature control system. The module temperature is collected by a temperature sensor and input to a conditioning board, which converts the current signal into a voltage signal recognizable by the DSP. After A/D conversion, the DSP calculates the actual temperature. If the converted temperature indicates the IGBT temperature exceeds 35°C, a D/A speed control signal is issued to start the fan for cooling the traction converter module.

The fan’ s on/off status is fed back to the DSP main control board through DI feedback signals.

An over-temperature fault of the traction converter module occurs when the system detects the module temperature exceeding 85°C, causing the module to immediately block pulses. The system can only restart after the fault is recovered. Based on the temperature control module system structure shown in [Figure 3: see original paper], a system fault tree is established, and the fault tree events for traction converter module over-temperature fault are listed in .

The fault tree is constructed as shown in [Figure 4: see original paper]. Using ite structure conversion from the bottom layer upward, the BDD corresponding to the fault tree is obtained as shown in [Figure 5: see original paper].

According to the properties of BDD, the converted BDD reflects all failure modes of the system. Searching from the root node downward through all “1” terminal points yields the disjoint expression of the fault tree structure function for traction converter module over-temperature fault. The minimal cut sets are identified as $\{E1\}$, $\{E2\}$, $\{E3, E4\}$, $\{E5\}$, $\{E6\}$, $\{E7\}$, $\{E8\}$, $\{E9\}$, $\{E10\}$.

From the complete set of minimal cut sets, except for $\{E3, E4\}$, all others are first-order minimal cut sets, indicating that other basic events are weak links in the system—each basic event can individually cause module over-temperature failure. In practical applications, efforts should focus on reducing the occurrence probability of events in first-order cut sets, such as adding insulation and anti-interference protection measures to the conditioning board to reduce its failure probability.

For quantitative analysis, the failure probabilities of system components must be determined through extensive theoretical and experimental data. In this example, lacking reliable data, hypothetical values are used for calculation and verification. Assuming the failure probability of each basic event $E_i (i = 1, 2, \dots, 10)$ is 0.2%, and using the derived expression, the top event probability is:

$$p(T) = p(E1) + p(E1)p(E2) + ... + p(E1)p(E2)p(E3)p(E4)p(E5)p(E6)p(E7)p(E8)p(E9)p(E10) \approx 1.589\%$$

To calculate the structural importance of basic events, the fault tree structure first reveals that $E1, E2, E5, E6, E7, E8, E9, E10$ share identical structural importance, while $E3$ and $E4$ have equal importance. By counting cases satisfying the condition $\Phi(0_i, X) = 0 \rightarrow \Phi(1_i, X) = 1$ for each basic event and applying the structural importance formula, the values can be computed.

The analysis shows that $E3$ changing from 0 to 1 causes system state to change from 0 to 1 in one case, while $E1$ causes this change in three cases. The system state changes induced by $E3$ and $E1$ are detailed in and , respectively.

The calculated structural importance values are:

$$I_1 = I_2 = I_5 = I_6 = I_7 = I_8 = I_9 = I_{10} > I_3 = I_4$$

The results indicate that $E1, E2, E5, E6, E7, E8, E9, E10$ have greater structural importance than $E3$ and $E4$, demonstrating that the former eight basic events contribute more significantly to traction converter module over-temperature failure. Therefore, prioritizing preventive measures for these events will be more effective.

Qualitative analysis of the traction converter module over-temperature fault reveals that the system' s weak links are first-order cut sets. When module over-temperature faults occur, prioritizing these weak links enables rapid fault location, providing important guidance for system maintenance and upgrades. The top event probability calculation predicts system failure frequency. The hypothetical data in this example yields a relatively high probability of 1.589%, suggesting thorough fault screening of all basic events before operation. For instance, strict requirements on wiring selection and installation processes can help avoid external circuit faults. The structural importance results confirm that basic events in first-order cut sets contribute most to system failure, reinforcing that reducing their occurrence probability is crucial for improving system safety and reliability.

6 Conclusion

This paper focuses on the principles of BDD-based fault diagnosis, including fault tree to BDD transformation, top event probability calculation, and basic event structural importance analysis. A case study demonstrates the application process. The results confirm that BDD-based fault tree analysis provides significant guidance for fault diagnosis.

References

[1] Marvin Rausand. System reliability theory: models, statistical methods, and applications[M]. 2nd ed. Translated by Guo Qiang, et al. Beijing: National Defense Industry Press, 2010.

[2] Xu Zhongchang. Supportability Engineering[M]. Beijing: Ordnance Industry Press, 2005.

[3] Akers S B. Binary decision diagrams[J]. IEEE Transactions on Computers, 1978, 27(6): 509-516.

[4] Tao Yongjian, Dong Decun, Ren Peng. Binary decision diagram of fault tree analysis[J]. Railway Computer Application, 2009, 18(9): 4-7.

[5] Zhou H, et al. The study on fault tree analysis method for building height fall accident[J]. Coastal Engineering, 2008.

[6] Xu Hengcheng, Zhang Jianguo. The importance analysis of fault tree based on the binary decision diagram[J]. Mechatronic Engineering, 2003, 19(6): 1-4.

Note: Figure translations are in progress. See original paper for figures.