

文章编号:1005-0523(2010)05-0053-05

基于二元决策图的系统可靠性模块分析方法

涂序跃

(华东交通大学 轨道交通学院,江西 南昌 330013)

摘要:为了克服二元决策图方法分析故障树所存在的不足,提出了基于二元决策图的模块方法分析系统可靠性。采用线性时间算法将故障树分解成相互独立的子故障树,然后用二元决策图方法对子故障树进行分析,递归综合相互独立子故障树的求解结果,得出整个系统的可靠性。基于二元决策图的模块分析方法克服了故障树分析与二元决策图方法存在的不足,适用于大型、复杂系统的可靠性分析。

关键词:二元决策图;模块;可靠性分析;故障树

中图分类号:TB114

文献标识码:A

故障树分析^[1-2](fault tree analysis, FTA)是一种广泛应用的系统可靠性建模方法。传统故障树分析方法随着系统元件数增多,最小割集数会呈指数关系增加,容易产生实践中常见的“组合爆炸”问题。采用二元决策图(Binary Decision Diagram, BDD)方法对故障树进行分析^[3-8],可以简化系统可靠性定性、定量的求解过程。但是, BDD 方法分析故障树亦有不足,即故障树向 BDD 的转化受到基本事件排序的影响^[3-5]。所以,为使 BDD 方法有效分析系统可靠性,需找出最优的基本事件排序方法,使所得 BDD 结构最小。可是,至今为止仍没有一种普遍适用的最优基本事件排序方法,这就给 BDD 方法求解大型、复杂系统的可靠性造成困难。针对 BDD 方法存在的不足,本文提出了一种基于 BDD 的模块方法对系统可靠性进行分析,基本研究思路如下:先把系统故障树分解成相互独立的模块;其次采用 BDD 方法对其进行分析;最后向上递归综合模块的求解结果,得到整个系统的可靠性。

1 基于 BDD 的系统可靠性模块分析方法

1.1 基于二元决策图(BDD)的故障树分析方法

1.1.1 二元决策图

二元决策图本质是布尔逻辑函数的图形表示,即用有向无环的二叉树图来表示布尔逻辑函数^[3](如图1所示)。二元决策图包括终结点和非终结点,结点间通过‘1’分支或‘0’分支连接。二元决策图中每条路径始于根结点,终于终结点,终结点的‘1’状态指系统失效,‘0’状态指系统工作^[10]。

二元决策图采用不变化形式表示系统失效逻辑函数 Top 。图2所示的系统失效逻辑函数为:

$$Top = X_1 + X_2 \cdot X_3 + X_2 \cdot X_4 \quad (1)$$

式中:“+”表示布尔算子 OR, “·”表示布尔算子 AND, $X_1 X_2 X_3 X_4$ 表示基本事件。

遍历图2所示的二元决策图,搜索从根结点到‘1’终结点的所有路径,并保留路径中沿‘1’分支发展的非终结点,则由这些保留的非终结点组成的一个集合就是系统的割集。因此,遍历图2所示的二元决策图

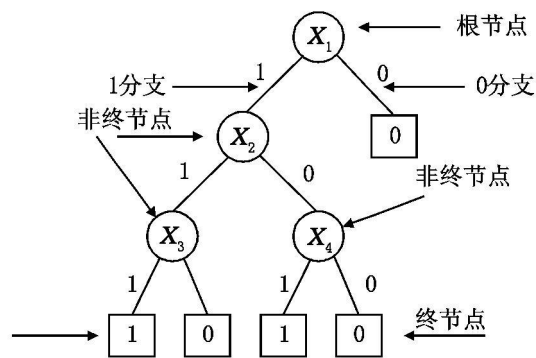


图1 BDD 示例

得到三个割集可得： $\{X_1\}$ 、 $\{X_2, X_3\}$ 、 $\{X_2, X_4\}$ 。

因为二元决策图中所有路径相互排斥，故相应故障树顶事件失效概率表示为

$$Q_{\text{SYS}} = qX_1 + (1 - qX_1)qX_2qX_3 + (1 - qX_1)qX_2(1 - qX_3)qX_4 \tag{2}$$

式中： Q_{SYS} 表示故障树顶事件失效概率； qX_i 表示基本事件 X_i 的失效概率。

1.1.2 故障树向 BDD 的转化方法及基本事件的排序

故障树转化成二元决策图普遍采用 if-then-else (ite) 规则^[4]。该规则由故障树最底层基本事件开始进行 ite 结构转化，消除最底层的逻辑门；然后逐层向上进行 ite 结构转化，直至故障树顶事件，消除故障树所有逻辑门，得到相应的二元决策图。

采用 if-then-else 规则将故障树转化成二元决策图的过程中，必须先确定基本事件的顺序。令 F 、 G 分别表示二元决策图中的两个节点，其中 $F = \text{ite}(X, f_1, f_2)$ ， $G = \text{ite}(Y, g_1, g_2)$ ， X 、 Y 是故障树的基本事件，如果 X 的排序在 Y 的前面，即 $X < Y$ ，则

$$F <_{op} G = \text{ite}(X, f_1 <_{op} G, f_2 <_{op} G) \tag{3}$$

如果 X 与 Y 的排序相同，即 $X = Y$ ，则

$$F <_{op} G = \text{ite}(X, f_1 <_{op} g_1, f_2 <_{op} g_2) \tag{4}$$

式(3)、(4)中， $<_{op}>$ 对应为故障树中门的类型(AND 或 OR)的布尔算子， f_1, f_2, g_1, g_2 分别为故障树的布尔变量。

由 ite 结构转化规则可知，二元决策图的结构大小受基本事件排序影响。不同的基本事件排序，得到不同大小的二元决策图^[9-12]。以非终节点数作为衡量二元决策图结构大小的标准，二元决策图结构越小(非终节点数越少)，则二元决策图技术较之传统故障树分析越具有优越性。在已有研究成果的基础上，本文提出下述 3 条基本事件排序的基本规则^[13-14]；

规则 1 根据从上至下的原则确定基本事件排序，即基本事件在故障树中的层次越高，排序越前；

规则 2 根据基本事件重复次数确定基本事件排序，即基本事件在故障树中重复出现越多，排序越前；

规则 3 故障树中相同层次、且没有重复的基本事件从左至右排序。

1.2 故障树的模块分解

故障树模块就是相互独立的子故障树，是故障树中至少两个底事件的集合，向上可到达同一逻辑门，而且必须通过此逻辑门才能到达故障树顶事件，该逻辑门称为模块输出或模块顶点，故障树的所有其它底事件向上均不能到达该逻辑门^[15]。模块不能有来自故障树其余部分的输入，而且不能有与故障树其余部分重复的事件。

模块分解就是把系统故障树分解成相互独立的模块(故障子树)。本文采用线性时间算法^[16]来确定系统故障树的模块。线性时间算法基本原则叙述如下：对系统故障树进行深度优先，从左至右遍历；假定 V 是一非根结点， t_1, t_2 分别对应结点 V 第一次和第二次的访问时间，如果结点 V 的后代结点不会早于 t_1 被访问，也不会晚于 t_2 被访问，则以结点 V 为顶事件的故障子树是一个模块。

线性时间算法确定系统故障树模块时，需要注意两点：(1) 每个非根结点至少被访问两次，第一次是由其父结点朝下遍历，第二次是由其最右子结点朝上遍历；(2) 以任一非根结点为顶事件的故障子树不会遍历两次^[11]。

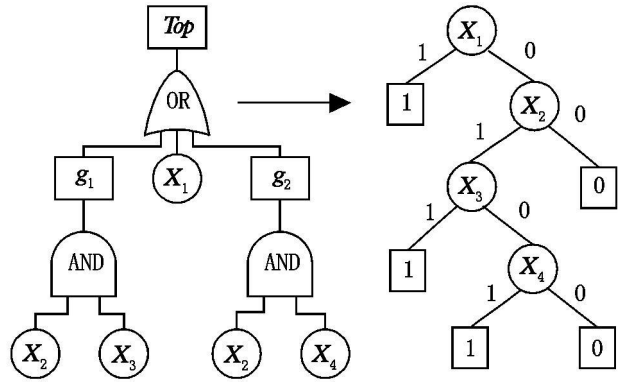


图 2 故障树转化成 BDD 范例
(变量排序： $X_1 < X_2 < X_3 < X_4$)

1.3 基于模块方法的故障树分析

1.3.1 基于模块方法的故障树定性分析

基于模块的故障树定性分析实质是一个自下而上的递归综合求解过程。先分析最底层故障树模块(该模块只包含基本事件),采用 BDD 方法求出该模块的所有最小割集和发生概率;然后将其用基本事件代替,该基本事件故障发生概率等于模块故障发生概率。对最底层模块被取代后的故障树再进行模块分析,分析新故障树最底层模块,同样用基本事件将其代替。分析过程递归进行,直至整个故障树分析完成。

基于模块方法的故障树定性分析依据上述原理,具体步骤为:

(1) 采用 BDD 方法求出故障树所有最底层模块的最小割集,将这些模块用相同故障发生概率的基本事件代替,得到新的故障树;

(2) 采用 BDD 方法求出新故障树所有最底层模块的最小割集,如果这些最小割集中包含代替模块的基本事件,则代替模块的基本事件用其最小割集代替,得到新故障树最底层模块的所有最小割集,并且其中只包含原故障树的基本事件;

(3) 递归向上执行相同分析过程,直至求出原故障树的所有最小割集,这些最小割集中只包含原故障树的基本事件。

1.3.2 基于模块方法的故障树定量分析

基于模块方法的故障树定量分析与定性分析原理相同,采用 BDD 方法求出故障树最底层模块的故障发生概率,用具有相同故障发生概率的基本事件代替最底层模块,得到新的故障树。对新故障树进行相同分析,求出新故障树的最底层模块,用具有相同故障发生概率的基本事件代替最底层模块,递归重复该过程,直至整个故障树分析完成。

基于模块方法的故障树定量分析具体步骤为:

(1) 采用 BDD 方法求出故障树所有最底层模块的故障发生概率,将这些模块用相同故障发生概率的基本事件代替,得到新的故障树;

(2) 采用 BDD 求出新故障树所有最底层模块的故障发生概率,将这些模块用相同故障发生概率的基本事件代替,得到新的故障树;

(3) 递归向上执行相同分析过程,直至求出原故障树顶事件发生概率。

2 分析实例

下面以文献[9]中所示故障树为例,详细阐述基于二元决策图的系统可靠性模块分析方法。图3是来自文献[9]的故障树范例,采用线性时间算法^[16]对其进行模块分解,模块分解结果如图4所示。整个故障树作为一个最大的模块 M ,其中又包括两个相互独立的子模块 M_1 、 M_2 。

采用二元决策图方法对所有模块进行可靠性分析,故障树基本事件排序为: $e_7 < e_{11} < e_1 < e_{15} < e_0 < e_8 < e_4 < e_3 < e_{12} < e_{10}$,模块 M_1 、 M_2 、 M 对应的 BDD 如图5所示。先求出模块 M_1 、 M_2 顶事件失效概率,用相同概率的同名基本事件代替构成新的故障树,并转化成相应的 BDD。

由式(2)可得:

$$Q_{M_1} = q_{e_{15}} q_{e_0} + q_{e_{15}} (1 - q_{e_0}) q_{e_8} \quad (5)$$

$$Q_{M_2} = q_{e_4} q_{e_3} q_{e_{12}} q_{e_{10}} \quad (6)$$

用相同概率且同名的基本事件代替模块 M_1 、 M_2 ,得出:

$$Q_M = q_{e_7} + (1 - q_{e_7}) q_{e_{11}} + (1 - q_{e_7}) (1 - q_{e_{11}}) q_{e_1} + (1 - q_{e_7}) (1 - q_{e_{11}}) (1 - q_{e_1}) Q_{M_1} + (1 - q_{e_7}) (1 - q_{e_{11}}) (1 - q_{e_1}) (1 - Q_{M_1}) Q_{M_2} \quad (7)$$

所得 Q_M 即为故障树顶事件失效概率。

模块分析方法求解系统故障树最小割集如表1所示。模块 M ,即系统故障树的最小割集为: $\{M_1\}$ 、 $\{M_2\}$ 、 $\{e_7\}$ 、 $\{e_{11}\}$ 、 $\{e_1\}$ 。 $\{M_1\}$ 、 $\{M_2\}$ 分别用其子模块的最小割集代替,得到系统故障树所有最小割集为

$\{e_{15}, e_{10}\}, \{e_{15}, e_8\}, \{e_4, e_3, e_{12}, e_{10}\}, \{e_7\}, \{e_{11}\}, \{e_1\}$ 。

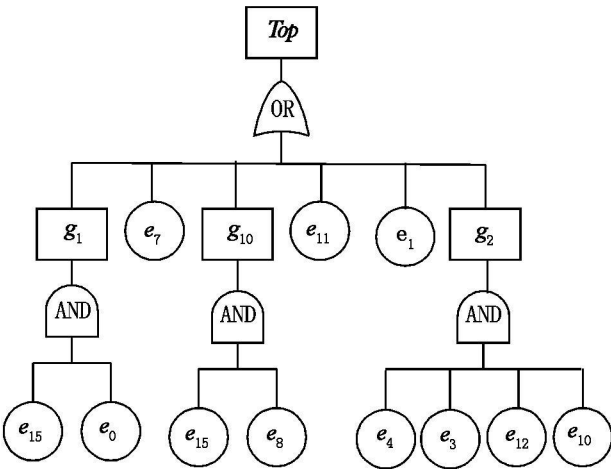


图 3 故障树范例

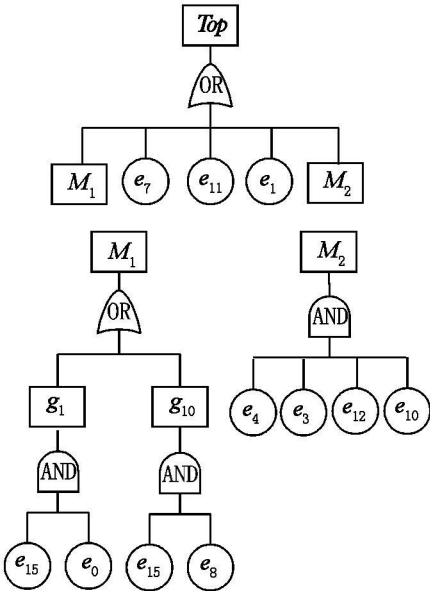


图 4 故障树模块分解

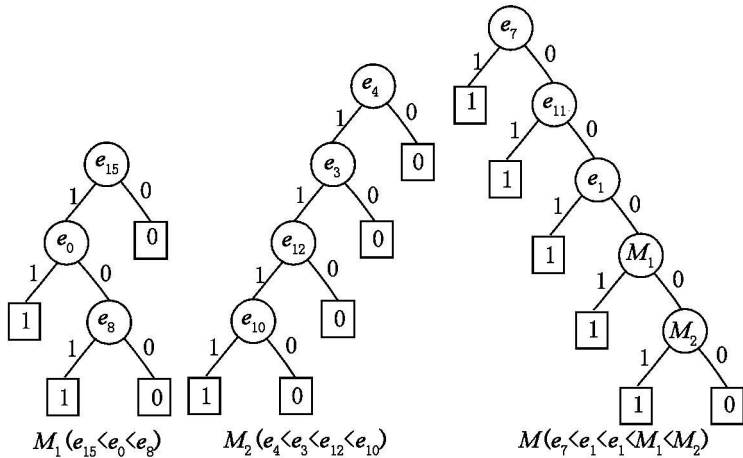


图 5 模块对应的二元决策图(基本事件排序: $e_7 < e_{11} < e_{15} < e_8 < e_4 < e_3 < e_{12} < e_{10}$)

表 1 故障树最小割集列表

$\{M_1\}$		$\{M_2\}$	$\{e_7\}$	$\{e_{11}\}$	$\{e_1\}$
$\{e_{15}, e_0\}$	$\{e_{15}, e_8\}$	$\{e_4, e_3, e_{12}, e_{10}\}$	$\{e_7\}$	$\{e_{11}\}$	$\{e_1\}$

基于 BDD 的系统可靠性模块分析实例表明:该方法在定量分析中,求解系统故障树顶事件失效概率更简单;定性分析过程中,仅得到所有最小割集,没有任何非最小割集,免除了 BDD 方法分析整个故障树易产生非最小割集问题。

3 结论

基于 BDD 的系统可靠性模块分析方法,采用线性时间算法将系统故障树分解成相互独立的模块,分别求出各模块顶事件发生概率,然后递归综合求解出系统故障树顶事件的失效概率。模块分析方法在定性、定量分析大型、复杂系统的可靠性过程中,避免了最小割集不交化求和的繁琐过程,又可简单、明了的直接得出所有最小割集。相比传统故障树分析方法和采用 BDD 方法直接分析整个系统故障树,模块方法

具有明显优越性,适用于大型、复杂系统可靠性分析。

参考文献:

- [1] 曾声奎. 系统可靠性设计分析教程[M]. 北京:北京航空航天大学出版社, 2001:117-144.
- [2] 李海泉, 李刚. 系统可靠性分析与设计[M]. 北京:科学出版社, 2003:136-172.
- [3] AKERS S B. Binary decision diagrams, IEEE Transaction on Computers[J]. 1978, 27(2):509-516.
- [4] RAUZY A. New Algorithms for Fault Trees Analysis[J]. Reliability Engineering and System Safety, 1993, 30(32):40:203-211.
- [5] SINNAMON R M, ANDREWS J D. Fault tree analysis and binary decision diagrams[C]. Proceedings Annual Reliability and Maintainability Symposium, 1996:215-222.
- [6] TOWHIDI F, LASHKARI A H, HOSSEINI R S. Binary Decision Diagram(BDD)[C]. International Conference on Future Computer and Communication, 2009:496-499.
- [7] CAI Yao, LIU Zhengjiang, WU Zhaolin. Improvement of Fault Tree Analysis in Formal Safety Assessment Using Binary Decision Diagram [C]. 1st International Conference on Information Science and Engineering, 2009:4 330-4 333.
- [8] MO Yuchang. New Insights Into the BDD-Based Reliability Analysis of Phased-Mission Systems[J]. IEEE Transactions on Reliability, 2009, 58(4):667-678.
- [9] DU Suguo, SUN Yan. A Novel Ordering Method of Binary Decision Diagram[C]. 2007 International Conference on Management Science & Engineering(14th), 2007:299-304.
- [10] 陶勇剑, 董德存, 任鹏. 故障树分析的三元决策图方法[J]. 铁路计算机应用, 2009, 18(9):4-7.
- [11] MO Yuchang. Variable Ordering to Improve BDD Analysis of Phased-Mission Systems With Multimode Failures[J]. IEEE Transactions on Reliability, 2009, 58(1):53-57.
- [12] EBENDT R, DRECHSLER R. Approximate BDD Minimization by Weighted A[C]. IEEE International Symposium on Circuits and Systems, 2009:2 974-2 977.
- [13] MOEINZADEH H, MOHAMMADI M, Mehrbakhsh A. Evolutionary-Reduced Ordered Binary Decision Diagram [C]. Third Asia International Conference on Modelling & Simulation, 2009:142-145.
- [14] BARTLETT L M, ANDREWS J D. An ordering heuristic to develop the binary decision diagram based on structure importance[J]. Reliability Engineering and System Safety, 2001, 72(3):31-38.
- [15] GULATI R, DUGAN J B. A modular approach for analyzing static and dynamic fault trees[C]. Proceedings Annual Reliability and Maintainability Symposium, 1997:57-63.
- [16] DUTUIT Y, RAUZY A. A linear-time algorithm to find modules of fault trees[J]. IEEE Transactions on Reliability, 1996, 45(3):422-425.

A Modular Analysis Method for System Reliability Based on Binary Decision Diagrams

Tu Xuyue

(School of Railway Transportation, East China Jiaotong University, Nanchang 330013, China)

Abstract: In order to overcome the deficiencies of Binary Decision Diagram, the modular method, based on Binary Decision Diagrams, is introduced to analyze system reliability. Fault trees are divided into independent subtrees using a very efficient linear time algorithm, then Binary Decision Diagram is applied to each subtree for reliability analysis. The solutions of various independent subtrees are integrated to obtain system reliability. The modular method, based on Binary Decision Diagrams, can overcome main deficiencies of fault tree analysis and Binary Decision Diagram, and is useful for assessing the reliability of large and complex systems.

Key words: binary decision diagram; modular; reliability analysis; fault tree