国 防 科 技 大 学 学 报
第21卷 第2期 JOURNAL OF NATIONAL UNIVERSITY OF DEFENSE TECHNOLOGY Vol.21 No.2 1999

非单调关联故障树顶事件失效概率的新算法*

周经伦 刘 波 孙 权

(国防科技大学系统工程与数学系 长沙 410073)

摘 要 提出了一种计算非单调关联故障树顶事件失效概率的新方法，其基本思想是利用BDD进行故障树中顶事件的不交化工作，可避免求PIS和PIS的最小覆盖的运算过程，为有效解决其运算的NP困难问题提供了一条新的途径。

关键词 非单调关联故障树，顶事件失效概率，BDD，FTA

分类号 TB114

A New Algorithm for Calculating Top-Event Failure Probability of Non-coherent Fault Tree

Zhou Jinglun LiuBo Sun Quan

(Department of System Engineering and Mathematics, NUDT, Changsha, 410073)

Abstract Calculating Top-Event failure probability is an important aspect of non-coherent FTA (Fault Tree Analysis). Because of the complexity of non-coherent system, all existing algorithms can't satisfy the requirements of exactness and speediness in application. In this article, we adopt a new algorithm for calculating Top-Event failure probability of non-coherent fault tree, which is based on BDD to decompose Top-Event of the fault tree and avoid seeking PIS and the smallest cover of PIS. So the new algorithm provides a new method for solving the NP problem effectively.

Key words non-coherent fault tree, top-event failure probability, BDD, FTA

非单调关联故障树的FTA (Fault Tree Analysis) 包括定性分析与定量分析两方面的研究内容。定性分析主要是寻找反映系统失效模式的PIS (Prime Implicants Sets)，根据PIS可采取相应措施以改善系统性能，达到提高系统可靠性的目的。定量分析有两方面的主要内容：一是求顶事件的失效概率，二是各种部件重要度的计算。顶事件失效概率在实际工程应用中有着重要的意义，精确、快速计算顶事件失效概率成为定量分析研究中的一个重要内容。

故障树顶事件失效概率的计算方法有两种：一种是利用容斥原理直接来计算；一种是先对故障树结构函数进行不交化，然后再利用互斥事件和的概率公式来计算。对于容斥原理的计算方法，由于它的效率低，我们一般不用。不交化方法的运算量是与故障树结构函数中积项的个数相关的，若故障树比较复杂，不交化的工作量也是巨大的。对于非单调关联故障树，为减少运算量，常用的方法是从故障树结构函数出发，先求出全部的PIS，再求出PIS的最小覆盖，得到故障树结构函数的最简表达式，再采用不交型布尔代数的有关知识来进行不交化工作。但困难在于，无论是求PIS还是求PIS的最小覆盖，都缺乏有效的算法，譬如求PIS的对合运算、Nelson原理(双取补)、双取对偶[1]等算法，本身运算效率就低，使得这种方法不能用来处理大规模的故障树。此外，曾有人提出了早期不交化[1]、Sharp算法[2]、矩阵法[3]等方法，这些方法同前一方法一样，都存在着运算效率低的问题，都不能用来处理大型复杂系统的故障树。

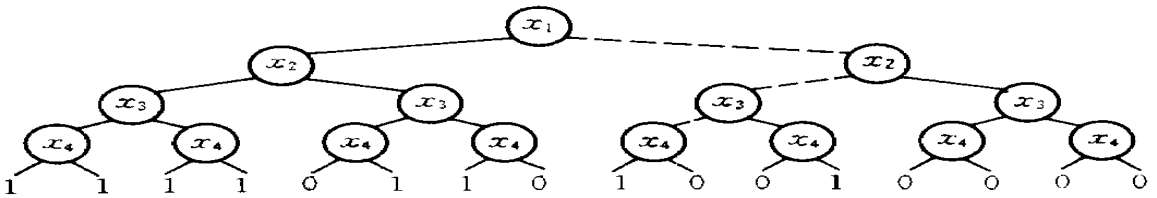
BDD (Binary Decision Diagrams) 是近几年发展起来的用于FTA研究的一种新方法，它为解决FTA研究中的NP问题提出了一条新思路。BDD方法与FTA的结合，已成功解决了单调关联故障树的定性与定量分析问题。本文利用BDD方法，提出了一种计算非单调关联故障树顶事件失效概率的新

* 1998年9月14日收稿
第一作者：周经伦，男，1955年出生，副教授

算法，可有效地提高处理大型复杂故障树时的运算效率。

1 Shannon 分解与 BDD

为介绍新算法，我们先引入 *Shannon* 分解与 *BDD* 的概念。



（图中虚线表示 x_1，x_2，x_3，x_4 的取值分别为0，1，1，0）

图1 函数 $f(x_1, x_2, x_3, x_4) = x_1x_2 + x_1(x_3\bar{x}_4 + \bar{x}_3x_4) + x_2(x_3x_4 + \bar{x}_3\bar{x}_4)$ 的 Shannon 分解树

Fig. 1 Shannon Tree of $f(x_1, x_2, x_3, x_4) = x_1x_2 + x_1(x_3\bar{x}_4 + \bar{x}_3x_4) + x_2(x_3x_4 + \bar{x}_3\bar{x}_4)$

1.1 *Shannon* 分解

Shannon 分解定理　设 $f(x_1, x_2, \cdots, x_n)$ 是一布尔函数，x_i $(i = 1, 2, \cdots, n)$ 是 f 的任一自变量，令 $f_{x_i} = f(x_1, x_2, \cdots, x_{i-1}, 1, x_{i+1}, \cdots, x_n)$，$f_{\bar{x}_i} = f(x_1, x_2, , x_{i-1}, 0, x_{i+1}, , x_n)$，则布尔函数 f 可分解为 $f(x_1, x_2, \cdots, x_n) = x_i f_{x_i} + \bar{x}_i f_{\bar{x}_i}$.

由于 f_{x_i} 和 $f_{\bar{x}_i}$ 仍然是布尔函数，我们可选取别的变量，继续利用 Shannon 分解定理对其进行分解，把这种过程一直进行到不能再进行为止，就实现了对原函数的不交化工作。从这里易看出这种分解过程的运算量是呈指数级增长的。若固定了 Shannon 分解中变量的顺序再进行分解，并且把最终分解结果用图形表示出来，可得到一棵唯一的二元树，我们称之为 Shannon 分解树，它有 $2^n - 1$ 个结点，叶结点都为0或1。例如，函数

$$f(x_1, x_2, x_3, x_4) = x_1x_2 + x_1(x_3\bar{x}_4 + \bar{x}_3x_4) + x_2(x_3x_4 + \bar{x}_3\bar{x}_4)$$

的 Shannon 分解树如图1所示。

1.2 二元决策图 BDD

二元决策图 BDD 最早是由 Sheldon B Akers 提出的[6]，它实质上是布尔函数的 Shannon 分解树的一种简化表示。文献[5] 指出 BDD 可由 Shannon 分解树经过一定的简化步骤得到，并用图例显示了它们之间的关系。图2为一 BDD 图，它和图1中的 Shannon 分解树表示的是同一函数。

无论是 Shannon 树还是 BDD 图，都是布尔函数不交化的图形表示，不同点在于 BDD 图所包含的结点数更少，图形更简洁明了。根据 BDD 图，我们就可以直接写出相应的布尔函数的不交化表达式，方法是回溯所有叶结点为1的路径，每一条路径代表一个蕴含集，全部蕴含集的积之和式即为所求的表达式。例如，由图2可写出相应的函数的不交化表达式为

$$f = x_1x_2 + x_1\bar{x}_2\bar{x}_3\bar{x}_4 + x_1\bar{x}_2x_3\bar{x}_4 + x_1x_2\bar{x}_3x_4 + \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4$$

BDD 除了能图形化表示布尔函数的不交化结果外，它更最重要的一个特性是能利用自身的结构特点，快速完成布尔函数中的“与”、“或”等逻辑运算。

2 算法

BDD 是布尔函数的不交化图形表示，而用于非单调关联系统可靠性分析的故障树，其结构函数表示的正是其顶事件与底事件之间的逻辑关系，它实质上是一个顶事件的布尔表达式，这使得我们可以运用 BDD 来进行故障树结构函数的不交化工作。

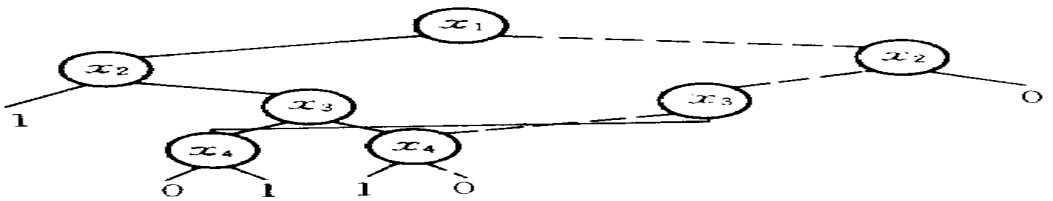


图2 函数 $f\ (x_1,\ x_2,\ x_3.\ x_4) = x_1x_2 + x_1\ (x_3\bar{x}_4 + \bar{x}_3x_4) + x_2\ (x_3x_4 + \bar{x}_3\bar{x}_4)$ 的 BDD 图

Fig. 2 BDD of $f\ (x_1,\ x_2,\ x_3.\ x_4) = x_1x_2 + x_1\ (x_3\bar{x}_4 + \bar{x}_3x_4) + x_2\ (x_3x_4 + \bar{x}_3\bar{x}_4)$

综上所述，我们提出如下计算非单调关联故障树顶事件失效概率算法:

第一步: 将故障树转化为 BDD。

随着近几年对 BDD 的深入研究, 将故障树转化为 BDD 已不是很困难的事, 常用的方法有递归法、模板法、减支法等, 这些方法已经可以用来处理大型复杂系统的故障树。文献 [4] 详细论述了这一步的转化过程, 它不但给出了 BDD 的一些基本概念, 而且给出了递归法的转化方法。运用这一方法, 能迅速完成从故障树到 BDD 的转化。

第二步: 回溯 BDD 中所有叶结点为1的路径, 写出相应函数的不交化表达式。

这一步可以采取深度优先的方式, 从根结点开始向下搜索, 每经过一个非叶结点 x_i (i= 1, 2, …, n) (n 为故障树中底事件的个数), 往左分支记为该结点的正事件, 用 x_i 表示; 往右分支记为该结点的逆事件, 用 $\bar{x}_i$ 表示, 记下所有叶结点为1的路径, 这些路径我们用 p_i (i= 1, 2, …, m) (m 为路径数) 表示, 其中 $p_i = \prod_{j=1}^{n_i} x_{i_j}^*$, $x_{i_j}^* \in \{x_{i_j},\ \bar{x}_{i_j}\}$, $x_{i_j} \in \{x_1,\ x_2,\ \cdots,\ x_n\}$, ($n_i$ 为路径 i 所含的结点数), 则故障树结构函数的不交化表达式为

$$T = \sum_{i=1}^{m} p_i = \sum_{i=1}^{m} \prod_{j=1}^{n_i} x_{i_j}^*$$

第三步: 利用互斥事件和的概率公式计算顶事件的失效概率。

若我们把每一个蕴含集 p_i 看作一个事件, 则相当于把顶事件化为了一簇互斥事件的和: $T = \bigcup_{i=1}^{m} p_i$, 至此, 可直接利用概率公式

$$P(T) = \sum_{i=1}^{m} P(p_i) = \sum_{i=1}^{m} \prod_{j=1}^{n_i} P(x_{i_j}^*)$$

进行计算, 其中 $P(\bar{x}_{i_j}) = 1 - P\ (x_{i_j})$, $P\ (x_{i_j})$ 表示底事件 x_{i_j} 的失效概率。

3 实例

设图3中的故障树中的底事件的失效概率都为0. 1, 求顶事件的失效概率。

第一步: 将故障树转化为相应的 BDD, 如图4所示。

第二步: 根据 BDD 图写出故障树结构函数的不交化表达为:

$$f = x_1x_2 + x_1\bar{x}_2\bar{x}_3x_4 + \bar{x}_1x_3 + \bar{x}_1\bar{x}_3x_4$$

第三步: 利用互斥事件的概率公式来计算顶事件的失效概率:

$$\begin{aligned} P(T) &= P(x_1x_2) + P(x_1\bar{x}_2\bar{x}_3x_4) + P(\bar{x}_1x_3) + P(\bar{x}_1\bar{x}_3x_4) \\ &= P(x_1)P(x_2) + P(x_1)\,\bar{P}(x_2)\bar{P}(x_3)P(x_4) + \bar{P}(x_1)P(x_3) + \bar{P}(x_1)\bar{P}(x_3)P(x_4) \\ &= 0.1 \times 0.1 + 0.1 \times (1 - 0.1) \times (1 - 0.1) \times 0.1 + \\ &\quad (1 - 0.1) \times 0.1 + (1 - 0.1) \times (1 - 0.1) \times 0.1 \\ &= 0.1891 \end{aligned}$$

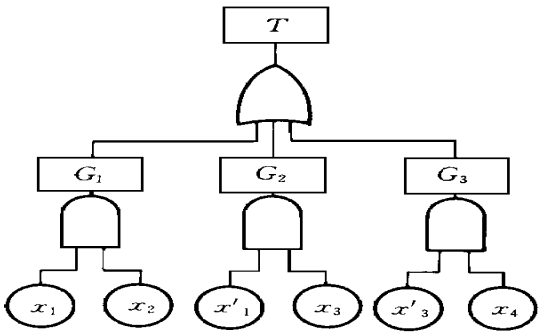


图3 实例分析所用的故障树

Fig. 3 A non- coherent fault tree

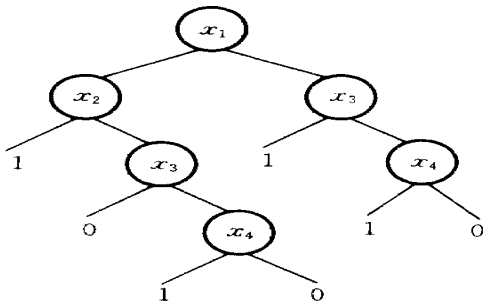


图4 所示故障树的 BDD 图

Fig. 4 BDD of the fault tree of Figure 3

4 结论

本文提出的运用BDD 方法来计算非单调关联故障树顶事件失效概率的算法, 避免了求非单调关联系统的 PIS 和 PIS 的最小覆盖, 经过实验测试, 其运算速度比传统算法快近一个数量级。所存在的问题是: 算法的运算量与得到的故障树结构函数的不交化表达式有直接关系, 而故障树结构函数的不交化表达式的繁简与 BDD 又有着直接的关系。但对于同一故障树, 在将其转化为BDD 时, 若采用不同的变量指标值, 得到的 BDD 也不相同, 有时甚至有很大的差别。因此, 有必要研究如何选取合适的变量指标值, 以缩小 BDD 的规模, 加快算法的运算速度。

参考文献

1 梅启智, 廖炯生, 孙惠中. 系统可靠性工程基础. 北京: 科学出版社, 1987

2 史定华, 王松瑞. 故障树分析技术方法和理论. 北京: 北京师范大学出版社, 1993

3 方逵, 罗强. 非单调关联系统 FTA 定量分析的新方法. 模糊系统与数学, 1997, 11 (3)

4 周经伦, 孙权. 一种故障树分析的新算法. 模糊系统与数学, 1997, 11 (3)

5 Coudert O, Madre J. Fault Tree Analysis: 10^{20} Prime Implicants and Beyond, Proceedings Annual Reliability and Maintainability Symposium, 1993

6 Akers S B. Bianry Decision Diagrams. IEEE Trans. On Comput., 6, C- 27, 1978