

基于多值决策图的动态故障树分析方法

王 斌 吴丹丹 莫毓昌 陈中育

(浙江师范大学数理与信息工程学院 金华 321004)

摘 要 针对具有动态故障模式的复杂系统,动态故障树分析一直是很重要的可靠性分析技术。为了提升可靠性分析效率,已有研究提出了各种模块化方法,但是对于实际动态故障树模型中由于事件关联导致的大型动态子树,这些模块化方法的状态空间爆炸问题仍然很突出。因此介绍了一种基于多值决策图(Multiple-valued Decision Diagrams, MDD)来分析动态故障树的方法,通过多值变量编码动态门,利用单一系统 MDD 模型刻画各种动态和静态可靠性行为,有效地缓解了状态爆炸问题。通过一个具体的实例说明了多值决策图方法的应用和优势。

关键词 多值决策图,动态故障树,马尔科夫,可靠性

中图法分类号 TP309 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.10.013

Dynamic Fault Tree Analysis Based on Multiple-valued Decision Diagrams

WANG Bin WU Dan-dan MO Yu-chang CHEN Zhong-yu

(College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, Jinhua 321004, China)

Abstract In the system reliability analysis, dynamic fault trees analysis has been used as a very important technique for many years. However, when these kinds of large dynamic subtrees appear, which abound in models of real-world dynamic software and embedded computing systems, the state explosion problem is too serious to be removed. In order to improve computing efficiency, this paper introduced an efficient, multiple-valued decision-diagram (MDD)-based DFT analysis approach. This approach restricts the state-space methods only to the subtree components associated with dynamic failure behaviors. By using multiple-valued variables to encode the dynamic gates, a single compact MDD is then generated. Finally, the failure probability is calculated to describe the reliability of the system. Applications and advantages of the proposed approach are illustrated through detailed analysis of a practical case study.

Keywords Multiple-valued decision diagram (MDD), Dynamic fault tree (DFT), Markov, Reliability

1 引言

随着信息时代的进步,科学技术尤其是电子技术、计算机技术的迅猛发展,实际应用对系统及设备性能的要求也越来越高。尤其是高水平冗余和动态冗余的应用增多,其结构也变得越来越多样化、复杂化,导致系统呈现出多种失效模式,给系统可靠性分析带来了新的问题。

自 20 世纪 60 年代,贝尔电话实验室采用故障树分析方法^[1,2]对民兵导弹系统进行可靠性分析,再到后来的“拉姆森报告”,故障树分析方法获得了迅速的发展,成为可靠性分析^[3]中相当重要的技术。故障树^[1-4]是一种逻辑因果关系图,它根据部件状态(基本事件)来描述系统(顶事件)状态。它是将系统故障模式图形化的方法,直观形象,逻辑清晰,在定性定量分析中都可以得到较好的应用。

早期的静态故障树^[5-7]采用与门、或门、表决门等,而它早已不适用于具有动态随机性故障和相关性的软件和嵌入式系统^[4],如容错系统、动态冗余、备件、具有顺序相关性的系统的可靠性分析。各种系统的复杂性越来越高,单纯地采用马尔

科夫方法伴随的状态空间爆炸问题也凸显出来。仅仅运用包括与门、或门、表决门等静态逻辑门的传统静态故障树模型已经远远不能满足现今对于计算机应用系统的分析建模的要求。动态故障树 DFT^[5,6,12]在静态故障树的基础上引入表征动态特性的新的逻辑门^[7]类型。

多值决策图^[8,9,13] (Multiple-valued Decision Diagram, MDD)是一种有向无环图,能反映部件状态和系统状态之间的关系。本文利用多值决策图来进行动态故障树的分析,其中对于动态失效行为相关的子树部件,采用状态空间方法来分析;并引入多值变量对动态门进行编码,生成一个简单明了的系统 MDD;综合运用组合方法和状态空间方法一定程度上解决了状态空间的爆炸问题。传统模块化^[10,14]方法中,大型动态子树需要用状态空间方法来分析,与此不同的是,本文提出的利用多值决策图来分析动态故障树的方法将马尔科夫模型应用在部件或者动态门这一层次,有助于减轻传统方法中的状态空间爆炸问题。本文对这种方法进行了详细的描述,并将其应用到实例中。

本文第 2 节简单介绍了基于 MDD 的 DFT 建模^[8-13]的

到稿日期:2015-09-10 返修日期:2016-01-05 本文受国家自然科学基金面上项目(61272007,61572442)资助。

王 斌 硕士生,主要研究方向为网络可靠性;吴丹丹 硕士生,主要研究方向为网络可靠性;莫毓昌 博士,副教授,主要研究方向为网络可靠性;陈中育 博士,教授,主要研究方向为规范化软件开发技术、需求建模。

基本知识,包含几种动态门及其编码、变量排序、MDD生成及评估;第3节通过具体实例对MDD分析DFT方法进行了形象描述,并提出了该方法的优点;最后总结全文,并提出未来研究的方向。

2 基于MDD的DFT分析

2.1 动态门

常用的典型动态逻辑门^[14-17]有3种:功能相关门(FDEP)、备件门(SP)、优先与门(PAND)。

在系统的功能相关门中,如果某个部件发生故障会导致与其相关的部件发生故障,称其为激发事件。如图1所示, T 为激发输入(可以为基本事件或其它门的输出事件), $X_1, X_2, \dots, X_n$ 为 n 个状态独立的基本事件。在功能相关门中,状态相关的基本事件是功能相关的,这表示了部件间一种相互关系,并没有实际输出。当激发事件发生时,所有的相关基本事件 $X_1, X_2, \dots, X_n$ 随之发生,但是任何单个基本事件的故障并不会对激发事件产生影响。

当系统具有备件时,此类故障模式无法在同一个事件框架下用基本事件的组合来表达,因此静态故障树技术无法精确地建模。如图2所示的SP门, P 为主要输入, $S_1, S_2, \dots, S_n$ 是 n 个备件可选择输入,所有输入事件都是基本事件。初始情况下,主要输入 P 为工作状态,备件输入为空闲状态。当所有输入事件发生后,SP门有输出,且输出为真。

当系统的故障模式要求事件必须以特定顺序发生时,需要用优先与门来表征。如图3所示的PAND门, $X_1, X_2, \dots, X_n$ 为 n 个输入。当它们以特定顺序发生时,门的输出就为真。

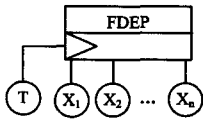


图1 FDEP门

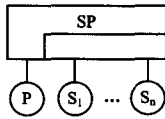


图2 SP门

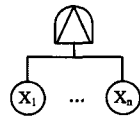


图3 PAND门

2.2 动态门编码

运用动态门对动态失效行为进行编码,有如下方案。

(1)FDEP门编码:FDEP门的激发事件为 T , n 个状态相关的部件 $X_1, X_2, \dots, X_n$ 的状态空间有 2^n+1 个状态。设 $t, x_1, x_2, \dots, x_n$ 表示布尔状态指示符变量, $t=1$ (或者 $x_i=1$)表示部件 T (或者 X_i)在任务完成前失效;否则, $t=0$ ($x_i=0$)。系统实现的状态矢量 $X=\{t, x_1, x_2, \dots, x_n\}$ 有多种情况: $(0, 0, 0, \dots, 0), (0, 1, 0, \dots, 0), (0, 0, 1, \dots, 0), (0, 1, 1, \dots, 0), \dots, (0, 1, 1, \dots, 1)$ 和 $(1, 1, 1, \dots, 1)$ 。

系统状态 $(0, 0, 0, \dots, 0)$ 代表没有部件失效;

系统状态 $(0, 1, 0, \dots, 0)$ 代表只有部件 X_1 失效;

系统状态 $(0, 1, 1, \dots, 1)$ 代表所有状态相关部件 $X_1, X_2, \dots, X_n$ 均失效。

FDEP门描述的动态失效行为可以通过一个由FDEP描述的变量来表示,它共有 2^n+1 个值。如图4所示,系统MDD模型中,每一个和FDEP相关的非汇节点有 2^n+1 个输出边; 0 -边对应系统状态 $(0, 0, 0, \dots, 0)$; j 边($0 < j < 2^n, j$ 有 $a_n a_{n-1} \dots a_2 a_1$ 的二进制形式)对应系统状态 $(0, a_1, a_2, \dots, a_{n-1}, a_n)$,如果 $a_i=1$,那么部件 X_i 失效; 2^n -边对应系统状态 $(1, 1, \dots, 1)$ 。

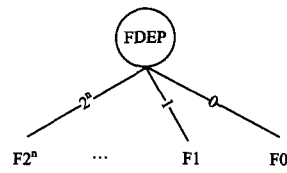


图4 FDEP相关的非汇MDD节点

(2)SP门编码:SP门包含主要部件 P 和 n 个备件 $S_1, S_2, \dots, S_n$,共有 $n+2$ 个状态。设 $p, s_1, s_2, \dots, s_n$ 为布尔状态指示符变量。 $p=1$ (或者 $s_i=1$)代表部件 P (或者 S_i)在任务完成前失效;否则, $p=0$ (或者 $s_i=0$)。系统实现的状态矢量 $X=\{p, s_1, s_2, \dots, s_n\}$ 有多种情况: $(0, 0, \dots, 0, 0), (1, 0, \dots, 0, 0), (1, 1, \dots, 0, 0), (1, 1, \dots, 1, 0), \dots, (1, 1, \dots, 1, 1)$ 。

系统状态 $(0, 0, \dots, 0, 0)$ 表示主要部件 P 处于工作状态,而备件处于空闲状态。

系统状态 $(1, 0, \dots, 0, 0)$ 表示当主要部件 P 失效,第一个可替换备件 S_1 立刻开始工作。

系统状态 $(1, 1, \dots, 1, 1)$ 表示所有备件 $S_1, S_2, \dots, S_n$ 都用完,即它们在一个特定时序中失效。

SP门描述的动态失效行为可由SP门描述的变量来表示,它共有 $n+2$ 个值。如图5所示,在系统MDD模型中,每个与SP相关的非汇节点具有 $n+2$ 个输出边; 0 -边对应系统状态 $(0, 0, \dots, 0, 0)$; 1 -边代表的是 P 失效,对应系统状态 $(1, 0, \dots, 0, 0)$; j 边($2 \leq j \leq n+1$)代表 P 和 $S_1, S_2, \dots, S_{j-1}$ 失效,对应系统状态 $(p, s_1, s_2, \dots, s_n)$,其中 $p, s_1, s_2, \dots, s_{j-1}$ 假设为1,其它元素假设为0。

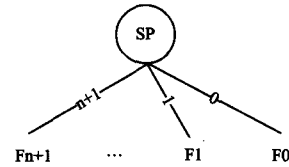


图5 SP门相关的非汇MDD节点

(3)PAND门编码:有 n 个状态相关部件 $X_1, X_2, \dots, X_n$ 的PAND门,共有 2^n+1 个状态。设 $t, x_1, x_2, \dots, x_n$ 为布尔状态指示符变量,其中 $x_i=1$ 表示在任务完成前部件 x_i 失效;否则, $x_i=0$ 。系统实现的状态矢量 $X=\{x_1, x_2, \dots, x_n\}$ 有多种情况: $(0, 0, \dots, 0, 0), (0, 0, \dots, 0, 1), (0, 0, \dots, 1, 0), (0, 0, \dots, 1, 1), \dots, (0, 1, \dots, 1, 1), (1, 1, \dots, 1, 1)$ 和 $(1, 1, \dots, 1, 1)^*$ 。

系统状态 $(0, 0, \dots, 0, 0)$ 代表没有发生状态相关部件的失效。

系统状态 $(0, 0, \dots, 0, 1)$ 代表只有一个部件 X_1 失效。

系统状态 $(1, 1, \dots, 1, 1)^*$ 代表所有状态相关部件 $X_1, X_2, \dots, X_n$ 均失效,但符合特定的失效顺序。

系统状态 $(1, 1, \dots, 1, 1)^*$ 代表所有状态相关部件 $X_1, X_2, \dots, X_n$ 均失效,且按照特定的顺序失效。

利用以上优先与门描述的动态失效行为,可以通过由PAND表示的变量来表示,它具有 2^n+1 个值。如图6所示,在系统MDD模型中,每个与PAND相关的非汇节点具有 2^n+1 个输出边; 0 -边对应系统状态 $(0, 0, \dots, 0, 0)$; j 边($0 < j < 2^n, j$ 有 $a_n a_{n-1} \dots a_2 a_1$ 的二进制形式)对应系统状态 $(0, a_1, a_2, \dots, a_{n-1}, a_n)$,代表如果 $a_i=1$,则部件 X_i 失效; 2^n-1 -边对应系统状

```

graph TD
    PAND((PAND)) --- F2n[F2n]
    PAND --- Ellipsis[...]
    PAND --- F1_slash[F1/]
    F1_slash --- F0[F0]
    F1_slash --- F1[F1]
  
```

2.3 变量排序

在构建系统 MDD 时,类似于静态故障树 BDD 的生成,需要首先对所有动态门和常规失效事件相关的变量进行排序。启发式排序很大程度上决定了 MDD 的大小。本文的启发式排序为 DFLM 遍历,主要规则如下:

- (1) 当与动态失效行为无关的常规事件发生时, 若其变量未在排序中, 则将其加入排序中;
- (2) 当动态门发生时, 若其变量未在序列中, 则将其加入排序;
- (3) 当与动态门相关的事件发生时, 若其变量未在序列中, 则将其加入排序。

将动态门和常规事件相关的变量重新排序,成为一个新的序列后,DFT 的遍历就完成了。然后以自底向上的方式构建系统 MDD 模型。规则如下。

- (1) 如果正在进行的 DFT 节点 X 是一个常规失效事件, 则将 X 的初始 MDD 设为 $case(x, 0, 1)$;
- (2) 如果正在进行的 DFT 节点 X 是一个与动态失效行为相关的动态门, 则按照 2.2 节中所描述的规则对 X 的初始 MDD 进行编码;
- (3) 如果正在进行的 DFT 节点 X 是一个静态门, 也就是 $X = X_1 \langle \rangle X_2 \langle \rangle$ 是一个 AND 或者 OR, X_1 和 X_2 是子树, 计算和子树 X_1 和 X_2 相关的 MDD (E 和 F), 然后在这两个 MDD 上执行操作 $\langle \rangle$ 。

$$\begin{aligned} E &= \text{case}(x, E_0, E_1, \dots, E_m) \\ F &= \text{case}(y, F_0, F_1, \dots, F_n) \end{aligned}$$

E 和 F 间的逻辑操作可以用如下 MDD 操作表示。相同的规则可以用于子 MDD 间的逻辑操作,直到它们中的一个变为汇点 0 或 1。表 1 给出了以上递归 MDD 操作的算法。

$$E \diamond F = \begin{cases} case(x, E_0 \diamond F_0, E_1 \diamond F_1, \dots, E_m \diamond F_m), & order(x) = order(y), m = n \\ case(x, E_0 \diamond F, E_1 \diamond F, \dots, E_m \diamond F), & order(x) < order(y) \\ case(y, E \diamond F_0, E \diamond F_1, \dots, E \diamond F_n), & order(x) > order(y) \end{cases}$$

```

0      Computation ( $\langle \rangle, E, F$ ) =
1      If ( $F=0$ ) or ( $F=1$ ) or ( $E=0$ ) or ( $E=1$ )
2          Return Exit_case( $\langle \rangle, E, F$ )
3      Else if (computation-table has entry  $\{ \langle \langle \rangle, E, F \rangle, T \}$ )
4          Return T
5      Else let x be the least variable of E and F, x has n+1 cases
6          For each case i,  $0 \leq i \leq n$ 
7               $H_i \leftarrow$  Computation ( $\langle \rangle, E_{(x=i)}, F_{(x=i)}$ );
8       $H \leftarrow$  find-or-add-unique-table (x,  $H_0, H_1, \dots, H_n$ )
9      Insert-in-computation-table ( $\{ \langle \langle \rangle, E, F \rangle, H \}$ )
10     Return H

```

MDD 模型生成后,通过计算系统失效的概率对该 MDD 进行评估。每个从根节点到汇点 1 的路径都是一种失效模式,即造成系统失效的事件组合;而每个从根节点到汇点 0 的路径都是系统正常运作的模式。到达汇点 1 路径的概率总和也就是系统的失效概率。

MDD 所有的边总是连接到统计独立部件的变量,因此在评估中不需要考虑相关性。每条路径的概率由出现在路径上的每条边的概率得出。下面考虑两个方案。

(1) 对一个常规失效事件 X , 考虑有两条输出边的 MDD 节点 x 。节点 x 的 1-边和事件 X 发生的概率 $F_X(t)$ 是相关的。节点的 0-边和事件 X 的不发生概率 $1 - F_X(t)$ 相关。记 $F_X(t)$ 为部件 X 的失效分布。

(2) 考虑具有 $n+1$ 输出边的动态门 DG 的 MDD 节点 DG 。节点 DG 的 j th-边 ($0 \leq j \leq n$) 与对应状态的状态发生概率相关。首先, 计算以马尔科夫模型动态门 DG 为基础的极小生成矩阵 $Q=[q, s']$ 。然后利用方案(1)来分析模型, 得到条件暂时概率矩阵 $V(t)=[V_{s,s'}(t)]$, 在时间 t 下, 每个状态 s 相关的事件发生概率由一个预先设定的初始状态分布矢量 V_0 决定。

在软件实现中,通过递归算法^[11]可以将评估的 MDD 表达得更加紧凑。MDD 中, $F = \text{case}(x, F_{x=0}, F_{x=1}, \dots, F_{x=m})$, 所有子表达式 $(x=i) * (F_{x=i})$ 都是不相交的。评价方程为:

$$\Pr\{F=1\}=\Pr\{x=0\} * \Pr\{F_{x=0}=1\}+\dots+\Pr\{x=m\} * \Pr\{F_{x=m}=1\}$$

其中, $\Pr\{F_{x=i}=1\}$ 是接受子 MDD $F_{x=i}$ 的概率。如果 X 是整个 MDD 的根节点, 那么 $\Pr\{F=1\}$ 给出了最终系统失效的概率。回归算法的条件如下:

- (1) 如果 $F=0$, 那么 $\Pr\{F=1\}=0$;
(2) 如果 $F=1$, 那么 $\Pr\{F=1\}=1$ 。

下面考虑一个高度可靠的网络计算系统,它为整个系统提供两个不同的数据进程服务。如图 7 所示,系统分为两个部分,Complex A 和 Complex B。每个子系统包含一个服务器(SVR)、一个存储子系统(STO)及一个网络子系统(NET)。存储子系统采用镜像存储方式,包含一个存储控制器及两个存储空间。当存储控制器故障时,该存储器故障。备用服务器 SVRC 作为失效备援,提高了系统的可靠性。当主要服务器发生故障时,备用服务器就会自动启动且没有警告。通常一个需要高可靠性和适用性的服务器都设计有失效备援。在两个服务器间,失效备援的备件服务器采用心跳连接。

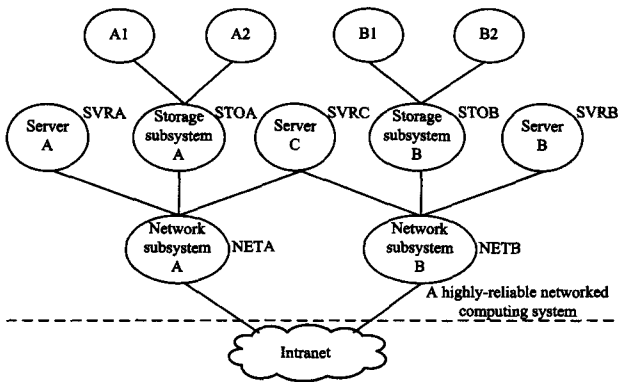


图7 高可靠性网络计算系统

该系统中的 SVRC 为一个温备件。SVRC 初始时就有-一个失效率,介于冷备件(具有 0 失效率,且在备用状态时不能失效)和热备件(具有与 SVRA,SVRB 相同的失效状态)。当 SVRA(或者 SVRB)单独失效时,SVRC 会接管已经失效的服务器工作,同时它具有失效率 λ_A (或 λ_B)。需要注意的是,当 SVRA 和 SVRB 都失效时,SVRC 会变得更加不可靠,它的失效率 λ_{AB} 比 λ_A 和 λ_B 都高。因为 SVRC 具有状态相关的失效率,它的失效标准不可以表示为同一时间框架下的基本事件组合。

图 8 为该系统的 DFT 模型。1)当两个数据进程服务都故障时,系统会完全失效。因此两个事件 Complex A 失效与 Complex B 失效和顶端与门连接。2)网络子系统或者存储子系统失效就会产生一种复杂的故障模式。因此 STOA(或者 STOB)和 NETA(或者 NETB)是与事件 Complex A 失效(或者 Complex B 失效)OR 门连接的。每个存储子系统中控制器的输入为激发输入,当它发生故障时,整个存储子系统会失效。而其中单个的存储空间故障并不会对系统产生影响。因此,此处用功能相关门 FDEP 连接。作为备件的 SVRC 与 SVRA 和 SVRB 的连接采用 SP 门。

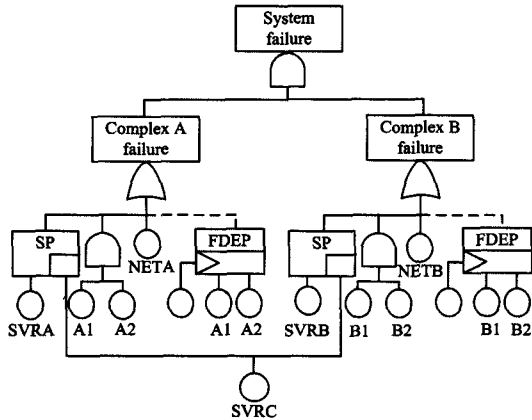


图8 高可靠性网络计算系统的 DFT 模型

利用两个 SP 门描述系统中与 SVRC 相关的动态失效行为。设 x_1, x_2, x_3 为布尔状态指示符变量。 $x_1=1$ 或者 $x_2=1$ 或者 $x_3=1$ 代表 SVRA 或者 SVRB 或者 SVRC 失效;否则,它们的值为 0。

这两个 SP 门的状态矢量 $X = \{x_1, x_2, x_3\}$ 有多种情况: $(0,0,0), (1,0,0), (0,1,0), (0,0,1), (1,1,0), (1,0,1), (0,1,1), (1,1,1)$ 。通过考虑 SP 门相关的故障模式及各个服务器的失效率,得到相应的马尔科夫链,如图 9 所示。显而易见,该 SP 门有一个比较大的状态空间,仅仅的 3 值变量不足

以进行编码,所以采用一个 8 值变量来编码。

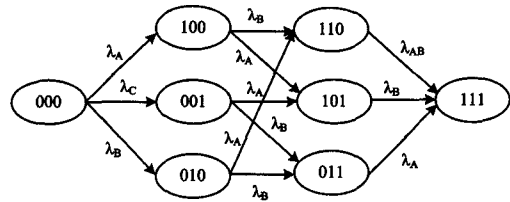
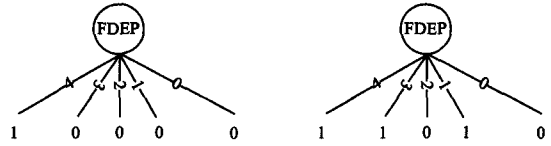
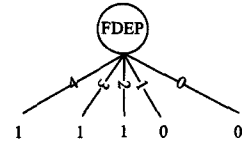


图9 高可靠性网络计算系统 SP 门的 CTMC

对于每个存储子系统 FDEP 门,以 A 子系统为例,存储控制器、存储器 A1、存储器 A2 由 FDEP 连接。存储控制器为激发输入,当它发生故障时,整个存储子系统就会失效。因此可以得到它的状态空间为: $(000), (010), (001), (011), (111)$ 。为表示方便,记为 0,1,2,3 和 4。这 3 个部件的 FDEP 编码的初始 MDD 如图 10 所示。



(a)与存储控制器相关的初始 MDD (b)与存储器 A1 相关的初始 MDD



(c)与存储器 A2 相关的初始 MDD

图10 3个部件的 FDEP 编码的初始 MDD

将排序启发式应用到 DFT 模型中,得到序列 $SP < FDEPA < NETA < FDEPB < NETB$ 。变量 SP 中有 3 个部件 SVRA,SVRB 和 SVRC 的状态指示符变量。由这个序列可以得到动态门的初始 MDD,如图 11 所示。

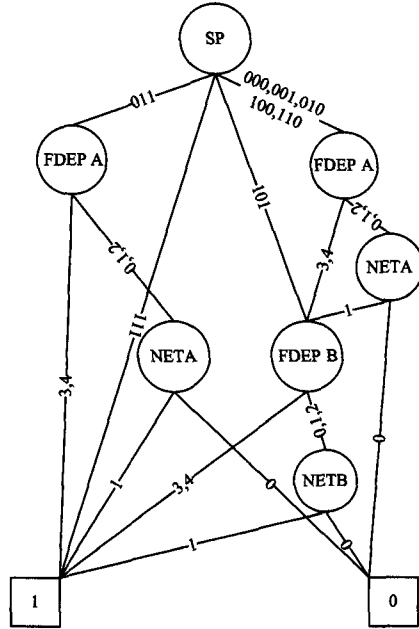


图11 高可靠性网络计算系统动态门的初始 MDD

对于静态失效行为相关的部件,例如存储子系统,可以按照指数失效分布来分析,而以上介绍的基于 MDD 的分析方

(下转第 92 页)

rical-based channel model arisen from scatterers on a hollow-disc for outdoor and indoor wireless environments[J]. Iet Communications, 2012, 6(17): 2775-2786

- [11] Le K N. On Angle-of-Arrival and Time-of-Arrival Statistics of Geometric Scattering Channels[J]. IEEE Transactions on Vehicular Technology, 2009, 58(8): 4257-4264
- [12] Pedersen K I, Mogensen P E, Fleury B H. Power azimuth spectrum in outdoor environments[J]. Electronics Letters, 1997, 33(18): 1583-1584
- [13] Borhani A, Patzold M. Time-of-arrival, angle-of-arrival, and angle-of-departure statistics of a novel simplistic disk channel model[C]// 2011 5th International Conference on Signal Processing and Communication Systems (ICSPCS). IEEE, 2011: 1-7

(上接第 73 页)

法依旧可以使用,但是半马尔科夫方法或者更新方法不可以使用;对于动态失效行为相关的部件,例如 3 个服务器 SVRA,SVRB 和 SVRC,依旧可以在指数失效分布中找到,以上基于 MDD 的方法也可以和传统的非马尔科夫模型一起使用。

结束语 本文详细介绍了基于多值决策图的故障树分析方法,从结果中看出,这种对大型动态子树的可靠性进行计算的方法是有效而且快速的,一定程度上解决了状态空间的爆炸问题。这种方法对与动态失效行为相关的子树部件运用状态空间方法进行分析。多值变量对动态门进行编码,生成的整个系统的 MDD 是简单且小型的,综合运用组合方法和状态空间方法,这使得状态空间的爆炸问题大大缓解。对现实生活中各种多功能动态软件和嵌入式系统的可靠性进行分析,该方法可以得到广泛应用。

与传统 BDD 相似的是,系统 MDD 的大小很大程度上取决于输入变量的顺序。因此,变量排序的优化仍是未来工作的一个重要方向。考虑到动态故障树分析更加高效,可以考虑构建动态故障树自动分析平台,借助软件平台进行建模,分析计算系统的可靠性。还有其它方面可以扩展研究,如 DFT 图形编辑打印、DFT 分析计算、优化改进方案的提出、系统数据库再到风险分析。

传统的马尔科夫方法是建立在假设事件的失效率保持不变的前提下,但是在现实情况中有些机械电子元件的失效率是随着时间变化的。因此,失效率的变化问题也是未来的研究方向。同一个部件或系统可能有多种状态,而不仅仅只有正常和故障两个状态,这也是未来需要考虑的。本文只讨论了动态模块独立的动态故障树,而动态门嵌套的情况较为复杂,尚未做出很好的分析。

参 考 文 献

- [1] Vemuri K K, et al. Automatic synthesis of fault trees for computer-based systems[J]. IEEE Trans. Rel., 1999, 48(4): 394-402
- [2] Sullivan J K, et al. The Galileo fault tree analysis tool[C]// Proc 29th Annu. Int. Symp. Fault-Tolerant Comput., Madison, WI, USA, 1999: 232-237
- [3] Meshkat L, et al. Dependability analysis of systems with on-demand and active failure modes, using dynamic fault trees[J]. IEEE Trans. Rel., 2002, 51(2): 240-251
- [4] Huang C Y, Chang Y R. An improved decomposition scheme for assessing the reliability of embedded systems by using dynamic fault trees[J]. Rel. Eng. Syst. Safety, 2007, 92(10): 1403-1412
- [5] Amari S, et al. A new approach to solve dynamic fault trees [C]// Proc. IEEE Annu. Rel. Maintainability Symp., Tampa, FL, USA, 2003: 374-379
- [6] Dugan J B, et al. Developing a low-cost high quality software tool for dynamic fault tree analysis[J]. IEEE Trans. Rel., 2000, 49(1): 49-59
- [7] Miller D M. Multiple-valued logic design tools[C]// Proc. 23rd Multiple-Valued Logic (ISMVL). 1993: 2-11
- [8] Xing L, Dugan J B. Dependability analysis using multiple-valued decision diagrams[C]// Proc. 6th Probabilistic Safety Assessment Manag., 2002
- [9] Xing L, Dai Y. A new decision diagram based method for efficient analysis on multi-state systems[J]. IEEE Trans. Dependable Secure Comput., 2009, 6(3): 161-174
- [10] Ou Y, Dugan J B. Sensitivity analysis of modular dynamic fault trees [C]// Proc. Compute. Perform. Dependability Symp., 2000: 35-43
- [11] Tarjan R. Depth first search and linear graph algorithms [J]. Sensors, 2006, 14(4): 114-121
- [12] Gao Shun-chuan. Methods and Implementation of Dynamic Fault Tree Analysis[D]. Changsha: National University of Defense Technology, 2005 (in Chinese)
- 高顺川. 动态故障树分析方法及其实现[D]. 长沙: 国防科技大学, 2005
- [13] Gu Ying-kui, Li Jing. Importance Analysis of Multi-state System Based on Multiple-valued Decision Diagram[J]. China Safety Science Journal, 2014, 24(6): 44-50 (in Chinese)
- 古莹奎, 李晶. 基于多值决策图的多状态系统重要度分析[J]. 中国安全科学学报, 2014, 24(6): 44-50
- [14] Zhang Hong-lin, Zhang Chun-yuan, Liu Dong. An Identification Method of Independent Module Applying to Dynamic Fault Tree with Interdependent Basic Events and Repeated Events [J]. Chinese Journal of Computers, 2012, 35(2): 229-243 (in Chinese)
- 张红林, 张春元, 刘东. 一种适用于具有相互依赖基本事件和重复事件的动态故障树独立模块识别方法[J]. 计算机学报, 2012, 35(2): 229-243
- [15] Cheng Ming-hua, Yao Yi-ping. Application of Dynamic Fault Tree Analysis to Software and Hardware Fault-tolerant Control Computer Systems[J]. Acta Aeronautica et Astronautica Sinica, 2000, 21(1): 34-37 (in Chinese)
- 程明华, 姚一平. 动态故障树分析方法在软、硬件容错计算机系统中的应用[J]. 航空学报, 2000, 21(1): 34-37