

基于动态熵搜索的故障隔离策略研究

常天庆¹, 李 勇¹, 王 栋²

(1 装甲兵工程学院 控制工程系, 北京 100072; 2 中国航天员科研训练中心, 北京 100094)

摘 要: 为了进一步缩短平均故障隔离时间, 在直接熵法和间接熵法的基础上, 提出了动态熵搜索的故障隔离策略. 该方法以系统最大动态熵作为测试选择的依据, 并以动态熵为核心遍历搜索而建立故障隔离决策树. 算例验证表明: 与直接熵法和间接熵法相比较, 该方法所需的测试少, 且减少了平均故障隔离时间.

关键词: 动态熵; 故障隔离策略; 直接熵法; 间接熵法

中图分类号: TB114. 3

文献标识码: A

文章编号: 1000-7180(2013)01-0013-04

Research on Fault Isolation Strategy Based on Dynamic Entropy Search

CHANG Tian-qing¹, LI Yong¹, WANG Dong²

(1 Department of Control Engineering, Academy of Armored Force Engineering, Beijing 100072, China;

2 Center of Astronaut Scientific Research and Training of China, Beijing 100094, China)

Abstract: In order to shorten the mean fault isolation time, the fault isolation strategy based on dynamic entropy search is proposed in this paper on the basis of entropy-directed algorithm and entropy-indirected algorithm. The strategy takes the maximum dynamic-entropy of the system for account to select tests and establishes fault isolation decision tree using ergodic search on the centre of dynamic entropy. It is showed by algorithm example that the strategy in this paper needs less tests and mean fault isolation time than entropy-directed algorithm and entropy-indirected algorithm.

Key words: dynamic entropy; fault isolation strategy; entropy-directed algorithm; entropy-indirected algorithm

1 引言

故障隔离策略是测试性设计的关键问题之一. 故障隔离策略的目的是当维修等级、工作环境、系统故障、测试数目和种类一定时, 通过合理安排测试顺序以隔离故障, 降低系统的平均故障隔离时间 (Mean Fault Isolation Time, MFIT). 已有的故障隔离策略算法包括: 动态规划、贪婪式搜索以及与或图搜索 AO* 算法三类^[1]. 动态规划方法计算复杂, 不适合工程应用. 贪婪式搜索算法主要有直接熵法和间接熵法两种. 直接熵法^[2]以寻找使系统不确定性降低最多的测试求解最优故障隔离策略, 该方法最简便, 但搜索速度最慢; 间接熵法^[3]虽然可以自动构造决策树, 但算法效率不高, 不适用于大型复杂系统的故障隔离. K. R. Pattipati 等人将决策树用二

值与或图表示, 由此发展了基于 AO* 与或树搜索的诊断序列生成算法^[4-5]. 与或图搜索算法需要构造启发式知识评价函数使算法朝最有希望接近目标的方向进行. 该算法的好坏受启发式函数影响, 启发式函数所携带的启发性信息越多, 搜索时扩展的节点就越少, 搜索效率越高, 因此, 如何建立适合的启发式函数是一个难点问题^[6].

本文提出了动态熵搜索算法. 通过算例分析, 比较了该算法与直接熵法和间接熵法所用的平均故障隔离时间. 结果表明, 动态熵法更能体现系统故障隔离过程的实质, 减少了平均故障隔离时间.

2 问题描述

故障隔离过程是通过一系列测试确定系统究竟处于哪一种状态, 构造一个完整的故障隔离策略就

是寻找一组测试序列,以区分系统所有状态.这些测试序列可形成一个外向的决策树,决策树的分枝点为测试,树叶为系统各种状态,树叶的数量等于系统状态树 K .

令决策树从树根至树叶 $l_k (k=0,1,\dots,K)$ 的路为 $w_k (k=0,1,\dots,K)$,而该路中所有分枝点对应的测试用 $w_k[i]$ 表示,所需时间用 $t_{w_k[i]}$ 表示,该路中测试总个数用 $|w_k|$ 表示,树叶 l_k 对应系统状态概率为 $p(l_k)$,那么该决策树的权值为

$$w(\text{tree}) = \sum_{k=0}^K \left(\sum_i |w_k| t_{w_k[i]} \right) p(l_k) \quad (1)$$

显然, $w(\text{tree})$ 为该决策树对应故障隔离策略的 MFIT. 这样,寻求最优故障隔离策略过程便可描述成一个根据故障隔离结论集 F 与测试集 T 的相关性矩阵 $F-T$,通过合理选择测试集安排测试顺序,使决策树的权值 $w(\text{tree})$ 最小的优化问题.

3 动态熵搜索算法

3.1 算法思想

故障隔离是沿着某种策略寻找测试并不断划分状态集,直到候选状态集中只剩一个故障状态的过程.故障隔离策略的优化是通过合理选择测试及测试顺序以实现快速故障隔离,是一个动态选择过程.要得到优化的诊断策略,仅考虑故障候选状态集的熵降是不够的,在动态故障隔离的每一步,应都进行测试选择.因此,在动态故障隔离系统中,故障候选状态集合测试都应被视为信息源.在故障隔离的每一步,应考虑故障候选状态集的变化和测试选择两个要素组成的动态系统的熵降,并选择使系统动态熵最大的测试.本文将执行某个测试后造成的整个动态系统熵降定义为测试的动态熵,以动态熵为核心遍历决策树来构成故障隔离策略的方法称为动态熵法.

3.2 算法流程

基于动态熵法的基本思想,可得如图 1 所示的算法流程.

3.3 动态熵的求解及测试选择

3.3.1 测试的信息量

测试 t_i 结果为正常(用“0”表示),得到的信息量用 a_i 表示,则有:

$$a_i = 1 + \sum_{j=1}^{|T|} \beta_{ij} + \sum_{j=1}^{|T|} \gamma_{ij} + \sum_{j=1}^{|T|} \delta_{ij} \quad (2)$$

测试 t_i 结果为故障(用“1”表示),得到的信息

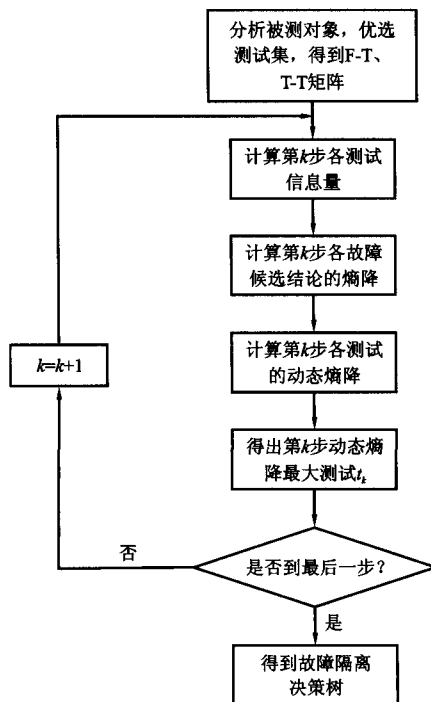


图1 动态熵搜索算法流程

量用 b_i 表示,则有:

$$b_i = 1 + \sum_{j=1}^{|T|} \beta'_{ij} + \sum_{j=1}^{|T|} \gamma'_{ij} + \sum_{j=1}^{|T|} \delta'_{ij} \quad (3)$$

其中, β_{ij} 为矩阵 $T_F - T_F$ 中第 i 行中不为“0”的元素, γ_{ij} 为矩阵 $T_F - T_F$ 中第 i 行中为“0”,并且矩阵 $T-T$ 第 i 列中也为“0”的测试.对二者进行计数(除自身外), δ_{ij} 为非多余测试个数.

3.3.2 故障候选结论的信息熵降

在第 k 步选择测试 t_i 之前,得到一个故障候选结论集合,用 x 表示.

第 k 步选择测试 t_i ,根据测试结果将 x 分为 $\{x_{k0}, x_{k1}\}$,分别对应测试结果为“0”、“1”的故障候选结论集,则选择测试 t_i 引起的熵降表示为:

$$\begin{aligned} \Delta I(x) &= \frac{\hat{p}(x_{k0})}{\hat{p}(x)} \log_2 \left(\frac{\hat{p}(x)}{\hat{p}(x_{k0})} \right) + \frac{\hat{p}(x_{k1})}{\hat{p}(x)} \\ &\quad \log_2 \left(\frac{\hat{p}(x)}{\hat{p}(x_{k1})} \right) \\ &= \Delta I(x_{k0}) + \Delta I(x_{k1}) \end{aligned} \quad (4)$$

3.3.3 动态熵降

在第 k 步选择测试 t_i ,系统动态熵降 $DE_i(x)$ 表示为:

$$DE_i(x) = \alpha a_i \Delta I(x_{k0}) + \beta b_i \Delta I(x_{k1}) \quad (5)$$

其中, α 和 β 分别为测试信息量与故障候选结论信息量乘积的待定系数,由设计者根据情况确定.

3.3.4 测试的选择

为简化计算,采用极大—极小准则选择测

试,即:

$$\varphi_i=\min\{a_i\Delta I(x_{k0}),b_i\Delta I(x_{k1})\}\tag{6}$$

$$\psi=\max_{i=1}^K\varphi_i\tag{7}$$

在故障隔离的每一步都选择使得 ψ 最大的测试 t_i ,极大一极小准则可以确保测试的选择具有一定的鲁棒性^[7].

3.4 等概率条件下的动态熵

在工程上,进行测试性设计时,往往事先很难获知系统各故障隔离结论概率的具体数据. 只能假设 F-T 中各故障隔离结论出现的概率相等,即相应的系统状态等概率. 状态集 S 中,每个状态的概率 $p(s_i)=1/K$. 假设 $\dim(x)=C,\dim(x_{k0})=C_{k0},\dim(x_{k1})=C_{k1}$,则第 k 步选择测试 t_i ,造成故障候选结论熵降的计算公式如下.

$$\Delta I(x)=\frac{C_{k0}}{C}\log_2(C/C_{k0})+\frac{C_{k1}}{C}\log_2(C/C_{k1})\tag{8}$$

在测试 t_i 选择前后,设 α 和 β 均为 1,则动态熵 $DE_i(x)$ 用 φ_i 表示,则有:

$$\varphi_i=a_iC_{k0}\log_2(C/C_{k0})+b_iC_{k1}\log_2(C/C_{k1})\tag{9}$$

$$\psi=\max_{i=1}^K\varphi_i\tag{10}$$

在故障隔离的每一步都选择使得 ψ 最大的测试 t_i ,即在等概率条件下,选择动态熵降最大的测试.

4 算例验证

为了验证动态熵搜索算法的优劣,对等概率条件下的算例建立决策树,并与直接熵法和间接熵法进行比较.

根据具体电路建立相关性矩阵. 本文以图 2 所示的电路建立其相关性矩阵 F-T 和 T-T,具体见表 1、表 2.

对 F-T 和 T-T 进行优选,删除多余测试,得到非多余测试为 $\{t_3,t_4,t_6\}$. 建立测试 t_3,t_4,t_6 之间的相关性矩阵,如表 3、表 4 所示.

利用公式(9)计算各测试的动态熵降,如表 5 所示,由此得到,在诊断第一步选择动态熵降最大的测试 t_3 .

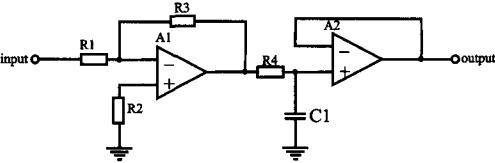


图 2 滤波放大电路

表 1 简化后 F-T 矩阵

F-T	t_1	t_2	t_3	t_4	t_5	t_6	t_7
f_1	1	0	1	1	1	0	1
f_2	0	0	1	0	0	0	1
f_3	1	1	1	1	1	1	1
f_4	0	0	0	1	1	0	0
f_5	0	0	0	0	1	1	1
f_6	1	1	1	1	1	1	1
f_7	0	0	0	1	1	1	1
f_8	0	0	0	0	1	1	1
f_9	0	0	0	0	0	0	0

表 2 T-T 矩阵

T-T	t_1	t_2	t_3	t_4	t_5	t_6	t_7
t_1	0	0	1	1	1	0	1
t_2	1	1	1	1	1	1	1
t_3	0	0	0	0	0	0	1
t_4	0	0	0	1	1	0	0
t_5	0	0	0	0	1	0	0
t_6	0	0	0	0	0	1	1
t_7	0	0	0	0	0	0	0

表 3 优选后的 F'-T' 矩阵

F-T	t_3	t_4	t_6
f_1	1	1	0
f_2	1	1	1
f_3	1	0	0
f_4	0	1	0
f_5	0	0	0
f_6	1	0	1
f_7	0	0	0
f_8	0	1	0
f_9	0	1	1

表 4 优选后的 T'-T' 矩阵

T-T	t_3	t_4	t_6
t_3	0	0	1
t_4	1	1	1
t_6	0	0	0

表 5 各测试的动态熵

测试	t_3	t_4	t_6
动态熵	16.32	11.43	14.17

在测试 t_3 得到不同结论的前提下,计算剩余测试的动态熵降,并选择动态熵降最大的作为下一步的测试. 表 6 为不同结论时各测试的动态熵.

如此反复,最后得到故障隔离决策树如图 3

所示.

表 6 t_3 为不同结论时各测试的动态熵

动态熵	t_4	t_6
$t_3 = "0"$	7.25	6.27
$t_3 = "1"$	2.75	4.34

在相同条件下,采用已有的直接熵法和间接熵法对图 2 的诊断建立故障隔离决策树,如图 4、图 5 所示.

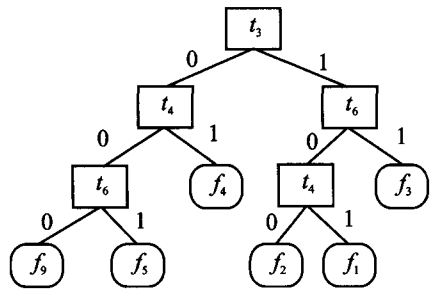


图 3 采用动态熵法建立的决策树

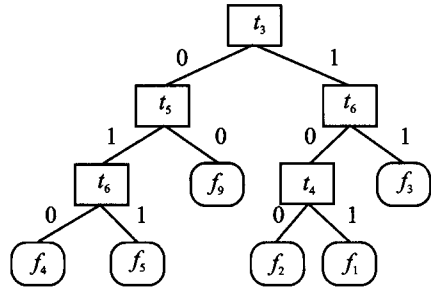


图 4 采用直接熵法建立的决策树

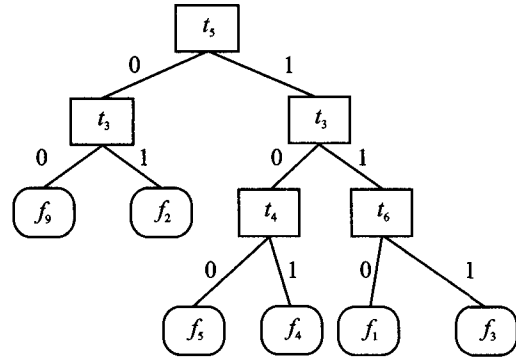


图 5 采用间接熵法建立的决策树

表 7 列出了分别采用直接熵法、间接熵法和动态熵法隔离系统所有状态所需要的测试数目和平均估值隔离时间.

表 7 三种隔离策略的比较

故障隔离策略	第一个测试	测试总数	平均隔离时间
直接熵法	t_3	4	2.56
间接熵法	t_5	4	2.67
动态熵法	t_3	3	2.44

由表 7 可得,采用动态熵法比间接熵法的平均间隔时间减少了 8.6%,比直接熵法的平均间隔时间减少了 4.7%,并且动态熵法所需测试总个数最少.

5 结束语

本文对故障隔离策略优化问题进行了深入探讨,提出了一种新的搜索算法——动态熵法.该算法通过计算优选测试集中各测试的动态熵,并以最大动态熵为依据进行测试选择,并建立故障隔离决策树.算例验证表明,采用动态熵法进行故障隔离能进一步缩短平均故障隔离时间,提高故障诊断效率.

参考文献:

[1] 王栋. 电子系统测试性分析与设计方法研究[D]. 北京:装甲兵工程学院,2009.

[2] 黎琼炜,易晓山,刘冠军. 系统级故障隔离的间接熵法[J]. 系统工程与电子技术,2001,23(2):51-54.

[3] 蒋荣华,王厚军,龙兵. 大型复杂系统故障隔离智能优化策略[J]. 系统仿真学报,2009,21(3):841-844.

[4] Pattipati K R, Alexandridis M G. Application of heuristic search and information theory to sequential fault diagnosis[J]. IEEE Transaction on System, Man, and Cybernetics, 1990,20(4):872-886.

[5] Vijay Raghavan, Mojdeh Shakeri,Krishna R Pattipati. Optimal and near-optimal test sequencing algorithms with realistic test models[J]. IEEE Transaction on System, Man, and Cybernetics-part A: Systems and Humans, 1999,29(1):11-25.

[6] 师军,曹茜. 启发式搜索问题研究[J]. 微型机与应用, 2004(10):10-12.

[7] Sheppard J W, Simpson W R. Performing effective fault isolation in integrated diagnostics (part 6) [J]. IEEE Design & Test of Computers, 1993, 10(2): 78-90.

作者简介:

常天庆 男,(1963—),博士,教授,博士生导师.研究方向为装备维修保障与智能火控系统.

李 勇 男,(1983—),博士研究生.研究方向为装备智能故障诊断.

王 栋 女,(1981—),博士,工程师.研究方向为智能 BIT、故障诊断、测试性设计.