

doi: 10.7690/bgzdh.2016.05.002

基于故障-测试相关性矩阵的测试性分析

马瑞萍¹, 董海迪², 马长李¹

(1. 海军装备研究院, 北京 100161; 2. 第二炮兵工程大学空间工程系, 西安 710025)

摘要: 为有效检测出系统测试中存在的不可测故障、冗余测试和隐含故障等, 提出一种基于故障-测试相关性矩阵的测试性分析方法。首先给出不可测故障、模糊组、不可隔离故障、冗余测试、隐含故障和伪故障的定义; 在此基础上, 提出基于故障-测试相关性矩阵的测试性分析流程和求解方法。算例分析结果表明: 该方法能有效完成系统测试性分析, 且操作简单, 具有较大的工程应用价值。

关键词: 相关性矩阵; 测试性分析; 隐含故障; 伪故障

中图分类号: TJ06 **文献标志码:** A

Testability Analysis Based on Fault-test Dependency Matrix

Ma Ruiping¹, Dong Haidi², Ma Changli¹

(1. Naval Academy of Armament, Beijing 100161, China;

2. Department of Space Engineering, The Second Artillery Engineering University, Xi'an 710025, China)

Abstract: To effectively detect the presence of undetectable faults, redundant tests and hidden faults, a testability analysis method based on fault-test dependency matrix is studied. First of all, definitions of undetectable faults, fuzzy sets, non-isolated faults, redundant tests, hidden faults and false fault are declined. On that basis, analysis process and solving method based on fault-test dependency matrix are proposed. Example analysis results show that the method can effectively realize system testability analysis and is easy to operate, owing great value in engineering application.

Keywords: dependency model; testability analysis; hidden fault; false fault

0 引言

武器装备在使用前通常需要进行全面、综合的测试, 以保证装备功能正常、性能良好、使用安全^[1-2]。为改善武器装备性能, 使武器功能更加先进, 装备研制过程中广泛采用现代先进技术, 也导致武器装备的结构和技术更加复杂。装备系统的测试流程设计面临着更加严峻的挑战^[3-4]。

测试流程设计前, 对系统测试性^[5-6]进行分析, 及早发现不可测故障、冗余测试和隐含故障等, 并进行针对性更改设计, 能够显著提高测试流程设计效率, 节约测试资源。

因此, 笔者研究了基于故障-测试相关性矩阵的测试性分析方法^[7], 给出了不可测故障、模糊组、不可隔离故障、冗余测试、隐含故障和伪故障的定义与求解方法, 提出了基于故障-测试相关性矩阵的测试性分析流程, 并通过案例应用分析验证方法的可行性和有效性。

1 故障-测试相关性矩阵构建

工程上常采用基于模型的测试性分析方法^[8], 方法原理是在建立相关性模型的基础上, 分析得到

系统的故障-测试相关性矩阵, 并对相关性矩阵进行测试性能分析。

笔者采用工程中广泛应用的信息流模型^[9]对复杂武器系统进行测试性能描述, 该模型主要包括可用测试、故障结论、可测试输入、不可测输入和无故障结论等基本元素。

在信息流模型中, 测试项和故障结论间的测试关系通过有向图表达, 图中节点表示可用测试项和故障结论, 边表示信息流流动方向。系统故障-测试相关性矩阵通过对信息流模型进行相关性分析得到, 逻辑关系表示如下:

$$FT = \begin{matrix} & T_1 & T_2 & \cdots & T_n \\ \begin{matrix} F_1 \\ F_2 \\ \vdots \\ F_m \end{matrix} & \begin{bmatrix} ft_{11} & ft_{12} & \cdots & ft_{1n} \\ ft_{21} & ft_{22} & \cdots & ft_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ ft_{m1} & ft_{m2} & \cdots & ft_{mn} \end{bmatrix} \end{matrix} \quad (1)$$

式中元素 ft_{ij} 均为布尔量, $ft_{ij}=1$ 表示测试项 t_j 能够检测隔离故障 f_i , $ft_{ij}=0$ 则表示不能。第 i 个行向量 $F_i=[ft_{i1} \ ft_{i2} \ \cdots \ ft_{in}]$ 表示故障 f_i 存在时所有可用测试项的输出, 描述了故障 f_i 的征兆; 第 j 个列向

收稿日期: 2016-01-12; 修回日期: 2016-02-27

作者简介: 马瑞萍(1965—), 女, 陕西人, 博士, 高级工程师, 从事导弹测试与故障诊断研究。

量 $T_j = [ft_{1j}, ft_{2j}, \dots, ft_{mj}]^T$ 表示可用测试项 t_j 能检测隔离的故障，描述了测试 t_j 的故障检测隔离能力。

2 测试性分析方法

笔者采用基于故障-测试相关性矩阵方法对系统的测试流程进行测试性能分析^[8-9]，主要分析测试流程设计中存在的不可检测故障、不可隔离故障、模糊故障组、冗余测试项、伪故障和隐含故障等。

不可检测故障：所有可用测试项及其组合均无法检测隔离的故障。 $F_i = \emptyset$ 表示对应的故障 f_i 为不可检测故障。

不可隔离故障：所有可用测试项及其组合均无法准确检测隔离的故障。 $T_j \wedge T_k = e_i$ 和 $T_j \wedge (T_j \oplus T_k) = e_i$ (其中 e_i 为第 i 个元素为 1 的单位向量， \wedge 为位与运算符， \oplus 为异或运算符) 表示测试项 t_j 和 t_k 组合起来能够检测隔离故障 f_i 。

模糊故障组：2 个故障具有相同的特征，无法被完全隔离分开。 $F_j = F_k (j \neq k)$ 表示故障 f_j 和故障 f_k 为模糊故障组。

冗余测试项：2 个测试项检测隔离的故障相同。 $T_j = T_k (j \neq k)$ 表示测试项 t_j 和 t_k 互为冗余测试项。

隐含故障：故障 f_k 的征兆与故障 f_j 和故障 f_k 共同的征兆完全相同。 $F_j \vee F_k = F_k (j \neq k)$ 表示故障 f_j 是 f_k 的隐含故障。

伪故障：满足 $\bigvee_{f_j \in X} F_j = F_k (f_k \notin X)$ 的组合 X 表示 f_k 的伪故障。

系统测试流程设计的测试性分析方法如图 1 所示，系统的故障-测试相关性矩阵表示为 FT 。

1) 按照 $ft_{ij} \geq ft_{kj} (i, k \geq j, i < k)$ 原则对矩阵 FT 中的所有行向量 F_i 进行排列，得到新的矩阵 FT' 。

2) 分析测试中存在的冗余测试项，矩阵 FT' 所有列中满足 $T_j = T_k$ 对应的测试 t_j 、 t_k 均为冗余测试，部分冗余测试是为提高系统可靠性有意增加的，无意引入的冗余测试项需要剔除。

3) 分析不可检测故障和不可隔离故障，矩阵 FT' 所有行中满足 $F_j = \emptyset$ 所对应的故障 f_j 为不可检测故障，满足 $F_j = F_k$ 所对应的故障 f_j 、 f_k 为模糊故障组，系统设计的测试项无法检测隔离出这些故障；因此，需要分析增加对应测试项。

4) 重复 1)~3)，直至不再进行剔除冗余测试项或增加测试项等操作，得到新的故障-测试相关性矩阵 FT'' 。

5) 分析隐含故障和伪故障，矩阵 FT'' 所有行向量 F_i 中元素值为 0 对应的列向量为 $T_j (ft_{ij}=0)$ ，去除 T_j 中元素值为 1 对应的行向量 $F_i (T_{ji}=1)$ 后矩阵余下的行向量为 $F_k (F_k \neq \bar{0}, k > i)$ ，则故障 f_k 为故障 f_i 的隐含故障，且满足 $\min_{\bigvee_{f_j \in X} F_j = F_k (f_k \notin X)}$ 的集合 X 为故障 f_i 的伪故障集。

6) 分析不可隔离故障，矩阵 FT'' 所有列向量中满足 $T_j \wedge T_k = e_i$ 或 $T_j \wedge (T_j \oplus T_k) = e_i$ ，删除 e_i 中元素值为 1 对应的行向量故障 f_i 后，剩余行向量对应的故障 f_k 为不可隔离故障。

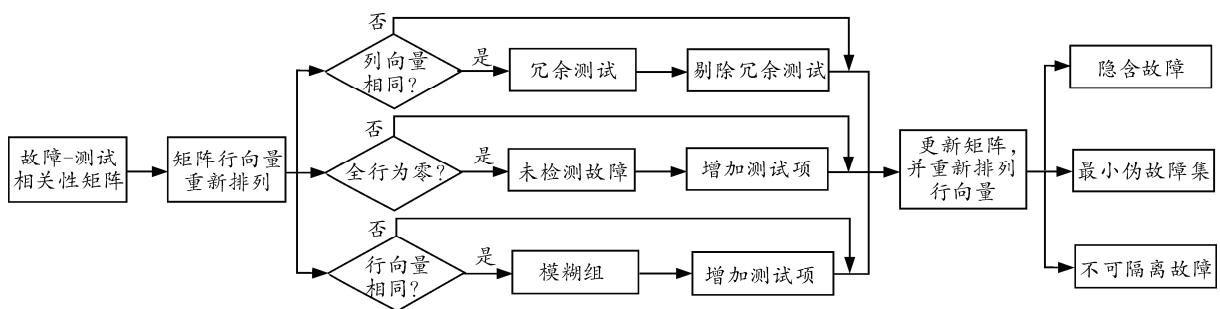


图 1 系统测试性分析流程

3 应用实例

笔者采用文献[10]中的某导弹系统级故障-测试相关性矩阵进行测试性分析，该系统主要由 32 个故障项和 22 个测试项组成，故障-测试相关性矩阵如表 1 所示。

采用文中提出的定义和方法，对系统不可测故障、模糊组、不可隔离故障、冗余测试、隐含故障和伪故障等进行分析，得出结论如下。

不可测故障有 4 项分别为故障 21、23、29 和 31；模糊组有 3 组，分别为故障(10, 18)、(11, 30)和(13, 16)；无冗余测试项；无不可隔离故障；隐含

故障和伪故障如表 2 所示。该分析结果与系统实际情况完全相符。

表 1 导弹系统级故障-测试相关性矩阵

| FT | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
|----|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 2 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 4 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 5 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 22 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 31 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 32 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

表 2 隐含故障和伪故障统计

| 故障 | 隐含故障 | 伪故障 |
|----|---------------------------------|---------|
| 1 | 9、13、14、15、17、22、24、25、27、28 | 无 |
| 2 | 12、13、19、20 | 无 |
| 3 | 9、10、11、12、13、14、15、19、20、22、24 | 无 |
| 4 | 10、13、22 | 无 |
| 5 | 11、13 | 无 |
| 6 | 13 | 无 |
| 7 | 13 | 无 |
| 8 | 13、19、32 | 无 |
| 9 | 13、22 | 无 |
| 10 | 13、22 | 无 |
| 11 | 13 | 无 |
| 12 | 13 | 无 |
| 13 | 无 | 无 |
| 14 | 22 | 无 |
| 15 | 22 | 无 |
| 17 | 9、13、22、27 | 9、27 |
| 19 | 13 | 无 |
| 20 | 12、13、19 | 12、19 |
| 22 | 无 | 无 |
| 24 | 9、13、14、15、22 | 9、14、15 |
| 25 | 13 | 无 |
| 26 | 11、13 | 无 |
| 27 | 无 | 无 |
| 28 | 无 | 无 |
| 32 | 13 | 无 |

4 结束语

笔者提出的基于故障-测试相关性矩阵的测试性分析方法，能够检测出系统的不可测故障、模糊组、不可隔离故障、冗余测试、隐含故障和伪故障等，为系统的测试项设置提供针对性建议，提高了

武器系统测试流程设计效率，节约了测试资源。

参考文献：

- [1] 胥辉旗, 田燕妮, 陈望达. 基于弹上总线的导弹装备测试性设计方法[J]. 兵工自动化, 2012, 31(1): 1-3.
- [2] 石君友, 王璐, 李海伟, 等. 基于设计特性覆盖的测试性定量分析方法[J]. 系统工程与电子技术, 2012, 34(2): 418-423.
- [3] 同江, 蔡远文, 伯伟, 等. BIT 技术的发展现状与应用分析[J]. 兵工自动化, 2008, 27(4): 5-7.
- [4] 连光耀, 黄考利, 陈建辉, 等. 复杂电子装备智能测试性设计技术[J]. 兵工自动化, 2006, 25(8): 71-72.
- [5] Williams T W, Parker K P. Design for testability-A survey[J]. Proceedings of the IEEE, 1983, 71: 98-112.
- [6] Fritzeimer R R, Nagle H T, Hawkins C F. Fundamentals of testability-a tutorial[J]. IEEE Transactions on Industrial Electronics, 1989, 36(2): 117-128.
- [7] 邓森, 景博. 基于测试性的电子系统综合诊断与故障预测方法综述[J]. 控制与决策, 2013(5): 641-649.
- [8] 连光耀, 黄考利, 郭瑞, 等. 基于结构模型的测试性设计与分析技术研究[J]. 系统工程与电子技术, 2007(10): 1777-1780.
- [9] 欧爱辉. 一种基于多信号流图的雷达系统测试性建模分析方法[J]. 兵工自动化, 2014, 33(4): 56-60.
- [10] 杨鹏. 基于相关性模型的诊断策略优化设计技术[D]. 长沙: 国防科学技术大学, 2008: 135-139.