

自动化测试技术在网络控制软件测试中的应用

游桂贞^{1,2}, 刘布麒^{1,2}, 周权强^{1,2}

(1. 中车株洲所电气技术与材料工程研究院, 湖南 株洲 412001;
2. 动车组和机车牵引与控制国家重点实验室, 湖南 株洲 412001)

摘要: 描述了对图形化编程软件进行自动化测试的方法, 包括设计、执行、数据记录和结果分析。该方法能自动分析测试数据并形成测试结果, 且能重复执行。与手工测试的方法进行对比, 有效证明了自动化测试不仅能高效地检测出软件中的错误, 还能减少测试时间、人力和硬件资源的投入, 提高测试效率, 节省测试成本。

关键词: 软件测试; 自动化测试; 图形化编程软件; 测试用例; 测试执行; 回归测试

中图分类号: TP31; TP393; U231

文献标识码: A

doi: 10.13890/j.issn.1000-128x.2017.05.114

Application of Automatic Test Technology in Network Control Software Testing

YOU Guizhen^{1,2}, LIU Buqi^{1,2}, ZHOU Quanqiang^{1,2}

(1. CRRC ZIC Research Institute of Electrical Technology & Material Engineering, Zhuzhou, Hunan 412001, China;

2. State Key Laboratory for Traction and Control System of EMU and Locomotive, Zhuzhou, Hunan 412001, China)

Abstract: The method of graphical programing software was described, including the design, implementation, data recording and results analysis, which could automatically analyze the test data and form the test results with repeated execution. Compared with the manual test method, it was proved that the automatic test not only could effectively detect the errors in the software, but also could reduce a lot of testing time, improve test efficiency, reduce the cost of human and hardware resources, and save test costing.

Keywords: software test; automated testing; graphical programing software; test case; test execution; regression testing

0 引言

近年来, 虽然国内软件测试技术有了很大的提高^[1-10], 但测试理论、测试方法和测试技术还无法满足当前软件开发的实际需要。因此, 对软件测试技术、测试方法和测试工具的研究有十分广阔的前景。软件测试的实质是根据软件开发各阶段的规格说明书和程序内部结构构造测试场景, 形成软件测试用例, 并用这些测试用例去驱动被测程序, 观察被测程序的运行结果, 验证所得结果与预期结果是否一致, 然后再做相应的调整。

1 自动化测试的引入

网络控制软件测试的基本需求是模拟被测软件需求的输入, 将其输入到被测软件, 再观测被测软件的输出结果, 并对结果进行判断分析。在测试网络控制软件时需要2个模块, 一个运行被测软件, 另一个运行陪试软件, 且通过MVB来发送和接收数据。

手工执行测试用例的流程为: 针对被测软件的软件需求编写测试需求, 然后编写测试用例, 构建实际值, 将其通过Multiprog的陪试软件发送至被测软件中, 再通过陪试软件观测输出结果, 并人工记录结果, 再与测试用例的预期输出进行对比分析, 从而得出测试是否通过的结论。如需回归测试, 则需再次执行一遍。

它有几个很明显的缺点：①执行每个测试用例时均需手动为每个信号赋值，而且一次只能执行一个用例；②执行结果只能靠人眼来观察并手工记录，无法自动保存；③每一轮测试均需将信号强制重置。

这样手工测试要花费很长的时间来进行测试和记录测试结果，并且有很多的重复劳动，回归一次就需要重新执行一次测试用例。

如果用自动化测试，则只需人工定义好测试用例及数据、设定好预期结果，无需编写陪试软件，即可实现存取测试用例并读取测试数据且能保存测试用例结果，这样就能大大节省测试人员对测试结果进行采集和分析的工作量，且回归测试时不用再编写测试用例，减少测试时所用的时间，节省测试成本。

2 自动化测试软件的设计及实施

基于人工测试无锡地铁网络控制软件的繁琐性，在测试长沙地铁 2 号线网络控制软件时就考虑用自动化测试。构想中的自动化测试软件应具有以下特殊要求：①能完成长沙地铁 2 号线网络控制软件测试的基本要求。②输入数据自动化，输入数据能用 txt 格式的文本自动地读入。③测试结果进行自动保存。④输出结果能自动与预期输出结果进行比较分析，得出测试结论。⑤能够处理延时的要求，对于特定情况能够实现指定时间的延时。⑥具有可重用性。a. 对于不同版本，回归测试时自动化测试软件是可重用的，在设计人员自己调试时也可使用。b. 对于地铁网络控制软件，如属于同一类型的软件，自动化测试脚本也可以进行重用，只需重新设计测试用例，修改测试数据即可。

2.1 总体思路

通过陪试模块的源端口按输入数据文本中内容发送数据，模拟被测模块的输入数据。

观测陪试模块的宿端口的数据，或查看输出结果文本，即可确定被测模块的输出数据。

实现方法如下：

第 1 步：读取测试输入数据文件；

第 2 步：通过陪试模块发送数据；

第 3 步：接收被测试程序的执行结果；

第 4 步：根据执行结果和预期结果进行初步判断，保存测试结果。

根据软件的基本功能，自动化测试软件执行程序的设计思路和流程图如图 1 所示。

其中文件类型主要分为以下 3 类：

1) 数据输入文件

输入数据文本文件包括 2 个文件：测试输入数据文件（test.txt）和生命信号文件（life.txt）。其中测试输入数据文件 test.txt 的格式为：标志（为 0 表示输入，为 1 表示输出，为 2 表示延时），端口号（与 mvb_confm.dat 里定义的端口号一致），字偏置，值，注释。生命信号文件 life.txt 的格式为：端口号，字偏置，注释。

其中端口号应与协议中规定的生命信号端口号一致。

2) 测试输出文件

测试输出结果文件为 result.txt，其格式为：日期时间，输入数据，日期时间，输出数据。其中日期时间为被测试模块的硬件时钟，输入数据为测试输入数据文件 test.txt 所列的输入及生命信号文件 life.txt 中的输入，输出结果为初步分析判断的结果和 test.txt 所列的输出。

3) 测试支持文件

测试支持文件（mvb_confm.dat）的格式按照 mvb 的要求配置为 slave 模式。其中，被测模块的源端口对应陪试模块的宿端口，陪试模块的宿端口对应被测模块的源端口。

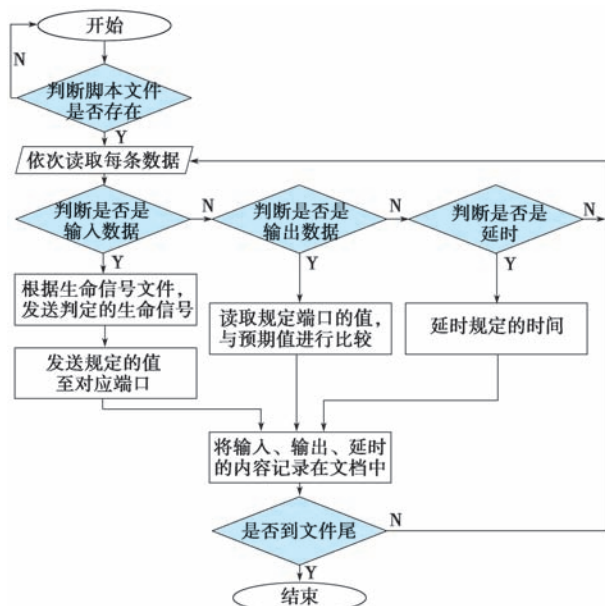


图 1 测试系统流程图

以上各类文件与程序的关系如图 2 所示。

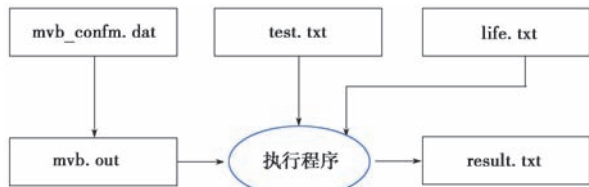


图 2 文件与程序关系图

2.2 自动化测试软件的实施

在测试长沙地铁 2 号线网络控制软件时，具体的实施过程以司机室占有功能为例来进行说明。

1) 软件需求

司机室占有功能的软件需求如下：

①当“Tc1 车司机室占有”和“Tc2 车司机室占有”同时有效，提示两端司机室同时有效故障；

②当仅“Tc1 车司机室占有”有效，输出“司机室占有状态有效”和“1 端司机室占有”；

③当仅“Tc2 车司机室占有”有效，输出“司机室占有状态有效”和“2 端司机室占有”。

2) 软件测试设计

根据软件需求进行分析，可以分为 4 种测试场景，见表 1。

表 1 测试场景

Tc1 车司机室占有	Tc2 车司机室占有
有效	有效
有效	无效
无效	有效
无效	无效

根据 4 个测试场景及相关协议，编写软件测试输入数据。

- ①测试输入数据文本文件 test.txt 的内容如下：
用例编号：TC-CSL2-0001
0,0x0131,0,0x42, 输入“Tc1 车司机室占有”有效
0,0x0631,0,0x42, 输入“Tc2 车司机室占有”有效
2,1000, 延时 1s
1,0x4006,2,0x1, 输出“两端司机室同时有效”故障
用例编号：TC-CSL2-0002
0,0x0131,0,0x42, 输入“Tc1 车司机室占有”有效
0,0x0631,0,0x0, 输入“Tc2 车司机室占有”无效
1,0x4001,11,0x4000, 输出“1 端司机室占有有效”
1,0x4006,10,0x2, 输出“司机室占有状态有效”
用例编号：TC-CSL2-0003
0,0x0631,0,0x42, 输入“Tc2 车司机室占有”有效
0,0x0131,0,0x0, 输入“Tc1 车司机室占有”无效
1,0x4001,11,0x8000, 输出“2 端司机室占有有效”
1,0x4006,10,0x2, 输出“司机室占有状态有效”
用例编号：TC-CSL2-0004
0,0x0131,0,0x0, 输入“Tc1 车司机室占有”无效
0,0x0631,0,0x0, 输入“Tc2 车司机室占有”无效
1,0x4006,10,0x2, 不输出“司机室占有状态有效”
- ②生命信号文本文件 life.txt 的内容。
0x0135,0,DIM1 生命信号
0x0635,0,DIM6 生命信号

3) 软件测试的环境搭建

完成测试设计后进行软件测试环境的搭建，测试环境示意图如图 3 所示。

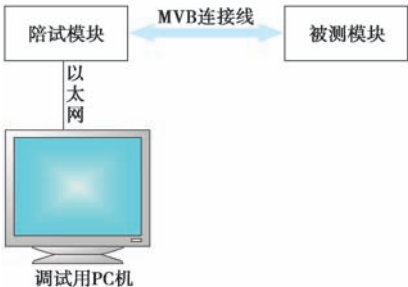


图 3 测试环境示意图

4) 软件测试自动化执行

完成测试环境的搭建后再执行软件测试，其步骤如下：

- ①将 test.out、life.txt、test.txt、mbv_confm.dat 这 4 个文件通过 FTP 工具下载至陪试模块 tffs 目录下；
- ②连接陪试模块，并用 SecureCRT 软件解锁

Flash；

③在 SecureCRT 软件中运行命令 ld</tffs/test.out，启动自动化测试软件；

④在 SecureCRT 软件中运行命令 main（执行全部测试用例）或 mian+ 测试用例编号（执行单个用例）；

⑤第 4 步中如果是执行全部测试用例，SecureCRT 软件将打印出 hello word 字样表示用例全部执行完毕，之后将 tffs 目录下的输出结果文件 result.txt 用 FTP 工具下载到本地电脑中，再通过 test_new.xlsm 文件可将 result.txt 导入到 Excel 中；

⑥第 4 步中如果是执行单个测试用例，则通过 Multiprog 工具查看被测试程序的输出，再手工记录测试结果，一般用于问题确认中。

当第一轮执行测试时，执行全部测试用例，导出测试结果文件 result.txt，转换成 Excel 格式后的测试结果如表 2。

表 2 第一轮测试结果

时间	标志	端口号	字偏置	输入值	实际值	说明
2014-12-16 T23:15:39	输入	0x131	0	0x42		输入“Tc1 车司机室占有”有效
2014-12-16 T23:15:39	输入	0x631	0	0x42		输入“Tc2 车司机室占有”有效
2014-12-16 T23:15:39	延时 1s					
2014-12-16 T23:15:39	pass	0x4006	2	0x1	0x1	输出“两端司机室同时有效”故障
2014-12-16 T23:15:43	输入	0x131	0	0x42		输入“Tc1 车司机室占有”有效
2014-12-16 T23:15:43	输入	0x631	0	0x0		输入“Tc2 车司机室占有”无效
2014-12-16 T23:15:43	pass	0x4001	11	0x4000	0x4000	输出“1 端司机室占有有效”
2014-12-16 T23:15:43	failed	0x4006	10	0x2	0x6	输出“司机室占有状态有效”
2014-12-16 T23:15:44	输入	0x631	0	0x42		输入“Tc2 车司机室占有”有效
2014-12-16 T23:15:44	输入	0x131	0	0x0		输入“Tc1 车司机室占有”无效
2014-12-16 T23:15:44	pass	0x4001	11	0x8000	0x8000	输出“2 端司机室占有有效”
2014-12-16 T23:15:44	failed	0x4006	10	0x2	0x6	输出“司机室占有状态有效”
2014-12-16 T23:15:43	输入	0x131	0	0x0		输入“Tc1 车司机室占有”无效
2014-12-16 T23:15:43	输入	0x631	0	0x0		输入“Tc2 车司机室占有”无效
2014-12-16 T23:15:43	pass	0x4006	10	0x2	0x0	不输出“司机室占有状态有效”

5) 问题跟踪及处理

通过测试结果统计分析，发现当仅有 Tc1 车司机室占有或 Tc2 车司机室占有有效时，本应输出 1/2 端司机室占有有效和司机室占有有效，但实际未输出，测试用例未通过。

分析未通过原因，通过查阅相关文档、协议和代码，发现按照协议规定，port 6 的第 10 个字偏置的第 0 位应输出“司机室占有状态有效”，但在编码时设计

人员在将司机室占有信号输出至此位置后，又将其他数值输出至此位置，导致输出的信息被覆盖，从而导致测试用例不通过。

6) 回归测试

提交问题报告给设计人员，设计人员根据问题报告来修改软件代码和相关协议，修改完成后，再将软件代码下载安装至被测模块中进行回归测试，执行全部测试用例，测试结果为全部通过。具体信息如表 3。

表 3 回归测试结果

时间	标志	端口号	字偏置	输入值	实际值	说明
2014-12-16 T23:15:39	输入	0x131	0	0x42		输入“Tc1 车司机室占有”有效
2014-12-16 T23:15:39	输入	0x631	0	0x42		输入“Tc2 车司机室占有”有效
2014-12-16 T23:15:39	延时 1 s					
2014-12-16 T23:15:39	pass	0x4006	2	0x1	0x1	输出“两端司机室同时有效”故障
2014-12-16 T23:15:43	输入	0x131	0	0x42		输入“Tc1 车司机室占有”有效
2014-12-16 T23:15:43	输入	0x631	0	0x0		输入“Tc2 车司机室占有”无效
2014-12-16 T23:15:43	pass	0x4001	11	0x4000	0x4000	输出“1 端司机室占有有效”
2014-12-16 T23:15:43	pass	0x4006	10	0x2	0x6	输出“司机室占有状态有效”
2014-12-16 T23:15:44	输入	0x631	0	0x42		输入“Tc2 车司机室占有”有效
2014-12-16 T23:15:44	输入	0x131	0	0x0		输入“Tc1 车司机室占有”无效
2014-12-16 T23:15:44	pass	0x4001	11	0x8000	0x8000	输出“2 端司机室占有有效”
2014-12-16 T23:15:44	pass	0x4006	10	0x2	0x6	输出“司机室占有状态有效”
2014-12-16 T23:15:43	输入	0x131	0	0x0		输入“Tc1 车司机室占有”无效
2014-12-16 T23:15:43	输入	0x631	0	0x0		输入“Tc2 车司机室占有”无效
2014-12-16 T23:15:43	pass	0x4006	10	0x2	0x0	不输出“司机室占有状态有效”

3 应用自动化测试的优势

以测试无锡地铁网络控制软件和长沙地铁 2 号线网络控制软件 2 个软件为例，其中在对无锡地铁网络控制软件测试时是手动执行测试用例，而在对长沙地铁 2 号线网络控制软件测试时采用自动化测试软件执行用例，2 个软件测试所用时间对比如表 4。

表 4 自动化测试及手动执行测试所用时间对比

测试流程	自动化测试软件	人工执行陪试软件
编写陪试软件	无需编写	10 天 × 2 人
编写测试数据	15 天 × 2 人	贯穿整个测试过程，大约需要 30 天 × 2 人
执行测试数据	执行命令 0.5 天	
统计分析测试结果	10 天 × 2 人	
回归测试	5 天 × 2 人	30 天 × 2 人
总计时间	30 天 × 2 人	70 天 × 2 人

经实践对比分析，自动化测试有如表 5 所列的几点优势。

表 5 自动化测试优势

手动测试	自动化测试
需编写配套的陪试软件	不需要编写陪试软件
每次执行测试用例时只能一条一条地执行	基于软件需求和协议，编写测试脚本，可重复执行，减少工作量
用例执行结果不进行保存和记录，需人工观察结果后再进行记录保存	无需人工干预，自动保存用例执行的结果，便于统计分析
软件运行时需 Multiprog 硬件狗	无需 Multiprog 硬件狗
回归测试时需重新一条一条地执行用例	回归测试时只需要输入执行命令即可全部执行用例

4 结语

本文描述的自动化测试方法主要针对网络控制软件，是对模拟物理信号进行测试的一种有益补充，有较强的实用性。通过以上说明，可以看出自动化测试软件在网络控制软件测试中不仅能高效地检测出软件中的错误，而且能减少大量的测试时间，提高测试效率，并能减少测试人力和物力投入，节省了测试成本。因此，自动化测试值得在后续相关产品的软件开发和测试中予以推广应用，并对其进行优化和完善，使之更好地应用于多种场合。

参考文献:

[1] 张志田, 向力. 自动化测试应用与创新 [J]. 中国金融电脑, 2016(10): 24-29.

[2] 王沛, 陆岳. 一种自动化测试工具使用问题的分析与解决 [J]. 电脑知识与技术, 2013(1): 77-79.

[3] 朱焕亮. 基于自动化测试框架的用例研究与实现 [J]. 制造业自动化, 2016(6): 147-149.

[4] 杨忆文. 一种自动化测试系统中为 I/O 建模及约束提取的方法 [D]. 北京: 北京邮电大学, 2015.

[5] 艾迪安. 软件自动化测试方法的研究与应用 [J]. 中国新通信, 2016(17): 16-17.

[6] 邓璐娟, 李金萌, 董东晓. 自动化测试框架技术及应用 [J]. 计算机测量与控制, 2016(9): 86-88.

[7] 欧洁. 基于 QTP 技术的自动化测试框架的研究及实现 [D]. 北京: 中国科学院大学, 2014.

[8] 王军, 孟凡鹏. 基于关键字驱动的自动化测试研究与实现 [J]. 计算机工程与设计, 2012(9): 3652-3656.

[9] 付晓, 杨海根. 自动化测试实现研究 [J]. 中国市场, 2016(34): 96-98.

[10] 胥枫. 软件自动化测试技术的研究 [D]. 无锡: 江南大学, 2015.

作者简介：游桂贞（1988-），女，主要从事轨道交通行业软件测试和软件工程的专业技术研究与应用工作。