

基于有色 Petri 网的测试案例生成方法 及其在 CTCS-3 车地通信中的应用

赵 天 时

(中国铁道科学研究院集团有限公司 标准计量研究所,北京 100081)

摘 要:CTCS-3 级列控系统作为高速铁路的中枢神经系统,是典型的安全苛求系统,对安全苛求系统进行有效测试是提高系统安全性的重要手段.鉴于车地通信在列控系统的重要性,提出一套基于有色 Petri 网的车地通信测试案例的生成方法.参考 CTCS-3 级列控系统技术规范,构建车地通信流程的有色 Petri 网模型,生成状态空间并搜索到达目标状态的路径,在路径的基础上自动生成测试案例.通过对比使用有色 Petri 网自动生成和现有的测试案例,验证了测试案例生成方法的正确性,提高了测试案例生成方法的自动化程度,为基于模型的系统测试提供了参考.

关键词:CTCS-3 级列控系统;有色 Petri 网;车地通信;状态空间;路径

中图分类号:U231 **文献标志码:**A

Test case generation method based on colored Petri net and its application in OnBoard-Trackside communication of CTCS-3

ZHAO Tianshi

(Standards & Metrology Research Institute, China Academy Railway Sciences Corporation Limited, Beijing 100081, China)

Abstract:As playing a key role in high-speed railway, CTCS-3 is a typical safety-critical system. Effective testing is an important mean to improve security of safety-critical system.Considering the importance of OnBoard-Trackside communication in train control system, a set of test case generation method for OnBoard-Trackside communication based on colored Petri net is proposed. According to CTCS-3 technical specification, models of OnBoard-Trackside communication procedure are established, then state space of the models are generated and path to target state is queried. The test case is generated automatically based on the path to target state. By comparing the test case generated by colored Petri net and the existing test cases, the correctness of test case generation method based on colored Petri net is verified, which could raise the automation degree of writing of test case. It provides the references for the model-based system test.

Keywords: CTCS-3; colored Petri net;OnBoard-Trackside communication; state space; path

收稿日期:2018-06-13

基金项目:国家自然科学基金(U1434209);中国铁道科学研究院集团有限公司科研项目(2017YJ114);中铁检验认证中心科研项目(1852ZJ1303);交控科技设计创新和学科发展基金(9907006510);中国铁道科学研究院集团有限公司科研项目(1851ZJ2904)

Foundation items:National Natural Science Foundation of China (U1434209); China Academy Railway Sciences Corporation Limited Science and Technology Foundation(2017YJ114); Chinese Railway Certification Center Funds (1852ZJ1303); TCT Funding Program (9907006510);China Academy Railway Science Corporation Limited Funding(1851ZJ2904)

第一作者:赵天时(1978—),男,辽宁铁岭人,副研究员.研究方向为高速铁路列控系统安全分析和测试方法.**email:** angel007_zhao@126.com.

引用格式:赵天时.基于有色 Petri 网的测试案例生成方法及其在 CTCS-3 车地通信中的应用[J].北京交通大学学报,2018,42(6):97—103.

ZHAO Tianshi. Test case generation method based on colored Petri net and its application in OnBoard-Trackside communication of CTCS-3[J].Journal of Beijing Jiaotong University,2018,42(6):97—103.(in Chinese)

铁路的建设发展进入一个全新的阶段,列车的速度和密度都比以前有了极大的提高.CTCS-3 级列控系统作为高速铁路的中枢神经系统,负责保证列车安全和快速的运行.为了验证其行为功能的正确性,并保证不同厂家设备的兼容性,测试成为了必不可少的环节.如何提高测试案例生成方法的自动化程度,提高人工编写测试案例的效率,已经成了目前测试行业亟需解决的问题.

近年来,CTCS-3 相关测试方法的研究主要集中于车载设备方面的测试.例如,文献[1-2]将车载设备模式转换的测试问题转化为中国邮路问题,使用遗传算法生成测试序列.文献[3]采用 Edmonds-Johnson 算法及 LINGO 工具解决车载模式转换的测试序列生成问题.文献[4]针对多端口协同测试问题提出了端口标记的时间输入输出自动机(Lp-TIOA).文献[5]提出了一种能够满足全状态、全变迁覆盖准则的测试套自动生成算法.上述关于测试方法的研究主要针对的是车载设备的测试,而 CTCS-3 采用无线通信(GSM-R)实现列车与地面之间连续、双向的信息传输,是典型的基于通信的列车运行控制系统.由于车地通信在列控系统中发展较晚,因此目前关于车地通信的测试方法的研究较少.而鉴于车地通信在列控系统的重要性,有必要提出一套针对车地通信的测试方法.

有色 Petri 网在层次化建模及处理数据方面具有很大优势,基于模型的测试方法能够直观体现系统的行为^[6].同时,有色 Petri 网模型能够生成状态空间,并将系统运行过程中会出现的状态建立成一系列由“状态”组成的序列,即“路径”^[7].因此,本文作者构建车地通信模型并生成状态空间,搜索到达目标状态的路径,其生成相应的测试案例,由于能够自动生成测试案例,该方法提高了测试案例编写的效率,从而使测试人员提前发现问题.

1 有色 Petri 网概述

有色 Petri 网是在 Petri 网的基础上加上模块化的功能,使模型能够分成不同的模块,降低了每个模块中模型的复杂程度,其相关的变量参数定义如下:

1)元素的类型用 $Type$ 表示,变量 v 的类型用 $Type(v)$ 表示;

2)变量集合 V 的绑定,即每个变量 $v \in V$ 与一个具体元素 $b(v) \in Type(v)$ 相关联,元素 $b(v)$ 属于 $Type(v)$ 类型.

有色 Petri 网模块集合可以表示为: $S = ((P^s,$

$T^s, A^s, \Sigma^s, V^s, C^s, G^s, E^s, I^s), T_{sub}^s, P_{port}^s, PT^s)$,其中,本文用到的元素为核心元素 P 和 T , P 和 T 分别代表库所和变迁, $t \in T$, 库所存储的令牌表示系统当前的状态,变迁的触发代表执行相应的动作.生成模型的状态空间.到达目标状态的路径搜索算法为:

$S_{open} \leftarrow root(M_0), S_{closed} = \varnothing, S_{path} = \varnothing / * M_0$ 表示有色 Petri 网模型 $*/$
for each node $(M) \in S_{open}$
if $\exists node', node'(M) \in S_{closed}$,
 $S_{closed} \leftarrow node(M)$, remove node (M) from S_{open} ,
 $root \rightarrow node's$ path is p ,
 $S_{path} \leftarrow p$, continue.
end if.
if no enabled transition at M ,
 $S_{closed} \leftarrow node(M)$, remove node (M) from S_{open} ,
 $root \rightarrow node's$ path is p ,
 $S_{path} \leftarrow p$, continue.
end if.
for each enabled transition t at M ,
generated $node_{child}$ who is $node's$ child node, mark t
on $node \rightarrow node_{child}'s$ path,
 $M[t > M', S_{open} \leftarrow node_{child}(M') / * M, M'$ 表示 M_0 的子模型 $*/$
 $root \rightarrow node's$ path is p ,
end for
 $S_{closed} \leftarrow node(M)$, remove node (M) from S_{open}
end for $/* S_{closed}$ 中的元素为搜索的路径 $*/$

2 基于有色 Petri 网的测试方法

本文提出路径搜索的方法生成测试案例.基本思想是:建立系统的有色 Petri 网模型,生成模型的状态空间,在状态空间上查找目标状态,搜索到达目标状态的路径.其算法如下:

输入:有色 Petri 网模型的状态空间
输出:目标状态的内容
1.查找目标状态在模型中的节点:EvalNodes(ListDead-Markings())
2.查看目标状态标识节点的具体内容:Print(NodeDescriptor n)
确定了目标状态就可以在状态空间上搜索到达目标状态的路径,路径搜索算法如下
输入:目标状态标识节点
输出:到达目标状态的路径
1.转换目标状态节点内容的表达形式:Mark.Page.Place
 $1\ n = [token]$
2.搜索到达该状态的路径:EvalArcs(ArcsInPath(1, hd(Endstate())))

3 有色 Petri 网的测试方法应用

3.1 CTCS-3 级列控系统车地通信流程

本文分析的 CTCS-3 车地通信流程参考文献[8],车载设备发起建立通信会话的流程见图 1.

1)车载设备收到通信会话管理消息(信息包

42),请求与无线闭塞中心(Radio Block Center, RBC)建立通信连接,不断重复请求直到连接成功或超过规定次数.如果不成功,应通过人机接口(Driver Machine Interface, DMI)通知司机未能建立连接;

2)一旦建立通信连接,车载设备应立即向 RBC 发送“通信会话开始”消息(消息 155);

3)车载设备接收到“系统版本”消息后(消息 32),应认为通信会话已经建立,若车载设备版本与 RBC 版本兼容,应向 RBC 发送“通信会话已建立”消息(消息 159),否则向 RBC 发送“版本不兼容”消息(消息 154).

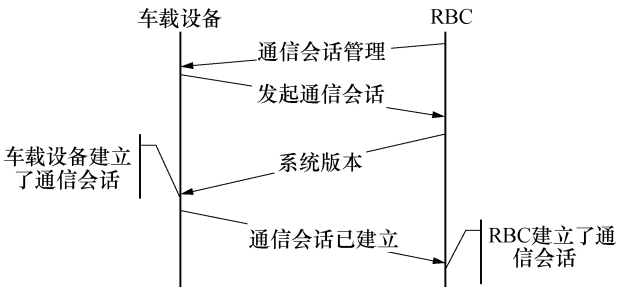


图 1 车载设备发起通信会话

Fig.1 Establishment of communication session initiated by OnBoard

车载设备终止通信会话的流程如图 2 所示^[8].

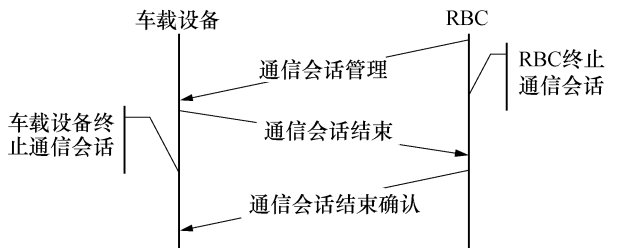


图 2 车载设备终止通信会话

Fig.2 Termination of a communication session by OnBoard

1)车载设备收到通信会话管理消息(信息包 42),向 RBC 发送“通信会话结束”消息(消息 156);

2)当接收到“通信会话结束确认”消息后(消息 39),车载设备应认为通信会话已经终止,并请求释放与 RBC 的通信连接.

3.2 有色 Petri 网模型

基于有色 Petri 网的模块化功能,可以分别构建通信会话建立和通信会话终止的模型.

本文考虑车载设备与 RBC 设备版本兼容的情况,构建的通信会话建立流程的模型如图 3 所示.

1)库所 OnBoard 表示车载设备,存储令牌“Pck42”、“Msg501”、“Msg506”、“Msg155”、“Msg32”、“Msg159”,分别对应信息包 42、消息 501、消息 506、消息 155、消息 32、消息 159.CTCS-3

中每种消息代表特定的动作,例如消息 501 代表请求建立通信连接、消息 506 代表确认建立通信连接、消息 32 代表系统版本^[9].

2)库所 Trackside 的初始令牌为“Pck42”,表明地面给车载设备发送通信会话管理消息,变迁 Manage 收到库所 Trackside 的令牌“Pck42”时激发,将令牌“Msg501”转移至库所 OnBoard 中,代表车载设备准备给地面发送请求通信连接消息(消息 501),变迁 Request 收到库所 OnBoard 中的令牌“Msg501”后激发,变迁 Indicate 收到库所 Trackside 中的令牌“Msg506”后激发,代表地面准备给车载设备发送确认通信连接消息(消息 506),一旦车载收到消息 506,表明车载和地面之间的连接建立成功.

3)库所 OnBoard 中的令牌“Msg506”激发变迁 Initial,将令牌“Msg155”转移至库所 Trackside;库所 Trackside 中的令牌“Msg155”激发变迁 Version,将令牌“Msg32”转移至库所 OnBoard;库所 OnBoard 的令牌“Msg32”激发变迁 Report,将令牌“Msg159+Msg136”转移至库所 Trackside.上述过程代表的含义为:车载给 RBC 传输发起通信会话消息,RBC 收到该消息后给车载发送系统版本消息,车载收到系统版本消息后给 RBC 发送通信会话建立消息,同时报告列车位置(消息 136,由令牌“Msg136”)表示.

4)在 CTCS-3 运行过程中,车载和地面之间传输的信息都需要下载到司法记录器(Judicial Record Unit, JRU),在模型中,车载和地面每进行一次消息的交互,代表消息的令牌都会转移至库所 JRU 中.

通信会话终止的模型如图 4 所示,由于篇幅限制,不再给出详细解释,模型中变迁的含义说明如下:SendTermMess 为车载向 RBC 发送终止消息;IsTerminated 为 RBC 结束通信会话;Record42 为 JRU 记录信息包 42;Record156 JRU 为记录消息 156;Record39 为 JRU 记录消息 39.

3.3 测试案例的生成

图 5 代表 CTCS-3 通信会话建立和终止流程的模型的状态空间,包含了模型中所有的状态信息和路径信息,因此可以在状态空间的基础上分析符合需要的状态和路径.

根据第 2 节的目标状态查找算法,各自目标状态的查找过程及结果如下:

```
输入 1:  
val_ = TextIO.output(fid, "List of target state: ")  
val _ = EvalNodes(ListDeadMarkings(), fn n =>
```


根据第 2 节的路径搜索算法,CTCS-3 通信会话建立和终止流程的路径搜索过程及结果如下:

输入 1:/* 通信会话建立流程的路径搜索过程 */

```
fun EndState a = (Mark.SessionEstablishment' OnBoard
1 a = [] andalso Mark.SessionEstablishment' Trackside 1 a
= [] andalso Mark.SessionEstablishment' JRU 1 a =
["Msg159 + Msg136"])
```

```
fun FinishCommunication() = PredNodes(ListDead-
Markings(), fn a => EndState a, NoLimit);
```

```
val _ = TextIO.output(fid, "Path to finish communica-
tion state: n")
```

```
val _ = EvalArcs(ArcsInPath(1, hd(FinishCommuni-
cation()))), fn a => STRING.output(fid, st_BE(ArcToBE
a)))
```

end;输出 1:/* 通信会话建立流程的路径搜索结果 */
Path to finish communication state:

"SessionEstablishment'Manage 1: {s="Pck42"}"

/* 车载收到通信会话管理消息—信息包 42 */

"SessionEstablishment'Record42 1: {s="Pck42"}"

/* JRU 下载记录信息包 42 */

"SessionEstablishment'S1 1: {s="Pck42"}"

"SessionEstablishment'Request 1: {s="Msg501"}"

/* 车载给 RBC 发送收到通信连接请求建立消息—消
息 501 */

"SessionEstablishment'Indicate 1: {s="Msg506"}"

/* RBC 给车载发送通信连接建立确认消息—消息
506 */

"SessionEstablishment'Initial 1: {s="Msg155"}"

/* 车载给 RBC 发送通信会话建立消息—消息 155 */

"SessionEstablishment'Record155 1: {s="Msg155"}"

/* JRU 下载记录消息 155 */

"SessionEstablishment'S2 1: {s="Msg155"}"

"SessionEstablishment'Version 1: {s="Msg32"}"

/* RBC 给车载发送系统版本信息——消息 32 */

"SessionEstablishment'Record32 1: {s="Msg32"}"

/* JRU 下载记录消息 32 */

"SessionEstablishment'S3 1: {s="Msg32"}"

"SessionEstablishment'Report 1: {s = "Msg159 +
Msg136"}"

/* 车载给 RBC 发送通信会话确认消息—消息 159,并
向 RBC 报告列车位置—消息 136 */

"SessionEstablishment'Record159 1: {s = "Msg159 +
Msg136"}"

/* JRU 下载记录消息 159 和消息 136 */输入 2:/* 通
信会话终止流程的路径搜索过程 */

```
fun EndState a = (Mark.SessionEnd' OnBoard 1 a = []
andalso Mark.SessionEnd' TrackSide 1 a = ["Msg502"] an-
dalso Mark.SessionEnd' JRU 1 a = [])
```

```
fun FinishTermination() = PredNodes(ListDeadMark-
```

```
ings(), fn a => EndState a, NoLimit);
```

```
val _ = TextIO.output(fid, "Path to finish termination
state: n")
```

```
val _ = EvalArcs(ArcsInPath(1, hd(FinishTermination
())), fn a => STRING.output(fid, st_BE(ArcToBE a)))
```

end;输出 2:/* 通信会话终止流程的路径搜索结果 */
Path to termination state:

"SessionEnd'Manage 1: {s="Pck42"}"

/* 车载收到通信会话管理消息—信息包 42 */

"SessionEnd'Record42 1: {s="Pck42"}"

/* JRU 下载记录信息包 42 */

"SessionEnd'S1 1: {s="Pck42"}"

"SessionEnd'SendTermMess 1: {s="Msg156"}"

/* 车载给 RBC 发送通信会话终止消息—消息 156 */

"SessionEnd'Record156 1: {s="Msg156"}"

/* JRU 下载记录消息 156 */"SessionEnd'S2 1: {s =
"Msg156"}"

"SessionEnd'IsTerminated 1: {s="Msg39"}"

/* RBC 给车载发送通信会话终止确认消息—消息
39 */

"SessionEnd'Record39 1: {s="Msg39"}"

/* JRU 下载记录消息 39 */

"SessionEnd'S3 1: {s="Msg39"}"

"SessionEnd'Disconnect 1: {s="Msg502"}"

/* 车载给 RBC 发送通信连接断开消息 502 */

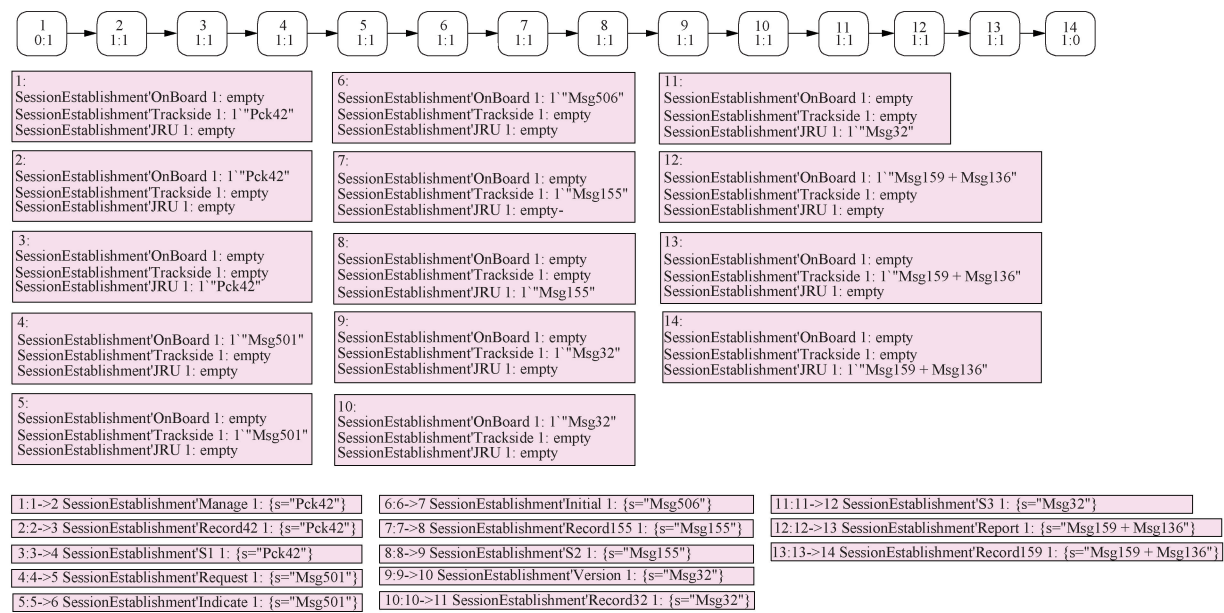
根据搜索出的路径,可以得到对应的测试案例:

1)通信会话建立流程:“车载接收信息包 42”→
“JRU 下载记录信息包 42”→“车载发送消息 501”→
“车载接收消息 506”→“车载发送消息 155”→“JRU
下载记录消息 155”→“车载接收消息 32”→“JRU 下
载记录消息 32”→“车载发送消息 159”→“JRU 下
载记录消息 159”→“车载发送消息 136”→“JRU 下
载记录消息 136”;

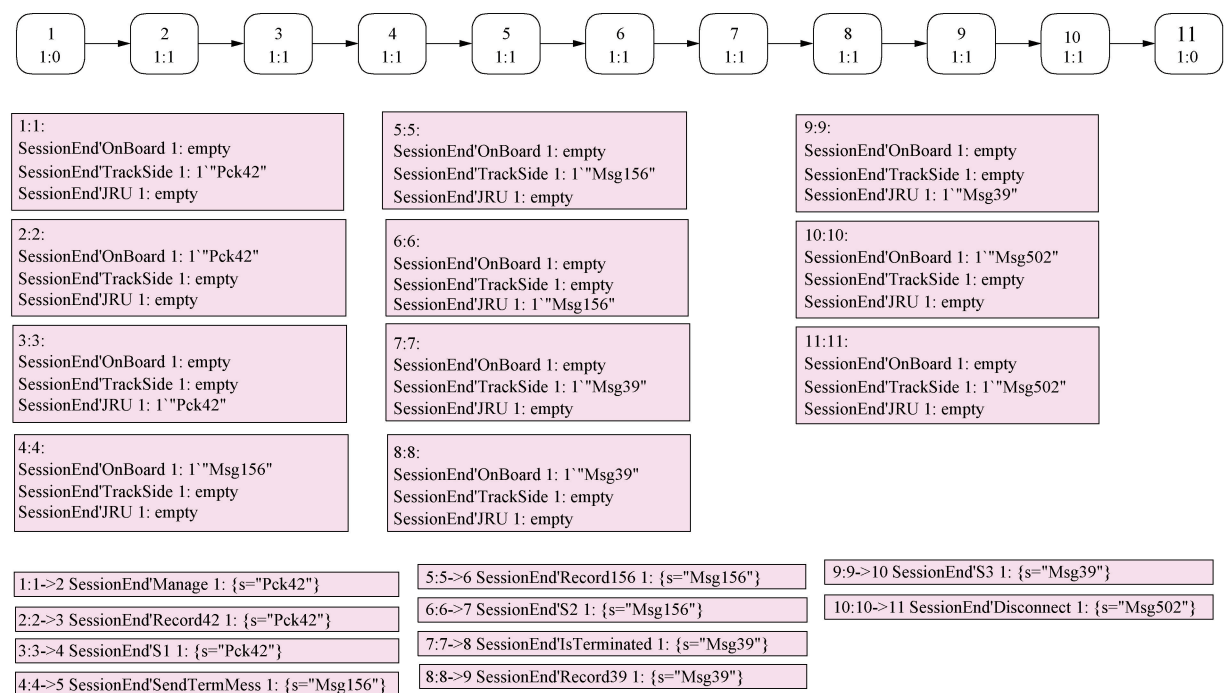
2)通信会话终止流程:“车载接收信息包 42”→
“JRU 下载记录信息包 42”→“车载发送消息 156”→
“JRU 下载记录消息 156”→“车载接收消息 39”→
“JRU 下载记录消息 39”→“车载发送消息 502”。

表 1 和表 2 是文献[10]中系统版本兼容的情况
下通信会话建立和通信会话终止的测试步骤,可以
看出,本文生成的测试案例包含表 1 和表 2 的测试
步骤,由此验证了本文提出的测试案例生成方法的
正确性.除此之外,由于文献[8]中未作具体规定,表
1 的步骤 3 和步骤 4 只写明“列车尝试建立通信连
接”和“建立通信连接成功”,表 2 的步骤 7 只写明
“列车尝试断开通信连接”,均未指明测试人员如何
实现这几步,本文参考文献[11],用消息 501(Sa-
CONNECTION.request)表示请求建立通信连接,

消息 506(Sa-CONNECTION.indication)表示确认



(a)通信会话建立模型



(b)通信会话终止模型

图 5 CTCS-3 车地通信模型的状态空间

Fig.5 State space of CTCS-3 OnBoard-Trackside communication

表 1 通信会话建立测试步骤

Tab.1 Test steps for establishment of communication session

步骤	事件描述	I/O	接口	步骤	事件描述	I/O	接口
1	接收信息包[42](通信会话管理)	I	BTM	7	接收消息[32](系统版本)	I	RTM
2	记录信息包[42](通信会话管理)	O	JRU	8	记录消息[32](系统版本)	O	JRU
3	列车尝试建立通信连接	O	RTM	9	发送消息[159](通信会话已建立)	O	RTM
4	建立通信连接成功	I	RTM	10	记录消息[159](通信会话已建立)	O	JRU
5	发送消息[155](通信会话开始)	O	RTM	11	车载系统发送消息[136] (列车位置报告)给 RBC	O	RTM
6	记录消息[155](通信会话开始)	O	JRU	12	记录消息[136](列车位置报告)	O	JRU

建立通信连接,消息 502 (Sa-DISCONNECTION,request)表示断开通信连接,生成的测试案例通过发送消息 501 和消息 506 实现通信连接的建立,发送消息 502 实现通信连接的断开.由此可以看出,本文的测试案例生成方法还可以避免自然语言编写测试案例出现的歧义.

表 2 通信会话终止测试步骤

Tab.2 Test steps for terminate of communication session			
步骤	事件描述	I/O	接口
1	接收信息包[42](通信会话管理)	I	BTM
2	记录信息包[42](通信会话管理)	O	JRU
3	发送消息[156](终止通信会话)	O	RTM
4	记录消息[156](终止通信会话)	O	JRU
5	接收消息[39](确认终止通信会话)	I	RTM
6	记录消息[39](确认终止通信会话)	O	JRU
7	列车尝试断开通信连接	O	RTM

4 结论

利用基于有色 Petri 网的测试案例生成方法,通过分析具体的 CTCS-3 车地通信流程,找出了适用于 CTCS-3 级列控系统的测试方法.

1)建立有色 Petri 网模型,包括系统存在的状态和运行过程中执行的动作.2)生成有色 Petri 网模型的状态空间,包含模型中所有的状态信息和路径信息.3)查找出目标状态的内容,在状态空间上搜索到达目标状态的路径,将路径的内容表示成消息的序列,从而得到对应的测试案例.4)对比有色 Petri 网自动生成的测试案例和现有的用自然语言描述的测试案例,验证使用有色 Petri 网生成测试案例的正确性.

在下一步研究中,将本文提出的方法用于功能更多、结构更复杂、模型更庞大的系统中,生成相应的测试案例,为测试人员提供便利.

参考文献 (References):

[1] 李伟,王海峰. CTCS-3 级列控系统车载设备测试序列的优化[J]. 北京交通大学学报, 2010, 34(2): 75—78.
LI Wei, WANG Haifeng. Optimization test sequence of CTCS-3 on-board equipment [J]. Journal of Beijing Jiaotong University, 2010, 34(2): 75—78. (in Chinese)

[2] 徐俊明. 图论及其应用[M]. 合肥: 中国科学技术大学出版社. 2010.
XU Junming. Graph theory and its application [M]. Hefei: China Science and Technology University Press, 2010.(in Chinese)

[3] 张勇,王超琦. CTCS-3 级列控系统车载设备测试序列优

化生产方法[J]. 中国铁道科学, 2011, 32(3): 100—106.

ZHANG Yong, WANG Chaoqi. The method for the optimal generation of test sequence for CTCS-3 on-board equipment [J]. China Railway Science, 2011, 32(3): 100—106. (in Chinese)

[4] 赵显琼,唐涛. 多端口形式化测试自动生成方法在 CTCS-3 车载系统中的应用[J]. 铁道学报, 2011, 33(7): 44—51.

ZHAO Xianqiong, TANG Tao. Multi-port based automatic formal testing generation and its application in CTCS-3 level on-board system [J]. Journal of the China Railway Society, 2011, 33(7): 44—51. (in Chinese)

[5] 袁磊,吕继东,刘雨,等. 一种全覆盖的列控车载系统测试用例自动生成算法研究[J].铁道学报, 2014, 36(8): 55—62.

YUAN Lei, LYU Jidong, LIU Yu, et al. Research on model-based test case generation method of onboard subsystem in CTCS-3 [J]. Journal of the China Railway Society, 2014, 36(8): 55—62. (in Chinese)

[6] 白锐,康随武. SCADE 在城市轨道交通 ATP 软件建模中的应用[J]. 铁路计算机应用,2014, 23(1): 37—40.

BAI Rui,KANG Suiwu. Application of SCADE in ATP software modeling for urban transit[J]. Railway Computer Application, 2014, 23(1): 37—40.(in Chinese)

[7] CPN ASK-CTL Manual [EB/OL].(1996-04-01)[2018-05-01]. <http://www.daimi.au.dk/designCPN/libs/askctl/ASKCTLmanual.pdf>.

[8] 中国铁路总公司. 自主化 CTCS-3 级列控车载设备暂行技术条件:TJDW 205—2018[S]. 2018.

China Railway Corporation. Interim technical conditions for ATP equipment in autonomous CTCS-3 train control system: TJDW 205—2018[S]. 2018.(in Chinese)

[9] 铁道部科学技术司铁道部运输局. CTCS-3 级列控系统系统需求规范(V1.0):TJDW 113—2008[S]. 2008.

Science and Technology Department, Transportation Bureau, Railway Ministry. Requirements specification for CTCS-3 train control system (V1.0): TJDW 113—2008 [S]. 2008.(in Chinese)

[10] 铁道部科学技术司铁道部运输局. CTCS-3 级列控系统测试案例(V3.0):TJDW 124—2009 [S]. 2009.

Science and Technology Department, Transportation Bureau, Railway Ministry. Test case for CTCS-3 train control system (V3.0): TJDW 124—2009 [S]. 2009. (in Chinese)

[11] Euroradio FIS: Class 1 Requirements [EB/OL].[2018-05-01]. [http:// www.atif.org/db/doe/com/SUBSET-037V225](http://www.atif.org/db/doe/com/SUBSET-037V225), 2003.