Archives of
Transport System
Telematics

Volume 6
Issue 4
November 2013

Model and tool based approach for the testing of cooperative systems within traffic

J. KRAUSE[a], S. MAGNUS[a], R. SCHÖNROCK[a], A. HERRMANN[a]
[a] INSTITUT F. AUTOMATION UND KOMMUNIKATION (IFAK), Department Intelligent Transport Systems, Werner-Heisenberg-Str. 1, 39106 Magdeburg, Germany
EMAIL: jan.krause@ifak.eu

ABSTRACT

The testing of cooperative transport systems is challenging because of their complexity and distributions. With TTCN3 a standard based procedure is available for specifying, executing and evaluating the testing of distributed systems such as cooperative transport systems. In the paper a standard, model and tool based solution for the testing of distributed cooperative transport systems is introduced. Based on the TTCN3 semantic, a specific model based approach for dealing with the complexity of the necessary distributed test systems was developed. With the help of some practiceoriented examples from the traffic sector this approach is presented.

KEYWORDS: cooperative systems, distributed testing, traffic simulation

1. Introduction

Cooperative systems are characterized by the connection of several (independent) actors/systems cooperating via suitable (standard based) communication mechanisms. In general, the implementation of an interesting functionality is the goal of this cooperation, for example information of road users as a result of cooperation of the (sub) system's traffic detection, strategy selection and/or traffic information distribution. Based on the ITS strategy of the EU [1, 2], the dissemination of standard based cooperative systems in the traffic sector will be strengthened and required as well as in other areas such as automotive and automation.

Currently the testing of cooperative systems is characterized by very great (manual) efforts. This great effort is one of the main reasons that occasionally testing activities cannot be carried out completely and as a result errors are only first detected and fixed at the runtime of the connected systems. With the expected transition from homogenous to heterogeneous cooperative systems of many vendors, the testing of every (sub)system and the complete functionality concerning their expected and specified functionality is becoming more and more important.

The challenge of testing cooperative systems is the fast and correct identification of the error prone (sub) system, if the response of the system compound is not conforming to the expected and specified behaviour. On the one hand, up to now there are no concepts for a standard based solution within this context. On the other hand, standard based concepts are required for the testing of cooperative systems. Based on these standard based concepts, tools and guidelines can be developed and presented dealing with the complexity of this context and supporting testing activities. Particularly in the traffic sector, the correct implementation of the systems for detection, processing and information as well as the correct cooperation of these systems are of enormous importance for their acceptance by the road user. Therefore concepts, tools and guidelines have to be developed which support the validation activities of cooperative systems.

In the paper a standard, model and tool based solution for the testing of distributed cooperative transport systems is introduced. The basis of this solution are the ETSI standard "Testing and Test Control Notation Version 3 TTCN3" [3] and results from [4, 5]. Based on the TTCN3 semantic, a traffic domain specific model based approach for dealing with the complexity of the necessary distributed test systems was developed. With the help of some practiceoriented examples from the traffic sector this approach will be presented. It is shown that this approach is useable for traffic control systems as well as Car2X applications as two important applications of cooperative systems technology.

With this approach, the handling of the complexity of the testing of cooperative transport systems should be improved. Each traffic engineer with some testing background should be able to specify, execute and evaluate complex test scenarios for the testing of cooperative transport systems.

2. Testing of traffic applications

Traffic applications are distributed systems with communicating, cooperating system components/subsystems. For the testing of distributed systems, distributed test systems are needed. With such test systems the stimulation, the observation as well as the evaluation of the behavior of the distributed test object at different points is possible. In a standard based approach, the user of such test systems specifies the required test components, test data and test the logic of the test cases at a high, abstract level. The encoding and decoding of the specified abstract test data and test logic is implemented and deployed by a special test adapter.

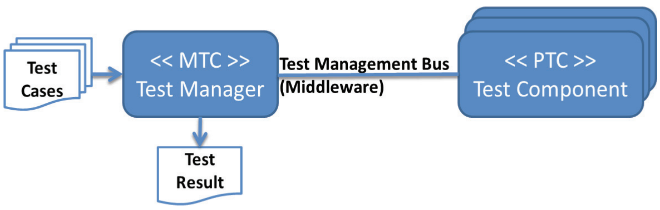

Fig. 1. Structure of a distributed test system

The European Telecommunications Standards Institute has an established test standard with TTCN3, especially in the telecommunication domain. In [6] the current application areas of TTCN3 are stated. In particular TTCN3 is well suited to describe and specify distributed test systems which are necessary for the testing of distributed test objects such as cooperative transport systems. TTCN3 is however also a very complex programming language, therefore TTCN3 is not widely used in the transport domain.

Based on the concepts of TTCN3, first steps of a testing framework for the specification, execution and evaluation of distributed test cases was developed. Thereby, it is focused on using this framework for the testing of cooperative traffic applications. The test cases are described at an abstract and implementation independent level allowing for reusing and library creation. In order to improve the usability for end users, mainly graphical methods for specification of the structure and behaviour of the test system as well as the test logic can be used. Thereby wellknown graphical concepts like message sequence charts/sequence diagrams and component diagrams [7] are used. With component diagrams the test system, the distributed test object and the connections are modelled. The sequence diagrams are used for simple specification of the distributed test cases. With these methods, the direct use of language TTCN3 is largely avoided, promising an improving and easy use of these test concepts.

The test system consists of a test manager and distributed test components. Test components are placed at accessible interfaces of the test object. The stimulation, observation and evaluation are the tasks of these test components. Other test components are used for the recording and evaluation of the communication between the distributed components/subsystems of the distributed test object.

Based on the general structure of distributed test systems (see Fig. 1, Fig. 2 illustrates the system architecture of a test system for an exemplary traffic control system at labor level. The sub systems of the traffic control system are tagged with the stereo type <<SUT>> (System Under Test). The distributed test system consists of the test manager, tagged with the stereo type <<MTC>> (Main Testing Component) and several test components, tagged with the stereo type <<PTC>> (Parallel Testing Component). The test manager is responsible for the configuration of the parallel test components as well as for the control and evaluation of the distributed test cases. The responsibility of the parallel test components are the stimulation of the detector components with realistic traffic data, the evaluation of the communication between the system components and the observation and checking of the actors (information displays, traffic lights, etc.). The scenario of Fig. 2 is realized within the research project UR:BAN(www.urbanonline.org)with the help of a traffic control system at labor level.

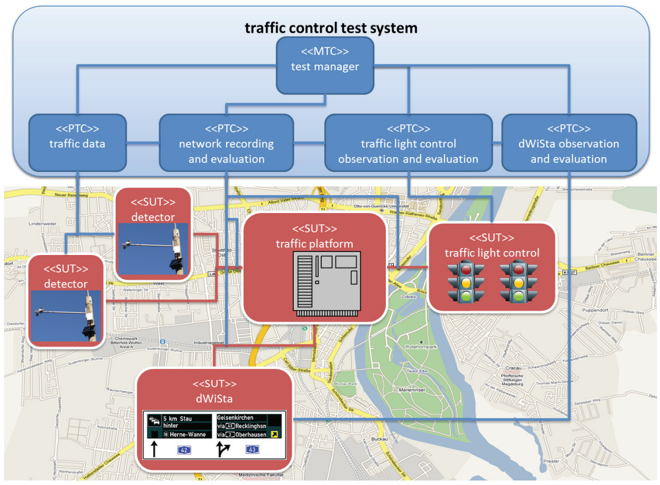

Fig. 2. Exemplary distributed test system for a traffic control system

One typical test case is illustrated in Fig. 3. In this test case the system components Detector, Traffic Platform and dWiSta (information display, german: "dynamischerWegweisermitintegriertenStauinformationen") are participated. With this test case it shall be checked whether the traffic users are rightly informed by the dWiSta about relevant traffic events (like e. g. traffic jam). The conditions and the demanded communication sequences between the subsystems/system components of the traffic control system are derived from the requirements/specification documents.

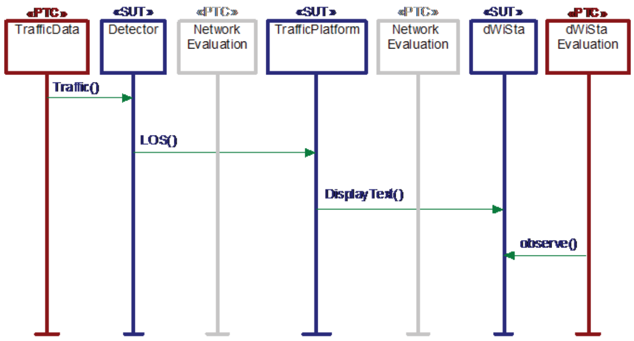

Fig. 3. Exemplary distributed test case for traffic control system of Fig. 2

First, the detectors are stimulated with suitable traffic data. After the stimulation of the detectors it is demanded that the detectors communicate a so called Level Of Service (LOS) about the current traffic state of the associated road sector to the Traffic Platform. The Traffic Platform evaluates the received LOS and sends a display text statement to the dWiSta displaying this text to inform the road users.

All sent messages are evaluated by the respective test components (PTCNetworkEvaluation). Therefore these test components must be configured from the MTCwith the expected behavior. Another test component (PTCdWiStaEvaluation) observes the display of the dWiSta and checks the observed against the expected behavior.

Test systems for Car2X applications have structurally the same architecture. Herein, distributed test components for stimulation, observation and evaluation are also necessary. But for these applications some different requirementsfor the test components, resulting from different communication systems and dynamically changing system architectures, are presented.

One example for a test system of a Car2X application is illustrated in Fig. 4. In this case the test object consists of a component RSU (Road Side Unit) and a mobile application for the relevant cyclist as the other component. The RSU shall send warnings with possible crashes to the cyclist based on the detected traffic.

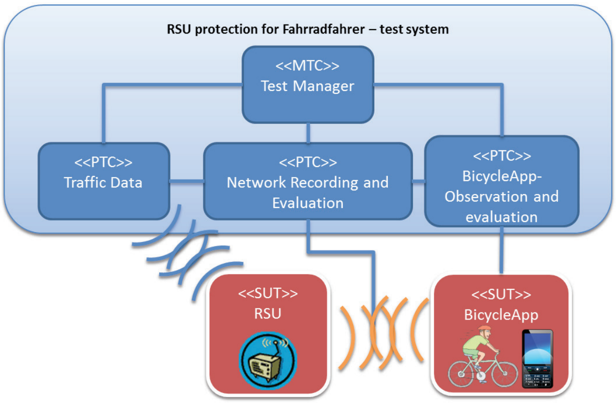

Fig. 4. Exemplary test system of a Car2X application Cyclist Protection

An exemplary test case for the Car2X application Cyclist Protection is illustrated in Fig. 5. The RSU is stimulated with suitable test data by the test component PTCTrafficData. Based on these data the RSU shall send warning messages to the relevant cyclists. These messages are recorded and checked by the test component PTCNetworkEvaluation. With another test component PTCBicycleAppEvaluation the mobile application of the cyclist Is observed in order to check whether the received warning message has been rightly displayed.

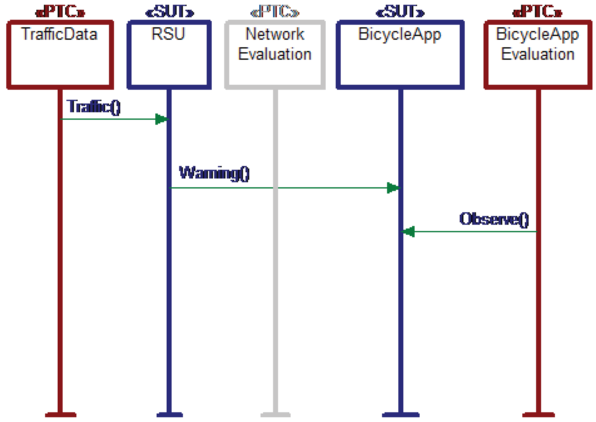

Fig. 5. Exemplary distributed test case for Car2X application CyclistProtection (Fig. 4)

3. Test data generation with traffic simulations

A further big challenge for testing of cooperative systems within the transport domain is the creation of suitable test data for the efficient stimulation of the test objects and/or components/ subsystems. Thereby especially realistic traffic data is demanded. Now there are several good traffic simulation tools (e. g. Vissim [7], SUMO [8], AnyLogic [9]) available, generally useable for the generation of realistic traffic data.

The challenge for a standard based test system using traffic simulation tools for test data generation is the parameterization, starting and the processing of the simulation results getting suitable test data. Within the research project UR:BAN first approaches for suitable concepts were realized. Thereby Vissim and Anylogic were used as simulation tools. Based on known topologies a minimum set of needed parameters for the configuration of the simulation scenarios were identified. With this approach it is possible to create and execute different suitable simulation scenarios only by changing the parameters. In order to get suitable test data the real execution of the simulation scenarios and the processing of the simulation results are done by the implemented test adapter. The implementation of the test adapter depends on the properties of the test object and can differ for each project.

Besides the test data creation the calculation of the expected behavior of the test object (test oracle) by using traffic simulations within standard based test systems is a huge challenge. It is not always possible to draw deterministic conclusions from the resulting test data. In these cases it is not possible to know the expected behavior of the test object before starting the test case. Here it is necessary to compute the needed unique test oracle at runtime of the test case based on the generated test data and the known requirement and specification documents. This computation is generally much simpler than the real application implementation because of the realized higher abstraction level. The general procedure with the use of traffic simulations for realistic test data generation and test oracle computation is illustrated in Fig. 6 for an exemplary test system.

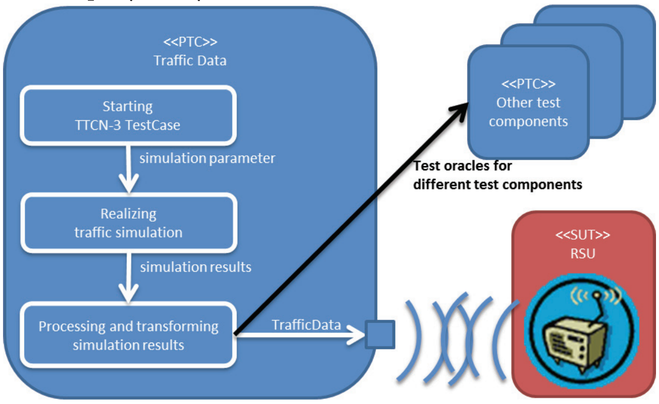

Fig. 6. Using of traffic simulations for test data generation and test oracle derivation

4. Conclusion

In this paper an approach for testing of cooperative systems of traffic/transport domain was introduced. Testing of cooperative systems extends the classical blackboxbased test procedures. Here the distribution of the test object has to be considered. It is not sufficient only to stimulate, observe and evaluate the accessible interfaces of the distributed test object. Also, the communication between the subsystems/components of the distributed test object must be considered within test activities in order to identify error prone subsystems/components of the distributed test object.

The approach of this paper is based on the wellknown standard test concepts of TTCN3. The test logic and test data are described at a high, abstract level as test model. With this, it is possible to concentrate on the test logic and to abstract from the projectspecific implementation technologies.

For the creation of relevant and realistic test data an approach for the use of traffic simulation tools and scenarios within standard based test systems was developed. Thereby for the configuration and controlling of the simulation scenarios a minimal set of needed parameters was identified. The necessary processing of the simulation results in order to get realistic and project specific test data as well as the computation of the test oracles for other test components is realized by implementations of project specific test adapters.

The developed approaches were applied to applications of traffic control systems as well as to Car2X applications. Within the research project UR:BAN the concepts will be extended and evaluated by using the approaches for testing more applications and systems from the traffic domain.

Bibliography

[1] European Commission, „Action Plan for the Deployment of Intelligent Transport Systems in Europe, COM(2008) 886," Brussels, 2008.

[2] Directive of the European Parliament and Council, „Laying down the Framework for the Deployment of Intelligent Transport Systems in the Field of Road Transport and for Interfaces with Other Transport Modes, COM(2008) 887," Brussels, 2008

[3] E. T. S. I. ETSI, 2009. Available: http://www.ttcn3.org/ (accesed 19.08.2013)

[4] Dmotion, „Düsseldorf in Motion," 08 Januar 2010. Available: http://www.dmotion.info/, (accessed 19.08.2013)

[5] OTS2, „Open Traffic Systems - OTS 2 Interface Specification," 2010. Available: www.opentrafficsystems.org, (accessed 19.08.2013)

[6] E. T. S. I. ETSI, „TTCN-3 Application Areas," 2011. Available: http://www.ttcn-3.org/ApplicationAreas.htm, (accessed 19.08.2013)

[7] PTV, „PTV Vissim," . Available: http://vision-traffic.ptvgroup.com/en-us/products/ptv-vissim/, (accessed 19.08.2013)

[8] DLR, „SUMO-Simulation of Urban Mobility," Available: http://sumo.sourceforge.net, (accessed 19.08.2013)

[9] AnyLogic Company, „Anylogic-Multimethod Simulation Software," Available: http://www.anylogic.com, (accessed 19.08.2013)

[10] (ETSI), European Telecommunications Standards Institute, 2009. Available: http://www.ttcn3.org/, (accessed 19.08.2013)