An experience on the Technology Transfer of CoFI Metodology to Automotive Domain

Fátima Mattiello-Francisco,
Ana Maria Ambrósio
National Institute for Space Research- INPE, Brazil
{fatima,ana}@dss.inpe.br

Emilia Villani
Aeronautics and Mechanical Dept. ITA, Brazil
evillani@ita.br

Eliane Martins
Computer Institute, University of Campinas, Brazil
eliane@ic.unicamp.br

Thiago Dutra,
Bruna Coelho
FIAT Betim, Brazil
{thiago.dutra, bruna.coelho}@fiat.com.br

Abstract

A large gap between the academic studies and the industry effective use of model-based testing approaches is still a reality, despite the academic effort in developing new techniques and tools. CoFI (Conformance and Fault Injection) is a model-based methodology developed to address the testing needs of embedded software in space missions. Beside others, CoFI deals with the systematization of the conformance testing and fault injection. Matured along the years, COFI is now being applied to the verification processes of automotive software testing in a big company. In this paper we present the experience on technology transfer of CoFI methodology to automotive industry aiming at vehicular software testing. In order to mitigate the risks of technology transfer we started with a feasibility study as part of the technology transfer process from the research to the practice. The challenges faced on the transference process and the lessons learned are discussed.

1. Introduction

The innovation of the automotive industry in the last decade is thanked to the growing of control and monitoring functions implemented by vehicular embedded software. The need of reducing the product time to market and achieving cost-effectiveness leads the industry to deal with software supplier's completion in the product chain.

Test activities are key elements for software verification and a high quality of the test environment is usually found in automotive industries as part of the design infrastructure. However, the specification of test cases is performed in an *ad hoc* manner, with low level of systematization.

Reliable methods and a well-established process for the automotive software acceptance testing are needed, despite the automotive industry investments in the standardization of the software specification and the interfaces.

CoFI (Conformance and Fault Injection) is a model-based testing methodology developed to address conformance and fault injection testing systematization of embedded software in space missions, aiming at the automatic test case generation. The first practical experience on using CoFI methodology was in the independent verification and validation process (IVV) established for the acceptance testing of the data handling software embedded in the space telescope developed at the Brazilian Institute for Space Research (INPE) for balloon missions [2] [3], with positive results.

CoFI guides the construction of a set of Finite State Machine (FSM) which models the expected behavior of the *services* provided by space onboard computer. From those FSM models test cases are automatically generated.

This paper reports the experience on the CoFI technology transfer to an automobilist factory, which addressed the following industrial problems: (i) Vehicular Function behavior's representativeness into FSM; (ii) modeling difficulties (iii) scalability of the artifacts as models and test-case sets (iv) need of automating (or not) the test execution.

The paper is organized as follows. Section 2 describes the technology transfer from the National Institute for Space Research to an automotive industry. Section 3 briefly introduces the CoFI methodology and the model-based process for test cases specification. Section 4 details the CoFI use in the feasibility study project following the model-based process to accomplish three vehicular functions test specification. Section 5 concludes the paper and presents the ongoing project that carries on the consolidation of CoFI technology transfer to the automotive factory.

LADC'2013, April 2-5, 2013, Rio de Janeiro, Brazil.
ISBN 978-85-7669-274-4.

2. Technology Transfer

The CoFI technology transfer to the automotive factory is the result of bilateral efforts between the V&V researcher's team from INPE, ITA and UNICAMP and the testing engineers at the automotive factory.

This transfer can be summarized in the following three steps: (1) motivation; (2) feasibility study; and (3) consolidation. These steps are respectively described in the subsections below.

2.1 Motivation

From the industry side, the interest on the CoFI methodology came up with the need to deal with the following drawbacks: (i) increasing number of vehicular software functions to be tested in short time, (ii) dependency of an expert engineer to concept the tests, (iii) anticipating the test activities, (iv) repeatability of tests, (v) deterministic actions in test activities. Moreover, it would be desirable to automate the test execution using the testing environment available in the factory plant. A testing process systematization for acceptance testing was a clear necessity to deal with increasing reduced time to market.

The critical nature of both vehicular embedded software and software onboard spacecraft attracted the automotive factory engineer team to look for current solutions being adopted in the space area.

During almost three years, informal contacts were carried on in order to, first, clarify the potential of CoFI model-based testing approach for solving the automotive industrial problem and, second, exercise particular automotive problems.

This step pointed out a number of challenges related to the adoption of CoFI methodology but it took the industry's team to give a step forward to formally establish a feasibility study project at the industry scale.

2.2 Feasibility study

The feasibly study comprised the application of the CoFI methodology in three Vehicular Function (VF). The criterion for the selection of the three VFs was the representativeness of their behavioral standard usually found in the set of VFs embedded in a car.

Focusing on modularity, the electric and electronic requirement specification of the car is broken down in parts named Vehicular Function (VF). The VF specification is textual including functional description, interfaces with other VFs, the response to the received stimulus. These stimulus are received from: external signals via CAN network or from physical devices, internal variables and physical distribution on the automobile network nodes. An example of the VF's distribution is illustrated in Figure 1.

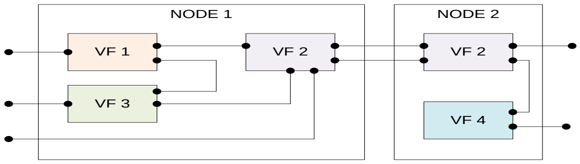

Figure 1: Exemplifying VF interactions.

The characterization of a VF in the automotive specification turns the VF similar to a *service* to be provided by a component as reactive responses to events related to a particular hardware, as expected in COFI methodology.

Three VFs were carefully chosen by the industry team, in order to assure the representativeness of the results to be obtained in the Feasibility Study. The criteria to choose the VFs were: (i) to address the challenges identified in the Contact Evolution phase and (ii) to model the most complex functions of a vehicle.

The application of CoFI methodology to the three chosen VFs was carried out during 9 months.

2.3 Consolidation

The positive results of the Feasibility Study led the industry team to give another step ahead: to apply COFI as a testing methodology as part of consolidation of the knowledge matured among the both teams in how to construct models, automatically generate and execute the model-based tests.

This step comprised the application of CoFI methodology to 13 VFs. The project has been conducted mostly by engineer's team at the automotive factory. The V&V researcher's team from INPE, ITA and UNICAMP has acted as consultants.

3. CoFI and the model-based testing process

In CoFI is a model-based testing methodology developed to guide the construction of FSM models aiming at automatic test cases generation. CoFI addresses conformance testing of reactive embedded software and robustness testing by means of fault injection.

The adoption of COFI imposes the testing specification process that comprises three phases (modeling, validation and test-case generation) as illustrated in Figure 2.

3.1 Modeling

Starting from a textual specification, CoFI proposes a systematic way to model the behavior of the system.

First, the *services* provided by the system under test must be identified.

For each identified *service*, the methodology guides the construction of models as FSMs, which represent the behavior of the *service* as response to a set of input events.

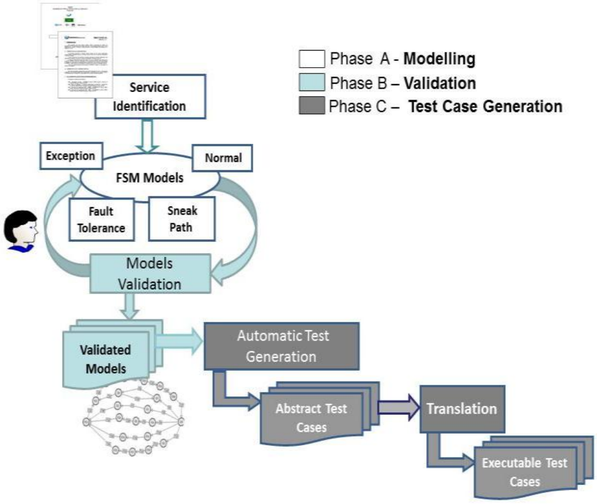

Figure 2: CoFI test specification process**.**

The *service* behavior is modeled in different perspectives: (i) normal, (ii) specified exceptions, (iii) inopportune inputs (i.e., corrects but occurring in wrong moments) and (iv) invalid inputs caused by hardware faults. The models are generally small because two levels of decomposition are taken into account: (i) the services provided by system under test and (ii) the *types of behavior*, which are named as: *Fault Tolerance*, *Sneak Path*, *Specified Exception* and *Normal*, respectively associated to the input events: invalid, inopportune, specified exceptions, normal. Moreover, it is possible to create more than one model to represent the same type of behavior of a service.

The selection of the inputs to be considered in the models must take into account the controllability and observability available in the test executing tools (or test environment). Thus, the test environment has to provide mechanisms for input events and observation of the system outputs.

3.2 Validation

The goal of the validation phase is to verify the conformance of the designed model with both the VF specification and the facilities available in the test environment. The model validation is an interactive activity with the modeling activity, which is performed in brainstorming meetings. It is assisted by human expertise in the system application domain. (See phases A and B in Figure 2). In this phase every input and output events used at least in one model are checked.

3.3 Test-case generation

In this phase, based on the FSMs abstract test-cases are automatically generated, by Condado tool [4]. After that they are translated into executable test-cases. The translation from *abstract* to *executable* test-cases requires a particular tool to be part of the test environment.

4. CoFI in the Feasibility Study

The process presented in Figure 2 was followed along the Feasibility Study step in order to evaluate the applicability of CoFI to automobilist domain.

The three chosen VFs were:

- **Tachometer**: controls the movement of the tachometer pointer in the car panel as a function of the information received by CAN networks about the car speed, the engine speed, and variables related to the occurrence of failure. It was selected because it deals with continuous variable as the engine speed and the car speed.
- **Internal Lights**: controls the internal lights as a response to user switch command, change in door status (opened/close), key status, among other events. This VF has been selected because it is characterized by a large number of timed functions that interact among them.
- **Door Locking**: controls the doors locking and unlocking as a function of mechanical pawn in the doors, doors status (open/closed), remote control commands, auto locking function related to speed, among others. It has been selected due to the complexity that arises from the interactions of the many events that affect the door locking/unlocking.

The Feasibility Study counted with the work of 2 testing engineers from automobilist factory. At research side, 2 engineers were dedicated to the modeling activities, 3 researchers composed the V&V team for the models review and report of lessons learned; and 1 engineer to be familiar with the test environment available in the automobilist factory in order to evaluate the needs for the automatic test execution.

At each phase of the test specification process (Figure 2), the team faced particular challenges. One of the main challenges identified was how to deal with temporization and real continuous variables (instead of Boolean or integer signals, i.e. discrete events). Another challenge was how to model complex VFs and deal with huge number of test cases automatically generated. For sake of space, we choose the tachometer VF to illustrate the process and relevant aspects concerning each phase.

At *modeling phase*, the project team dealt with continuous variables in the FSM models, the tachometer pointer outputs the engine speed (continuous variable). In order to avoid high pointer oscillation, the output value follows a function similar to that graphically presented in Figure 3. It is important to observe that for the same input more than one output is possible when the input variable is in the zone of curves C2 and C4. If the value of engine speed is increasing (coming from C1 curve) then curve C2 shall be used to determine the tachometer output. If the value of engine speed is decreasing (coming from C3 curve) then curve C4 shall be used.

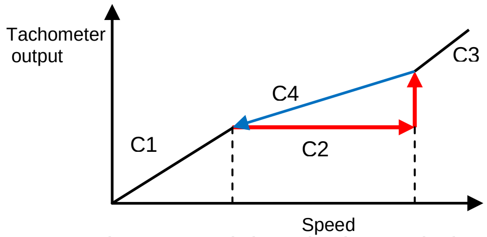

Figure 3: Continuous variables in the VF behavior**.**

The solution was the discretization of the continuous variable into intervals associated with each curve. Each interval is modeled as a different state. The possible transitions are between the states and a self-loop in each state. They represent the arrival of a new value of engine speed. After the test case generation, the speed interval is replaced by real values selected according to predefined criteria (e.g. one value near each border of the interval and one selected at random in the input interval). The corresponding output is also replaced in the output event.

In order to deal with complexity, CoFI recommends that a *type of behavior* being decomposed in two or more models. The behavior of tachometer which focus is the pointer controlling (referred as NQS) was represented by 9 (nine) FSM models: two model the *Normal* behavior; six model the *Specified Exception;* and one models the *Sneak Path* which includes the known events that occur at unexpected time. No *Fault Tolerance* models were built because the hardware faults feasible for emulation had been considered in the *Specified Exception* models.

At *validation phase*, the project team proceeded a detailed review of all relevant model parameters identified for the tachometer VF: (i) events (external input to NQS); (ii) observable outputs; (iii) actions (activities to be performed during the test execution); and (iv) specified exceptions events. Actually, the expertise of the 2 testing engineers from automobilist factory is essential for this phase accomplishing. The knowledge of the facilities and the constraints of the test environment (Test System) available in the factory plant are essential to validate the inputs and outputs that characterize every state transition in each model.

At *test case generation phase*, a set of abstract test cases were automatically generated by Condado tool for each model [4]. The transformation from abstract into executable test cases was necessary to convert each input of all Condado-generated test cases into an Excel-file input to the VeriStand tool for the test execution. For input messages from CAN, particular values were set in the Excel-file for each data. These test script efforts were required for compatibility with the test environment available at the automotive design infrastructure for test execution purposes. Table 1 presents the number of FSM models and respective test cases (TCs) generated from those models for each VF.

Table 1. FSM models and test cases produced.

Behavior Services		Normal	SpecExc	SneakPat	Total
Tachometer	Models	8	10	5	23
	#TCs	204	511	155	870
Doors Lock	Models	14	1	8	23
	#TCs	588	23	272	883
Internal Lights	Models	13	1	12	26
	#TCs	1157	57	66	1280

5. Conclusion and ongoing project

The high fidelity of the test cases set automatically generated was cost-effective, although the hard effort on the modeling and validation phases. The required detailed and systematic readings of the VF specification contribute to identify ambiguities and lack in the textual description. The high quantity of test cases produced (Table 1) is manageable by the automatic facilities available for test execution.

The feasibility study grounded the effective transfer of CoFI technology to the automobilist factory. Currently a new project comprising 13 VFs is carried on, as a consolidation of this technology transfer.

6. References

[1] M. Broy, B. Jonsson, J.P. Katoen, A. Pretschner, *Model-Based Testing of Reactive Systems*, Advanced Lectures, LNCS 3472, Springer, Germany, 2005.

[2] Mattiello-Francisco M.F.; Santiago V.; Ambrosio A.M.;.Jogaib L.; Costa R., "*A Brazilian Software Industry Experience in Using ECSS for Space Application Software Development*". 14th International Conference on Concurrent Engineering (CE2007) pg.167-174. (ISBN-978-1-84628-975-0)

[3] Ambrosio A. M., Mattiello-Francisco M. F., Martins E. "*An Independent Software Verification And Validation Process For Space Applications*". Spaceops 2008.

[4] Martins, E.; Sabião, S.B.; Ambrosio, A.M. - "*ConData: a Tool for Automating Specification-based Test Case Generation for Communication Systems*". Software Quality Journal, Vol. 8, No.4, 303-319, 1999.