# NAT2TEST: Generating Test Cases from Natural Language Requirements based on CSP

Hugo Araújo[1], Gustavo Carvalho[1], Bruno Medeiros[1], and Augusto Sampaio[1]

Universidade Federal de Pernambuco - Centro de Informática, 50740-560, Brazil[1]
{hlsa,ghpc,bmo,acas}@cin.ufpe.br

During the last fifty years, there has been a significant increase of embedded HW-SW components in critical systems. This trend increases software size and complexity, and strongly impacts systems safety and reliability. Currently, many researches are focusing on how to achieve the trustworthiness levels required for these systems. To this end, model-based testing (MBT) techniques play an important role generating tests from specification models. When MBT is applied to analyse critical systems, it is also desired to formally state the conditions to guarantee that the SUT is correct with respect to the specification by means of a conformance relation, which can be used to prove that the test generation strategy is sound. In this situation, we say we have a formal MBT strategy.

***Problem description*** Despite the benefits of formal MBT, those who are not familiar with the models syntax and semantics may be reluctant to adopt theses formalisms. Moreover, most of these models are not available in the very beginning of the project, when usually natural-language requirements are available. One possible alternative to overcome these limitations is to employ natural-language processing (NLP) techniques to derive the required models from natural-language specifications automatically.

***Proposed approach*** In this light, we propose a formal MBT strategy for generating test cases from natural-language requirements: NAT2TEST. It is a general approach that can be specialised via the adoption of different formal models and test generation tools. We dispense the need to know the syntax and semantics of the underlying notations, besides enabling early use of MBT, by means of NLP: the formal and semi-formal models internally used by the strategy are automatically generated from the natural language requirements.

The NAT2TEST strategy is tailored to generate tests for data-flow reactive systems (DFRSs [6]): a class of embedded systems whose input signals can be seen as data provided by sensors, whereas the output data are provided to system actuators. These systems can also have timed-based behaviour, which may be discrete or continuous. Our test-generation strategy comprises a number of phases. The three initial phases are fixed: syntactic analysis, semantic analysis, and DFRS generation; the remaining phases depend on the target formalism, such as SCR [7], IMR [5] or CSP [8]. This strategy is automated by the NAT2TEST tool[1] [4], which is written in Java, and its GUI is built using the Eclipse RCP framework.

[1] Available for download at: http://www.cin.ufpe.br/~ghpc/

***Empirical analyses*** The NAT2TEST strategy was evaluated considering examples from the literature, but also from the aerospace and the automotive industry. We analysed the ability to detect defects by means of mutation analysis. In general, our strategy outperformed the considered baseline: a random testing approach. We also compared our strategy with relevant commercial tools [3].

***Current and future work*** Now, we are evolving the NAT2TEST strategy to deal with hybrid systems. Addressing the generation of test cases for hybrid systems will have a vertical impact on our strategy, since each of its constituent phases needs to evolve in the light of this new domain. We have already adapted the SysReq-CNL (our controlled natural language for editing system requirements) to support references to difference and differential equations, as well as the derivation of hybrid DFRS models.

Therefore, the next major step is to develop a testing theory based on a conformance relation for hybrid systems. Some relations have already been proposed [1,9,10,2], but we would like to combine in a single relation aspects such as support for partial specifications, non-determinism, and tolerance margins. Therefore, dealing with hybrid systems within the NAT2TEST strategy seems to be an interesting, challenging and relevant research topic.

## References

1. Abbas, H., Hoxha, B., Fainekos, G.E., Deshmukh, J.V., Kapinski, J., Ueda, K.: Conformance testing as falsification for cyber-physical systems. CoRR (2014)
2. Aerts, A., Mousavi, M.R., Reniers, M.: A tool prototype for model-based testing of cyber-physical systems. In: ICTAC (2015)
3. Carvalho, G.: NAT2TEST: Generating Test Cases from Natural Language Requirements based on CSP. Ph.D. thesis, UFPE, Brazil (2016)
4. Carvalho, G., Barros, F., Carvalho, A., Cavalcanti, A., Mota, A., Sampaio, A.: NAT2TEST Tool: from Natural Language Requirements to Test Cases based on CSP. In: SEFM (2015)
5. Carvalho, G., Barros, F., Lapschies, F., Schulze, U., Peleska, J.: Model-Based Testing from Controlled Natural Language Requirements. In: FTSCS (2014)
6. Carvalho, G., Carvalho, A., Rocha, E., Cavalcanti, A., Sampaio, A.: A Formal Model for Natural-Language Timed Requirements of Reactive Systems. In: ICFEM (2014)
7. Carvalho, G., Falcão, D., Barros, F., Sampaio, A., Mota, A., Motta, L., Blackburn, M.: NAT2TEST$_{SCR}$: Test case generation from natural language requirements based on SCR specifications. Science of Computer Programming (2014)
8. Carvalho, G., Sampaio, A., Mota, A.: A CSP Timed Input-Output Relation and a Strategy for Mechanised Conformance Verification. In: ICFEM (2013)
9. Dang, T., Nahhal, T.: Coverage-guided test generation for continuous andhybrid systems. Formal Methods in System Design (2009)
10. van Osch, M.: Hybrid input-output conformance and test generation. In: FATES/RV (2006)