

# Performance metamorphic testing: A Proof of concept

Sergio Segura, Javier Troya, Amador Durán, Antonio Ruiz-Cortés

Universidad de Sevilla, Spain. {sergiosegura, jtroya, amador, aruiz}@us.es

**Palabras Clave:** Metamorphic testing, Performance testing, Search-based testing

**Lugar de publicación:** Information and Software Technology, Vol. 98, 1-4

**Impact factor:** JCR IF: 2.921 (Q1, categoría Computer Science – Software Engineering, posición 17/107, año 2018)

**DOI:** <https://doi.org/10.1016/j.infsof.2018.01.013>

**Resumen(Abstract).** *Context:* Performance testing is a challenging task mainly due to the lack of *test oracles*, i.e. mechanisms to decide whether the performance of a program is acceptable or not because of a bug. Metamorphic testing enables the generation of test cases in the absence of an oracle by exploiting the so-called *metamorphic relations* between the inputs and outputs of multiple executions of the program under test. In the last two decades, metamorphic testing has been successfully used to detect functional faults in different domains. However, its applicability to performance testing remains unexplored.

*Objective:* We propose the application of metamorphic testing to reveal performance failures.

*Method:* We define *Performance Metamorphic Relations (PMRs)* as expected relations between performance measurements of multiple executions of the program under test. These relations can be turned into assertions for the automated detection of performance bugs, removing the need for complex benchmarks and domain experts guidance. As a further benefit, PMRs can be turned into *fitness functions* to guide search-based techniques on the generation of test data.

*Results:* The feasibility of the approach is illustrated through an experimental proof of concept in the context of the automated analysis of feature models.

*Conclusion:* The results confirm the potential of metamorphic testing, in combination with search-based techniques, to automate the detection of performance bugs.