An experimental study of hyper-heuristic selection and acceptance mechanism for combinatorial $t$-way test suite generation

Kamal Z. Zamli\textsuperscript{a,}\textsuperscript{*}, Fakhir Din\textsuperscript{a}, Graham Kendall\textsuperscript{b}, Bestoun S. Ahmed\textsuperscript{c}

\textsuperscript{a} IBM Centre of Excellence, Faculty of Computer Systems and Software Engineering, Universiti Malaysia Pahang, Lebuhraya Tun Razak, 26300 Kuantan, Pahang Darul Makmur, Malaysia
\textsuperscript{b} School of Computer Science, University of Nottingham Malaysia Campus, Jalan Broga, 43500 Semenyih, Selangor Darul Ehsan, Malaysia
\textsuperscript{c} Department of Computer Science, Faculty of Electrical Engineering, Czech Technical University Karlovo n'am 13, 121 35 Praha 2, Czechia

ARTICLE INFO

Article history:
Received 21 July 2016
Revised 1 March 2017
Accepted 3 March 2017
Available online 6 March 2017

Keywords:
Software testing
t-way testing
Hyper-heuristics
Meta-heuristics
Fuzzy Inference Selection

ABSTRACT

Recently, many meta-heuristic algorithms have been proposed to serve as the basis of a $t$-way test generation strategy (where $t$ indicates the interaction strength) including Genetic Algorithms (GA), Ant Colony Optimization (ACO), Simulated Annealing (SA), Cuckoo Search (CS), Particle Swarm Optimization (PSO), and Harmony Search (HS). Although useful, meta-heuristic algorithms that make up these strategies often require specific domain knowledge in order to allow effective tuning before good quality solutions can be obtained. Hyper-heuristics provide an alternative methodology to meta-heuristics which permit adaptive selection and/or generation of meta-heuristics automatically during the search process. This paper describes our experience with four hyper-heuristic selection and acceptance mechanisms namely Exponential Monte Carlo with counter (EMCQ), Choice Function (CF), Improvement Selection Rules (ISR), and newly developed Fuzzy Inference Selection (FIS), using the $t$-way test generation problem as a case study. Based on the experimental results, we offer insights on why each strategy differs in terms of its performance.

\textcopyright 2017 Elsevier Inc. All rights reserved.

1. Introduction

Testing is an important process in software development to help identify areas where the software is not performing as expected. This process is often expensive owing to the time taken to execute the set of test cases. It has been a research focus to find suitable sampling strategies to generate a small yet efficient set of test cases for testing a software system.

Over the years, a plethora of sampling strategies has been proposed in the literature (including that of boundary value analysis, equivalence partitioning, and decision tables; to name just a few). Although useful for some classes of software testing problems, these sampling strategies have not been designed to effectively deal with faults due to interaction. For this reason, many (sampling) $t$-way strategies (where $t$ indicates the interaction strength) have been proposed in the scientific literature. Some early algebraic based $t$-way strategies exploit exact mathematical properties of orthogonal arrays. These $t$-way strategies are often fast and produce optimal solutions, yet, they impose restrictions on the supported configurations and interaction strength. The emergence of computational based $t$-way strategies ease these restrictions allowing for

\textsuperscript{*} Corresponding author.
E-mail addresses: kamalz@ump.edu.my (K.Z. Zamli), fakhruddin@uom.edu.pk (F. Din), Graham.Kendall@nottingham.edu.my (G. Kendall), albeybes@fel.cvut.cz (B.S. Ahmed).

http://dx.doi.org/10.1016/j.ins.2017.03.007
0020-0255/\textcopyright 2017 Elsevier Inc. All rights reserved.