Hyper-heuristic based Strategy for Pairwise Test Case Generation

Fakhrud Din1,2, Kamal Z. Zamli1, *
1 Faculty of Computer Systems & Software Engineering, Universiti Malaysia Pahang, Kuantan Pahang, Malaysia
2 Department of Computer Science & IT, University of Malakand, KPK Pakistan
Corresponding author Email: kamalz@ump.edu.my
Received: 31 August 2017   Accepted: 25 September 2017

Pairwise testing significantly reduces testing efforts of contemporary software systems by efficiently sampling their exorbitant number of parameter configurations. Meta-heuristic based pairwise test generation strategies appeared effective in the recent literature for pairwise testing. However, meta-heuristics require substantial information of the problem domain before producing optimal results. As alternative to meta-heuristics, hyper-heuristics have been introduced. Hyper-heuristics promote generality by using a high-level heuristic as chief selector from a set of low-level heuristics. The suitability of hyper-heuristics for optimization problems motivated us to adopt the Exponential Monte Carlo hyper-heuristic as a basis for our proposed pairwise test case generation strategy called Pairwise_HHH. Based on the published benchmarking results, Pairwise_HHH gives competitive results in many of the parameter configurations considered. Pairwise_HHH serves as our research vehicle to investigate the effective use of hyper-heuristic based algorithm for pairwise test case generation.

Keywords: Pairwise Testing, Hyper-heuristic, Exponential Monte Carlo

1. INTRODUCTION

Recent developments suggest that in many real software and hardware applications, there are potentially many pairwise interactions between software parameters, operating system as well as hardware operating environments. Although useful, most traditional existing test generation strategies (such as boundary value analysis equivalence class partitioning, decision table, etc.) are not appropriate for exposing faults that may occur due to pairwise interactions. As such, many researchers have opted to develop effective strategies to address the generation of test data for pairwise interaction testing.

For the past 15 years, computational based strategies have been popular resulting into a number of strategies (including Automatic Efficient Test Generator (AETG)1, In-Parameter-Order (IPO)2, All Pairs3, etc.).

These strategies are suitable for a range of applications but their results are not always optimal.

Lately, in line with the emerging field of Search Based Software Engineering (SBSE)4,5, strategies based on meta-heuristic algorithms (e.g. Simulated Annealing (SA)7, Genetic Algorithm (GA)6, Ant Colony Algorithm (ACA)6, Pairwise Particle Swarm Optimization Test Generation (PPSTG), Pairwise Harmony Search Strategy (PHSS)9, etc.) have started to emerge. This sudden interest for adopting meta-heuristic algorithms stemmed from the fact that the problem of finding pairwise test cases can be viewed as an optimization problem.

As expected, owing to the No Free Lunch Theorem10, the performance of each meta-heuristic algorithm is mixed. One algorithm may excel in one configuration while others in another. Typically, each algorithm comes with a set of parameters that needs to be tuned in order to balance between exploration and exploitation. As an example, a GA13 may require substantial tuning for population size and mutation/crossover rate. Particle