Migrating Birds Optimization based Strategies for Pairwise Testing

Hasneeza L. Zakaria
School of Computer and Communication Engineering
Universiti Malaysia Perlis(uniMAP)
02600 Arau, Perlis, MALAYSIA
hasneeza@unimap.edu.my

Kamal Z. Zamli
Faculty of Computer Systems and Software Engineering
Universiti Malaysia Pahang(UMP)
Lebuhraya Tun Razak, 26300 Kuantan, Pahang,
MALAYSIA
kamalz@ump.edu.my

Abstract—Exhaustive testing of all possible combinations of input parameter values of a large system is impossible. Hence, pairwise testing technique is often chosen owing to its effectiveness for bug detection. For pairwise testing, test cases are designed to cover all possible pair combinations of input parameter values at least once. In this paper, we investigate the adoption of Migrating Birds Optimization (MBO) algorithm as a strategy to find an optimal solution for pairwise test data reduction. Two strategies have been proposed; the first strategy implements the basic MBO algorithm, called Pairwise MBO Strategy (PMBOS) and the second strategy implements an improved Pairwise MBO strategy, called iPMBOS. The iPMBOS enhances the PMBOS with multiple neighborhood structures and elitism. Based on the published benchmarking results, these two strategies offer competitive results with most existing strategies in terms of the generated test size. We also noted that iPMBOS outperforms PMBOS in several parameter configurations, especially when the test size generated is relatively small.

Keywords—pairwise testing; MBO Algorithm; PMBOS; iPMBOS

1. INTRODUCTION

The past three decades has cultivated human reliance towards technology. Simple task such as making a phone call to intricate task such as launching spacecraft were done with the help of computer software. As we are more and more dependence on software, failures greatly reduced system availability and can cause huge money loss. Thus, software companies invest enormous effort and resources on software testing and bug detection prior to releasing software[1].

Exhaustive testing i.e. testing all possible combinations of inputs and execution paths is impossible for real-world software testing as the number of test input combinations is enormous. If we were to test exhaustively we need more money, time and effort. As a solution, combinatorial testing is being adopted to generate the required tests with minimum possible combinations. The fundamental rule of this form of testing is that not every parameter contributes to every failure and most failures are caused by interactions between relatively few parameters[2].

Pairwise testing (also referred to as all-pairs or 2-way testing) is a popular approach to combinatorial testing problems. Pairwise testing is a test case generation technique that is based on the observation that most faults are caused by interactions of two parameter values. According to Othman and Zamli[3], a significant number of work have focused on pairwise (t=2) strategies (e.g. Orthogonal Array Test System (OATS), IRPS, AllPairs, In- Parameter-Order (IPO), Test Case Generator (TCG), OATSGen, ReduceArray2, Deterministic Density Algorithm (DDA), CTE- XL, rdExpert, and SmartTest). As interaction strength is limited to t=2, pairwise strategies often yield the minimum test set as compared to other interactions. However, the problem of finding the minimum number of test cases for pairwise testing is an NP-complete problem[4]. Hence most of the existing solutions although has been found in reasonable time but it is not necessarily an optimal solution.

Strategies based on nature inspired metaheuristic algorithms for test data generation has been investigated by several authors. Examples of older works are by Shiba[5] with both Genetic Algorithm(GA) and Ant Colony Algorithm(ACA) and Cohen[6] with Simulated Annealing(SA). Examples of recent works are by Ahmed and Zamli[7] with Particle Swarm Optimization(PSO) and Alsewari and Zamli[8] with Harmony Search(HS). Most of the strategies have been proven to produce a good performance.

Complementing existing work, we adopt the Migrating Birds Optimization (MBO) algorithm[9] for our pairwise strategies called Pairwise MBO Strategy(PMBOS) and improved PMBOS(iPMBOS). Among the advantages of MBO which justifies our choice are:

- MBO offers a unique sharing mechanism where the best unused solutions are shared with the next solutions. The benefit mechanism makes it possible to explore the more promising areas of the search space in more detail.
- MBO offers parallel processing where a number of solutions run in parallel.
- Previous research shows that MBO perform well as compared to other nature inspired metaheuristic algorithms for various engineering problems[10].

This paper discusses the design, implementation and assessments of MBO algorithm based strategies: PMBOS and iPMBOS. Based on the published benchmarking results, the