A Pairwise Test Suite Generator based on Melody Search Algorithm

Toh S. Yuen[1], Foo W. Wen[1], Goh G. Hau[1], AbdulRahman A. Alsewari[1*]
Software Engineering Research Group, Software Engineering Department
[1] Faculty of Computer Systems and Software Engineering
Universiti Malaysia Pahang
alsewari@ump.edu.my

Abstract. Melody Search Algorithm (MS) has been adopted to solve many optimization problems. This paper will introduce MS as a pairwise testing strategy called a Pairwise Test suite generator based MS (PTMS). A pairwise testing is an operative approach in the combinatorial test suite construction. It will reduce the construct test suite size to save the testing time with effective defects detecting. The proposed strategy generate the test suite with real-value variables. A comparative evaluation shows the PTMS efficient in constructing a minimum test suite.

1 Introduction

In the information technology era, there is a huge influence of high technology and artificial intelligence when creating new software products in the whole world. This brand new method provides an effective way from bringing high quality software products to the end user. Many fields also rely on this method, especially in the Research and Development (R&D) area. As an evidence, there are many manual processes has been taking place by certain software products or artificial intelligence. Basically, every created product is operating by the combination of hardware and software to implement each feature [1]. There is a closely relationship between hardware and software, both are playing an important roles to avoid failure exists.

Failure of products will happen when a human action produces some error or bug in the software and this will lead to the defects which will cause a failure occur when executed. This problem will cause serious damage of system function, and will involve higher cost and loss of time especially for a critical system. Therefore, software testing takes first priority in any Software Development Life Cycle (SDLC) to make sure the quality of software and to prevent the failure of the software.