

The Measurement of Software Complexity through Program Decomposition

Chi-Ming Chung

TamKang University, R.O.C.

Timothy K. Shih

TamKang University, R.O.C.

Ying-Hong Wang

TamKang University, R.O.C.

Wei-Chuan Lin

TamKang University, R.O.C.

Ying-Feng Kou

TamKang University, R.O.C.

Abstract

When the cost of software continues to increase, software development techniques become more important. In software life cycle, about two-thirds of software development cost is spent on testing and maintenance [2]. The purpose of our research is to improve the quality and reliability of software thus software cost is reduced [4]. One of them is utilizing software metrics to monitor software development process and to improve the quality of the program. In software engineering, various "metrics" have been introduced as an attempt to measure the complexity of programs. For example, the "Lines of code" strategy has been widely applied in measuring software complexity. But it is not precise enough. The "Halstead's metrics" is too complex to implement [17], so it is rarely used. The McCabe's cyclomatic complexity [21] is well known metric and has been widely applied because it is more accurate and easier to implement. However, the method still has shortcomings [3]. In this paper, a more accurate and easier measuring methodology - *Path complexity* is proposed and an algorithm of the methodology is presented. Moreover, the methodology points out that how many individual complete paths of a program need to be tested.

Keywords: Software Engineering, Software Metric, Line of Code, McCabe's Cyclomatic Complexity, Path Complexity, Structural Programming, Control Flow Analysis.