

Measuring Complexity of Object Oriented Programs

Sanjay Misra and Ibrahim Akman

Department of Computer Engineering, Atilim University, Ankara, Turkey
smisra@atilim.edu.tr, akman@atilim.edu.tr

Abstract. In this paper, a metric for object oriented language is formulated and validated. On the contrary of the other metrics used for object oriented programming (OOPs), the proposed metric calculates the complexity of a class at method level and hence considers the internal architecture of the classes, subclasses and member functions. The proposed metric is evaluated against Weyuker's proposed set of measurement principles through examples and validated through experimentation, case study and comparative study with similar measures. The practical usefulness of the metric is evaluated by a practical framework.

Keywords: Software metrics, method complexity, object oriented programming (OOP), cognitive weights, validation.

1 Introduction

Software metrics have always been important for software engineers to assure software quality. Normally, software engineers attempt to derive a set of indirect measures that lead to metrics that provide an indication of quality of some representation of software. The quality objectives may be listed as performance, reliability, availability and maintainability [20], and are closely related to software complexity. Today, the relevant literature provides a variety of metrics to compute the complexity of software for OO software development. OO software development normally requires different software metrics than conventional ones since it adopts different approaches [15]. We refer the reader to [1], [2], [6], [7], [12], [14], [15],[19] for OO complexity metrics. All the reported complexity metrics are supposed to cover the correctness, effectiveness and clarity of software and to provide good estimate of these parameters. Out of the numerous proposed metrics, selecting a particular complexity metric is again a problem, as every metric has its own advantages and disadvantages. There is an ongoing effort to find such a comprehensive complexity metric, which addresses most of the parameters of software.

Although complexity of methods of OO software development directly effects understandability of the software, complexity metrics based on the method level have not been studied carefully yet. For OO system, traditional metrics are applied to methods that comprise the operation of a class. However, available metrics on the method level like cyclomatic number, line of code etc., do not capture the features of object-oriented system, which differentiate program structure than that of procedural language and are under several criticisms [18], [22], [23],[27]. This may be the underlying reason for the failure of the traditional complexity metrics on method level for OO code.