A Theoretical and Empirical Analysis of a TTCN-3 Coupling Metric

M.S. ABOUTRAB, S. COUNSELL AND T. HALL
Department of Information Systems and Computing
Brunel University
Uxbridge, Middlesex, UB8 3PH, UK

Abstract: - Testing and Test Control Notation (TTCN-3) is a testing language for test suite specification. A Template Coupling Metric (TCM) was recently proposed by Zeiss et al. for measuring the degree of template coupling in TTCN-3 test suites. In this paper, we validate the TCM against validation criteria (both theoretical and empirical). The former was achieved through applying a set of validation properties - the latter by running experiments against a large TTCN-3 file. Results suggest that the TCM did not satisfy the majority of the theoretical validation properties and showed various flaws empirically. As such, the TCM may not useful in extracting the degree of template coupling and should be modified to improve its characteristics.

Key-Words: TTCN-3, TCM, metrics, coupling, theoretical validation, empirical validation.

1 Introduction
Testing and Test Control Notation (TTCN-3) is a testing language developed by the European Telecommunications Standardization Institute (ETSI) as a result of the need to handle complex test cases [16, 11, 25]. Assessing the quality of TTCN-3 test suites is still a problem and suitable metrics for the TTCN-3 context are needed [26]. Testing systems including TTCN-3 test suites remain largely untouched by use of metrics or empirical validation [1, 13, 14]. Recently, Zeiss et al. proposed a new metric, called the Template Coupling Metric (TCM), in a TTCN-3 context. The TCM analyses the degree of coupling within a TTCN-3 template structure in order to improve overall TTCN-3 structure quality and make it easy to read and maintain [26]. The research we conduct validates the TCM and tries to answer two questions: a) Does the TCM help developers identify precisely the degree of coupling within TTCN-3 templates structure and the estimation of effort for the maintenance process? b) How does the TCM react to different configurations of the TTCN-3 file?

2 Foundations and Related Work
2.1 Software Metrics
Generally, software metrics are considered a necessary feature not only in planning a project, but also for supporting managerial decision-making, predicting development process cost, evaluating productivity and assessing quality [13, 17]. Many metrics have been proposed for measuring coupling between software entities; coupling is an important concept in any maintenance process and its reduction can contribute to the avoidance of faults in a system [4]. Coupling can be defined as the degree of interdependence between objects and should be kept to a minimum level for high quality software [4, 6, 9, 19]. The more coupling, the higher changes code may require, the more difficult to reuse ‘modules’ it will be and the more complex software maintenance will also be [7].

On the other hand, any proposed metric should be validated theoretically, empirically (or both) for several reasons. Firstly, each proposed metric may make different underlying assumptions about the systems it studies. Secondly, since the majority of metrics are based on a researchers’ intuition only, software metrics are often misunderstood and misused. Finally, metrics are often vague with respect to what they are supposed to measure [1, 3, 4]. Theoretical validation has been addressed various times through frameworks against which the metrics can be considered valid or invalid [5, 8, 9, 18, 20]. Many examples can also be extracted from the literature regarding empirical validation [6, 9, 18].

2.2 TTCN-3
The Testing and Test Control Notation v.3 (TTCN-3) combines the best parts of the two previous versions and extends them with strong textual syntax for classifying test properties [16, 24]. TTCN-3 constructs are built like other programming languages and allow concepts such as built-in data matching, concurrent execution of test cases and distributed test system architecture. As a standardized test language, TTCN-3 components are clearly defined, specified
and suitable for a wide range of applications and are not just restricted to telecommunication systems. TTCN-3 can also be used during the entire product life-cycle [11, 16]. Consequently, TTCN-3 has become a favourite testing language in several industrial domains [2, 23]. However, TTCN-3 test files suffer from problems related to readability, usability and maintainability. For instance, Motorola developers who converted a legacy test suite to TTCN-3 produced a TTCN-3 file with 60,000 lines of code [26]. Moreover, since TTCN-3 test data definitions occupy 60% - 70% of the test specification, TTCN-3 modules are generally large. That leads to multiple problems such as increases in the length of the compilation and inherent complexity [21, 25]. As a result, practitioners have problems maintaining TTCN-3 test suites suffering from high coupling between component definitions and references [2, 10]. Removing redundant and unused data definitions and reducing coupling between test data and test behaviour requires using suitable metrics and therefore increases generated test suite quality and maintainability [21, 25].

3 TTCN-3 TCM
Zeiss et al. proposed the Template Coupling Metric (TCM) shown in equation (1) and measures the degree of coupling between template definitions and template references [26].

\[
TCM = \frac{\sum_{i=1}^{n} \text{score}(\text{stmt}(i))}{n} \tag{1}
\]

\[
\text{where}
\text{score}(\text{stmt}(i)) =
\begin{cases}
1, & \text{stmt references a template without parameters} \\
2, & \text{stmt references a template with parameters} \\
3, & \text{stmt uses an inline template}
\end{cases}
\]

\[n: \text{the number of templates references}\]

Templates (defined as constructs used to either send a set of distinct data or test whether the received data matches template specifications) have several types [12]. The TCM thus considers templates without-parameters, parameterized and inline. Each contributes in a different way to the degree of coupling between test data and test behaviour. From equation (1), the TCM ‘threshold’ values fall between 1 (when behaviour statements use just without-parameter template references) and 3 (when behaviour statements just use inline template references).

According to [26], to improve TTCN-3 maintainability, the aim should be to increase the number of without-parameter template references (the value of TCM should be as close to 1 as possible). When readability has high priority, the TCM value should be close to 3. However, Zeiss et al. admit that they have only a rough idea about appropriate boundaries for the TCM metric and suggest further work needs to be done on this.

4 TCM Theoretical Validation
Our research addresses the validation of the TCM against criteria proposed by Kitchenham and Pfleeger’s [20] properties. The approach by which the TCM was validated was similar to that used in [18]. Kitchenham’s and Pfleeger’s properties comprise four properties for direct measurements and a further four for indirect measurements. Since the main purpose of TCM is to measure maintainability and readability of a TTCN-3 system, the TCM is considered an indirect measurement. We thus validated the TCM using all eight properties. To satisfy Property 1, two TTCN-3 systems with different amount of templates references should have different TCM values. Consider two TTCN-3 systems with different amounts of references.

System 1:
No. of without-parameter template references = 100.
No. of parameterized template references = 100.
No. of inline template references = 100.

\[
TCM = \frac{(1 \times 100) + (2 \times 100) + (3 \times 100)}{100 + 100 + 100} = 2
\]

System 2:
No. of without-parameter template references = 150.
No. of parameterized template references = 0.
No. of inline template references = 150.

\[
TCM = \frac{(1 \times 150) + (2 \times 0) + (3 \times 150)}{150 + 150} = 2
\]

The TCM fails to distinguish between these two systems despite having different numbers of references. This was an interesting feature of the TCM metric revealed by our analysis. As a result, TCM does not satisfy Property 1. Property 2 is satisfied since the greater number of template references, the greater the value of the TCM (from the TCM equation). With regards to Property 3, each added reference does not contribute in an equivalent way to the TCM value due to the existing threshold of
the TCM. The closer the TCM value is to the threshold, the smaller the interval is between two TCM values. As a result, Property 3 is not satisfied. Mathematically, Property 4 is satisfied because two systems could exist with the same amount of references and therefore generate the same TCM value. Property 5 is not satisfied because Zeiss et al. does not provide the underlying model upon which the TCM was based. Properties 6 and 8 are satisfied since the TCM uses scale correctly and it uses units consistently. Finally, Property 7 is not satisfied because the TCM without any template references will be undefined and therefore unexpected discontinuities will exist. From validating TCM against Kitchenham’s properties, the TCM is not theoretically valid. However, accepting that some of the validation criteria were satisfied suggests that our understanding of the TCM might benefit empirical analysis to support this theoretical analysis.

5 TCM Empirical Validation
The Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) system was used to calculate the TCM. The system consisted of three files: the first two comprised the template definitions and its component definitions, respectively. The third file was the main module which imports definition files and executes test cases. The TCM calculation process was automated using a Java-based tool implemented using the NetBeans environment [22]. In our experiment, the independent variables were: Number of Parameterized Template References (NPTR), Number of Without-Parameter Template References (NW PTR) and Number of Inline Template References (NITR). The dependent variable in each case was the TCM. Our investigation of the TCM metric was therefore underpinned by analysing the effect on its value of adding and removing the number of occurrences of each template type in turn (whilst keeping the number of the other two template types constant). The main module was modified in a controlled way in each case.

Firstly, we note from Table 1 that the TTCN-3 testing system was large. The Lines of Code (LOC) of the main module, 28485, was far larger than the LOC in the template definition file, 5619. This was because the main module file held all behaviour statements including executable test cases, whereas the template definitions file held just the template definitions. Secondly, no inline templates existed within the system (a surprising and revealing aspect of the study). The rationale beyond that might be simply to ease system maintainability and enhance the reusability of the data within templates by only using the other two types of template. Using inline templates might pose a maintenance problem and limit reuse of those templates. Thirdly, it was observed that many template references were repeated within the main module file. The repetition could simply be related to reusability of system components. However, it is not clear yet whether this is a general attribute of TTCN-3 systems or it is specific for the system under study. This again was an interesting feature to emerge from our analysis. Fourthly, the TCM value was found to be approximately 1.74 (see Table 1). Since this value falls mid-way between the threshold values of 1 (the TTCN-3 file is easy to maintain) and 3 (it is easy to read), it is difficult to interpret this TCM value.

The next step was achieved by incrementing and decrementing the Number of Parameterized Template References (NPTR) while keeping the other two variables (NW PTR = 394 AND NITR = 0) constant to examine how sensitive the TCM metric is to NPTR changes. Table 2 clarifies the main properties of NPTR and the TCM due to increments in the NPTR.

<table>
<thead>
<tr>
<th>Description Statistics</th>
<th>NPTR</th>
<th>TCM Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Maximum</td>
<td>1608</td>
<td>1.8032</td>
</tr>
<tr>
<td>Minimum</td>
<td>1108</td>
<td>1.7377</td>
</tr>
<tr>
<td>Interval</td>
<td>500</td>
<td>0.0655</td>
</tr>
<tr>
<td>Median</td>
<td>1358</td>
<td>1.7751</td>
</tr>
<tr>
<td>Mean</td>
<td>1358</td>
<td>1.7735</td>
</tr>
<tr>
<td>Std. Deviation</td>
<td>146.5</td>
<td>0.0191</td>
</tr>
</tbody>
</table>

Table 2: TTCN-3 System characteristics

We notice that the TCM value interval (the difference between the maximum and the minimum values) was only 0.0655 due to the stepped increases in NPTR. Realistically, the total change in the TCM values was therefore too small to be of any real note and illustrated the insensitivity of the TCM metric when the main module was significantly changed in...
composition. Figure 1 shows the relationship between the TCM and the NPTR. A linear relationship is not evident from the graph and signifies a positive correlation between TCM and NPTR through the 100 samples. The larger the NPTR values become, the more the gradient of the curve decreases and the smaller the changes in the TCM values. In other words, the more the TCM approaches the NPTR threshold of 2 (i.e., the TTCN-3 system is, in theory, equally maintainable as it is readable) the smaller the gap between the TCM values and the more difficult it becomes to distinguish the differences between TCM values.

As well as observing the values of the TCM when values of the NPTR were incremented, we also observed the TCM when values of the NPTR were successively decremented. From Table 3, we note that the TCM value range was 0.7377 due to our stepped decrease in the NPTR. Although this is significantly larger than the TCM interval due to the NPTR increment, the total change in the TCM values was too small for any real conclusions to be drawn.

The same relationship between the TCM and the NPTR (when decremented) is shown in Figure 2. As the TCM approaches the NWPTR threshold of 1 (i.e., the TTCN-3 system is easy to maintain but difficult to read), the greater the gap is between the TCM values. This is in complete contrast to the result for incremented changes where the gap between TCM values became smaller.

After studying the incremented and decremented changes in NPTR and the corresponding TCM values, we note that the TCM values did not reflect NPTR changes properly. Although there is a positive correlation between the TCM and NPTR, it is still difficult to interpret the TCM value to identify the degree of template coupling and therefore the effort the developer might need to spend on maintenance or refactoring [15].

The next step of our experiment was to examine how sensitive the TCM metric was to changes (increments and decrements) in the Number of Without-Parameter Template References (NWPTR) while keeping the other two variables (NPTR = 1108 AND NITR = 0) constant. Table 4 shows the main statistical properties of NWPTR and the TCM due to increments in the NWPTR. We notice that the TCM value interval was only 0.1843 due to the stepped increases in the NWPTR. Realistically, the total change in the TCM values was once again too small to be of any real note and again illustrated the insensitivity of the TCM metric when the main module was significantly changed in composition.
the more the gradient of the curve decreases and the smaller the changes in the TCM values. As a result, the more the TCM approaches the NWPTR threshold of 1 (i.e., the TTCN-3 system is easy to maintain but difficult to read), the smaller the gap between the TCM values and the more difficult it becomes to distinguish differences between TCM values.

As well as observing the values of the TCM when values of the NWPTR were incremented, we also observed the TCM when values of the NWPTR were successively decremented. Table 5 shows the main statistical properties of the NWPTR and the TCM following decrements in the NWPTR. We note that the TCM value interval was 0.2623 due to our stepped decrease in NWPTR. Although this is greater than the TCM range due to the NWPTR increments, the total change in the TCM values was again small.

As the TCM approaches the NWPTR threshold of 2, the greater the gap between the TCM values.

From examining the incremented and decremented changes in NWPTR and the corresponding TCM values, we conclude that the TCM values do not reflect NWPTR changes properly. In addition, there was a negative correlation between the TCM and NWPTR even though NWPTR contributes positively to the degree of template coupling. In other words, the TCM in some circumstances does not measure coupling at all. As a result, it is still difficult to interpret the TCM value to identify the degree of template coupling and therefore the effort a developer might need to spend on maintenance or refactoring [15].

We examined next how sensitive the TCM metric was to changes (in this case just incremented changes since the original value of NITR is 0) in the Number of Inline Template References (NITR) while keeping the other two variables (NWPTR = 394 AND NPTR = 1108) constant. Table 6 shows the main properties of NITR and the TCM due to increments in the NITR. The TCM value interval as a result was 0.3152 due to those stepped increases. The total change in the TCM values was thus too small to be of any interest.

The relationship between the TCM and NITR is shown in Figure 5. The curve signifies a positive correlation between TCM and NITR through the 100 samples. The larger the NITR values become, the more the gradient of the curve decreases and the smaller the changes in the TCM values.

---

### Table 5: TCM due to decrements in NWPTR

<table>
<thead>
<tr>
<th>Descriptive Statistics</th>
<th>NWPTR</th>
<th>TCM Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>Maximum</td>
<td>394</td>
<td>2</td>
</tr>
<tr>
<td>Minimum</td>
<td>0</td>
<td>1.7377</td>
</tr>
<tr>
<td>Interval</td>
<td>389</td>
<td>0.2623</td>
</tr>
<tr>
<td>Median</td>
<td>194</td>
<td>1.8510</td>
</tr>
<tr>
<td>Mean</td>
<td>194.01</td>
<td>1.8576</td>
</tr>
<tr>
<td>Std. Deviation</td>
<td>114.7</td>
<td>0.0763</td>
</tr>
</tbody>
</table>

### Table 6: TCM due to increments in NITR

<table>
<thead>
<tr>
<th>Descriptive Statistics</th>
<th>NITR</th>
<th>TCM Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Maximum</td>
<td>500</td>
<td>2.0529</td>
</tr>
<tr>
<td>Minimum</td>
<td>0</td>
<td>1.7377</td>
</tr>
<tr>
<td>Interval</td>
<td>500</td>
<td>0.3152</td>
</tr>
<tr>
<td>Median</td>
<td>252.5</td>
<td>1.9194</td>
</tr>
<tr>
<td>Mean</td>
<td>252.5</td>
<td>1.9120</td>
</tr>
<tr>
<td>Std. Deviation</td>
<td>145.057</td>
<td>0.0907</td>
</tr>
</tbody>
</table>
As a result, the more the TCM approaches the NITR threshold of 3 (i.e., the TTCN-3 system is readable but difficult to maintain), the smaller the gap is between the TCM values and the more difficult it becomes to distinguish the differences between TCM values.

Studying the incremented changes in NITR and the corresponding TCM values, we conclude that the TCM values do not reflect NITR changes properly. There is a positive correlation between the TCM and NITR even though the NITR contributes negatively to the degree of template coupling. As a result, it is still difficult to interpret the TCM value to identify the degree of template coupling and therefore the effort the developer needs to spend on maintenance or potential refactoring.

Finally, it is worth noting that if we change any parameter within a template, it should be followed by changing its corresponding references. In other words, the number of parameters contributes significantly to the degree of template coupling. However, because the TCM equation only considers the number of templates references without taking into account the number of parameters, it can be concluded that the TCM does not reflect the changes in numbers of parameters.

In summary, although we note from the analysis that the TCM values were more sensitive to the decremented changes in TCM variations (NPTR, NWPTR and NITR) than the incremented changes, the conclusion which could be then made is that the TCM is invalid empirically and should thus be modified to improve its characteristics.

6 Conclusions and Further Work
The research addressed and discussed the results of the theoretical and empirical validation of the template coupling metric in a TTCN-3 context. The research succeeded in applying a set of validation properties - those of Kitchenham and Pfleeger and found that the TCM did not meet many of their validation properties. From the empirical investigation of the TCM we found that, firstly, the TCM showed a clear insensitivity towards changes in template variations. Secondly, the overall changes in the TCM values did not reflect the significant changes in composition of the main module. Thirdly, the TCM in some circumstances does not measure coupling at all (e.g., although inline templates contribute negatively to the degree of template coupling, TCM values correlate positively to inline templates). Fourthly, despite the fact that the number of parameters significantly contribute to the degree of template coupling, the TCM does not consider the changes in the number of parameters as part of its definition.

This research could be considered as a first step towards developing a well-defined template coupling metric within a TTCN-3 context. It could be extended by modifying the TCM structure to improve its properties. Such modification can be done by firstly, increasing the distance between the TCM thresholds for the TCM values to be more noticeable due to any change in its variation; secondly, forcing the TCM to consider other template types and the number of parameters in its formula. Finally, by adjusting the TCM to reflect each template type separately, since each type contributes differently to the degree of coupling. We also see significant potential for applying refactoring and eradication of ‘bad smell’ principles to TTCN-3 artefacts through application of a modified version of the TCM [15].

References:


