Functional Measurement of a Real-time Software System

Jean-Marc Desharnais[1,2], Alain Abran[2],
Pınar Efe Dikici[3], Mert Can İliş[3], İrfan Nuri Karaca[3]

[1]Bogaziçi University, Turkey [2]École de technologie supérieure, Canada
[3]Middle East Technical University, Turkey

Abstract. The main objective of this paper is to explore, through a case study, the issue of the measurement adequacy of the COSMIC and IFPUG FPA measurement methods to capture the functional size of real-time software. The key issue for practitioners is that the measurement result adequately represent functional size. More specifically, this measure, which is a number, should take into consideration the particularities of specific real-time software and be sensitive to small variations in functionality. These two functional size measurement methods were applied separately to measure the same real-time software, and their results compared and analyzed.

Keywords: COSMIC, FPA, functional measure, function points, real-time system, ISO 19761, ISO 20926

1. Introduction

Functional size is the functionality-based measure of software size. ISO defines it as the size of software derived by quantifying the Functional User Requirements[6]. This functional size must be independent of the development methodology, programming language, and capabilities of the project team developing the application, so that it entirely depends on the functionality delivered to the user of the software. It provides an objective measure that assists in the evaluation, planning, management, and control of software development. Functional Size Measurement (FSM) methods have been used and studied since the publication of the first one in 1979 by Alan Albrecht [1]. Numerous FSM methods have since been proposed.

In 1996, the International Organization for Standardization (ISO) established a working group to develop an international series of documents on characteristics and criteria for FSM methods: ISO 14143 [6]. The ISO has since published five standards for FSM: IFPUG FPA: ISO 20926, NESMA FPA: ISO 24570, Mark II FPA: ISO 20968, FISMA: ISO 29811, and the COSMIC Functional Size Measurement Method: ISO 19761.

The IFPUG FPA method claims in its ISO version that it is applicable to all types of software, and to real-time and embedded software in particular. While both methods yield a measure, the key issue for practitioners is that such a number adequately represent functional size. More specifically, it should take into consideration the particularities of specific real-time software and be sensitive to small variations in functionality.

This issue is investigated in this paper through a case study using the two measurement methods, looking in detail into the measurement processes and the results obtained to compare the findings.

This paper is organized as follows. The scope of the real-time software application measured is described in the Case Study section (section 2), including the software characteristics, the measurement process, and some of the measurement problems encountered. The summary results by functional process are presented in section 3. The analysis of the different measurement results across the two measurement methods is discussed in section 4. Finally, the Conclusion summarizes the findings in section 5, which includes suggestions for future work.

2. Case Study

2.1 Dealing with ambiguities in the documentation

Identifying data groups and their persistence is challenging: data persistence concepts in real time are not quite the same as their corresponding persistence concepts in MIS. There is a possibility that some implicit Read operations could exist in the software, even though they are not mentioned in the documents. Since the project documentation available on the Web was the only source of information, it was not possible to obtain clarification of ambiguities. Consequently, assumptions had to be made with respect to unclear points, and, of course, the same assumptions were made in the measurement using both FSM methods.

2.2 Application of the measurement method

Four of the co-authors of this paper performed the size measurement, one of whom is double-certified by IFPUG. Only unadjusted IFPUG FP were considered for the study, since they reflect the specific functions provided to users by project or application; moreover, it is only this portion of FP that is recognized by the ISO.

The same four people performed the measurement with the COSMIC method: they are all COSMIC-certified at the Entry Level, and one of them is a co-author of the COSMIC method.

2.3 *Case study execution (high level)*

The concepts of Purpose, Scope, and Boundary are almost identical in the IFPUG FPA and COSMIC methods. COSMIC has a method-specific concept of Layer, and IFPUG FPA has specific concepts of Function Point Count Type.

Purpose

The purpose of the measurement is to determine the size of the software's Automatic Production Environment as determined by the IFPUG FPA and COSMIC methods. The objective in this case study is to measure the size of the Functional User Requirements as specified in the Software Requirements Specification document, pages 10-17 (http://www.rt.db.erau.edu/ BLUE/index.htm).

Scope

Scope is the real-time software functionality allocated to software in the Software Requirements. The user interface part is not included in the scope of this study, since the aim here is to compare the measurement results of real-time systems using the two sizing methods (IFPUG FPA and COSMIC).

Layer

There is only a single layer in the Automatic Production Environment (APE) software. This is not a source of difference in the measurement results of this case study, since IFPUG FPA handles a single layer.

Boundary

Figure 1 illustrates the software boundary of this case study, in particular the information in input and in output, the information in persistent storage, and the triggering event (i.e. the timer).

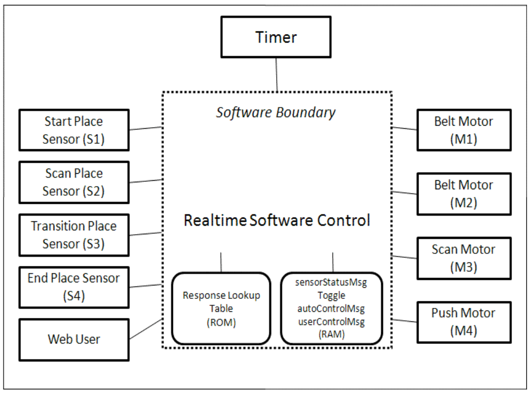

Figure 1 Software Boundary

Type of Function Point Count (used only in IFPUG FPA)

In this case study, the type of IFPUG FPA count is a Development Project.

3. Measurement Results at the Function Type Level

This section presents the measurement results at the function type level for both FSM methods.

- IFPUG FPA: 5 types of elementary process, comprising two at the Data level and three at the Transactions level.
- COSMIC: the functional processes.

Data functions

IFPUG FPA requires the measurement of data functions before the transactions are measured at the elementary process level: in this case study, five Internal Logical Files (ILF) and one EIF were identified – see Table 1.

In IFPUG FPA, the weights assigned to the files are high, in comparison to the transactional functions: each file has a size of at least 5 points for an EIF and 7 points for an ILF, even when there is only a single attribute in the EIF or ILF.

In addition, the transactional functions of IFPUG FPA also take into account both the ILF and EIF as a part of its measurement procedure, which leads to some duplication.

Internal Logical Files – ILF	**IFPUG FPA Function Points**
sensorStatusMsg	7
Toggle	7
autoControlMsg	7
userControlMsg	7
sysStatusMsg	7
External Interface Files – EIF	
Response Lookup Table	5
TOTAL	**40 FP**

Table 1 IFPUG FPA data function sizes for the case study

COSMIC recognizes these persistent data groups, but does not consider them separately in its measurement process (Table 2).

Functional processes (COSMIC) and transactional functions (IFPUG FPA)

The measurement results at the functional process level are presented in Table 2: the list of functional processes is the same for both FSM methods, but the measurement results differ in each method.

ID	Functional Process	COSMIC Size (CFP)	IFPUG FPA Size (FP)
A1	Poll All Sensors process	16	3
A2	Auto Control process	7	4
A0	Motor Control process	8	NA[1]
A3	User Control process	6	4
A4	Manage Status process	10	4
	TOTAL	**47**	**15**

Table 2 Functional processes (transactional function) sizes

The detailed measurement results are presented in Appendices A and B. The measurers identified five functional processes for COSMIC and four for IFPUG FPA. The measurer using IFPUG FPA initially identified five transactional functions, but had to drop one that did not have a data group (ID process = A0).

Figure 2 graphically illustrates the differences in the measurement results for COSMIC and IFPUG FPA for the functional process (transactional functions).

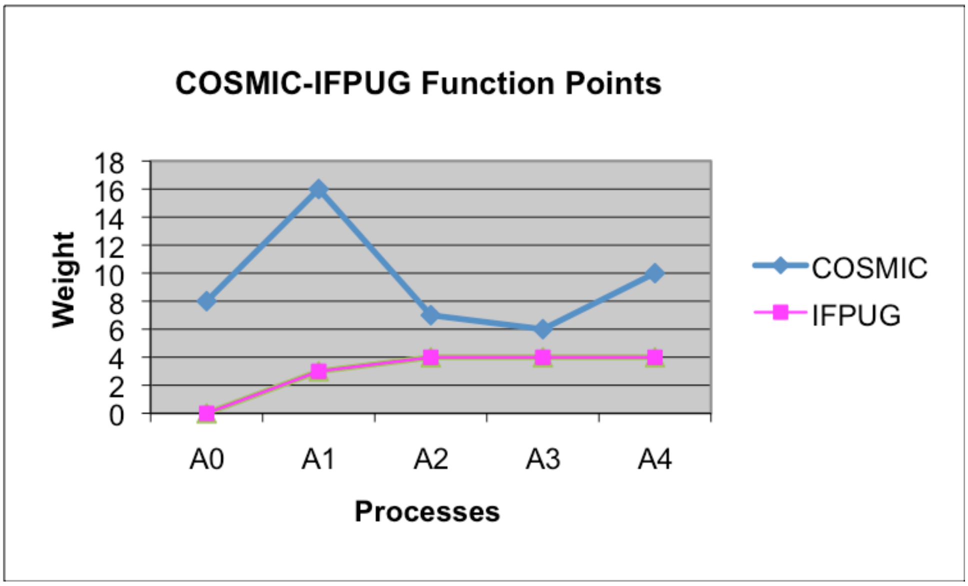

Figure 2 COSMIC and IFPUG FPA functional process sizes

It can be observed from Figure 2 that, for this case study, there is a larger variation in size with COSMIC for each functional process (from 6 to 16 CFP), while there is a variation of only 1 FP (from 4 to 5 FP[2]) with the IFPUG FPA method: this indicates that COSMIC provides a measure that is considerably more sensitive to differences in

[1] There is no data group in this process.
[2] Excluding A0, which is not measured with the IFPUG FPA method.

functionality than the IFPUG FPA method (COSMIC does not have an arbitrary upper limit on the size of a functional process). Table 2 shows that the size of this set of functional processes is more than three times greater with COSMIC than the transactional function size with IFPUG FPA.

Table 3 presents the total functional size for both FSM methods, including the size of both the data and the transactions for the IFPUG FPA method.

	COSMIC Function Points	**IFPUG FPA Function Points**
Data Functions	Not Applicable[3]	40
Transactional Functions (process)	47	15
TOTAL	47	55

Table 3 IFPUG FPA and COSMIC summary results

It can be observed from Table 3 that the total size is greater for IFPUG FPA, because of the inclusion of the IFPUG FPA data sizes.

4. Sensitivity analysis

4.1 Analysis

The total size of all transactional functions with IFPUG FPA (equivalent to a functional process for COSMIC) is 15 points (Table 2), but only because the IFPUG FPA tables assigns a maximum of 4 points when there is one DET.

COSMIC size is much greater for each functional process, from 6 to 16 (Table 2), since COSMIC does not have an arbitrary upper limit on the size of a functional process.

It is important to remember that one functional transaction was excluded (A0: Motor control) with IFPUG FPA, because there was no persistent data group for the A0 function (see detailed FPA measurement rules). By contrast, COSMIC rules consider all data movements, without taking into account persistence. A functional process without a Read or a Write (from/to a persistent data group) can still be measured with COSMIC, while it cannot with IFPUG FPA.

If a persistent data group were to be added to this functional process later, this would add only a single size unit in COSMIC, while 4 FP would be added all at once with IFPUG FPA, which is more sensitive to this type of error: the IFPUG FPA approach is a step-wise framework of intervals and weights, which leads to size steps for the transactional functions of 3, 5, and 6 points.

[3] COSMIC does not count data groups directly.

In addition, across real-time functional processes with potentially significant variations in data movements, the IFPUG FPA measurement results are within 1 to 2 points of each other, while in real-time software the variation in the number of data movements can be much larger, which in turn permits larger increases in the size of a functional process.

It can also be observed that for this dataset the IFPUG FPA data size is over-represented: a data group with a single attribute per data group is automatically assigned 7 points (7 is the minimum in the IFPUG FPA ILF table of weights).

The impact of this over-representation is difficult to determine, because of the distinct mix of transactions and data in the software measured. This issue was identified and discussed a few years ago for MIS-type software by Desharnais and Morris [8].

In summary, while the difference between the two sizes at the total level is less than 20% (47 CFU by COSMIC and 55 FP by IFPUG FPA), the difference at the detailed level is much greater. While COSMIC size is the result of the direct sizing of the functional process size (equivalent of transactional size for IFPUG FPA), the IFPUG FPA size is the result of the measurement of both the data groups and the transactions. Therefore, at the level of the transactional size, the difference is more than 70%.

4.2 Discussion

The COSMIC method allows for a finer granularity in the measurement of functional size, and is not burdened by the large step functions represented by the IFPUG FPA weight tables: the IFPUG FPA method is on the one hand much less able to discriminate between the sizes of very small functional processes, and on the other hand much less able to assign large sizes to large and very large functional processes. Furthermore, the IFPUG FPA method significantly over-represents the size of the data functionality when the data groups are very small (having 1 attribute, for example).

In "extreme" case studies (such as software with a large number of very small functional processes, or with a large number of very large functional processes), COSMIC and IFPUG FPA would produce functional sizes diverging considerably more than in the case study presented here: the COSMIC measurement results would be more sensitive to both small and large differences in the measured size than IFPUG FPA. The IFPUG FPA results are often within 1 to 2 points of either the lower or upper limits of the corresponding weight table[4]. In real-time software, the variation in the size of a functional process has no lower or upper limit. The COSMIC method would therefore provide a more useful quantitative functional size with much more quantitative discriminative measurement power.

[4] The IFPUG tables show a minimum of 4 points for an Input and an Enquiry transaction, and a maximum of 7 points. For the Output transactions, the minimum is 4 and the maximum is 7.

When comparisons are made across methods to evaluate the adequacy of the measurement method for sizing real-time software, the COSMIC method should be used as the reference point, since it is more sensitive at the detailed level. The same comment holds true for MIS software.

5. Conclusion

The COSMIC method captures functionality directly at the process level, and does so by producing quantitative results that capture the size of both very small functional processes and very large functional processes with greater sensitivity, while being able to represent functional size differences across the whole spectrum with better sensitivity.

COSMIC can measure smaller software without the distortion of the large minimum size of data groups in the IFPUG FPA method. This is particularly true with real-time systems, where a data group can often contain a single attribute.

The measurement results of the case study presented here have illustrated the distinct sensitivities of the COSMIC and IFPUG FPA methods to both small and large functionality variations in real-time software processes. They explain why there is no direct and simple convertibility ratio across methods: convertibility depends on both the particular functional profile of the software being measured and the distinct sensitivity of each method to variations in the sample being measured.

What could happen with a large-scale project with a large number of transactions reusing existing data groups? In that specific context, the number of transactions would be much larger than the number of data groups: the number of transactions with IFPUG FPA would then have proportionally more points than the data groups [8], while with COSMIC the total size of the functional processes would increase in proportion to the number of functional processes: the COSMIC size would probably be larger than the IFPUG FPA size. Further work on measuring large-scale software could help investigate this issue in greater detail.

6. References

[1] Albrecht, A. J., Measuring Application Development Productivity, IBM Applications Development Symposium, Monterey, CA, 1979, 10 p.

[2] COSMIC group, The COSMIC Implementation Guide for ISO/IEC 19761: 2003, www.gelog.etsmtl.ca/cosmic-ffp

[3] COSMIC Group Case Study, Rice Cooker, www.gelog.etsmtl.ca/cosmic-ffp

[4] COSMIC Group Case Study, Automatic Line Switching, www.gelog.etsmtl.ca/cosmic-ffp

[5] IFPUG FP CPM, (2000), International Function Point Users Group (IFPUG) Function Point Counting Practices Manual, Release 4.1.1

[6] ISO/IEC 14143-1, Information technology, Software measurement, Functional size measurement, Part 1: Definition of concepts, International Organization for Standardization, Geneva, Switzerland, 2007..

[7] Symons, C., Sizing and Estimating for Real-time Software – the COSMIC-FFP method (2006), DOD Software Tech News, Editor: Data & Analysis Center for Software, USA DOD, Rome NY, Vol. 9 No. 3, pp. 5-11,

[8] Desharnais J. M., Morris P., Comparison between FPA and FFP: a field experience, 8th International Workshop on Software Measurement – IWSM 1998, Magdeburg, Germany, 1998, p. 22.

[9] Abu Talib, M., Khelifi, A., Abran, A., Ormandjieva, O., (2007) A case study using the COSMIC-FFP Measurement Method for Assessing Real-Time System Specifications, International Workshop in Software Measurement and International Conference on Software Process and Product Measurement – IWSM/MENSURA 2007, Palma de Mallorca, Spain.