Two Surveys of Software Development Project Managers' Use of and Attitudes towards Modeling Techniques

Andrew R. Gray and Stephen G. MacDonell

Department of Information Science
University of Otago, PO Box 56, Dunedin, New Zealand
+64 3 4795282 (ph.) +64 3 4798311 (fax), agray@infoscience.otago.ac.nz

Abstract

This paper describes the results from two surveys of project managers in New Zealand that asked them various questions about their use of and attitudes towards modeling techniques for supporting the management of software development projects, especially fuzzy logic. Each survey is summarized separately and then some overall conclusions are drawn. The results give some indication of how new modeling techniques, and especially fuzzy logic, can be best presented to managers. The positive attitude of many managers towards the use of fuzzy logic models is especially promising and we are currently working with some of the organizations to see if fuzzy logic can be used within their current software development management practices.

1. INTRODUCTION

This document reports the results of two surveys conducted by the Software Metrics Research Laboratory at the University of Otago during 1998 and 1999. The focus of both surveys was on contemporary practices in software development effort estimation, addressing in particular the model-building methods used. The first survey was conducted during the months May–July, 1998 and the second during November 1998–February 1999.

The many questions on general software development practices contained in the first survey, and a considerable number from the second, are omitted, with only those questions related to demographics (in order to provide some context for the results) and modeling covered in this paper.

2. FIRST SURVEY

The first survey was sent out as part of a larger data collection exercise to a very large sample - a list of nearly 1000 companies in New Zealand. In hindsight this was an ineffective approach. Many of the organizations in the list undertook no software development in-house, rendering the survey useless for this large segment of the sample. Moreover, it is likely that the larger collection exercise may have put some potential respondents off replying at all. Ultimately, we received just 38 useful replies (almost certain to be a self-selected sample of those interested in effort estimation) to our one-page survey on estimation practices. Their responses are summarized in the following section.

2.1 Frequencies

Most of the respondents came from organizations with between 1 and 10 full-time equivalent development personnel (see Table 1), an expected outcome given the proliferation of small organizations in New Zealand.

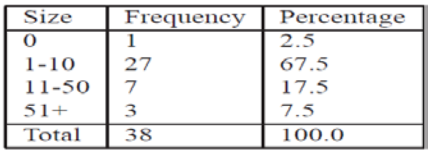
Software metrics is the field concerned with measuring aspects of software development processes, products, and resources; it is also concerned with the utilization of these measurements in developing models to support software development processes [1].

Of the 38 survey respondents over half stated that they made no use of software metrics. Of the 17 respondents using software metrics, 15 said that they used metric models for estimating development effort.

A much more interesting question is how these models are created. Table 2 shows that the vast majority use expert judgment (76.5%) [2]. This is partially due to the difficulties faced by empirical model developers (low quantities of data if any, frequently contaminated data, very high dimensionality, complex interactions between variables, and a necessity for explanation) and is a reason for the interest in knowledge based methods including case-based reasoning [4] and fuzzy logic.

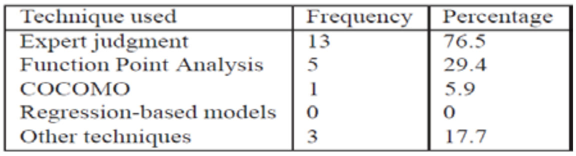
3. SECOND SURVEY

The second survey was sent out to specifically address issues related to software development effort estimation, with a more focused sample made up of organizations more likely to have in-house development groups, or software houses. The total sample was made up of 333 organizations. Forty-four replies were received - their responses are summarized in the following section.



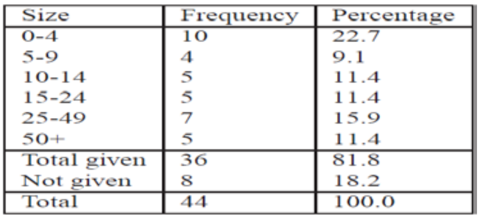
Size	Frequency	Percentage
0	1	2.5
1-10	27	67.5
11-50	7	17.5
51+	3	7.5
Total	38	100.0

Table 1: First survey: breakdown of organization sizes in terms of full-time equivalent development personnel



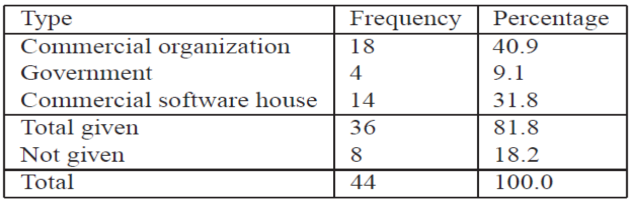
Technique used	Frequency	Percentage
Expert judgment	13	76.5
Function Point Analysis	5	29.4
COCOMO	1	5.9
Regression-based models	0	0
Other techniques	3	17.7

Table 2: First survey: breakdown of techniques used by the organizations for software metrics



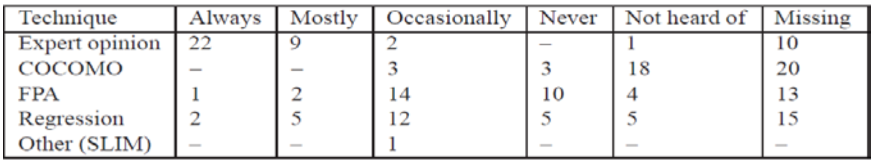
Size	Frequency	Percentage
0-4	10	22.7
5-9	4	9.1
10-14	5	11.4
15-24	5	11.4
25-49	7	15.9
50+	5	11.4
Total given	36	81.8
Not given	8	18.2
Total	44	100.0

Table 3: Second survey: breakdown of organizations by size in terms of full-time development personnel



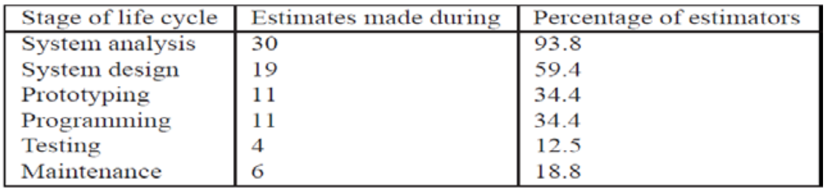
Type	Frequency	Percentage
Commercial organization	18	40.9
Government	4	9.1
Commercial software house	14	31.8
Total given	36	81.8
Not given	8	18.2
Total	44	100.0

Table 4: Second survey: breakdown of respondents by type of organization



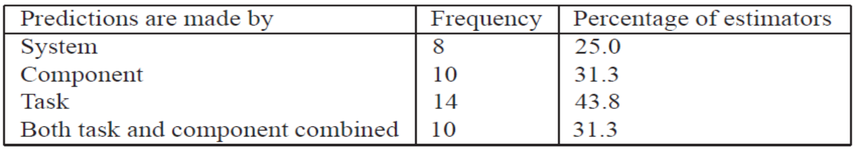
Technique	Always	Mostly	Occasionally	Never	Not heard of	Missing
Expert opinion	22	9	2	–	1	10
COCOMO	–	–	3	3	18	20
FPA	1	2	14	10	4	13
Regression	2	5	12	5	5	15
Other (SLIM)	–	–	1	–	–	–

Table 5: Second survey: Frequency of metric modeling technique usage by organizations



Stage of life cycle	Estimates made during	Percentage of estimators
System analysis	30	93.8
System design	19	59.4
Prototyping	11	34.4
Programming	11	34.4
Testing	4	12.5
Maintenance	6	18.8

Table 6: Second survey: Stages of life cycle where effort estimates are made



Predictions are made by	Frequency	Percentage of estimators
System	8	25.0
Component	10	31.3
Task	14	43.8
Both task and component combined	10	31.3

Table 7: Second survey: Levels of prediction for organizations using effort estimation

3.1 Frequencies

The sizes of organizations responding to the second survey are shown in Table 3. While the organizations are still generally quite small, they appear to be larger than the sample from the first survey. Table 4 shows the nature of the organizations.

Of the 36 organizations who replied to the question about attempting to predict development effort, 32 said that they did and only four that they did not. Of those who said ‘no’, one gave a reason that they did “not know how to go about making such predictions”, two said that there was “no need for such predictions”, and one gave a comment that their development environment was too unstable.

In terms of the modeling methods actually employed by those respondents who performed estimation, expert opinion again totally dominated those others considered (see Table 5). Given that this sample included a significant number of software houses we were genuinely surprised at the lack of infiltration of the FPA and COCOMO techniques [1].

The next question was concerned with the stage at which estimates are actually made—clearly the need for predictive capability at the analysis phase is most evident (see Table 6). Estimation at the development task level and in the main for individual components are also undertaken by most of the respondents (see Table

7). The importance of estimation during the very early stages of the system life cycle (analysis and design) are well indicated which further supports the use of qualitative estimation techniques since it is impossible to accurately estimate many tradition software metrics used in effort estimation models at these stages.

Problems with expert opinion as noted by respondents are shown in Table 8. Additions to this question included a lack of experts, leading to an assertion that the technique should not be used as the sole or final approach. Another respondent noted the problems caused by inconsistencies between staff. One respondent suggested that the technique was reasonably accurate for small to medium sized projects which, for them, was up to three months duration.

The same ratings of problems are shown for COCOMO (Table 9), FPA (Table 10), Regression (Table 11), SLIM (Table 12), and three other less used methods (Tables 13, 14, and 15).

An additional problem with FPA mentioned by one respondent is that productivity rates fluctuate too much from project to project due to continuous change. Another mentioned that there was resistance from personnel given its “seemingly weird basis”. One of the respondents who checked the box that FPA is too complex felt sufficiently strongly about this to use five ticks! This supports the need for simple to understand modeling techniques in this domain.

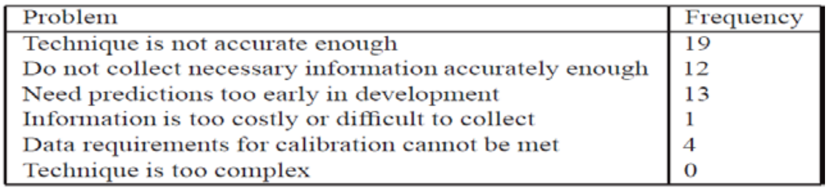
3.2 Analysis of knowledge of fuzzy logic

It was found that a surprisingly high 31 out of the 44 information system managers who responded had heard of fuzzy logic (Table 16). Of the 36 managers who were actively involved in managing development projects, 11 were interested in using fuzzy logic techniques, 23 stated that they would need to know more about the technique before making a decision, and only two did not think that fuzzy logic techniques would be useful to them. See the table below for more details (Table 17). One organization was currently using fuzzy logic and another had in the past.

Three advantages of fuzzy logic were proposed to readers of the survey and they were asked to indicate their interest or disinterest in each feature (Table 18). These were being able to use expert knowledge for model development, using linguistic labels before numerical values are known, and having less precise estimates from the model. Interestingly, those expressing some interest in the use of fuzzy logic found each of the three advantages equally appealing. The percentages do not sum to one hundred since most respondents selected more than one advantage.

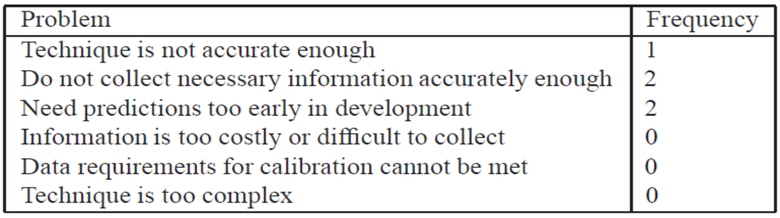
No relationships were found between the organization’s software development department size in terms of equivalent full-time personnel (six levels for full-time equivalent employees) and type (commercial, government, and software-house), and their knowledge or interest in fuzzy logic. These associations were all initially tested using X^2 tests at the 0.05 level.

While the survey results reported above certainly reflect a self-selected sample since many surveys were not returned, they are encouraging in that they suggest that a significant number of project managers are prepared to use such a technique. Some of these organizations are now being approached to evaluate the FULSOME system, as described in [3], in a more practical setting. Such feedback will be essential to the development of a truly usable system for practitioners. They will also be involved in developing a set of standard practices for the use of fuzzy logic for software metric model development.



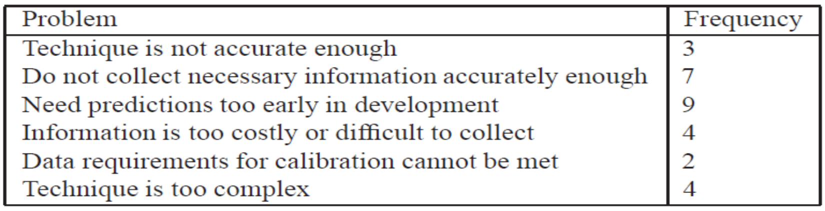
Problem	Frequency
Technique is not accurate enough	19
Do not collect necessary information accurately enough	12
Need predictions too early in development	13
Information is too costly or difficult to collect	1
Data requirements for calibration cannot be met	4
Technique is too complex	0

Table 8: Second survey: Problems with expert opinion as an estimation method



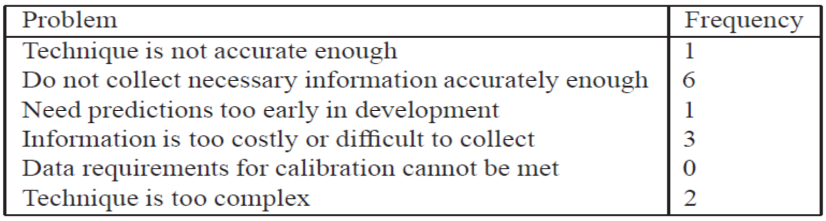
Problem	Frequency
Technique is not accurate enough	1
Do not collect necessary information accurately enough	2
Need predictions too early in development	2
Information is too costly or difficult to collect	0
Data requirements for calibration cannot be met	0
Technique is too complex	0

Table 9: Second survey: Problems with COCOMO opinion as an estimation method



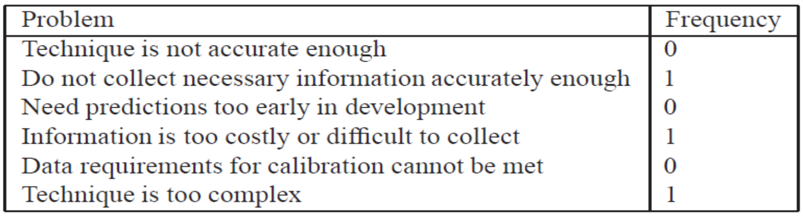
Problem	Frequency
Technique is not accurate enough	3
Do not collect necessary information accurately enough	7
Need predictions too early in development	9
Information is too costly or difficult to collect	4
Data requirements for calibration cannot be met	2
Technique is too complex	4

Table 10: Second survey: Problems with FPA as an estimation method



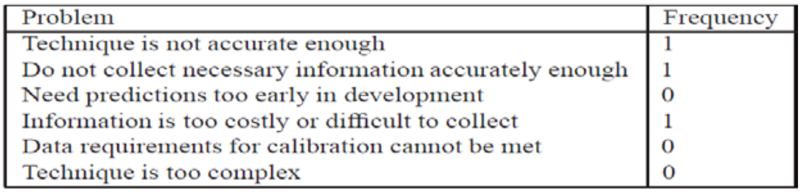
Problem	Frequency
Technique is not accurate enough	1
Do not collect necessary information accurately enough	6
Need predictions too early in development	1
Information is too costly or difficult to collect	3
Data requirements for calibration cannot be met	0
Technique is too complex	2

Table 11: Second survey: Problems with regression models as an estimation method



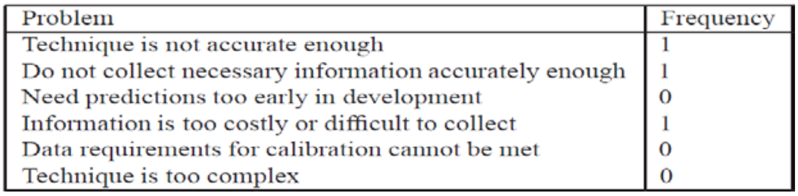
Problem	Frequency
Technique is not accurate enough	0
Do not collect necessary information accurately enough	1
Need predictions too early in development	0
Information is too costly or difficult to collect	1
Data requirements for calibration cannot be met	0
Technique is too complex	1

Table 12: Second survey: Problems with other (SLIM) as an estimation method



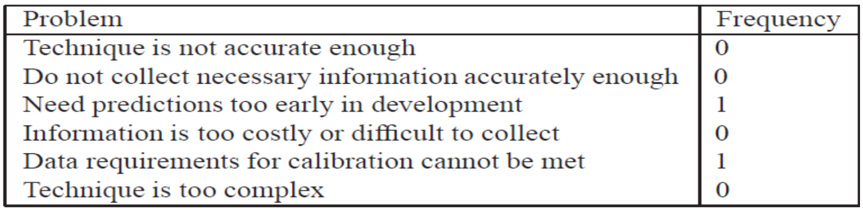
Problem	Frequency
Technique is not accurate enough	1
Do not collect necessary information accurately enough	1
Need predictions too early in development	0
Information is too costly or difficult to collect	1
Data requirements for calibration cannot be met	0
Technique is too complex	0

Table 13: Second survey: Problems with other (own prior experience) as an estimation method



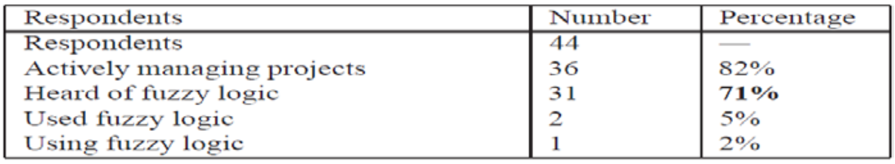
Problem	Frequency
Technique is not accurate enough	1
Do not collect necessary information accurately enough	1
Need predictions too early in development	0
Information is too costly or difficult to collect	1
Data requirements for calibration cannot be met	0
Technique is too complex	0

Table 14: Second survey: Problems with other (own data) as an estimation method



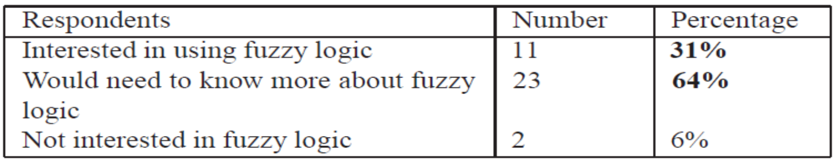
Problem	Frequency
Technique is not accurate enough	0
Do not collect necessary information accurately enough	0
Need predictions too early in development	1
Information is too costly or difficult to collect	0
Data requirements for calibration cannot be met	1
Technique is too complex	0

Table 15: Second survey: Problems with other (own measures) as an estimation method



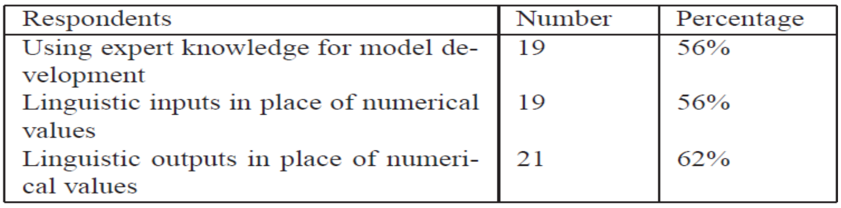
Respondents	Number	Percentage
Respondents	44	—
Actively managing projects	36	82%
Heard of fuzzy logic	31	**71%**
Used fuzzy logic	2	5%
Using fuzzy logic	1	2%

Table 16: Second survey: knowledge of fuzzy logic



Respondents	Number	Percentage
Interested in using fuzzy logic	11	**31%**
Would need to know more about fuzzy logic	23	**64%**
Not interested in fuzzy logic	2	6%

Table 17: Second survey: interest in fuzzy logic by active managers



Respondents	Number	Percentage
Using expert knowledge for model development	19	56%
Linguistic inputs in place of numerical values	19	56%
Linguistic outputs in place of numerical values	21	62%

Table 18: Second survey: advantages of fuzzy logic

4. CONCLUSIONS

The survey results suggest that the problems noted by project managers with existing modeling techniques would make fuzzy logic models a very useful addition. It is also pleasing to see such as high level of awareness of fuzzy logic and, what we felt to be, a surprisingly high level of interest.

The question therefore has to be why there is such as very low levels of usage. The disparity could be seen as due to a low level of software availability for non-specialists, lack of guidelines (note that Function Point Analysis has a very well developed education and certification framework), and the lack of successful case studies which is an all too common catch-22 situation in software engineering.

REFERENCES

[1] N. E. Fenton and S. L. Pfleeger. *Software Metrics: A Rigorous & Practical Approach*. PWS, 1997.

[2] R. Hughes. Expert judgement as an estimating method. *Information and Software Technology*, 38:67–75, 1996.

[3] S. MacDonell, A. Gray, and J. Calvert. Fulsome: A fuzzy logic modelling tool for software metricians. In *Proceedings of the 18 International Conference of the North American Fuzzy Information Processing Society - NAFIPS*, pages 263–267. IEEE, 1999.

[4] M. Shepperd and C. Schofield. Estimating software project effort using analogies. *IEEE Transactions on Software Engineering*, 23(12):736–743, 1997.