Interpretation of Qualitative Data in XP using Fuzzy Logic

Sian Hope
University of Wales, Bangor
School of Informatics
Dean Street, Bangor
Gwynedd, UK
+44 1248 382695

sian@informatics.bangor.ac.uk

Robert Gittins
University of Wales, Bangor
School of Informatics
Dean Street, Bangor
Gwynedd, UK
+44 1248 382724

rgittins@informatics.bangor.ac.uk

Abstract

Currently, the software engineering industry is reluctant to embrace new software development methodologies, because of the inherent risk of failures. Soft-computing techniques open up possibilities for increasing confidence in software development, where classical analysis alone does not satisfy the thirst for credible information. Companies looking for direction and wishing both to improve their software development performance and return-on-investment, will find soft computing useful.

This paper applies fuzzy-logic techniques to assist in the interpretation of qualitative data arising from studies with companies using extreme Programming (XP). In particular, sample results from questionnaires returned by companies embracing the change from 'traditional' software practices to XP are analysed. The merit of selected XP practices is investigated, and fuzzy techniques applied to assist in the interpretation of qualitative data. The results are examined and demonstrate that these techniques of interpretation, appreciably contribute to the value of the findings.

Key Words
Software development, fuzzy logic, qualitative methods, extreme programming.

1.INTRODUCTION

The researcher is primarily interested in improving the analysis process of 'qualitative' (as opposed to 'quantitative') data. It is essential to reflect on the results from previous experiments to accurately interpret 'cause and effect' in empirical studies. Qualitative data is presently evaluated using crisp techniques (classical sets that wholly include or wholly excludes elements), inadequate in providing confident conclusions from sparse and often small empirical software development studies. It is hoped that from this on-going investigation, fuzzy logic will eventually provide significantly richer results to the qualitative researcher and thus to the body of knowledge on case studies using such agile techniques.

It is important perhaps to gain some insight into the nature of qualitative research before describing the analysis process adapting fuzzy logic techniques. Qualitative evaluation allows the researcher to study selective issues in detail, without the pre-determined constraints of >categorised= analysis. The researcher is instrumental in the gathering of data from open-ended interview questions and questionnaires. Direct quotations and opinions gleaned from the questionnaires are the basic source of raw materials, revealing the respondent's depth of concern. This contrasts with the crisp statistical results of quantitative methods, recognised by their encumbrance of predetermined procedures. Individuals do not live in a vacuum but within the context of their accumulated knowledge, experiences and surroundings. Engineering research currently relies upon 'quantitative' information and crisp analysis that is inadequate for capturing the context, experiences and expectations of users or designers in the context of an engineering development process. Qualitative research methods have been developed in order to explore, characterise and assess phenomena involving a complex human dimension.

Software engineers are often constrained to attribute strong significance to a single statistical imprecise finding, simply because empirical findings are so scarce and hence design decisions are taken based upon inadequate information. The rational for qualitative studies is to build up a convincing *weight of evidence* to discover and support propositions (hypotheses) by a process of data collection and analysis. Supported or refuted evidence for such propositions can be gathered quantitatively or qualitatively, but is best achieved by a combination of these methods. The reason for employing qualitative methods is that they have the added advantage of

providing more explanatory information, and so help in refining a proposition to better support the accumulated data.
Fuzzy logic is recognised by its imprecise, non-crisp nature and reflects in many ways the characteristics of qualitative data.

2. RELATED WORK

Ali Idri et al [4] have conducted research into software metrics, considering factors such as the experience of programmers, particularly in software cost estimation and the COCOMO model. They investigate the issue of compatibility of COCOMO with fuzzy logic.

From a series dedicated to 'Studies in Soft Computing' Ludmila Kuncheva [5] introduces the fundamentals of fuzzy set recognition and fuzzy set theory. Ludmila defines fuzzy 'if-then' classifiers and in a chapter on multiple classifier combination discusses fuzzy and non-fuzzy models for fusion selection.

Carolyn B. Seaman [1] describes an empirical study that addresses the issue of communication among members of a software development organization. In recent papers Carolyn has presented several qualitative methods for data collection and analyses, and described them in terms of how they might be incorporated into empirical studies of software engineering. In particular Carolyn shows how they might be combined with quantitative methods.

Helen Sharp et al. [6] emphasise 'how' the relevant aspects of *context*, within the rich cultural setting of Software development, can be identified. Their paper *'The role of 'culture' in successful software process improvement.'* Their paper covers the influences on software development and the techniques for software development practices traditionally associated with the social sciences: ethnography and discourse analyses. Their work found recurring themes, throughout the discourse of the department studied, which provided the cultural context for software quality initiatives. They found that an approach combining *ethnography* and *discourse analysis* successfully uncovered implicit influences operating within a culture and affecting the acceptance and application of the organisation's software quality management system.

Sharp, Robinson and Woodman [7] in their article *'Software Engineering: Community and Culture'* rather than integrate methods from the social sciences into the systems design process, as others have done, attempt what they describe as a 'cross-pollination' and strive to inform and improve the development of software engineering itself through a deeper understanding of our community's implicit values and beliefs.

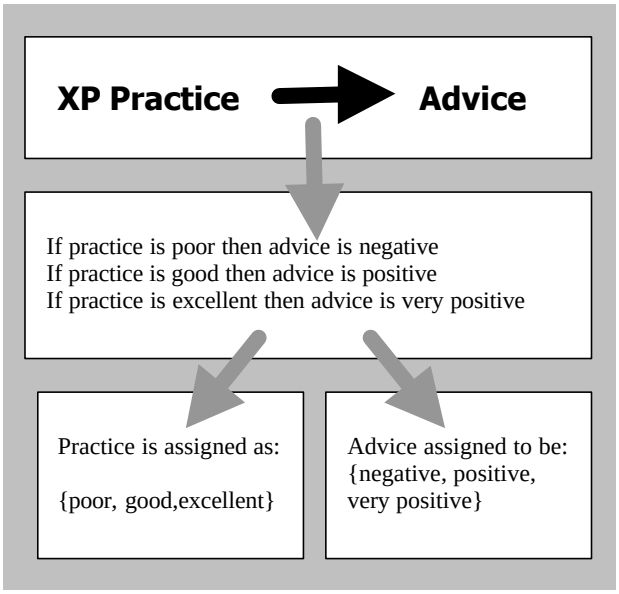
Previous work has investigated combining software metrics with fuzzy logic and qualitative researchers have begun to address the human issues influencing the software development process. This work bridges the gap between qualitative and quantitative techniques by investigating soft computing as a means of improving the evaluation of the XP process.

3. FUZZY LOGIC CONCEPTS

What are the relevant concepts of fuzzy logic that may apply to the analyses of qualitative research questionnaires?

Fuzzy Inference is a process of mapping a method that interprets the values in the input vector and based on some set of rules, assigns values to the output vector. The principle here is to map an input space to an output space via a black box that contains a fuzzy system. The primary mechanism for doing this is a series of 'if-then' statements called rules. Rules are evaluated in parallel but their order is not important. Figure 1 shows a fuzzy inference map for the evaluation of an Extreme Programming practice.

Fig 1. Fuzzy Inference Mapping



Fuzzy logic is a superset of Boolean Logic, and therefore standard logical operators apply. However, as truth in fuzzy logic is a matter of degree, the functions of a truth table need to be preserved..

There are two basic types of fuzzy inference systems: Mamdani-type [8] and Sugeno-type [9] that vary in the way their outputs are determined. Mamdani inference expects the output membership functions to be fuzzy sets. After the aggregation process there is a fuzzy set output for each output variable that needs defuzzification. However it is often more efficient to use a spike or *singleton* output membership function,

which can be described as a pre-defuzzified fuzzy set. This greatly simplifies the computation required by the more general Mamdani method.

4. METHOD

- ? Fuzzification of the inputs
- ? Application of fuzzy operator (antecedent)
- ? Implication (from the antecedent to the consequent)
- ? Aggregate all outputs (across rules)
- ? Defuzzify

Binary logic has the crisp antecedents and consequents, true or false. Fuzzy logic is concerned with partial antecedents that provide partial consequents. There can be multiple antecedents and the consequent can have multiple parts. However one rule isnt much good, two or three are needed to play off one another and produce a richer result. The output of each rule is a fuzzy set, the sets are aggregated to a single output fuzzy set. The set is then defuzzied, or resolved to a single number.

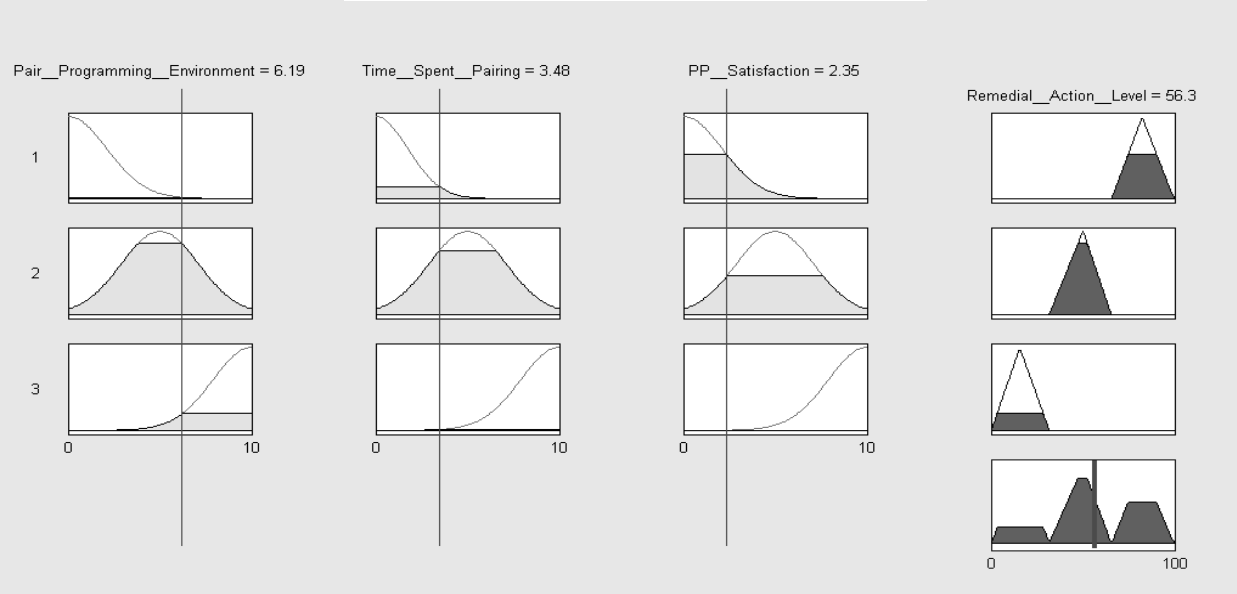
To summarise; from the input values, the output is an entire fuzzy set. This set is later defuzzified assigning one value to the output.

It was soon apparent that designing suitable rules to the results of questionnaires created before any thought to using fuzzy analyses, was problematic. Future questionnaires may need to consider both the wording of the questions and the scaling of the answers, to reap the benefits that soft computing has to offer.

Having determined the input values, rules that determine the behaviour of the system and the output value range, the input membership functions (MF) or 'universe of discourse' were specified. The choice of MF Type is very much a matter of experience, personal preference, and much conjecture. For the purposes of this exercise three Gaussian curves were added to each of the input variables. i.e. For 'Environment' – *Poor*, *Good* and *Excellent*. For the output a triangular MF type was used with a range 0 to 100 percent with the output defined as 'Remedial-Action-Level' of '*Low*', '*Standard*' and '*High*' values, with ranges set between 0 to 33, 33 to 66 and 66 to 100 respectively.

Many of the questions from the XP survey had answers in the format - *Very effective*, *Effective*, *Ineffective*, *Very Ineffective* or *Strongly Agree*, *Agree*, *Disagree*, *Strongly Disagree*, which are easily entered as input variables. However, some questions had multiple-choice answers, which were too verbose and loosely connected to be of use as MF variables.

Figure 2. Pair Programming - Rule View



5. RESULTS

The first step is to write down the 'rules', 'if-then statements', and determine the 'outputs'. In the following example, data returned from a group of software developers who practice 'Pair Programming' as part of their XP methodology [3] has been adapted to examine the fuzzy inference system and gain experience of the mechanism.

With data from the XP questionnaire, a simple example was run using Matlab's Fuzzy logic toolbox. It was hoped to gain insight into the mechanism and reveal problems and opportunities arising from the exercise.

In Figure 2, 'Pair Programming – Rule View', the diagram shows three simple rules, from left to right leading to the output column.

The aggregation occurs down the fourth column, with the result shown bottom right as a defuzzified output from the aggregate fuzzy set. A vertical line shows the input values on each column against the respective rules. The heavy vertical line in the lower right output column is the defuzzified result.

6. COMMENTS ON FINDINGS

Previous results [3], have provided information such as ".. the time to market for our software products has halved as a result of using XP". Is this enough, for example in supporting the uptake of XP in another organisation. The work presented here is a first approach to using soft computing techniques, in an attempt to add a richer dimension to traditional methods measuring the success(or otherwise) of software development approaches. Fuzzy logic is shaping future questionnaires to consider the structure and scaling of questions. In this way it is expected that it will assist in providing "agile measurement" to sit alongside our agile processes.
An appropriate output must be considered carefully – therefore customer involvement, knowledge and experience is an all-important factor. Adopting fuzzy logic may be a natural extension to enrich survey results by adding a new perspective from qualitative data returned.

7. CONCLUSIONS

Is it enough to say "50% of Pair programmers prefer to work alone 10% - 50% of the time" and "96% … enjoy there jobs more when pair programming" [2] or " 71% of developers regarded unit testing…. to be very poor" [3]. Qualitative Research presupposes examination upon *processes* and *meanings* that do not gain sufficient description for the investigator by using quantitative methods or where quantitative methods alone are inappropriate. Isolated results alone may not provide sufficiently satisfying information that perhaps a balance of related and disproportionate factors through fuzzy logic might provide. Further research will no doubt provide more information to the body of knowledge and thus the acceptance and uptake of agile processes. We are still looking for better ways of evaluating (and measuring) the results.

REFERENCES

1. Seaman, C.B. 'Qualitative methods in empirical studies of software engineering', *IEEE Transactions on Software Engineering*, Vol.25 (4):557- 572, July/August 1999.

2. Williams, L. A. and Kessler, R.R. 'All I Really Wanted to Know About Pair Programming I Learned in Kindergarten'. *Communications of the ACM*. Vol.43, No5. May 2000

3. Gittins, R.G., Hope, S., Williams, I., 'Qualitative Studies of XP in a Medium Sized Business'. *Proceedings of the 2nd International Conference on Extreme Programming and Flexible Processes in Software Engineering*. 20-23 May 2001
Villasimius, Sardinia, Italy.

4. Idri, A., Abran, A. and Kjiri, L. 'COCOMO Cost model using fuzzy logic' *7th International Conference on Fuzzy Theory Technology* Atlanta City, New Jersey, February 27 – March 3, 2000

5. Kuncheva, L.I. 'Fuzzy Classifier Design' Heidelberg; New York: Physica-Verlag Publishers 2000

6. Sharp. H.,Woodman, M., Hovenden, F., & Robinson, H. 'The Role of Culture in Successful *Software Process Improvement.*' 25th Euromicro 99 2, p17 IEEE Computer Society.

7. Sharp. H.,Woodman, M., Hovenden, F., & Robinson, H. 'Software Engineering Community and Culture' *IEEE Software* 17:(1) 40-47 Jan-Feb 2000

8. Mamdani, E.H. and Assilian, S. 'An experiment in linguistic synthesis with a fuzzy logic controller'. *International journal of man machine studies*, Vol. 7, No1, pp. 1-13, 1975

9. Sugeno, M. 'Industrial applications of fuzzy logic'. *Elsevier Science Publishers*. 1985