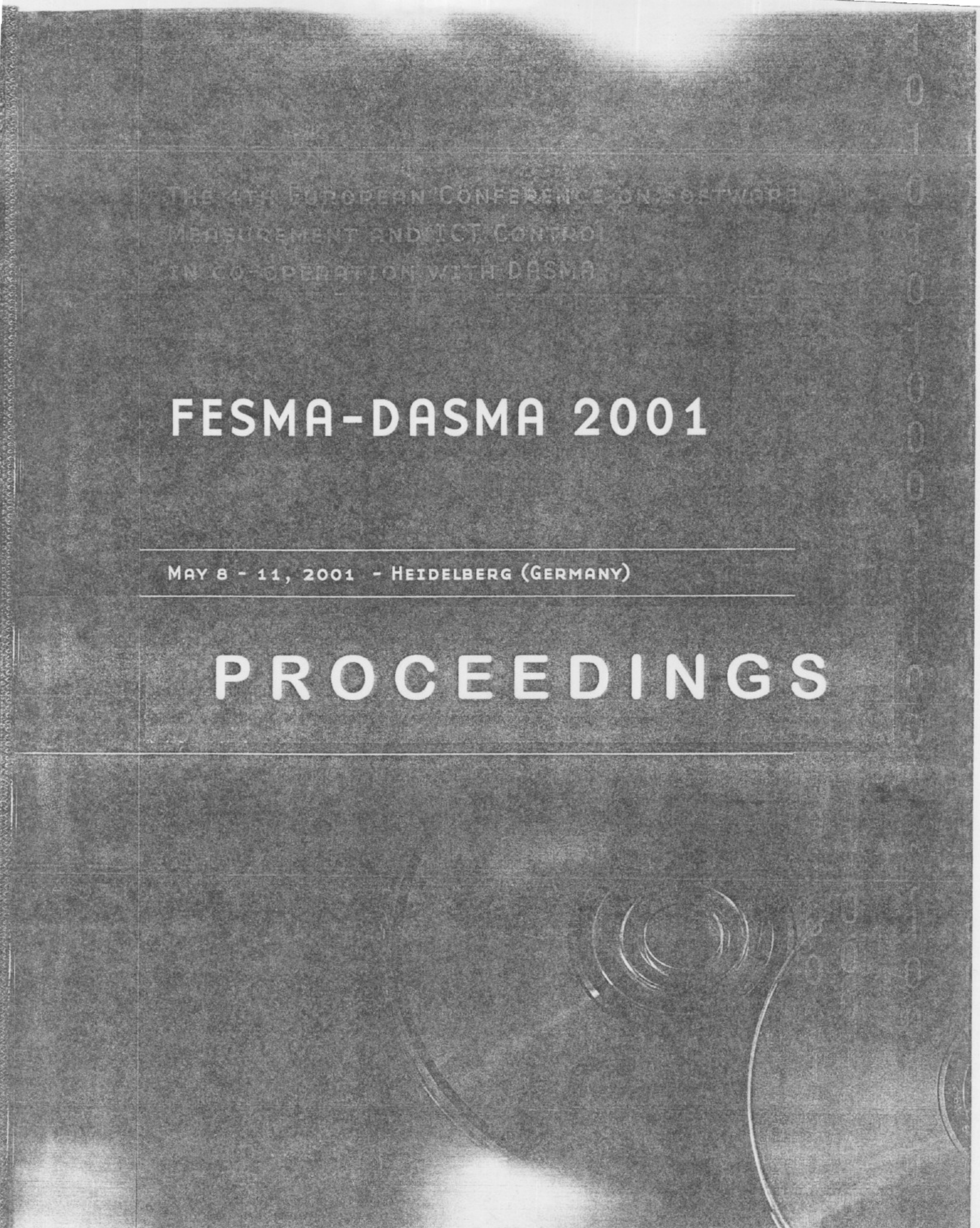

The 4th European Conference on Software
Measurement and ICT Control
in co-operation with DASMA

FESMA-DASMA 2001

May 8 - 11, 2001 - Heidelberg (Germany)

PROCEEDINGS

Technologisch Instituut

FESMA

ISBN 90-76019-15-0
D 2001/0277/1

An Extension of the original KDD Process for Predicting Entity Relationship Diagram Maintainability

Marcela Genero, José A. Olivas, Mario Piattini, Francisco P. Romero
Department of Computer Science
University of Castilla-La Mancha
Ronda de Calatrava, 5
13071, Ciudad Real (Spain)
E-mail: {mgenero, jaolivas, mpiattin, fpromero}@inf-cr.uclm.es

ABSTRACT

Within the information system (IS) field it is generally accepted that the quality of the IS is highly dependent on decisions made early in the development. The construction of conceptual data models is often an important part of this early development, so their quality could heavily influence the quality of the IS which is ultimately implemented. We are interested in measuring the quality of conceptual data models such as entity relationship diagrams (ERD), especially focussing on their maintainability (ISO, 1999). But we are aware that maintainability is an external quality attribute that can only be measured once an IS is finished, so it is highly important to have of metrics for measuring an internal quality attribute, such as structural complexity, early in the IS life cycle. And based on those metrics build a prediction model for ERD maintainability. We have proposed in Genero et al. (2000a) a set of measures for measuring the ERD structural complexity. As in other aspects of Software Engineering, proposing techniques and metrics is not enough; it is also necessary to perform an empirical validation of them to assure their utility in practice. Empirical validation is critical to the success of software measurement (Kitchenham et al., 1995; Fenton and Pflegeer, 1997; Schneidewind, 1992; Basili et al., 1999). Therefore we have carried out a controlled experiment, with the objective of predicting each of the sub-characteristics of ERD maintainability (understandability, simplicity, analysability, modifiability, stability, and testability) at the initial phases of the IS life cycle. For building the prediction model based on the ERD structural complexity metrics we have used an extension of the traditional Knowledge Discovery in Databases (KDD) (Fayyad, 1996): the Fuzzy Prototypical Knowledge Discovery (FPKD) (Olivas, 2000a; 2000b) that consists of the search for fuzzy prototypes (Zadeh, 1982) which characterise the maintainability of an ERD. These prototypes form the foundation of the prediction model that allows us to predict ERD maintainability.

Keywords: entity relationship maintainability, structural complexity, metrics, maintanability prediction, prediction models, fuzzy prototypical knowledge discovery, fuzzy deformable prototypes

1. INTRODUCTION

Within the information system (IS) field it is generally accepted that the quality of the IS is highly dependent on decisions made early in development. The construction of conceptual data models is often an important part of this early development, so their quality could heavily influence the quality of the IS which is ultimately implemented. We are interested in measuring the quality of conceptual data models such as entity relationships diagrams (ERD), specially focussing on their maintainability (ISO, 1999). We focus on maintainability, because maintainability has been and continues to be one of the most pressing challenges facing any software development organisation. But we are aware that maintainability is an external quality attribute that can only be measured once an IS is finished, so it is highly important to have of metrics for measuring an internal quality attribute, such as structural complexity, early in the IS life cycle. And based on those metrics build a prediction model for ERD maintainability. We have proposed in Genero et al. (2000a) a set of measures for measuring ERD structural complexity. We focus on ERD because in today's IS design world it is still the dominant method of conceptual modelling (Muller, 1999).

The early availability of significant measures allows for better management of the later phases, and more effective quality assessment when quality can be more easily affected by corrective actions (Briand et al., 1999c). They allow IS designers:

1. a quantitative comparison of design alternatives, an therefore and objective selection among several class diagram alternatives with equivalent semantic content.

they act together. The term process implies that there are several steps, like the preparation of data or the search for patterns.

The KDD process (interactive and iterative) described by Fayyad and his collaborators is shown in figure 1.

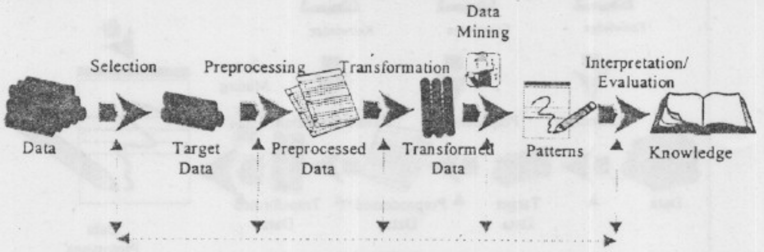

Figure 1. KDD process

The term pattern (in this work it will be denominated prototype of data) talks about a subgroup of data, along with a description and a model applicable to the same . The prototypes of data discovered must be valid for new data with some degree of certainty. , These patterns must be new, at least for the system and preferably for the user, and potentially useful. Finally, these patterns must be comprehensible, if not immediately, after postprocessing. This definition implies that they must be defined measures of the goodness of the prototypes of data; in many cases it is possible to define measures of certainty (capability of classification of new data) or utility (quality of the predictions on the basis of theses prototypes of data).

Taking the prototype theory of psychology as a reference, a single representation of ERD Maintainability could be seen as prototypical. However, in a previous approximation of the knowledge acquisition process we were able to observe that this representation excessively simplifies the behavioural guidelines of the experts. When a technician is confronted with a real situation he handles a range of prototypes determined by a series of factors and must decide which type of ERD maintainability is to be expected. Therefore, the prototype "ERD maintainability" is not unique.

Zadeh (1982) mentions the classical prototype theories from the point of view of psychology, criticizing precisely what we have just pointed out: that these theories do not fit the function that a prototype should have. Zadeh's approach to what must be taken as a prototype is less intuitive than the conceptions of psychological theories but is more rational and closer to the meaning of a prototypical concept displayed in a more detailed examination. In our case, we have observed that Zadeh's idea suggests a concept that encompasses a set of prototypes, which represent the high, medium, or low compatibility of the samples with the concept A. "The prototype is not a single object or even a group of objects in A. Rather, it is a fuzzy schema for generating a set of objects which is roughly coextensive with A" (Zadeh, 1982).

Based on these suggestions, modifications of the original KDD process are proposed, as represents fig 2. Which they involve incorporation of a new knowledge in different points and decisions of the users or experts. The aim must be to generate conceptual prototypes (Zadeh's approach: fuzzy schemas) that allow us to evaluate new situations from these patterns, and to establish predictions if these prototypes represent ordered series. The stages of the modified KDD (from now on Fuzzy Prototypical Knowledge Discovery: FPKD) are the following (see figure 2):

- SELECTION: Applying the knowledge of the dominion and excellent knowledge *a priori*, considering the objectives of the global process of FPKD, target data is created that will include selected sets of data or subgroups of excellent variables or examples.
- PRE-PROCESSING: Data cleaning, noise elimination, handling of empty fields, lost data, unknown values or by defect. Standard techniques of data bases are applied.
- TRANSFORMATION: Reduction of the number of variables. Location of useful forms to express the data depending on the later use and on the objectives of the system. The expert knowledge and techniques of transformation and information in data bases are used.
- DATA MINING: Selection of the algorithms of Data Mining. Decisions about the model that is derived from the algorithm of Data Mining (classification, summary of data, prediction). Search for interest patterns, as far as concerns classification, decision trees, regression, dependency, heuristics, uncertainty, etc.

- NAggR. Is the total number of aggregations in an ERD. In this case, we consider one relationship for each pair whole-part within the aggregation relationship.
- RR METRIC. Is the total number of relationships that are redundant in an ERD.
- SD METRIC. We define SD metric thus:

$$SD = \sum_{i=1}^{N^e} a_i^2$$

a_i is the number of entities which can be directly reached from the entity "i" through relationships.
N^e is the number of entities in an ERD.

To calculate this metric we only consider common relationships (neither generalisation nor aggregation). In the case of reflexive relationships, between an entity and itself, we consider the value 1.
These metrics are open-ended metrics (Lethbridge, 1998), i.e. they are not bounded in an interval. Close-ended metrics (percentage metrics), such as the following: RR/NR; DA/NA; M.NR/NR; 1:NR/NR, BinaryR/NR, N-AryR/NR, etc...could also be useful
We have theoretically validated these metrics (Genero et al., 2000a) following the formal measurement framework proposed by Zuse (1998), with the objective of ascertaining the scale type of each metric, concluding that all of these open-ended metrics are in ratio scale and the close-ended metrics are in absolute scale.

4. A COMPREHENSIVE CONTROLLED EXPERIMENT FOR PREDICTING ERD MAINTAINABILITY

Taking into account some suggestions provided in Briand et al. (1999a; 1999b) and Perry et al. (2000=)about how to do empirical studies in Software Engineering, we carried out a controlled experiment with the aim of predicting ERD maintainability based on ERD structural complexity metrics (see section 3), obtained in the early phases of IS life cycle.

4.1 Subjects

The experimental subjects used in this study were: 9 professors and 7 students enrolled in the final-year of Computer Science in the Department of Computer Science at the University of Castilla-La Mancha in Spain. All of the professors belong to the Software Engineering area and they have on average five years of experience in the design of ERD. By the time the experiment was done all of the students had had two courses on Software Engineering and one course on Databases, in which they learnt in depth how to build ERD. Moreover, subjects were given an intensive training session before the experiment took place.

4.2 Experimental materials and tasks

The subjects were given twenty - seven ERD (Ruíz Gómez-Nieto, 1997, De Miguel and Piattini, 1999) of the different universes of discourse, general enough to be easily understood by each of the subjects. Each diagram has a test enclosed which includes the description of maintainability sub-characteristics, such as: understandability, simplicity, analysability, modifiability, stability and testability. Each subject has to rate each sub-characteristic using a scale consisting of seven linguistic labels. For example for understandability we proposed the following linguistic labels:

Extremely difficult to understand	Very difficult to understand	A bit difficult to understand	Neither difficult nor easy to understand	Quite easy to understand	Very easy to understand	Extremely easy to understand

We allowed one week to do the experiment, i.e., each subject had to carry out the test alone, and could use unlimited time to solve it. After completion of the tasks subjects were asked to complete a debriefing questionnaire. This questionnaire included (i) personal details and experience, (ii) opinions on the influence of different components of ERD, such as: entities, relationships, attributes, etc... on their maintainability.

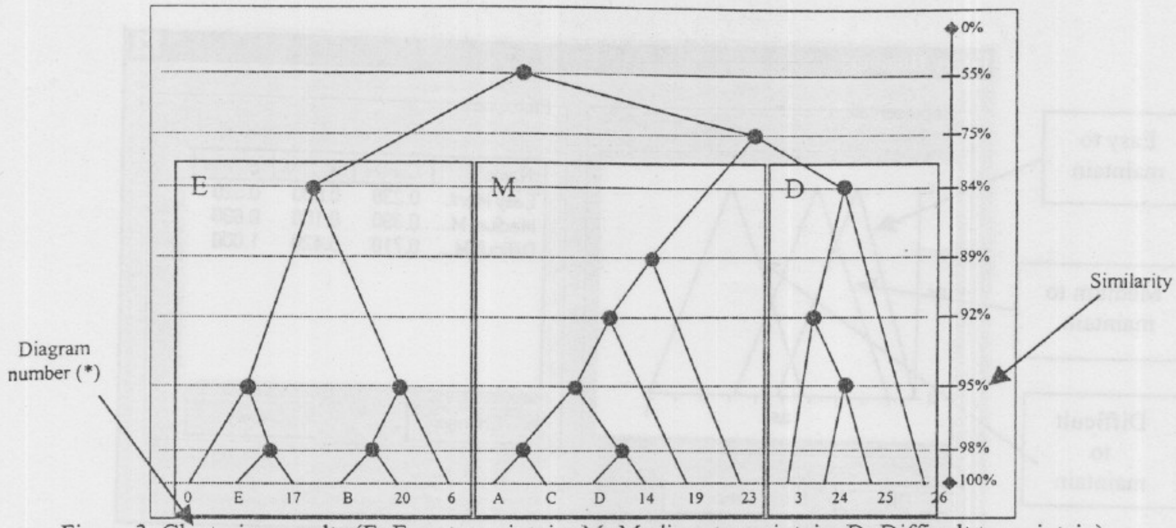

Figure 3. Clustering results (E: Easy to maintain, M: Medium to maintain, D: Difficult to maintain)
(*) We have grouped some ERD assigning them one letter because they have 100% of similarity (see appendix A)

- DATA MINING. The selected algorithms for the data mining process were summarise functions. Table 1 shows the parametric definition of the prototypes. These parameters will be modified taking into account the degree of affinity of a new ERD with the prototypes. With the new modified prototype we will be able to predict the maintainability of a new ERD.

Difficult	Understandability	Simplicity	Analisability	Modifiability	Stability	Testability
Average	6	6	6	5	6	6
Max.	6	6	6	5	6	7
Min.	5	5	5	5	5	5
Medium						
Average	3	4	4	4	4	4
Max.	4	4	4	4	5	4
Min.	3	4	3	3	3	4
Easy						
Average	3	3	3	3	3	3
Max.	3	3	3	3	4	3
Min.	2	2	2	2	2	2

Table 1. Prototypes "Easy, Medium and Difficult to maintain"

- FORMAL REPRESENTATION OF CONCEPTUAL PROTOTYPES. The prototypes have been represented as fuzzy numbers, which are going to allow us to obtain a degree of membership in the concept. For the sake of simplicity in the model, they have been represented by triangular fuzzy numbers. Therefore, in order to construct the prototypes (triangular fuzzy numbers) we only need to know their centrepoints ("centre of the prototype"), which are obtained by normalising and aggregating the metric values corresponding to the ERD of each of the prototypes (see figure 4).

- FATIGUE EFFECTS. On average the experiment lasted for less than one hour, so fatigue was not very relevant. Also, the different order of the tests helped to prevent these effects.
- PERSISTENCE EFFECTS. In order to avoid persistence effects, the experiment was carried out by subjects who had never done a similar experiment.
- SUBJECT MOTIVATION. All the professors who were involved in this experiment participated voluntarily, in order to help us in our research. We motivated students to participate in the experiment, by explaining to them that similar tasks to the experimental ones could be done in exams or practice by students, so they wanted to make the most of the experiment.
- OTHER FACTORS. Plagiarism and influence between students really could not be controlled. Students were told that talking to each other about the experiment was forbidden, but they did the experiment alone without any supervision , so we had to trust them as far as that was concerned.

Seeing the results of the experiment we can conclude that empirical evidence of the existing relationship between the independent and the dependent variables exists. But only by replicating controlled experiments, where the measures would be varied in a controlled manner and all factors would be kept constant, could we really demonstrate causality.

4.5.3 Threats to External Validity

External validity is the degree to which the results of the research can be generalised to the population under study and other research settings . The greater the external validity, the more the results of an empirical study can be generalised to actual software engineering practice. Two threats of validity have been identified which limit the ability to apply any such generalisation:

- MATERIALS AND TASKS USED. In the experiment we tried to use ERD and tasks which can be representative of real cases, but more empirical studies taking "real cases" from software companies must be done.
- SUBJECTS. To solve the difficulty of obtaining professional subjects, we used professors and advanced students from software engineering courses. We are aware that more experiments with practitioners and professionals must be carried out in order to be able to generalise from these results. However, in this case, the tasks to be performed do not require high levels of industrial experience, so, experiments with students could be appropriate (Basili et al., 1999).

In general in order to extract a final conclusion we need to replicate this experiment with a greater number of subjects, including practitioners. After carrying out the replication we will have a cumulative body of knowledge; which will lead us to confirm if the presented metrics could really be used as early quality indicators, and could be used to predict ERD maintainability.

5. Prediction of ERD maintainability

Using Fuzzy Deformable Prototypes (Olivas, 2000a; 2000b), we can deform the most similar prototype to a new ERD, and define the factors for a new situation, using a linear combination with the degrees of membership as coefficients. We will show an example of how to deform the fuzzy prototypes found in section 4.4.

Given the metric values corresponding to a new ERD (see table 2) and their normalised values (see tbale 3), the final average is 0.69, and the affinity with prototypes is shown in figure 5.

NE	NA	NR	M:NR	1:NR	N_ARY	BINARY_R	NIS_AR	SD
11	30	9	4	5	2	7	7	65

Table 2. Metric values for a new ERD

NE	NA	NR	MN_R	1:NR	N-ARY	BINARY_R	NIS_AR	SD
0.8	0.7	0.7	0.5	0.5	1.0	0.6	0.8	0.9

Table 3. Normalised values of metrics shown in table 2

focussing on static diagrams like class diagrams. The work related to dynamic diagrams is scare(Poels, 2000; Poels and Dedene, 2000), so it is an area that needs further investigation (Brito e Abreu et al., 1999). Furthermore, we will not only address maintainability sub-characteristics, we also have to focus our research on measuring other quality factors as proposed in the ISO 9126 (1999).

Acknowledgements

This research is part of the MANTICA project, partially supported by CICYT and the European Union (1FD1997-0168TIC), the DOLMEN project supported by CICYT (TIC 2000-1673-C06-06) and the CIPRESES project supported by CICYT (TIC 2000-1362-C02-02).

References

- Basili, V., Shull, F. and Lanubile, F. (1999). Building knowledge through families of experiments. *IEEE Transactions on Software Engineering*, 25(4), 435-437.
- Briand, L., Arisholm, S., Counsell, F., Houdek, F., and Thévenod-Fosse, P. (1999a). Empirical Studies of Object-Oriented Artifacts, Methods, and Processes: State of the Art and Future Directions. *Technical Report IESE 037.99/E, Fraunhofer Institute for Experimental Software Engineering*, Kaiserslautern, Germany.
- Briand, L., Bunse, C. and Daly, J. (1999b). A Controlled Experiment for evaluating Quality Guidelines on the Maintainability of Object-Oriented Designs. *Technical Report IESE 002.99/E, Fraunhofer Institute for Experimental Software Engineering*, Kaiserslautern, Germany.
- Briand L., Morasca S. and Basili V. Defining and Validating Measures for Object-Based high-level design. *IEEE Transactions on Software Engineering.* Vol. 25 No. 5, (1999) 722-743.
- Brito e Abreu, F., Zuse, H., Sahraoui, H. and Melo, W. (1999). Quantitative Approaches in Object-Oriented Software Engineering. *Object-Oriented technology: ECOOP'99 Workshop Reader*, Lecture Notes in Computer Science 1743, Springer-Verlag, 326-337.
- De Miguel, A. and Piattini, M. (1999). *Fundamentos y Modelos de Bases de Datos*. 2nd ed. Ra-Ma. (In Spanish).
- Fayyad, U., Piatetsky-Shapiro, G. and Smyth, P. (1996).The KDD Process for Extracting Useful Knowledge from Volumes of Data. *Communications of the ACM*, 39(11), 27 – 34.
- Fenton, N. and Pfleeger, S. (1997). *Software Metrics: A Rigorous Approach*. 2nd. edition. London, Chapman & Hall.
- Fenton, N. And Neil, M. (2000). Software Metrics: a Roadmap. *Future of Software Engineering.* Ed:Anthony Finkelstein, ACM, 359-370.
- Genero, M., Piattini, M. and Calero, C. (2000a). Measures to get better quality databases. *2nd International Conference on Enterprise Information Systems - ICEIS 2000,* Stafford, 49-55.
- Genero, M., Piattini, M. and Calero, C. (2000b). An Approach To Evaluate The Complexity Of Conceptual Database Models. 3rd *European Software Measurement Conference - FESMA 2000,* Madrid.
- Genero, M., Piattini, M. and Calero, C. (2000a). Early Measures For UML class diagrams. *L'Objet.* 6(4), Hermes Science Publications.
- ISO/IEC 9126-1 (1999). Information Technology- Software product quality – Part 1: Quality Model.
- Kitchenham, B., Pflegger, S. and Fenton, N. (1995). Towards a Framework for Software Measurement Validation. *IEEE Transactions of Software Engineering*, 21(12), 929-943.
- Muller, R. (1999). *Database Design For Smarties: Using UML for Data Modeling*. Morgan Kaufman.
- Olivas, J. A. and Romero, F. P. FPKD. (2000a). Fuzzy Prototypical Knowledge Discovery. Application to Forest Fire Prediction. *Proceedings of the SEKE'2000*, Knowledge Systems Institute, Chicago, Ill. USA, 47 – 54.
- Olivas, J. A. (2000b). Contribution to the Experimental Study of the Prediction based on Fuzzy Deformable Categories, PhD Thesis, University of Castilla-La Mancha, Spain.
- Poels G. On the Measurement of Event-Based Object-Oriented Conceptual Models. (2000) *4th International ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering*, June 13, Cannes, France.
- Poels, G. and Dedene, G. Measures for Assessing Dynamic Complexity Aspects of Object-Oriented Conceptual Schemes. (2000). *In: Proceedings of the 19th International Conference on Conceptual Modeling (ER 2000),* Salt Lake City, USA, 499-512.