Software Cost Estimation with Granular Models

Petr Musílek, Witold Pedrycz, Giancarlo Succi, Marek Reformat

Abstract--Estimation of effort/cost required for development of software products is inherently associated with uncertainty. In this paper, we are concerned with a fuzzy set-based generalization of the COCOMO model (f-COCOMO). The inputs of the standard COCOMO model include an estimation of project size and an evaluation of other parameters. Rather than using a single number, the software size can be regarded as a fuzzy set (fuzzy number) yielding the cost estimate also in form of a fuzzy set. The paper includes detailed results with this regard by relating fuzzy sets of project size with the fuzzy set of effort. The analysis is carried out for several commonly encountered classes of membership functions (such as triangular and parabolic fuzzy sets). The issue of designer-friendliness of the f-COCOMO model is discussed in detail. Here we emphasize a way of propagation of uncertainty and ensuing visualization of the resulting effort (cost). Furthermore we augment the model by admitting software systems to belong partially to the three main categories (namely embedded, semidetached and organic) and discuss key implications of this generalization and highlight its links with a generalized sensitivity analysis. The experimental part of the study illustrates the approach and contrasts it with the standard numeric version of the COCOMO model.

Index Terms--Software Cost Estimation, Uncertainty, Fuzzy Numbers, Information Granularity

I. INTRODUCTION

It is believed that better understanding of economy of software development would reduce the current difficulties of software production resulting in cost overruns or even project cancellations. This prospect has led to the development of many different models for software cost estimation [1][2][3][4][10][13][15]. In general, these models are based on measuring certain size or function related attributes of the software and relating these measurements to the cost or effort necessary for its development.

In this paper, we propose to extend cost estimation models by incorporating the concept of fuzziness into the measurements, models, and their parameters. The rationale for such approach stems from the vagueness present in all the data entering cost estimation process: size, function points, development modes, and other metrics and attributes are matter of (informed) guessing rather than exact measurements.

In our initial study, we have decided to extend the original COCOMO model [3]. The reasons for our choice are threefold. First, COCOMO is based on log-linear formula considered the most plausible for software cost modeling [5]. Second, the basic version of COCOMO is simple and, therefore, suitable for illustration we intend to offer. Third, the database used for construction and testing the COCMO model is readily available and allows comparison of results. Furthermore, the ideas presented in this paper are general enough to be easily extended to other cost estimation models.

The standard model we are interested in, and which forms an initial point of our investigation, assumes the form

$$E = aK^b, \qquad (1)$$

and relates effort E (expressed in person-month) with the size of software product K (that is expressed in thousand of delivered source instructions, KDSI). In the simplest version, the COCOMO model distinguishes between three basic development modes: organic (ORG), semidetached (SD) and embedded (E), cf. [3]. This classification is commonly used throughout the Software Engineering community [3] [11]. For each software project handled in this setting, the resulting cost estimation model comes with some specific values of its parameters a and b. The dataset of the COCOMO Software Project [3] contains 63 projects of different development modes and different application domains. The ensuing experiments presented in this study will be based on this dataset. To gain a better understanding into the nature of the dataset, a histogram of size of the projects is shown in Figure 1.

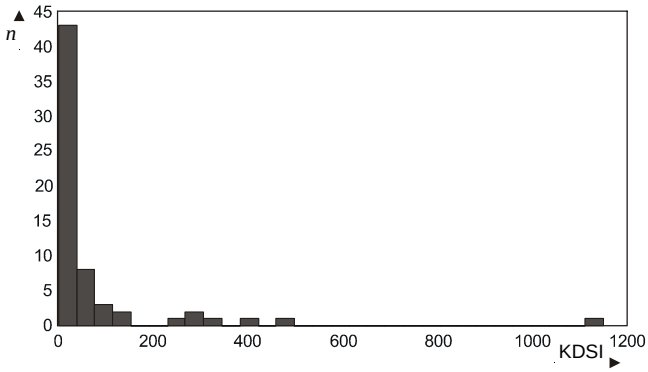

Figure 1. Histogram of the size of projects included in the COCOMO Software Project

This paper is arranged into six sections. Section 2 identifies sources of uncertainty occurring in models of cost estimation. In Section 3, we review basic notions of fuzzy sets and fuzzy arithmetic. In Sections 4 and 5, respectively, we describe simple and augmented versions of the proposed f-COCOMO model. Finally, in Section 6, we draw some important conclusions and show possible directions for future research.

Department of Electrical and Computer Engineering University of Alberta, Edmonton, Alberta, Canada T6G 2G7, Phone: (780) 492-5368, E-mail: musilek@ee.ualberta.ca

II. SOURCES OF UNCERTAINTY IN THE COST ESTIMATION MODELS

Traditionally, problem of software cost estimation relies on a single (numeric) value of size of given software project to predict the ensuing effort or cost. However, the size of the code is, especially at the beginning of the project, a matter of estimation (e.g., based on some previously completed projects that resemble the current one). Obviously, correctness and precision of such estimates are limited. It is of paramount importance to recognize this situation and come up with a technology using which we can evaluate the associated imprecision residing within the final results of cost estimation. The technology endorsed here deals with fuzzy sets.

Using fuzzy sets, size of a software project can be specified by distribution of its possible values. Commonly, this form of distribution is represented in the form of a fuzzy set. For example, a *small* software project can be described by a fuzzy set *K* in the form shown in Figure 2. What becomes apparent here is a continuous character of belongingness (membership) of elements to the given concept (being here a small software project). In this sense, fuzzy sets help alleviate a dichotomy problem (yes-no evaluation) that could be very artificial to grasp using the language of set theory. The grades of membership capture a notion of partial membership of an element to the concept (fuzzy set).

In general, a fuzzy set *K* is described by its membership function $K(x)$ which expresses the degree of membership of *x* to the fuzzy set *K* describing a certain concept (say, small project, high reliability, etc.).

III. FUZZY NUMBERS

Fuzzy sets defined in **R**, or *fuzzy numbers* (FN), can easily represent various concepts with partial membership. The membership grades can be captured through membership functions. There are several basic models of membership functions of fuzzy numbers such as triangular, parabolic, Gaussian, to name the most commonly used. Generally, membership functions may not be symmetric. Quite often this is the case as we may anticipate a certain tendency as to the estimation of the size of the code. From this perspective, triangular and parabolic fuzzy sets are of special interest in our study.

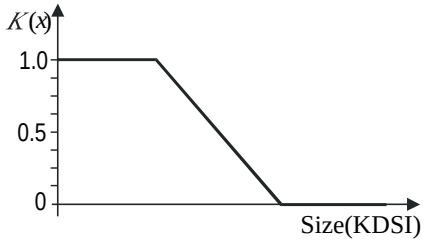

Figure 2. The size of a small software project modeled as a fuzzy set

A. Triangular Fuzzy Numbers

A triangular fuzzy number (TFN) *K*, see Figure 3, is described by a triplet $\{\alpha, m, \beta\}$, where *m* is the modal value of the fuzzy number *K*, and α and β are its left and right boundary values, respectively

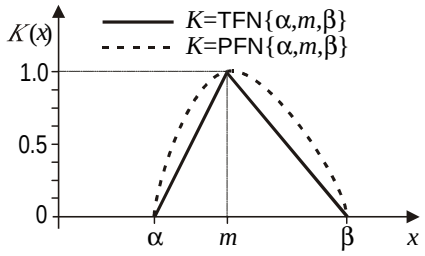

Figure 3. Triangular and parabolic fuzzy numbers

The membership function of the triangular fuzzy number *K* is defined in the form

$$K(x) = \begin{cases} \dfrac{x-\alpha}{m-\alpha} & \text{for } x \in [\alpha, m] \\ \dfrac{\beta - x}{\beta - m} & \text{for } x \in [m, \beta] \\ 0 & \text{for } x \notin [\alpha, \beta] \end{cases} \qquad (2)$$

In the sequel, we will be using a shorthand notation TFN{α, m, β} that explicitly summarizes the parameters describing this fuzzy number.

1) Parabolic Fuzzy Numbers

Parabolic fuzzy number (PFN) is composed of two parabolic segments of the membership grades delimited by modal and boundary values. It is described by a triplet PFN$\{\alpha, m, \beta\}$ with modal value *m*, and parameters of the left and right boundaries α and β.

The membership function of the parabolic fuzzy number *K* is defined in the form

$$K(x) = \begin{cases} 1 - \dfrac{(x-\alpha)^2}{(m-\alpha)^2} & \text{for } x \in [\alpha, m] \\ 1 - \dfrac{(\beta - x)^2}{(\beta - m)^2} & \text{for } x \in [m, \beta] \\ 0 & \text{for } x \notin [\alpha, \beta] \end{cases} \qquad (3)$$

see also Figure 3.

B. Fuzziness of Fuzzy Numbers

Fuzzy numbers are special fuzzy sets representing uncertain quantitative information. They are convex and normal, usually with single modal value. They are also associated with some vagueness or *fuzziness*. For the evaluation of the uncertainty of fuzzy quantities in software cost estimation problem, we use measure of relative fuzziness *f* defined as

$$f(A) = \frac{\int_{\alpha}^{\beta} A(x)\mathrm{d}x}{\mathrm{m}}, \qquad (4)$$

where *A* is a fuzzy number with support $[\alpha, \beta]$. The higher the value of $f(A)$, the more "uncertain" *A* is.

C. Fuzzy Arithmetic

Computations involving fuzzy numbers are carried out in the setting of so-called fuzzy arithmetic [6][12]. It dwells on the extension principle [16]. In a nutshell, given a function (mapping) f such that f: **R** -> **R**, the extension principle deals with the generalization of this mapping to the case of arguments being fuzzy numbers that is $B = f(A)$, where A and B are fuzzy numbers.

For instance consider that $C = f(A, B)$. Then the membership function of C is computed as

$$C(z) = \sup_{x,y \in \mathbf{R}: z = f(x,y)} [\min(A(x), B(y))] \qquad (5)$$

Essentially, the above is nothing but a problem of nonlinear programming

$$\sup_{x,y \in \mathbf{R}} [\min(A(x), B(y))] \qquad (6)$$

with constraints represented by the mapping f under consideration

$$z = f(x, y). \qquad (7)$$

The extension principle generalizes to mappings involving any number of the variables.

IV. SIMPLE F-COCOMO MODEL

Let us pose the simplest case of granular COCOMO model considering only size of the project K to be fuzzy while the coefficients a and b are crisp. We will call this model Simple f-COCOMO model. As the input variable (size K) is a fuzzy set (fuzzy number), so is the effort E.

The determination of the membership function of effort is based on the extension principle as shown in the previous section. More specifically, we get

$$E(e) = \sup_{x \in \mathbf{R}: e = ax^b} [K(x)] \qquad (8)$$

where $K(x)$ and $E(e)$ denote membership functions of the size of the code and membership function of the effort, respectively. We rewrite (8) in a different format by eliminating the constraints and moving them directly into the membership function (note that in this problem the constraint is a one-to-one mapping). We get

$$E(e) = \sup_{x \in \mathbf{R}: e = ax^b} [K(x)] = K((\frac{e}{a})^{1/b}). \qquad (9)$$

In the following, we show detailed computation of effort in simple f-COCOMO model for two types of fuzzy numbers.

A. Simple f-COCOMO Model: Triangular Numbers

For triangular fuzzy numbers, we substitute (2) for $K(x)$ in (9) to get (10).

This formula can be used directly for estimation of effort from size of software project described by triangular fuzzy set. The results of TFN based simple f-COCOMO model for a selected software project are summarized in Figure 5.

$$E(e) = \begin{cases} \dfrac{\left(\dfrac{e}{a}\right)^{1/b} - \alpha}{m - \alpha} & \text{for } e \in \left[a\alpha^b, am^b\right] \\ \dfrac{\beta - \left(\dfrac{e}{a}\right)^{1/b}}{\beta - m} & \text{for } e \in \left[am^b, a\beta^b\right] \\ 0 & \text{for } e \notin \left[a\alpha^b, a\beta^b\right] \end{cases} \qquad (10)$$

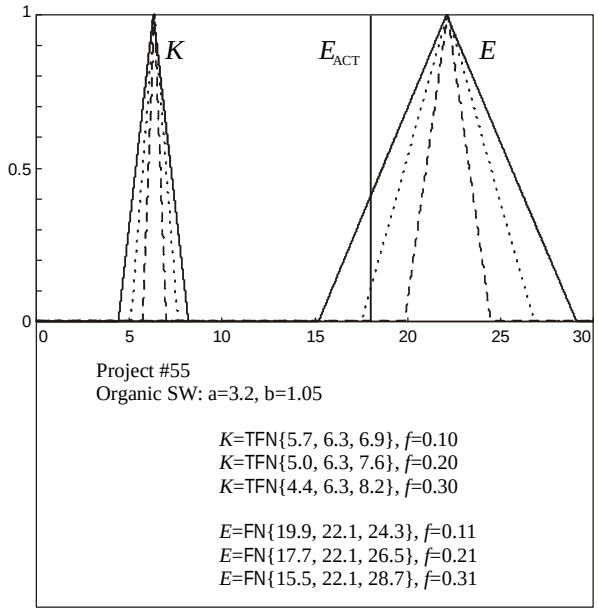

Figure 5. Simple f-COCOMO model with triangular fuzzy numbers

B. Simple f-COCOMO Model: Parabolic Numbers

For parabolic fuzzy numbers, we substitute (3) for $K(x)$ in (9) and get

$$E(e) = \begin{cases} 1 - \dfrac{\left[\left(\dfrac{e}{a}\right)^{1/b} - m\right]^2}{(m - \alpha)^2} & \text{for } e \in \left[a\alpha^b, am^b\right] \\ 1 - \dfrac{\left[\left(\dfrac{e}{a}\right)^{1/b} - m\right]^2}{(\beta - m)^2} & \text{for } e \in \left[am^b, a\beta^b\right] \\ 0 & \text{for } e \notin \left[a\alpha^b, a\beta^b\right] \end{cases} \qquad (11)$$

The obtained formula can be used directly for estimation of effort from size of software project described by parabolic fuzzy set.

The relationships (10) and (11) quantify the way in which propagation of uncertainty (granularity) occurring at the input of the model is realized. It is important to stress that uncertainty at the input level of the COCOMO model yields uncertainty at the output. This becomes obvious and, more importantly, bears a substantial significance in any practical endeavor. By changing the size of the fuzzy set of size of the code (that reflects a level of designer's confidence as to the estimate), we can easily model how such estimate impacts the effort. Obviously, a certain monotinicity property holds, that is less precise estimates of size give rise to less detailed effort estimates (cf. again Figure 5).

In addition, different categories of software require the use of different values of coefficients *a* and *b*. This also leads to different fuzziness of the estimated effort *E*. Series of plots in Figure 6 illustrate this effect.

It is evident that f-COCOMO model provides estimates of software development effort in terms of possibility distributions described by fuzzy numbers *E*. Such form of results is more confined when compared to a single numerical value obtained using common COCOMO model. In other words, even if the mean value of the fuzzy estimate *E* does not correspond to the actual value of development effort E_{ACT}, it may be covered by the estimate as possible with degree $E \in (0,1)$. This phenomenon is illustrated in Figure 7 by series of histograms of possibility values of E_{ACT}, as estimated by the f-COCOMO model for various values of spread of the input fuzzy numbers *K*. The relation between the value of spread (and, therefore fuzziness *f*) of *K* and coverage of the actual values E_{ACT} is shown in Figure 8. The two curves depict relative number of nonzero possibilities and average possibility of E_{ACT} for each value of spread $\delta = (\beta - \alpha)/m$.

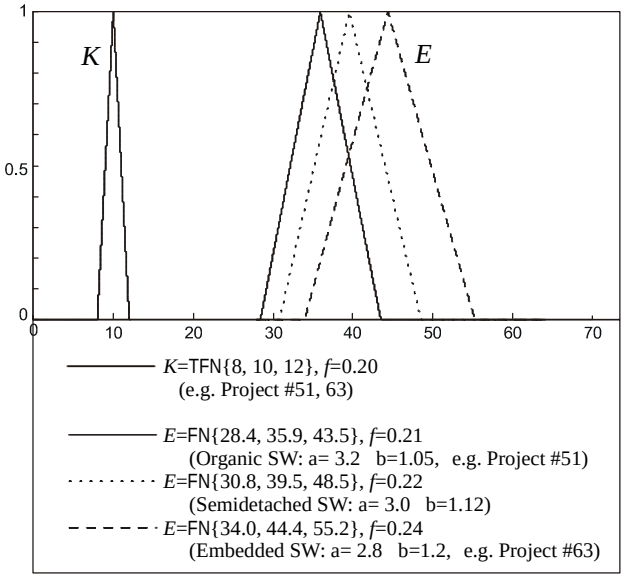

Figure 6. Simple f-COCOMO model with TFN for different development modes

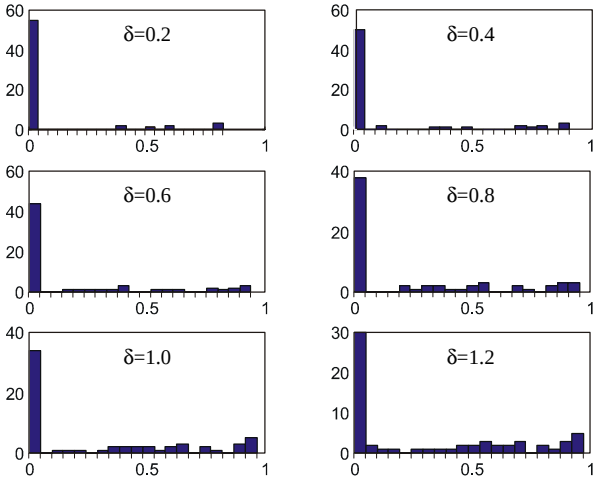

Figure 7. Histograms of possibilities of E_{ACT}

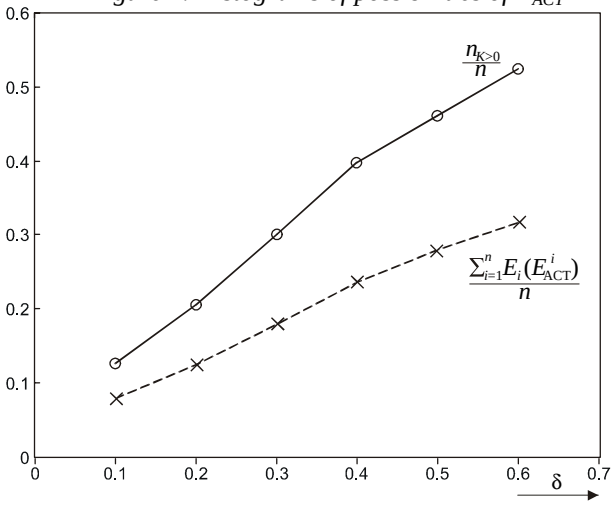

Figure 8. Spread δ vs. coverage of E_{ACT}

Similar experiments have been run among different groups of software projects to explore potential contrasts in suitability of the proposed method for some of them. The results obtained for δ=1 are summarized in Table 1.

δ=1	CAT		ORG		SD		E	
APP	(63)		0.14	(23)	0.32	(10)	0.24	(30)
BUS	0.21	(7)	0.28	(4)	0	(1)	0.18	(2)
CTL	0.23	(10)	0	(0)	0	(1)	0.26	(9)
HMI	0.44	(13)	0	(1)	0.48	(2)	0.48	(10)
SCI	0.21	(17)	0.21	(11)	0.40	(2)	0.11	(4)
SUP	0.27	(8)	0.32	(4)	0.17	(3)	0.37	(1)
SYS	0.31	(8)	0.22	(3)	0.88	(1)	0.24	(4)

Table 1: Average value of possibility measures obtained using simple f-COCOMO model for various categories of software (CAT) and application domains (APP). The data in parentheses indicate the number of cases for each group.

V. AUGMENTED F-COCOMO MODEL

An interesting generalization arises when we admit that the software project may concern a system whose membership to one of the three system categories is not obvious. Then

we may view this as a fuzzy set and view this as such in the above calculations. For instance, the fuzzy set in Figure 9 concerns the system that is predominantly embedded (with membership degree equal to 0.9) with some far lower membership (0.3) in the category of semidetached software. In this sense, the coefficient *a* is a fuzzy set with discrete membership function of the form *A* = {0.9, 0.3, 0}.Similarly, we treat the second parameter *b* as a discrete fuzzy set *B* equal to {0.9, 0.3, 0}, see again Figure 10.

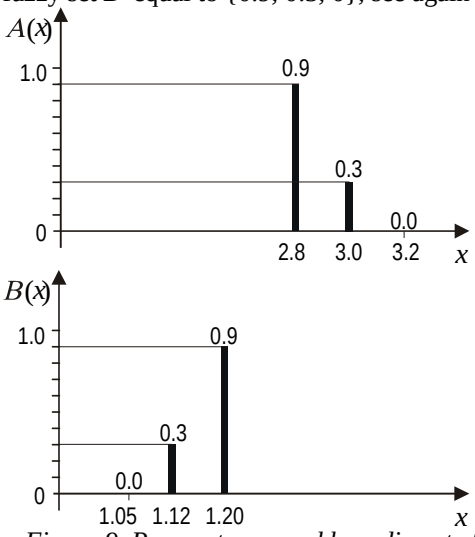

Figure 9. Parameters a and b as discrete fuzzy numbers

Taking this into consideration, we use the extension principle treating now *K*, *A*, and *B* as fuzzy sets (fuzzy numbers). More specifically, the calculations are carried out in the form

$$E(e) = \sup_{x,a,b \in \mathbf{R}: e=axb} [\min(K(x), A(a), B(b))] \quad (12)$$

This leads to relation

$$E(e) = \max_{a,b \in \mathbf{R}} \left\{ \min\left[K\left((e/a)^{1/b} \right), A(a), B(b) \right] \right\} \quad (13)$$

that can be directly used to determine the effort estimate *E*. As fuzzy numbers *A* and *B* are discrete fuzzy sets, it can be further simplified to the form

$$E(e) = \max_{\substack{a \in \{2.8,3.0,3.2\} \\ b \in \{1.2,1.12,1.05\}}} \left\{ \min\left[K\left((e/a)^{1/b} \right), A(a), B(b) \right] \right\} \quad (14)$$

The results of the augmented f-COCOMO model applied to a project with *K*={8,10,12} and development mode parameters *A* = *B*={0.9, 0.3, 0} (cf. Figure 10) are shown in Figure 11.

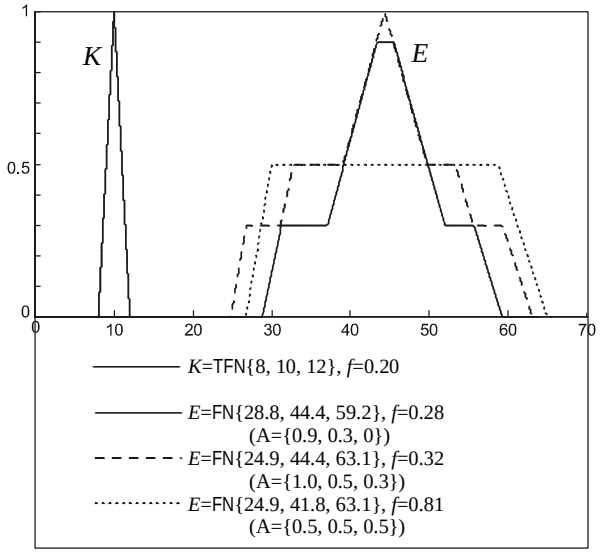

Figure 11. Augmented f-COCOMO model with triangular fuzzy numbers

The figure shows also two other interesting examples with development mode parameters equal to {1, 0.5, 0.3}, and {0.5, 0.5, 0.5}. This series of results also demonstrates the growth of uncertainty of resulting estimate of effort *E* with growing vagueness of determination of the development modes.

VI. CONCLUSIONS AND FUTUTRE RESEARCH

In this study, we have introduced an important generalization of the COCOMO software cost estimation model by augmenting by the technology of fuzzy sets. The detailed formulas involving several main classes of fuzzy numbers were derived and a set of intensive experiments provided a detailed insight into the conceptual and algorithmic essence of this generalization.

One should stress that the f-COCOMO model supports an important aspects of the "*what - if* " analysis [14]. The model admits inputs (say, size of code) that are non-numeric and therefore more acceptable to the designer and project manager. Fuzzy sets (as opposed to standard interval analysis) create a more flexible, highly versatile development environment. First, they help articulate the estimates and their essence (e.g., by exploiting fuzzy numbers described by asymmetric membership functions). Second, they generate a feedback as to the resulting uncertainty (granularity) of the results. The decision-maker is no longer left with a single number estimate which could be highly misleading in many cases and lead to the belief as to the relevance of the obtained results. Moreover, by capturing the uncertainty of the initial data (estimates), one can monitor the behavior (quality) of the cost estimates over the course of the software project. This facet adds up a new conceptual dimension to the models of software cost estimation by raising awareness of the decision making with regard to the quality of the initial data needed by the model.

The methodology of fuzzy sets giving rise to the f-COCOMO extension is sufficiently general to be applied to other models of software cost estimation (such as the well-known function point method [13]) and to other areas of quantitative software engineering.

VII. ACKNOWLEDGEMENTS

Support from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Alberta Software Engineering Research Consortium (ASERC) is gratefully acknowledged.

VIII. REFERENCES

1. A. J. Albrecht, Measuring application development productivity, *SHARE/GIDE IBM Application Development Symposium*
2. V. R. Basili, K. Freburger, Programming Measurement and Estimation in the Software Engineering Laboratory, *Journal of Systems and Software*, 2, 2(1981), 47-57
3. B. W. Boehm, *Software Engineering Economics*, Prentice Hall, 1981
4. B. W. Boehm et al., *Software Cost Estimation with Cocomo II, Prentice Hall*, 2000
5. L. C. Briand, K. El Emam, I. Wieczorek, Explaining the Cost of European Space and Military Projects, *ICSE' 99: Proceedings of the International Conference on Software Engineering*, Los Angeles, CA, ACM, 1999, 303-312
6. D. Dubois, H. Prade, Fuzzy real algebra - some results, *Fuzzy Sets and Systems*, 2, 2, (1978), 327-348.
7. G. W. Jones, *Software Engineering*, J. Wiley, N. York, 1990.
8. E. M. Hall, *Managing Risk. Methods for Software Development*, Addison-Wesley, Reading, MA, 1998.
9. A. Kaufmann, M. M. Gupta, Introduction to fuzzy arithmetic : theory and applications, Van Nostrand Reinhol, NY, 1985.
10. B. Kitchenham, Software Development Cost Models, in P. Rook, ed., *Software Reliability Handbook*, Elsevier, NY, 1990
11. P. Laplante, *Dictionary of Computer Science, Engineering and Technology*, CRC Press, 2000
12. M. Mares, *Computation over fuzzy quantities*, CRC Press, Boca Raton, FL, 1994.
13. J. E. Matson, B. E. Barrett, J. M. Mellichamp, Software Development Cost Estimation Using Function Points, *IEEE Trans. on Software Engineering*, 20, 4 (1994), 275-287
14. W. Pedrycz, F. Gomide, *An Introduction to Fuzzy Sets. Analysis and Design*, MIT Press, Cambridge, MA, 1998.
15. L. H. Putnam, A General Empirical Solution to the Macro Software Sizing and Estimating Problem, *IEEE Trans. on Software Engineering*, 4, 4 (1978), 345-361
16. L. A. Zadeh The Concept of a Linguistic Variable and Its Application to Approximate Reasoning *Information Sciences*, 8 (1975),199--251