A Systematic Review of Software Development Cost Estimation Studies

Magne Jørgensen and Martin Shepperd

Abstract—This paper aims to provide a basis for the improvement of software estimation research through a systematic review of previous work. The review identifies 304 software cost estimation papers in 76 journals and classifies the papers according to research topic, estimation approach, research approach, study context and data set. A Web-based library of these cost estimation papers is provided to ease the identification of relevant estimation research results. The review results combined with other knowledge provide support for recommendations for future software cost estimation research, including 1) increase the breadth of the search for relevant studies, 2) search manually for relevant papers within a carefully selected set of journals when completeness is essential, 3) conduct more studies on estimation methods commonly used by the software industry, and 4) increase the awareness of how properties of the data sets impact the results when evaluating estimation methods.

Index Terms—Systematic review, software cost estimation, software effort estimation, software cost prediction, software effort prediction, research methods.

1 Introduction

This paper reviews journal articles on software development cost[1] estimation with the goal of supporting and directing future estimation research. Our review differs from previous reviews [1], [3], [16] with respect to the following elements:

- *Different goal.* While the main goal of this review is to direct and support future estimation research, the other reviews principally aim at introducing software practitioners or novice estimation researchers to the variety of formal estimation models. This difference in goal leads to a different focus. For example, our review focuses on research methods and does not include a comprehensive description of the different estimation methods.
- *More comprehensive and systematic review.* We base the analysis on a systematic search of journal papers, which led to the identification of 304 journal papers. The review in [3] is based on about 130, that in [16] on about 65, and that in [1] on about 40 journal and conference papers. None of the previous reviews describe a systematic selection process or state clear criteria for inclusion or exclusion.
- *Classification of studies.* We classify the software development estimation papers with respect to estimation topics, estimation approach, research approach, study context and data set. We found no classification, other than that with respect to estimation method in the other reviews.

Based on what we believed were interesting issues to analyze, we posed the eight research questions described in Table 1. The underlying motivation for all questions was our goal of improvement of the software cost estimation research. These research questions guided the design of the review process.

The remaining part of this paper is organized as follows: Section 2 describes the review process. Section 3 reports the review results. Section 4 summarizes the main recommendations for future research on software cost estimation.

2 Review Process

2.1 Inclusion Criteria

The main criterion for including a journal paper in our review is that the paper describes *research on software development effort or cost estimation.* Papers related to estimation of software size, assessment of software complexity, or identification of factors correlated with software development effort, are only included if the main purpose of the studies is to improve software development effort or cost estimation. We exclude pure discussion/opinion papers.

There were examples of papers describing, essentially, the same study in more than one journal paper. Fortunately, the number of such cases was small and would not lead to important changes in the outcome of our analysis. We decided, therefore, not to exclude any papers for that reason. However, when conducting a review of a particular phenomenon, e.g., the robustness of a particular finding, we would recommend a clearer distinction between paper, study, and data set.

[1] The main cost driver in software development projects is typically the effort and we, in line with the majority of other researchers in this field, use the terms "cost" and "effort" interchangeably in this paper.

- *M. Jørgensen is with Simula Research Laboratory, Marin Linges v 17, Fornebu, PO Box 134, NO-1325 Lysaker, Norway.*
E-mail: magnej@simula.no.
- *M. Shepperd is with the School of IS, Computing and Mathematics, Brunel University, Room SJ048, Uxbridge, UB8 3PH, UK.*
E-mail: martin.shepperd@brunel.ac.uk.

Manuscript received 7 Nov. 2005,; revised 13 Mar. 2006; accepted 26 Oct. 2006; published online 30 Nov. 2006.
Recommended for acceptance by D. Rombach.
For information on obtaining reprints of this article, please send e-mail to: tse@computer.org, and reference IEEECS Log Number TSE-0299-1105.

0098-5589/07/$20.00 © 2007 IEEE Published by the IEEE Computer Society

TABLE 1
Research Questions

Research Question	Main Motivation
RQ1: Which and how many journals include papers on software cost estimation?	Support cost estimation researchers with a list of journals with potentially relevant papers.
RQ2: To what extent are software cost estimation researchers aware of the breadth of potential estimation study sources?	Identify possible shortcomings of cost estimation researchers' searches for related work.
RQ3: Which journal is the dominant software cost estimation journal? To what extent does this journal have research topic biases?	Identify the most important software cost estimation journal and the extent to which this journal reflects the totality of software cost estimation research.
RQ4: How easy is it to identify relevant software cost estimation journal papers?	Identify possible shortcomings of internet and library-based searches to identify cost estimation papers.
RQ5: How many researchers are there who have a long term interest in software cost estimation? To what extent do the interests of these researchers affect the distribution of research topics?	Assess the vulnerability of software cost estimation research; for example, there being few researchers on particular topics may increase vulnerability.
RQ6: What are the most investigated software cost estimation research topics and how has this changed over time?	Identify trends and possible shortcomings / opportunities for research topic focus.
RQ7: What are the most investigated estimation methods and how has this changed over time?	Identify trends and possible shortcomings / opportunities for estimation method focus.
RQ8: What are the most frequently applied research methods, and in what study context (including use of data sets)? How has this changed over time?	Identify trends and possible shortcomings / opportunities for the use of research methods.

2.2 Identification of Papers

The search for papers was based on an issue-by-issue, manual reading of titles and abstracts of all published papers, starting with volume 1, in more than 100 potentially relevant, peer-reviewed journals with papers written in English. These journals were identified through reading reference lists of cost estimation papers, searching on the Internet for previously not identified, relevant journals, and using our own previous experience. Both authors constructed a list of potentially relevant journals independently. These two lists were merged together. In spite of the high number of identified journals, it is possible that there are, e.g., national or company-specific journals with software cost estimation papers that we have missed.

Papers that were potential candidates for inclusion in our review were read more thoroughly to decide whether to include them or not. In total, 304 relevant papers were found in 76 of these journals. The 76 journals and 304 papers are listed in, respectively, Appendices A and C. The search was completed in April 2004.

2.3 Classification of Papers

For the purpose of our analysis, i.e., addressing the research questions, the authors of this paper classified the papers according to the properties and categories listed in Table 2. The categories are explained in Appendix B.

The categories were based on categories applied in previous estimation studies, e.g., [2], [3], [5], [8], [16] and adapted to the needs of our analysis. In particular, the classification aims at answering the research questions RQ6, RQ7, and RQ8. The classification schema was developed for the purpose of our review and is not intended to be a general-purpose classification of software effort estimation studies. We believe, however, that the classification may be useful for other researchers searching for relevant papers on, for example, a particular estimation approach. Note that most of the categories are nonexclusive, e.g., a paper may focus on more than one estimation approach and apply more than one research approach.

The initial classification was performed by the first author of this paper. The robustness of the classification schema and process was evaluated by testing a random sample of 30 papers (about 10 percent of the total). This classification test was performed by the second author of this paper. The classification test showed that several of the initial descriptions of categories were vague. In fact, there were disagreements over 59 out of the 150 classifications, i.e., 39 percent. Most of the disagreements were due to

TABLE 2
Classification of Papers

Property	Categories
Research topic	Estimation method, production function, calibration of models, size measures, organizational issues, uncertainty assessments, measures of estimation performance, data set properties, other
Estimation approach	Regression, analogy, expert judgment, work break-down, function points, classification and regression trees, simulation, neural network, theory, Bayesian, combination of estimates, other, not relevant
Research approach	Theory, survey, experiment, case study, development of estimation method, history-based evaluation, own experience, real-life evaluation, review, simulation, other
Study context	Students and/or student projects, professionals and/or industrial software projects, not relevant
Data sets	Reference to data set applied

recurring issues, e.g., different interpretations regarding how much review of previous studies a paper should include to be classified as a review paper. Fortunately, fewer than 3 percent of the initial classifications were clearly inconsistent with the category descriptions. In other words, the main problem was the clarity of the descriptions and not so much the initial classification itself. We therefore decided that the initial classification had a degree of accuracy sufficiently high for the purpose of this paper, given that we 1) clarified the descriptions that led to disagreements (12 of the category descriptions were clarified), and 2) reclassified the papers that belonged to the problematic categories (109 papers were reread and considered for change of classification; 21 reclassifications were made in total). This clarification and reclassification was then completed by the first author.

In spite of this effort to improve the reliability of the classification, it is likely that several of our classifications are subject to discussion, that the descriptions could be improved further, and that some of the papers are classified incorrectly. However, we believe that on the whole, the current classification is of sufficiently high quality to serve the purpose of our analysis. Furthermore, BESTweb (www.simula.no/BESTweb), the underlying bibliographic database, is publicly available and other researchers are free to reanalyze the data and draw their own conclusions.

2.4 Analysis

The classification of research papers provided a general picture of the characteristics of the software estimation research. This general picture served as a starting point for deeper investigation of findings that seemed, from the authors' perspectives, to suggest important shortcomings in estimation research and possibilities for improvement.[2]

2.5 Threats to Validity

The main threats to the validity of our review we have identified are these:

Publication bias: The exclusion of conference papers and reports is based mainly on practical concerns, including workload, e.g., the problems of identifying all relevant conferences and the amount of analysis needed to handle the fact that many journal papers are improvements of previously published conference papers. This exclusion of conference papers would be more difficult to defend if we had studied a particular estimation phenomenon, e.g., whether the estimation method COCOMO or Function Points has been shown to be more accurate. In that type of study, all relevant papers should be identified and reviewed, regardless of type of source. However, we are interested mainly in properties of the research into software cost estimation. The main bias of our inclusion of journal papers is simply one toward papers with high scientific quality. We analyzed informally a selection of software cost estimation papers (about 50) published at conferences and found that the research topics, methods, study designs, and study contexts of these papers were similar to those of the journal papers. An important difference between conference and journal papers, relevant for our review, was that estimation experience reports not written by research scientists are typically only published at industry conferences. This means that we probably have excluded a major source of information about the software industry's experience in using different estimation approaches. Our analyses and recommendations try to reflect this bias. Apart from this difference, we believe that our analyses and recommendations would not be very different if we had included conference papers.

Another potential publication bias is that which might result from *not* publishing estimation research that has nonsignificant results, results that did not yield the desired outcome, company-confidential results, or results that were conducted on topics that do not fit into the common software engineering journals. The size and effect of the potential publication biases would be interesting to study but would require a study design different from ours.

Vested interests of the authors: Both authors of this paper are active researchers on software cost estimation. One of the authors mainly publishes papers on expert estimation and the other on formal, mainly analogy-based, estimation models. We are not aware of biases we may have had when categorizing the papers, but the reader should be aware of the possible impact of our own interests on the analyses. In particular, it is possible that the recommendations we make are affected by our interests and opinions.

Unfamiliarity with other fields: Clearly, estimation is a topic that is relevant to many fields; thus, it is possible that we have overlooked essential work and relevant journals published in another discipline, e.g., civil engineering.

3 Results

3.1 Relevant Software Cost Estimation Research Journals (RQ1)

We found papers on software cost estimation in as many as 76 journals, i.e., the total number of journals with such papers is high, or at least higher than we expected. The 76 journals are listed in Appendix A. The 10 journals with five or more papers on software cost estimation are displayed in Table 3, together with the corresponding number, proportion, and cumulative proportions of papers. These 10 journals include two-thirds of all identified journal papers on software cost estimation. Reading *only* the 10 most relevant journals means that important research results may be missed.

3.2 Researcher Awareness of Relevant Journals (RQ2)

We were interested in the degree to which software estimation researchers were aware of and systematically searched for related research in more than a small set of journals. An indication of this awareness was derived through a random selection of 30 software cost estimation journal papers (about 10 percent of the total). These 30 papers are marked with (S) in Appendix C. The reference lists of each of these papers were examined. From this examination we found that:

- The typical (median) software cost estimation study relates its work to and/or builds on cost estimation

[2] Our analysis examines study properties one by one, i.e., is a rather reductionist analysis. An alternative approach for this type of review, pointed out by one of the reviewers of this paper, would be one based on "prototypical sets of properties," i.e., to find typical software cost estimation study profiles and categorize and analyze papers according to profiles instead of single properties. That type of review may have several advantages and should be considered when conducting future reviews.

TABLE 3
Most Important Software Cost Estimation Journals

Rank	Journal	Number	Proportion	Cumulative Proportion
1	IEEE Transactions on Software Engineering	51	17%	17%
2	Information and Software Technology	47	15%	32%
3	Journal of Systems and Software	42	14%	46%
4	Empirical Software Engineering	12	4%	50%
5	IEEE Software	11	4%	54%
6	Communications of the ACM	9	3%	57%
7	Software Quality Journal	9	3%	60%
8	American Programmer	6	2%	62%
9	Information and Management	6	2%	64%
10	Journal of Software Maintenance and Evolution	5	2%	66%

studies found in only three different journals. We examined the topics of the papers and found that some of the papers did not refer to previously published, seemingly relevant, papers on the same research topic. This indicates, we believe, that many research papers on software cost estimation are based on information derived from a rather narrow search for relevant papers.

- The most referenced journal, with respect to related cost estimation work, was the *IEEE Transactions on Software Engineering (IEEE TSE)*. Estimation papers from this journal were referred to in as many as 70 percent of the papers. Relative to the number of cost estimation papers in each journal, there were surprisingly many references to the *Communications of the ACM*; fully 30 percent of the papers made references to at least one of the journal's nine papers on software cost estimation. There were, relative to the number of cost estimation papers available, few references to papers in *Information and Software Technology* (20 percent) or the *Software Quality Journal* (3 percent).
- Papers published in Information System (IS) journals, e.g., *Information and Management*, contained mainly references to other IS journals, while papers published in Software Engineering (SE) journals mainly contained references to other SE journals. The only exception here was *IEEE TSE*. This journal was referred to frequently by both communities. To communicate software cost estimation results to other researchers from both communities, results may benefit from being published in *IEEE TSE*.
- Few papers referred to estimation results outside the software community, e.g., to studies in forecasting, human decision making, and project management. The main portion of references to sources outside the software community seems to be to literature on statistics.
- We made a separate test on references to two journals outside the software engineering field: the *International Journal of Forecasting* and the *International Journal of Project Management*. The former includes many relevant results on how to predict future outcomes (forecasting) and contains papers and results that are highly relevant for many papers on software cost estimation. The latter is the major journal on project management and provides results related to the project context of software cost estimation as well as project cost estimation results. Out of the 30 journal papers, only one referred to the *International Journal of Project Management* and none to the *International Journal of Forecasting*! Both journals can be accessed by using digital libraries, e.g., Inspec and Google Scholar, but may be hard to identify due to different use of terminology.

The above evidence indicates that several authors use narrow criteria when searching for relevant cost estimation papers. The most important issue, however, is whether papers on software cost estimation miss prior results that would improve the design, analysis, or interpretation of the study. This cannot be derived from our review alone. Our impression, however, based on the review presented in this paper and previous experience, is that the major deviation from what we assess as best research practice is the lack of identification and integration of results from journals outside the "top five" journals in Table 3. This is exemplified by the documented lack of reference to relevant papers in the *International Journal of Forecasting* and the *Journal of Project Management*.

An example of the incomplete summary of related work can be found in one of the authors' (Shepperd's) own papers [12]. That paper claims that *"A number of researchers have used this type of approach* [estimation by analogy] *with generally quite encouraging results."* This claim was based on references to 6 studies (out of 20 relevant studies). Of the 6 studies, 3 were conducted by one of the authors. Including all relevant studies would have led to less optimism. Out of the 20 studies, 9 were in favor of estimation by analogy, four were inconclusive, and seven were in favor of regression-based estimation. An incomplete identification of relevant studies may, consciously or unconsciously, lead to conclusions that are biased toward the researcher's own vested interests [13].

3.3 Most Important Software Cost Estimation Journal (RQ3)

IEEE TSE was found to be the dominant software cost estimation research journal when considering both the number of papers (see Table 3) and citations made by other researchers (see analysis in Section 3.2). It is, therefore, an interesting question whether *IEEE TSE* has publication biases, e.g., whether it favors certain topics and rejects papers on other topics. A strong publication bias in *IEEE TSE* could, for example, have the unfortunate consequence of directing software cost estimation researchers' focus towards the topics most easily accepted by the journal. To analyze this, we compared the distribution of research topics, estimation approaches, and research methods of *IEEE TSE* with the total set of estimation papers published in journals.

We found that the distributions of *IEEE TSE* cost estimation topics, estimation methods, and research methods were similar to the corresponding distributions of the total set of papers. *IEEE TSE* papers had a somewhat stronger focus on "Function Point"-based estimation methods and less focus on "Expert Judgment," but even here the difference was small. Moreover, there may be a time effect, since this topic was more popular in the 1980s and 1990s while not all other journals, e.g., *Empirical Software Engineering,* were publishing during this time. This suggests that the *IEEE TSE* software cost estimation papers reflect the total set of software cost estimation papers reasonably well. Notice that we have only studied high level types of publication bias and do not exclude the possibility that there are other types of difference regarding journal publication, for example, differences in the formalism used when describing estimation methods. We have had no access to information about papers rejected by *IEEE TSE*. There may, consequently, be publication biases that are not visible from a study of only the published papers.

3.4 Identification of Relevant Software Cost Estimation Research Journal Papers (RQ4)

Our search for estimation papers was, as described earlier, based on a manual issue-by-issue search of about 100 potentially relevant journals. This is, we believe, an accurate method of identifying relevant research papers, given that the people conducting the search possess sufficient expertise. It does, however, require much effort and, if possible, it should be replaced with more automated search and identification methods. The main tool for this is the use of digital libraries. To indicate the power of the digital libraries we conducted the following evaluation:

1. The search term: *"software cost estimation" OR "software effort estimation"* was applied in the digital research libraries Google Scholars (scholar.google.com) and Inspec. Wider searches would, obviously, lead to more complete searches. The number of "false alarms" would, however, also increase strongly and the benefit of automatic search may easily disappear. The search "software" AND ("cost" OR "effort") AND "estimation," for example, led to the identification of about 278,000 records in Google Scholar. A similar example using Inspec is presented later in this section.
2. The papers identified by using the above searches were compared with the set of papers from our manual search.

The main conclusion from this simple test was that the use of the search facilities of digital libraries to search for common software cost estimation terms is not sufficient for the identification of all relevant software cost estimation research. The search in Google Scholars (October 2005) resulted in 998 records. However, only 92 out of the 304 journal papers were identified, i.e., a recall rate of only about 30 percent. The search in Inspec identified 763 journal papers. As expected, Inspec performed better and identified 177 of the 304 papers, i.e., a recall rate of about 60 percent. The joint set of Google Scholar and Inspec led to the identification of 204 of the 304 papers, i.e., a recall rate of almost 70 percent. Nevertheless, even the use of both libraries missed a substantial part of relevant papers.

A closer examination of the titles and abstracts of the journal papers not identified by Google Scholar or Inspec suggests that the most typical reasons for nonidentification in our test were:

- A variety of substitutes for the term "estimation," e.g., "prediction," "scheduling," "cost modeling," "use of costing models," "application of cost estimation techniques," "calculating the cost," and "use of estimation technology."
- A variety of terms used instead of "software," e.g., "system," "maintenance," "project," and "task."
- Use of more specific terms derived from particular estimation methods, e.g., "function points," instead of more general estimation terms.
- Studies dealing with specific estimation topics, e.g., studies on the use of accuracy measures or cost interval predictions, may not use the general estimation terms.

In many cases, a software cost estimation researcher will use more specific terms when searching for relevant papers. This may reduce the effect of the last two of the above reasons and higher coverage rates can be expected. However, there remains the problem that a number of synonyms are used for the terms "estimating" and "software." A paper written by one of this paper's authors (Jørgensen) titled "Experience with the Accuracy of Software Maintenance Task Effort Prediction Models" [9], illustrates this problem. The paper compares different formal software development effort estimation models. Assume that a researcher wants to summarize related work on regression-based software cost estimation models. Our paper is clearly relevant for this summary but turns out to be difficult to identify since we use "prediction" instead of "estimation" and "maintenance task" instead of, e.g., "software development." A wider search, e.g., the search ("software" AND ("prediction OR "estimation")) in Inspec, identified the paper. The set of identified records was, however, as high as 11,303. This is too large a set to be meaningful for an identification of relevant papers. In addition, if we had used variants of the terms, e.g., "predict" instead of "prediction," we would not have identified the paper in spite of a very wide search. It is evident that searches in digital libraries that are sufficiently wide to identify relevant software cost estimation research can easily lead to higher workload than purely manual search processes.

Manual searches do not guarantee completeness, either. It is easy to make errors and to miss relevant journals. To illustrate the point, when we conducted this evaluation of Google Scholar and Inspec, we found three journals with potentially relevant papers that we had not identified during our manual search. These journals were country- and company-specific IT journals.

The current situation, with a lack of standardized terminology, may require a manual search of titles and abstracts in a carefully selected set of journals to ensure the proper identification of relevant research on software cost estimation. In the longer term, enforcing a more standard scheme of classification for software cost estimation or a more standardized use of keywords should be an important goal for digital libraries and researchers on software cost estimation. As an illustration on the current state, the ACM Computing Classification System prescribes the keyword "cost estimation" for the types of studies included in our review. As documented earlier this section, the use of this keyword (even when used together with the keyword "effort estimation") identified only 60 percent of the estimation papers found by manual search. This search included the title and the abstract of the papers, i.e., a search for papers applying the term "cost estimation" as keyword will identify even less relevant papers.

3.5 Researchers with a Long-Term Interest in Software Cost Estimation (RQ5)

There seem to be few researchers with a long-term focus on software cost estimation research. We found, for example, only 13 researchers with more than five journal papers on software cost estimation. Most of these researchers are still active; nine of them, for example, published journal papers on software cost estimation in the period 2000-2004.

The potential importance of researchers with a long-term focus on software cost estimation can be illustrated through an analysis of the papers covering the topics "measures of estimation performance" and "data set properties." These topics are basic research on cost estimation necessary for meaningful analyzes and evaluations of estimation methods [6], [11], [13], [15]. There is, for example, currently no precise use of estimation terminology in research studies [7], which easily leads to problems when interpreting estimation accuracy measurement. Many of these basic topics require more than average experience and maturity in software cost estimation research. Not surprisingly, almost two-thirds of the papers on these topics were authored or coauthored by the nine active, long-term focus researchers identified above.

The current number of active researchers with a long-term focus on software cost estimation is low compared to the number of research topics and estimation approaches, and there could be a strong research bias toward these researchers' interests, e.g., through the active researchers' positions as reviewers, supervisors, and project leaders and from ideas generated from their research. Fortunately, these researchers seem to cover a wide spectrum of research topics, estimation approaches, and research methods: They publish papers on most topics and on most estimation approaches, and they apply most types of research methods. Several cost estimation topics have been the long-term focus of only one active researcher or research group.

A consequence of the low number of active researchers with a long-term focus on software cost estimation is that a substantial proportion of the research papers have authors (and reviewers) with a shorter-term focus on software cost estimation, e.g., as part of a PhD. The body of knowledge within software cost estimation and related areas is very large, and without supporting environments we cannot expect fresh researchers to deliver high-quality research. We recommend that research groups with an interest in software cost estimation should address this challenge, e.g., by developing and using supporting tools such as BESTweb and by cooperation with relevant estimation research expertise in software engineering and other disciplines.

TABLE 4
Research Topics

Research topic	-1989	1990-1999	2000-2004	Total
Em	30 (73%)	96 (59%)	58 (58%)	184 (61%)
Pf	8 (20%)	7 (4%)	3 (3%)	18 (6%)
Cm	3 (7%)	13 (8%)	4 (4%)	20 (7%)
Sm	5 (12%)	39 (24%)	16 (16%)	60 (20%)
Oi	9 (22%)	25 (15%)	14 (14%)	48 (16%)
Un	2 (5%)	10 (6%)	13 (13%)	25 (8%)
Ep	2 (5%)	8 (5%)	6 (6%)	16 (5%)
Ds	0 (0%)	1 (1%)	2 (2%)	3 (1%)
Ot	0 (0%)	3 (2%)	1 (1%)	4 (1%)

The abbreviations used are: Estimation method = Em, Production function = Pf, Calibration of models = Cm, Size measures = Sm, Organizational issues = Oi, Uncertainty assessments = Un, Measures of estimation performance = Ep, Data set properties = Ds, Other = Ot.

One paper may discuss more than one topic.

3.6 Distribution of Research Topics (RQ6)

Table 4 shows the distribution of topics for three periods and in total.

The distribution in Table 4 suggests that:

- The most common research topic, with 61 percent of the papers, is the introduction and evaluation of estimation methods (Em).
- The distribution of topics over time is quite stable, with a few exceptions. Papers on the production functions were more common before 1990, and there has been an increase in focus on the uncertainty of effort estimates since 2000.

Table 4 indicates that the most papers examine cost estimation from a technical point of view. This is connected with a strong focus on research on formal estimation model; see Section 3.7. Sixteen of the papers (category Oi) examine the organizational context of cost estimation. This focus on technical issues does not correspond well with what we have experienced are basic problems in many software companies. Contextual factors, such as the mix of planning and estimation processes [4], may be prerequisites for meaningful use of estimation methods.

3.7 Distribution of Estimation Methods (RQ7)

Table 5 shows the distribution of papers on different estimation approaches per period and in total.

The distribution in Table 5 suggests that:

- Regression-based estimation approaches dominate. Notice that regression-based estimation approaches include most common parametric estimation models, e.g., the COCOMO model. Roughly half of all estimation papers try to build, improve or compare with regression model-based estimation methods.

TABLE 5
Estimation Approaches

Estimation approach	-1989	1990-1999	2000-2004	Total
Rg	21 (51%)	76 (47%)	51 (51%)	148 (49%)
An	1 (2%)	15 (9%)	15 (15%)	31 (10%)
Ej	3 (7%)	22 (13%)	21 (21%)	46 (15%)
Wb	3 (7%)	5 (3%)	4 (4%)	12 (4%)
Fp	7 (17%)	47 (29%)	14 (14%)	68 (22%)
Ct	0 (0%)	5 (3%)	9 (9%)	14 (5%)
Si	2 (5%)	4 (2%)	4 (4%)	10 (3%)
Nn	0 (0%)	11 (7%)	11 (11%)	22 (7%)
Th	20 (49%)	14 (9%)	5 (5%)	39 (13%)
By	0 (0%)	1 (1%)	6 (6%)	7 (2%)
Cb	0 (0%)	3 (2%)	2 (2%)	5 (2%)
Ot	2 (5%)	7 (4%)	16 (16%)	25 (8%)

The abbreviations used are: Rg = Regression, An = Analogy, Ej = Expert judgment, Wb = Work break-down, Fp = Function Point, Ct = Classification and regression trees, Si = Simulation, Nn = Neural network, Th = Theory, By = Bayesian, Cb = Combination of estimates, Ot = Other

One paper may discuss more than one approach.

TABLE 6
Research Approaches

Research approach	-1989	1990-1999	2000-2004	Total
Th	6 (15%)	6 (4%)	6 (6%)	18 (6%)
Sv	1 (2%)	20 (12%)	6 (6%)	27 (9%)
Ex	1 (2%)	9 (6%)	9 (9%)	19 (6%)
Ca	1 (2%)	3 (2%)	4 (4%)	8 (3%)
Dm	18 (44%)	73 (45%)	50 (50%)	141 (46%)
Hv	22 (54%)	90 (55%)	54 (54%)	166 (55%)
Ox	4 (10%)	9 (6%)	5 (5%)	18 (6%)
Rl	0 (0%)	8 (6%)	3 (3%)	11 (4%)
Re	5 (12%)	17 (10%)	4 (4%)	26 (9%)
Si	2 (5%)	8 (5%)	7 (7%)	17 (6%)
Ot	0 (0%)	0 (0%)	2 (2%)	2 (1%)

The abbreviations used are: Th = Theory, Sv = Survey, Ex = Experiment, Ca = Case study, Dm = Development of estimation method, Hv = History-based evaluation, Ox = Own experience, Rl = Real-life evaluation, Re = Review, Si = Simulation, Ot = Other.

One paper may apply more than one approach.

- The proportion of papers on analogy-based estimation models is increasing.
- The proportion of papers on expert judgment-based estimation models is increasing.
- The popularity of research on the function point-based estimation approaches was at its highest in the 1990s. In that period, almost one-third of the papers tried to improve upon, or make comparisons with, function point-based estimation methods.
- Theory-based estimation approaches, e.g., the SLIM model and Halstead's software science, were initially very popular, but research interest has faded considerably since 1990.
- The diversity of estimation approaches is very high and increasing, especially in the period 2000-2004. This increase is illustrated by the increased proportion of "other approaches" (16 percent in 2000-2004). "Other approaches" include, for example, the use of lexical analysis of requirement specifications, genetic programming, linear programming, economic production models, soft computing, fuzzy logic modeling, and bootstrap-based analogy cost estimation.

Table 5 shows that expert judgment-based approaches are discussed in 15 percent of the papers. This rather contrasts with the fact that the performance of more formal estimation techniques has been somewhat erratic to date, has not documented higher accuracy than expert judgment, and that expert judgment-based estimation approaches are, by far, the most common used approaches by the software industry (see [14] for a review of papers on this issue). The relative lack of focus upon expert judgment-based approaches suggests, we believe, that most researchers either do not cooperate closely with the software industry or that they believe it is better to focus on replacement, rather than improvement, of judgment-based approaches currently employed in industry.

3.8 Research Approach and Study Context (RQ8)

Table 6 shows the distribution of papers applying different research approaches, per period, and in total.

Table 6 suggests that:

- There are few case studies that focus upon the actual effort estimation process published in the journals, i.e., in-depth studies of software effort estimation processes and outcomes. Overall, we identified only eight such estimation case studies. Even in the IS journals, which may have a stronger tradition for case studies, estimation case studies seem to be rare. There may be much to learn from well-described and well-understood real-life cases. It may, therefore, be unfortunate that so few estimation case studies are reported as journal papers.
- Most papers evaluate estimation methods by employing historical data; few evaluations are completed in real-life estimation situations. Of particular concern is that the already-small proportion of real-life evaluations of estimation methods seems to be decreasing.

In short, there seems to be a lack of in-depth studies on the actual use of estimation methods and real-life evaluations published as journal papers. We investigated the papers that proposed a new estimation method or evaluated an existing, i.e., those with topic = Em. We found that most of these papers (more than 60 percent) were of the research approach Hv, i.e., they evaluated an estimation method by applying historical data. This is an understandable choice of evaluation method, given the extra noise added with more realism. However, there is a problem when there are few or no groups that conduct research on the actual use and effect of an estimation method in real-life settings. We believe this is the case for software cost estimation methods. We analyzed all the papers with topic *Em* and research approach *Rl.* We could find no study that had an in-depth data collection and analysis of how estimation methods were actually applied! Even for well-known models, such as the COCOMO-model, this type of study was missing.

As mentioned earlier, our exclusion of conference paper excludes a high number of cost estimation experience

TABLE 7
Study Contexts

Study context	-1989	1990-1999	2000-2004	Total
St	1 (2%)	10 (6%)	11 (11%)	22 (7%)
Pr	24 (59%)	123 (76%)	70 (70%)	217 (71%)
Nr	16 (39%)	30 (18%)	22 (22%)	68 (22%)
The abbreviations used are: St = Students and/or student projects, Pr = Professionals and/or industrial software projects, Nr = Not relevant				

One paper may apply more than one study context.

papers (case studies) published at conferences. These conference papers provide valuable information, but are, as far as we have experienced, typically lower in scientific rigor. We believe, therefore, that the lack of journal papers on real-life use of estimation methods point at a potentially important shortcoming, i.e., the lack of high quality research studies with in-depth, real-life evaluation of the use and impact of estimation methods.

Table 7 shows the distribution of papers applying different study contexts per period and in total.

Table 7 suggests that the study context is usually professionals and/or professional projects. This increases the realism of the studies. However, this does not necessarily mean that problems with realism are absent. In particular, the dominant use of historical data instead of real-life evaluation results in a decrease in realism. The "Not relevant" category includes papers where there are no empirical studies, e.g., the development of an estimation method without empirical evaluation or analytical comparison of properties of estimation methods.

Evaluation of estimation methods based on historical data sets obviously requires that these data sets be representative for current or future projects; or, at least, that their analysis should be a useful means of evaluating the strengths and benefits of an estimation method. We analyzed the use of historical data sets in the journal papers and found that there are good reasons to claim that the *availability* of a data set is more indicative for its use than its representativeness or other properties. An additional weakness related to the use of the data sets, is that the data quality is frequently not known and/or discussed in the papers.

The *Boehm-Cocomo* data set [2], for example, was published in 1981 and is readily available. However, it is based on software development projects that may be quite different from the most recent development projects, e.g., the technology used might well be different. In spite of that, the data set has been used in as many as 12 journal papers to evaluate estimation methods, calibration of methods, and estimation model size measures since 1995.

If a data set is not representative, it may still be useful for evaluating properties of an estimation method relative to properties of data sets Unfortunately, we found that very few papers (including most of our own) paid any attention at all to properties of the data set. The evaluations typically pick one or more easily available data sets and leave it to the reader to decide the degree to which it is possible to generalize the results to other contexts and other sets of projects. This is hardly a research method that leads to a systematic aggregation of results.

4 Summary

This paper reviews software cost estimation papers published in journals and tries to support other software cost estimation researchers through a library of estimation papers classified according to research topic, estimation approach, research approach, study context, and use of data sets. Based on our, to some extent subjective, interpretation of the review results and other knowledge, we recommend the following changes in estimation research:

- **Increase the breadth of the search for relevant studies.** Software cost estimation research studies seem, on average, to be based on searches for relevant previous studies that use few sources. If the goal is to find relevant research on software cost estimation, it is *not* sufficient to conduct searches in digital libraries or to conduct manual searches of the most important software engineering journals. We identified as many as 76 journals with studies on software cost estimation, several software cost estimation papers that would have been hard to find through searches in digital libraries, and many journals from other domains that contained relevant results.
- **Search manually for relevant papers in a carefully selected set of journals when completeness is essential.** There is a lack of standardized use of terms pertaining to software cost estimation. We believe that such a lack makes it is easy to miss important papers when relying on automatic searches in digital libraries. The search term "software cost estimation" OR "software effort estimation," for example, did not identify more than 60 percent of the papers on software cost estimation identified by our manual search. We are aware of the practical limitations related to the use of manual search process, e.g., the required search effort. In cases where completeness is not essential, a combination of manual search of the most relevant journals and use of digital libraries to cover the other journals may be sufficient.
- **Conduct more studies on estimation methods commonly used by the software industry.** In spite of the fact that formal estimation models have existed for many years, the dominant estimation method is based on expert judgment. Further, available evidence does not suggest that the estimation accuracy improves with use of formal estimation models. Despite these factors, current research on, e.g., expert estimation is relatively sparse and we believe that it deserves more research effort.
- **Increase the awareness of how properties of the data sets impact the results when evaluating estimation methods.** Currently, a typical evaluation of an estimation method is based on an arbitrarily chosen data set, where the representativeness and other properties are not analyzed or much discussed. Even worse, quite a few papers are based on data sets that are clearly too old to be representative for more recent or future projects. We recommend that researchers on software cost estimation change their focus from the availability of project data sets to understanding the relationship between project characteristics (data set properties) and estimation methods.

We cannot claim, based on empirical evidence, that use of our library and adherence to our recommendations actually leads to better software cost estimation methods and fewer cost overruns in the software industry. However, adherence to the recommendations will, we believe, increase the probability that future cost estimation research builds on existing knowledge, is relevant for the software industry and is easier to generalize to other contexts. This in turn, we think, will increase the probability of better estimation methods and industry practice.

APPENDIX A
LIST OF INCLUDED PAPERS

(The numbers of papers identified in each paper are in (). The titles of journals with more than five papers appear in italic.)

ACM Transactions on Computer Personnel (1)
Ada User Journal (1)
Advances in Computers (3)
Advances in Information Systems (1)
American Programmer / Cutter IT Journal (6)—Rank 7
Annals of Software Engineering (3)
Applied Computing Review (1)
Australian Journal of Information Systems (3)
Automated Software Engineering (2)
Communications of the ACM (9)
Computers & Operations Research (2)
Computing and Control Engineering Journal (1)
Concurrent Engineering: Research and Applications (1)
Datamation (1)
Embedded Systems Programming (1)
Empirical Software Engineering (12)—Rank 4
Engineering Economist (1)
Engineering Intelligent Systems for Electrical Engineering and Communications (1)
European Journal of Information Systems (3)
Expert Systems (1)
Expert Systems with Applications (2)
GEC Journal of Research (1)
Human Factors (1)
IBM Systems Journal (1)
ICL Technical Journal (1)
IEE Proceedings Software (4)
IEE Proceedings Software Engineering (2)
IEEE Aerospace and Electronic Systems Magazine (1)
IEEE Computer (1)
IEEE Multimedia (1)
IEEE Software (11)—Rank 5
IEEE Transactions on Computers (1)
IEEE Transactions on Software Engineering (51)—Rank 1
IEEE Transactions on Systems, Man, and Cybernetics (3)
IIE Transactions (1)
Industrial Management & Data Systems (2)
Information and Management (6)—Rank 6
Information and Software Technology (47)—Rank 2
Information Resources Management Journal (1)
Information Strategy: The Executive's Journal (1)
Information Systems Journal (1)
Information Systems Management (1)
Information Systems Research (1)
Information Technology & Management (1)
International Journal of Project Management (4)
International Journal of Software Engineering and Knowledge Engineering (2)
International Journal of Systems Science (2)
Journal of Computer and Software Engineering (1)
Journal of Computer Information Systems (1)
Journal of Defense Software Engineering (1)
Journal of End User Computing (1)
Journal of Experimental and Theoretical Artificial Intelligence (1)
Journal of Information Technology (3)
Journal of Management Information Systems (3)
Journal of Parametrics (2)
Journal of Software Maintenance and Evolution: Research and Practice (5)
Journal of Systems and Software (42)—Rank 3
Journal of Systems Management (1)
Management Science (4)
MIS Quarterly (4)
New Review of Hypermedia and Multimedia (1)
Pakistan Journal of Information and Technology (1)
Programming and Computer Software (1)
R.F.-Design (1)
Scandinavian Journal of Information Systems (1)
SIGPLAN Notices (1)
Software—Practice and Experience (1)
Software Engineering Journal (4)
Software Engineering Notes (4)
Software Quality Journal (9)—Rank 6
Software World (1)
Technometrics (1)
Texas Instruments Technical Journal (1)
The Australian Computer Journal (1)
Transactions of the Information Processing Society of Japan (1)
Vitro Technical Journal (1)

APPENDIX B
CLASSIFICATION DESCRIPTIONS

Category: Research topic

1. Estimation methods: Studies of methods producing effort estimates. This includes formal estimation models, expert estimation processes, decomposition-based estimation processes, and processes for combining different estimation methods and sources. It does not include estimation methods with a focus on size.
2. Production functions: Studies of properties of the "software production function." This includes studies on linear versus nonlinear relationship between effort and size, and on the relationship between effort and schedule compression. Studies on production functions differ from studies on estimation methods in that they focus on selected properties of formal estimation models, not on complete estimation methods.
3. Calibration of models: Studies on calibration of estimation models, e.g., studies on local versus multiorganizational data or calibration of the COCOMO model to certain types of projects.

4. Size measures: Studies with a main focus on the validity and improvements of size measures that are important in estimation models, e.g., inter-rater validity of function points counting. Studies discussing size measures but with a main focus on developing effort estimation methods are not classified as Sm.
5. Organizational issues: Studies on estimation processes in a wide organizational context, e.g., surveys on estimation practice, reasons for cost overruns, impact of estimate on project work, and estimation in the general context of project management.
6. Effort uncertainty assessments: Studies on uncertainty of effort or size estimates, e.g., methods providing minimum-maximum intervals for effort.
7. Measures of estimation performance: Studies on evaluation and selection of estimation methods, e.g., studies on how to measure estimation accuracy or how to compare estimation methods.
8. Data set properties: Studies on how to analyze data sets for the purpose of estimation methods, e.g., data sets with missing data.
9. Other topics: Unclassified topics.

Category: Estimation approach

1. Regression: Studies on regression-based estimation methods, including most algorithmic models, e.g., studies on the COCOMO estimation model. Studies applying regression analysis for other purposes, e.g., to validate or test relationships, are *not* included.
2. Analogy: Studies on analogy- and case-based reasoning estimation methods, e.g., studies evaluating estimation by analogy on different data sets.
3. Expert judgment: Studies on expert judgment-based estimation methods. This includes structured processes for expert judgment, such as AHP and paired comparisons. Studies where expert judgment is input to a formal estimation method or where expert judgment based on output from other types of estimation approaches is *not* classified as Ej.
4. Work breakdown: Studies on work breakdown structure-based and other activity decomposition-based estimation methods.
5. Function Point: Studies on "function point"-based estimation methods, including studies on estimation methods based on feature points and use case points.
6. CART: Studies on estimation methods based on classification and regression trees.
7. Simulation: Studies based on simulation-based/derived models, e.g., system dynamics studies applying the Monte Carlo simulation.
8. Neural network: Studies on artificial neural network-based estimation methods.
9. Theory: Studies on theory-derived estimation models, e.g., the SLIM estimation model and estimation models based on "software science."
10. Bayesian: Bayesian or Markow-based estimation models.
11. Combination of estimates: Studies on the combination of estimates from difference sources, e.g., the combination of expert and formal model-based estimates.
12. Other: Estimation methods based on other techniques for estimation modeling, e.g., genetic programming, to derive estimation models.
13. Not relevant: Studies on estimation methods in general and studies where the approach to estimation is not relevant.

Category: Research approach

1. Theory: Nonempirical research approaches or theoretical evaluation of properties of estimation models. Most studies apply theories, but only studies that rely heavily on nonempirical research methods in their evaluation and development of estimation approach are included here.
2. Survey: Survey-based studies, e.g., questionnaire- and interview-based surveys of industry practice.
3. Experiment: Experiment-based studies.
4. Case study: Case-based studies, e.g. in-depth study of the estimation processes of one, or a very small number, of software projects.
5. Development of estimation method: Studies where new effort (or size) estimation models, processes, or tools are developed. This includes new methods for combining estimates.
6. History-based evaluation: Studies evaluating estimation methods or other estimation-relevant relationships on previously completed software projects.
7. Own experience/lessons learned: Studies where the *only* reference is one's own experience, without any scientific documentation of the experience as a description of case studies, observations, experiments, etc.
8. Real-life evaluation: Studies evaluating estimation methods/expert estimation in real estimation situations.
9. Review: Studies that review other estimation papers/estimation methods. Studies where a review of other papers is not the main purpose of the paper but is included mainly as a discussion of related work or as input to theory or is included in discussions of results are not classified as reviews.
10. Simulation: Simulation-based studies. This category relates to the research method, e.g., the evaluation method, not to the estimation method itself.
11. Other: Studies with other research approaches.

Category: Study context

1. Students: Studies where the subjects are students and/or student projects.
2. Professionals: Studies where the subjects are software professionals and/or industrial software projects.
3. Not relevant: Studies where the study context is not relevant.

Category: Data set

1. Own data, not included: Own data set applied to develop and/or evaluate estimation methods. The data set is not included in the paper.
2. Own data, included: Own data set, but the data set is not included in the paper.
3. Xx: Data sets borrowed from other studies, e.g., the data set Bc refers to the Barry Boehm, COCOMO 81 data set. (An updated description of the list of data set abbreviations are provided at

www.simula.no/BESTweb or upon request to magnej@simula.no.)

4. No data set: No data set applied.

Appendix C

Journal Papers on Software Cost Estimation

(Search completed April 2004. For a more updated list that includes more recently published journal papers, see www.simula.no/BESTweb. Each reference is followed by the categorization suggested by the authors of this paper. Category 1-x is related to research topic, category 2-x to estimation approach, category 3-x to research approach, category 4-x to study context, and, category 5-x to data set. Please report possible misclassifications and missing papers to magnej@simula.no and they will be updated in the Web library BESTweb. An explanation of the categories is provided in Appendix B. An (S) in the reference shows that the paper was used to assess researcher awareness of relevant journals in Section 3.2.)

1. Abdel-Hamid, T., Investigating the Cost/Schedule Trade-Off in Software Development. IEEE Software, 1990. 7(1): p. 97-105. 1-Pf, 2-Si, 3-Si, 4-Pr, 5-On.
2. Abdel-Hamid, T.K., Adapting, Correcting, and Perfecting Software Estimates: A Maintenance Metaphor. IEEE Computer, 1993. 26(3): p. 20-29. 1-OiEm, 2-RgSi, 3-DmSi, 4-Nr, 5-No. (S)
3. Abdel-Hamid, T.K. and S.E. Madnik, The Dynamics of Software Project Scheduling. Communications of the ACM, 1983. 26(5): p. 340-346. 1-Oi, 2-Si, 3-Si, 4-Nr, 5-No.
4. Abdel-Hamid, T.K. and S.E. Madnik, Impact of Schedule Estimation on Software Project Behavior. IEEE Software, 1986. 3(4): p. 70-75. 1-Oi, 2-Si, 3-Si, 4-Nr, 5-On.
5. Abdel-Hamid, T.K., K. Sengupta, and D. Ronan, Software Project Control: An Experimental Investigation of Judgment with Fallible Information. IEEE Transactions on Software Engineering, 1993. 19(6): p. 603-612. 1-Oi, 2-Si, 3-SiEx, 4-St, 5-On.
6. Abdel-Hamid, T.K., K. Sengupta, and C. Swett, The Impact of Goals on Software Project Management: An Experimental Investigation. MIS Quarterly, 1999. 23(4): p. 531-555. 1-Oi, 2-Ej, 3-SiEx, 4-St, 5-On.
7. Abran, A., et al., Adapting Function Points to Real-Time Software. American Programmer, 1997. 10(11): p. 32-43. 1-Sm, 2-Fp, 3-DmRl, 4-Pr, 5-On.
8. Abran, A. and P.N. Robillard, Function Points: A Study of Their Measurement Processes and Scale Transformations. Journal of Systems and Software, 1994. 25: p. 171-184. 1-Sm, 2-Fp, 3-Th, 4-Nr, 5-No.
9. Abran, A. and P.N. Robillard, Function Points Analysis: An Empirical Study of Its Measurement Processes. IEEE Transactions on Software Engineering, 1996. 22(12): p. 895-910. 1-Sm, 2-RgFp, 3-Hv, 4-Pr, 5-Oi.
10. Abran, A., I. Silva, and L. Primera, Field Studies Using Functional Size Measurement in Building Estimation Models for Software Maintenance. Journal of Software Maintenance and Evolution: Research and Practice, 2002. 14(1): p. 31-64. 1-Em, 2-RgFp, 3-DmHv, 4-Pr, 5-Oi.
11. Adrangi, B. and W. Harrison, Effort Estimation in a System Development Project. Journal of Systems Management, 1987. 36(8): p. 21-23. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-On.
12. Aguilar-Ruiz, J.S., et al., An Evolutionary Approach to Estimating Software Development Projects. Information and Software Technology, 2001. 43(14): p. 875-882. 1-Em, 2-CtSiOt, 3-DmHv, 4-Pr, 5-On.
13. Ahn, Y., et al., The Software Maintenance Project Effort Estimation Model Based on Function Points. Journal of Software Maintenance and Evolution: Research and Practice, 2003. 15(2): p. 71-85. 1-EmSm, 2-FpRg, 3-DmHv, 4-Pr, 5-Oi.
14. Albrecht, A.J. and J.E. Gaffney Jr, Software Function, Source Lines of Code, and Development Effort Prediction: A Software Science Validation. IEEE Transactions on Software Engineering, 1983. 9(6): p. 639-648. 1-Em, 2-FpTh, 3-Hv, 4-Pr, 5-Oi.
15. Angelis, L. and I. Stamelos, A Simulation Tool for Efficient Analogy Based Cost Estimation. Empirical Software Engineering, 2000. 5(1): p. 35-68. 1-EmUn, 2-RgAnSi, 3-DmHv, 4-Pr, 5-AgAb.
16. Antoniol, G., R. Fiutem, and C. Lokan, Object-Oriented Function Points: An Empirical Validation. Empirical Software Engineering, 2003. 8(3): p. 225-254. 1-EmSm, 2-FpRg, 3-DmHv, 4-Pr, 5-Oi.
17. Antoniol, G., et al., A Function Point-Like Measure for Object-Oriented Software. Empirical Software Engineering, 1999. 4(3): p. 263-287. 1-EmSm, 2-FpRg, 3-DmHv, 4-Pr, 5-Oi.
18. Arifoglu, A., A Methodology for Software Cost Estimation. Software Engineering Notes, 1993. 18(2): p. 96-105. 1-Em, 2-ThFpRg, 3-Re, 4-Nr, 5-No.
19. Balcom, O., Estimating Program Complexity. Embedded Systems Programming, 1998. 11(3): p. 96-100. 1-Em, 2-Fp, 3-Ox, 4-Pr, 5-On.
20. Banker, R.D., H. Chang, and C.F. Kemerer, Evidence on Economies of Scale in Software Development. Information and Software Technology, 1994. 36(5): p. 275-282. 1-Pf, 2-Rg, 3-Hv, 4-Pr, 5-BeBcYoBaWiAgBhKtMe.
21. Banker, R.D., et al., Software Complexity and Maintenance Costs. Communications of the ACM, 1993. 36(11): p. 81-94. 1-Em, 2-Rg, 3-Hv, 4-Pr, 5-On.
22. Banker, R.D., et al., Automating Output Size and Reuse Metrics in a Repository-Based Computer-Aided Software Engineering (CASE) Environment. IEEE Transactions on Software Engineering, 1994. 20(3): p. 169-187. 1-Sm, 2-Fp, 3-Hv, 4-Pr, 5-On.
23. Banker, R.D. and S.A. Slaughter, A Field Study of Scale Economies in Software Maintenance. Management Science, 1997. 43(12): p. 1709-1725. 1-Pf, 2-Rg, 3-Hv, 4-Pr, 5-On.
24. Barki, H., S. Rivard, and J. Talbot, An Integrative Contingency Model of Software Project Risk Management. Journal of Management Information Systems, 2001. 17(4): p. 37-69. 1-Un, 2-Nr, 3-Sv, 4-Pr, 5-No. (S)
25. Barry, E.J., T. Mukhopadhyay, and S.A. Slaughter, Software Project Duration and Effort: An Empirical Study. Information Technology & Management, 2002. 3(1-2): p. 113-136. 1-EmOi, 2-Rg, 3-Dm, 4-Pr, 5-On.

26. Basili, V.R. and J. Beane, Can the Parr Curve Help with Manpower Distribution and Resource Estimation Problems? Journal of Systems and Software, 1981. 2(1): p. 59-69. 1-Pf, 2-Th, 3-Hv, 4-Pr, 5-Oi. (S)
27. Basili, V.R. and K. Freburger, Programming Measurement and Estimation in the Software Engineering Laboratory. Journal of Systems and Software, 1981. 2(1): p. 47-57. 1-PfEm, 2-Rg, 3-Hv, 4-Pr, 5-On.
28. Behrens, C.A., Measuring the Productivity of Computer Systems Development Activities with Function Points. IEEE Transactions on Software Engineering, 1983. 9(6): p. 648-652. 1-PfEm, 2-RgFp, 3-HvDm, 4-Pr, 5-On.
29. Benediktsson, O. and D. Dalcher, Effort Estimation in Incremental Software Development. IEE Proceedings Software Engineering, 2003. 150(6): p. 351-357. 1-Em, 2-Rg, 3-Th, 4-Nr, 5-No.
30. Benediktsson, O., D. Dalcher, and K. Reed, COCOMO-Based Effort Estimation for Iterative and Incremental Software Development. Software Quality Journal., 2003. 11(4): p. 265-281. 1-Em, 2-Rg, 3-Th, 4-Nr, 5-No.
31. Benyahia, H., Costs and Productivity Estimation in Computer Engineering Economics. Engineering Economist, 1996. 41(3): p. 229-241. 1-Em, 2-Fp, 3-Hv, 4-Pr, 5-On.
32. Bergeron, F. and J.Y. St-Arnaud, Estimation of Information Systems Development Efforts: A Pilot Study. Information and Management, 1992. 22(4): p. 239-254. 1-EmOi, 2-RgAnEj, 3-SvRl, 4-Pr, 5-On.
33. Betteridge, R., Successful Experience of Using Function Points to Estimate Project Costs Early in the Life-Cycle. Information and Software Technology, 1992. 34(10): p. 655-658. 1-Sm, 2-Fp, 3-Rl, 4-Pr, 5-On.
34. Bielak, J., Improving Size Estimates Using Historical Data. IEEE Software, 2000. 17(6): p. 27-35. 1-Sm, 2-Rg, 3-DmHv, 4-Pr, 5-On. (S)
35. Boehm, B., C. Abts, and S. Chulani, Software Development Cost Estimation Approaches - A Survey. Annals of Software Engineering, 2000. 10: p. 177-205. 1-Em, 2-Rganejwbfpnnthby, 3-Re, 4-Nr, 5-No.
36. Boehm, B., et al., The COCOMO 2.0 Software Cost Estimation Model: A Status Report. American Programmer, 1996. 9(7): p. 2-17. 1-Em, 2-Rg, 3-Dm, 4-Nr, 5-No.
37. Boehm, B. and K. Sullivan, Software Economics: Status and Prospects. Information and Software Technology, 1999. 41(14): p. 937-946. 1-Em, 2-EjRgCbNnOt, 3-Re, 4-Nr, 5-No. (S)
38. Boehm, B.W., Software Engineering Economics. IEEE Transactions on Software Engineering, 1984. 10(1): p. 4-21. 1-Em, 2-Ejrganfpthot, 3-Re, 4-Nr, 5-No.
39. Boehm, B.W., et al., Cost Models for Future Software Life Cycle Processes: COCOMO 2.0. Annals of Software Engineering, 1995. 1: p. 57-94. 1-EmSm, 2-FpRg, 3-Dm, 4-Nr, 5-No.
40. Boehm, B.W. and P.N. Papaccio, Understanding and Controlling Software Costs. IEEE Transactions on Software Engineering, 1988. 14(10): p. 1462-1477. 1-Oi, 2-Wbot, 3-Re, 4-Nr, 5-No.
41. Boehm, B.W. and R.W. Wolverton, Software Cost Modeling: Some Lessons Learned. Journal of Systems and Software, 1980. 1: p. 195-201. 1-Ep, 2-Nr, 3-Ox, 4-Nr, 5-No.
42. Bollinger, T.B. and S.L. Pfleeger, Economics of Reuse: Issues and Alternatives. Information and Software Technology, 1990. 32(10): p. 643-652. 1-Em, 2-Th, 3-ThDm, 4-Nr, 5-No.
43. Bowden, P., M. Hargreaves, and C.S. Langensiepen, Estimation Support by Lexical Analysis of Requirements Documents. Journal of Systems and Software, 2000. 51(2): p. 87-98. 1-Em, 2-Ot, 3-Ex, 4-St, 5-On.
44. Bradley, M. and R. Dawson, Whole Life Cost: The Future Trend in Software Development. Software Quality Journal, 1999. 8(2): p. 121-131. 1-Em, 2-Nr, 3-Re, 4-Nr, 5-No.
45. Briand, L.C., V.R. Basili, and W.M. Thomas, A Pattern Recognition Approach for Software Engineering Data Analysis. IEEE Transactions on Software Engineering, 1992. 18(11): p. 931-942. 1-Em, 2-CtRg, 3-DmHv, 4-Pr, 5-BcKe.
46. Briand, L.C. and J. Wüst, Modeling Development Effort in Object-Oriented Systems Using Design Properties. IEEE Transactions on Software Engineering, 2001. 27(11): p. 963-986. 1-Em, 2-RgCt, 3-DmHv, 4-Pr, 5-On. (S)
47. Burgess, C.J. and M. Lefley, Can Genetic Programming Improve Software Effort Estimation? A Comparative Evaluation. Information and Software Technology, 2001. 43(14): p. 863-873. 1-Em, 2-AnRgNnOt, 3-DmHv, 4-Pr, 5-De.
48. Card, D.N. and W.W. Agresti, Resolving the Software Science Anomaly. Journal of Systems and Software, 1987. 7(1): p. 29-35. 1-Em, 2-Th, 3-Hv, 4-Pr, 5-HaOn.
49. Carmel, E., Time-to-Completion Factors in Packaged Software Development. Information and Software Technology, 1995. 37(9): p. 515-520. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-On.
50. Carson, C., Using the TQC Problem-Solving Process to Develop an Improved Estimation Technique. Texas Instruments Technical Journal, 1996. 13(6): p. 101-106. 1-Oi, 2-Nr, 3-Ca, 4-Pr, 5-On.
51. Chan, T., S.L. Chung, and T.H. Ho, An Economic Model to Estimate Software Rewriting and Replacement Times. IEEE Transactions on Software Engineering, 1996. 22(8): p. 580-598. 1-Oi, 2-Th, 3-Thsi, 4-Pr, 5-On.
52. Chang, C.K., M.J. Christensen, and Z. Tao, Genetic Algorithms for Project Management. Annals of Software Engineering, 2001. 11: p. 107-139. 1-Ot, 2-Ot, 3-Dm, 4-Nr, 5-No.
53. Chatzoglou, P.D. and L.A. Macaulay, A Review of Existing Models for Project Planning and Estimation and the Need for a New Approach. International Journal of Project Management, 1996. 14(3): p. 173-183. 1-Em, 2-RgOt, 3-DmHv, 4-Pr, 5-On.
54. Chatzoglou, P.D. and L.A. Macaulay, A Rule-Based Approach to Developing Software Development Prediction Models. Automated Software Engineering, 1998. 5(2): p. 211-243. 1-Em, 2-Ct, 3-DmHv, 4-Pr, 5-On.
55. Chrysler, E., Some Basic Determinants of Computer Programming Productivity. Communications of the ACM, 1978. 21(6): p. 472-483. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-On.

56. Chulani, S., B. Boehm, and B. Steece, Bayesian Analysis of Empirical Software Engineering Cost Models. IEEE Transactions on Software Engineering, 1999. 25(4): p. 573-583. 1-EmCm, 2-RgEjBy, 3-DmHv, 4-Pr, 5-On.

57. Cockcroft, S., Estimating CASE Development Size from Outline Specifications. Information and Software Technology, 1996. 38(6): p. 391-399. 1-Sm, 2-Rg, 3-DmHv, 4-Pr, 5-On.

58. Cole, A., Runaway Projects: Cause and Effects. Software World, 1995. 26(3): p. 3-5. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-On. (S)

59. Connolly, T. and D. Dean, Decomposed versus Holistic Estimates of Effort Required for Software Writing Tasks. Management Science, 1997. 43(7): p. 1029-1045. 1-EmUn, 2-Ej, 3-Ex, 4-St, 5-On.

60. Constantine, G. and W.J. Vitaliano, Structured Estimating—A New Approach to a Critical Project Parameter. R.F.-Design, 2001. 24(5): p. 60-68. 1-Em, 2-Rg, 3-DmOx, 4-Pr, 5-No.

61. Conte, S.D., H.E. Dunsmore, and V.Y. Shen, Software Effort Estimation and Productivity. Advances in Computers, 1985. 24: p. 1-60. 1-EpEmSm, 2-RgTh, 3-Re, 4-Nr, 5-No.

62. Costagliola, G., et al., A Metric for the Size Estimation of Object-Oriented Graphical User Interfaces. International Journal of Software Engineering and Knowledge Engineering, 2000. 10(5): p. 581-603. 1-Sm, 2-Rg, 3-DmHv, 4-Pr, 5-On.

63. Cuelenaere, A.M.E., M.J.I.M. Genuchten, and F.J. Heemstra, Calibrating a Software Cost Estimation Model: Why and How. Information and Software Technology, 1987. 29(10): p. 558-567. 1-Cm, 2-Th, 3-Th, 4-Nr, 5-No.

64. Czuchra, W., Optimizing Budget Spendings for Software Implementation and Testing. Computers & Operations Research, 1999. 26(7): p. 731-747. 1-Em, 2-Ot, 3-Dmth, 4-Nr, 5-No.

65. Day, R., Software Project Estimation: Experiences from a Large Scale Object-Oriented Ada Project. Ada User Journal, 1999. 20(2): p. 154-161. 1-Em, 2-Ot, 3-Ox, 4-Pr, 5-On.

66. De Souza Lima Júnior, O., P.P.M. Farias, and A.D. Belchior, Fuzzy Modeling for Function Points Analysis. Software Quality Journal., 2003. 11(2): p. 149-166. 1-Sm, 2-Fp, 3-DmHv, 4-Pr, 5-On.

67. Dolado, J.J., A Study of the Relationships among Albrecht and Mark II Function Points, Lines of Code 4GL and Effort. Journal of Systems and Software, 1997. 37(2): p. 161-172. 1-Sm, 2-Fp, 3-Hv, 4-St, 5-Oi.

68. Dolado, J.J., A Validation of the Component-Based Method for Software Size Estimation. IEEE Transactions on Software Engineering, 2000. 26(10): p. 1006-1021. 1-Sm, 2-Rgfpnnot, 3-DmHv, 4-Pr, 5-On.

69. Dolado, J.J., On the Problem of the Software Cost Function. Information and Software Technology, 2001. 43(1): p. 61-72. 1-Pf, 2-RgOt, 3-Hv, 4-Pr, 5-AgAbBcBaBeDhDaHeKeKtMsSh. (S)

70. Dutta, S., L.N.V. Wassenhove, and S. Kulandaiswamy, Benchmarking European Software Management Practices. Communications of the ACM, 1998. 41(6): p. 77-86. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-On.

71. Ebrahimi, N.B., How to Improve the Calibration of Cost Models. IEEE Transactions on Software Engineering, 1999. 25(1): p. 136-140. 1-EmUn, 2-Rg, 3-ThDmHv, 4-Pr, 5-La.

72. Edwards, J.S. and T.T. Moores, A Conflict between the Use of Estimating and Planning Tools in the Management of Information Systems. European Journal of Information Systems, 1994. 3(2): p. 139-147. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-On.

73. Evanco, W.M., Modeling the Effort to Correct Faults. Journal of Systems and Software, 1995. 29(1): p. 75 - 84. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-On. (S)

74. Ferens, D. and D. Christensen, Calibrating Software Cost Models to Department of Defense Databases: A Review of Ten Studies. Journal of Parametrics, 1999. XIV(1): p. 33-51. 1-Cm, 2-Rg, 3-Hv, 4-Pr, 5-On.

75. Ferens, D.V., Software Support Cost Models: Quo Vadis? Journal of Parametrics, 1984. 4(4): p. 64-99. 1-Em, 2-Rgth, 3-Rehv, 4-Pr, 5-On.

76. Ferens, D.V., Software Cost Estimation in the DoD Environment. American Programmer, 1996. 9(7): p. 28-34. 1-Em, 2-Rgth, 3-Re, 4-Nr, 5-No.

77. Ferens, D.V., The Conundrum of Software Estimation Models. IEEE Aerospace and Electronic Systems Magazine, 1999. 14(3): p. 23-29. 1-Cm, 2-Rg, 3-Hv, 4-Pr, 5-On. (S)

78. Finnie, G.R., G.E. Wittig, and J.-M. Desharnais, A Comparison of Software Effort Estimation Techniques: Using Function Points with Neural Networks, Case-Based Reasoning and Regression Models. Journal of Systems and Software, 1997. 39(3): p. 281-289. 1-Em, 2-AnRgNn, 3-Hv, 4-Pr, 5-Dh.

79. Finnie, G.R., G.E. Wittig, and J.M. Desharnais, Reassessing Function Points. Australian Journal of Information Systems, 1997. 4(2): p. 39-45. 1-Sm, 2-Fp, 3-Hv, 4-Pr, 5-Dh.

80. Fleck, R.A., Jr., Managing Programmer Resources in a Maintenance Environment with Function Points. Industrial Management + Data Systems, 1998. 98(2): p. 63-70. 1-Em, 2-Fp, 3-Hv, 4-Pr, 5-On.

81. Foss, T., et al., A Simulation Study of the Model Evaluation Criterion MMRE. IEEE Transactions on Software Engineering, 2003. 29(11): p. 985-995. 1-Ep, 2-Nr, 3-Si, 4-Pr, 5-Dh.

82. Francalanci, C., Predicting the Implementation Effort of ERP Projects: Empirical Evidence on SAP/R3. Journal of Information Technology, 2001. 16(1): p. 33-48. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-On.

83. Genuchten, M.V. and H. Koolen, on the Use of Software Cost Models. Information and Management, 1991. 21: p. 37-44. 1-EmCm, 2-CbRgEj, 3-Ex, 4-Pr, 5-On.

84. Gilb, T., Estimating Software Attributes: Some Unconventional Points of View. Software Engineering Notes, 1986. 11(1): p. 49-59. 1-Oi, 2-Nr, 3-Ox, 4-Nr, 5-No.

85. Gonzales, M. and S. Paravastu, Correction on the Software Science Length Estimator Skewness for 'C' Language Programs. Empirical Software Engineering, 2000. 5(2): p. 155-159. 1-Sm, 2-Th, 3-Hv, 4-Pr, 5-On.

86. Goodman, P.A., Application of Cost-Estimation Techniques: Industrial Perspective. Information and Software Technology, 1992. 34(6): p. 379-382. 1-Em, 2-EjAn, 3-CaDm, 4-Pr, 5-On.

87. Granju-Alvarez, J.C. and M.J. Barranco-Garcia, A Method for Estimating Maintenance Cost in a Software Project: A Case Study. Journal of Software Maintenance and Evolution: Research and Practice, 1997. 9(3): p. 161-175. 1-Em, 2-Rg, 3-Ca, 4-Pr, 5-On.

88. Graves, T.L. and A. Mockus, Identifying Productivity Drivers by Modeling Work Units Using Partial Data. Technometrics, 2001. 43(2): p. 168-179. 1-Em, 2-Rg, 3-DmHvSi, 4-Pr, 5-No.

89. Gray, A. and S. MacDonell, Software Metrics Data Analysis-Exploring the Relative Performance of Some Commonly Used Modeling Techniques. Empirical Software Engineering, 1999. 4: p. 297-316. 1-EmEp, 2-RgNn, 3-Hv, 4-Pr, 5-MzLiDa. (S)

90. Gray, A.R. and S.G. MacDonell, A Comparison of Techniques for Developing Predictive Models of Software Metrics. Information and Software Technology, 1997. 39(6): p. 425-437. 1-Em, 2-RgNnAnCt-Ot, 3-Re, 4-Nr, 5-No.

91. Gulezian, R., Reformulating and Calibrating COCOMO. Journal of Systems and Software, 1991. 16(3): p. 235-242. 1-Cm, 2-Rg, 3-Th, 4-Nr, 5-No. (S)

92. Hakuta, M., F. Tone, and M. Ohminami, A Software Size Estimation Model and Its Evaluation. Journal of Systems and Software, 1997. 37(3): p. 253-263. 1-SmUn, 2-Fp, 3-DmHv, 4-Pr, 5-Oi.

93. Hall, B., G. Orr, and T.E. Reeves, A Technique for Function Block Counting. Journal of Systems and Software, 2001. 57(3): p. 217-220. 1-Sm, 2-FpRg, 3-DmHv, 4-Pr, 5-On.

94. Halstead, M.H., Advances in Software Science. Advances in Computers, 1979. 18: p. 119-172. 1-Em, 2-Th, 3-ThDm, 4-Nr, 5-No.

95. Harrison, W. and B. Adrangi, The Role of Programming Language in Estimating Software Development Costs. Journal of Management Information Systems, 1987. 3(3): p. 101-110. 1-EmPf, 2-Rg, 3-HvDm, 4-Pr, 5-On.

96. Hastings, T.E. and A.S.M. Sajeev, A Vector-Based Approach to Software Size Measurement and Effort Estimation. IEEE Transactions on Software Engineering, 2001. 27(4): p. 337-350. 1-SmEm, 2-RgFp, 3-DmHv, 4-Pr, 5-Oi.

97. Haynes, P. and B. Henderson-Sellers, Cost Estimation of OO Projects: Empirical Observations, Practical Applications. American Programmer, 1996. 9(7): p. 35-41. 1-Em, 2-Fp, 3-Ox, 4-Pr, 5-On.

98. Heemstra, F.J., Software Cost Estimation. Information and Software Technology, 1992. 34(10): p. 627-639. 1-Oi, 2-EjAnRgFp, 3-Re, 4-Nr, 5-No. (S)

99. Heemstra, F.J. and R.J. Kusters, Function Point Analysis: Evaluation of a Software Cost Estimation Model. European Journal of Information Systems, 1991. 1(4): p. 223-237. 1-SmEm, 2-Fp, 3-SvHv, 4-Pr, 5-On.

100. Heiat, A., Comparison of Artificial Neural Network and Regression Models for Estimating Software Development Effort. Information and Software Technology, 2002. 44(15): p. 911-922. 1-Em, 2-RgNn, 3-DmHv, 4-Pr, 5-KeOi.

101. Heiat, A. and N. Heiat, A Model for Estimating Efforts Required for Developing Small-Scale Business Applications. Journal of Systems and Software, 1997. 39(1): p. 7-14. 1-Em, 2-Fp, 3-DmHv, 4-Pr, 5-Oi.

102. Helander, M.E., Z. Ming, and N. Ohlsson, Planning Models for Software Reliability and Cost. IEEE Transactions on Software Engineering, 1998. 24(6): p. 420-434. 1-Em, 2-Wb, 3-DmHv, 4-Pr, 5-On.

103. Hill, J., L.C. Thomas, and D.E. Allen, Experts' Estimates of Task Durations in Software Development Projects. International Journal of Project Management, 2000. 18(1): p. 13-21. 1-Oi, 2-EjWb, 3-Sv, 4-Pr, 5-On.

104. Horgan, G., S. Khaddaj, and P. Forte, Construction of an FPA-Type Metric for Early Lifecycle Estimation. Information and Software Technology, 1998. 40(8): p. 409-415. 1-Em, 2-Fp, 3-DmHv, 4-Pr, 5-AgAb.

105. Hu, Q., Evaluating Alternative Software Production Functions. IEEE Transactions on Software Engineering, 1997. 23(6): p. 379-387. 1-Pf, 2-Rg, 3-Hv, 4-Pr, 5-AgBcBaBeKeKtWiYo.

106. Hu, Q., R.T. Plant, and D.B. Hertz, Software Cost Estimation Using Economic Production Models. Journal of Management Information Systems, 1998. 15(1): p. 143-163. 1-EmPf, 2-RgThOt, 3-DmHv, 4-Pr, 5-Ke. (S)

107. Huang, S.J., Early Project Estimation in the Formal Communication Protocol Development. Information and Management, 2001. 38(7): p. 449-458. 1-Em, 2-Rg, 3-Ca, 4-Pr, 5-On.

108. Hughes, R.T., Expert Judgement as an Estimating Method. Information and Software Technology, 1996. 38(2): p. 67-75. 1-Em, 2-Ej, 3-ReSv, 4-Pr, 5-On.

109. Hughes, R.T., A. Cunliffe, and F. Young-Martos, Evaluating Software Development Effort Model-Building Techniques for Application in a Real-Time Telecommunications Environment. IEE Proceedings Software, 1998. 145(1): p. 29-33. 1-EmEp, 2-RgNn-AnOt, 3-Hv, 4-Pr, 5-On.

110. Humphrey, W.S., The PSP and Personal Project Estimating. American Programmer, 1996. 9(6): p. 2-15. 1-Em, 2-RgEj, 3-DmRl, 4-Pr, 5-On.

111. Humphrey, W.S. and N.D. Singpuwally, Predicting (Individual) Software Productivity. IEEE Transactions on Software Engineering, 1991. 17(2): p. 196-207. 1-EmUn, 2-Rg, 3-HvDm, 4-Pr, 5-On.

112. Höst, M. and C. Wohlin, A Subjective Effort Estimation Experiment. Information and Software Technology, 1997. 39(11): p. 755-762. 1-EmUn, 2-Ej, 3-Rl, 4-St, 5-On.

113. Idri, A., T.M. Khoshgoftaar, and A. Abran, Investigating Soft Computing in Case-Based Reasoning for Software Cost Estimation. Engineering Intelligent Systems for Electrical Engineering and Communications, 2002. 10(3): p. 147-157. 1-EmUn, 2-AnOt, 3-DmHv, 4-Pr, 5-Bc.

114. Jeffery, D.R., Time-Sensitive Cost Models in the Commercial MIS Environment. IEEE Transactions on Software Engineering, 1987. 13(7): p. 852-859. 1-PfOi, 2-RgTh, 3-HvDm, 4-Pr, 5-On.

115. Jeffery, D.R. and G. Low, Calibrating Estimation Tools for Software Development. Software Engineering Journal, 1990. 5(4): p. 215-221. 1-CmEm, 2-Rg, 3-Hv, 4-Pr, 5-On.

116. Jeffery, D.R., G.C. Low, and M. Barnes, A Comparison of Function Point Counting Techniques. IEEE Transactions on Software Engineering, 1993. 19(5): p. 529-532. 1-Sm, 2-Fp, 3-Hv, 4-Pr, 5-On.

117. Jeffery, D.R., M. Ruhe, and I. Wieczorek, A Comparative Study of Two Software Development Cost Modeling Techniques Using Multi-Organizational and Company-Specific Data. Information and Software Technology, 2000. 42(14): p. 1009-1016. 1-EmCm, 2-AnRg, 3-Hv, 4-Pr, 5-IsOn.

118. Jeffery, R. and L. Graham, Function Points and Their Use. The Australian Computer Journal, 1997. 29(4): p. 148-156. 1-SmEm, 2-Fp, 3-DmHv, 4-Pr, 5-On. (S)

119. Jeffery, R. and J. Stathis, Function Point Sizing: Structure, Validity and Applicability. Empirical Software Engineering, 1996. 1(1): p. 11-30. 1-EmSm, 2-Fp, 3-DmHv, 4-Pr, 5-AgKeKkOi.

120. Jenkins, A.M., J.D. Naumann, and J.C. Wetherbe, Empirical Investigation of Systems Development Practices and Results. Information and Management, 1984. 7(2): p. 73-82. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-On.

121. Jenson, R.L. and J.W. Bartley, Parametric Estimation of Programming Effort: An Object-Oriented Model. Journal of Systems and Software, 1991. 15(2): p. 107-114. 1-PfEm, 2-Rg, 3-HvDm, 4-Pr, 5-On.

122. Johnson, P.M., et al., Empirically Guided Software Effort Guesstimation. IEEE Software, 2000. 17(6): p. 51-56. 1-Em, 2-EjRg, 3-DmRl, 4-St, 5-On.

123. Jongmoon, B., B. Boehm, and B.M. Steece, Disaggregating and Calibrating the CASE Tool Variable In COCOMO II. IEEE Transactions on Software Engineering, 2002. 28(11): p. 1009-1022. 1-CmEm, 2-RgBy, 3-DmHv, 4-Pr, 5-Co.

124. Jun, E.S. and J.K. Lee, Quasi-Optimal Case-Selective Neural Network Model for Software Effort Estimation. Expert Systems With Applications, 2001. 21(1): p. 1-14. 1-Em, 2-RgNn, 3-Hv, 4-Pr, 5-On.

125. Jung, H.-W., Y.-W. Ahn, and G.-J. Kim, Optimal Schedule and Effort Considering the Penalty Cost of Schedule Delay in Software Development. International Journal of Systems Science, 2000. 31(1): p. 91-95. 1-Pf, 2-Nr, 3-Th, 4-Nr, 5-No. (S)

126. Jørgensen, M., Experience with the Accuracy of Software Maintenance Task Effort Prediction Models. IEEE Transactions on Software Engineering, 1995. 21(8): p. 674-681. 1-Em, 2-RgAnCtNn, 3-DmHv, 4-Pr, 5-On.

127. Jørgensen, M., How Much Does a Vacation Cost? Or What Is a Software Cost Estimate? Software Engineering Notes, 2003. 28(6): p. 5-5. 1-OiEp, 2-Nr, 3-Ox, 4-Nr, 5-No.

128. Jørgensen, M., Realism in Assessment of Effort Estimation Uncertainty: It Matters How You Ask. IEEE Transactions on Software Engineering, 2004. 30(4): p. 209-217. 1-Un, 2-Ej, 3-DmRl, 4-Pr, 5-On.

129. Jørgensen, M., A Review of Studies on Expert Estimation of Software Development Effort. Journal of Systems and Software, 2004. 70(1-2): p. 37-60. 1-Em, 2-Ej, 3-Re, 4-Nr, 5-No.

130. Jørgensen, M., Top-Down and Bottom-Up Expert Estimation of Software Development Effort. Information and Software Technology, 2004. 46(1): p. 3-16. 1-Em, 2-EjWb, 3-Ex, 4-Pr, 5-Oi. (S)

131. Jørgensen, M., U. Indahl, and D.I.K. Sjøberg, Software Effort Estimation by Analogy and "Regression toward the Mean". Journal of Systems and Software, 2003. 68(3): p. 253-262. 1-Em, 2-AnEj, 3-ExDmHv, 4-PrSt, 5-JeOiKm.

132. Jørgensen, M. and D.I.K. Sjøberg, Impact of Effort Estimates on Software Project Work. Information and Software Technology, 2001. 43(15): p. 939-948. 1-Oi, 2-Ej, 3-ExCa, 4-Pr, 5-On.

133. Jørgensen, M. and D.I.K. Sjøberg, Software Process Improvement and Human Judgement Heuristics. Scandinavian Journal of Information Systems, 2001. 13: p. 99-121. 1-Em, 2-Ej, 3-Re, 4-Nr, 5-No.

134. Jørgensen, M. and D.I.K. Sjøberg, An Effort Prediction Interval Approach Based on the Empirical Distribution of Previous Estimation Accuracy. Information and Software Technology, 2003. 45(3): p. 123-136. 1-Un, 2-RgEj, 3-ExDmHv, 4-Pr, 5-On.

135. Jørgensen, M. and D.I.K. Sjøberg, The Impact of Customer Expectation on Software Development Effort Estimates. International Journal of Project Management, 2004. 22: p. 317-325. 1-Oi, 2-Ej, 3-Ex, 4-StPr, 5-On.

136. Jørgensen, M., K.H. Teigen, and K. Moløkken, Better Sure Than Safe? Over-Confidence in Judgement Based Software Development Effort Prediction Intervals. Journal of Systems and Software, 2004. 70(1-2): p. 79-93. 1-UnOi, 2-Ej, 3-ExSv, 4-PrSt, 5-On.

137. Kalipsiz, Y. and S. Kurnaz, Adapting Function Point Analysis to Object Oriented Systems. Pakistan Journal of Information and Technology, 2002. 1(1): p. 5-11. 1-CmSm, 2-Fp, 3-Dm, 4-Nr, 5-No.

138. Kamatar, J. and W. Hayes, An Experience Report on the Personal Software Process. IEEE Software, 2000. 17(6): p. 85-89. 1-EmOi, 2-Ej, 3-Ca, 4-Pr, 5-On.

139. Kaplan, H.T., The Ada COCOMO Cost Estimating Model and VASTT Development Estimates vs. Actuals. Vitro Technical Journal, 1991. 9(1): p. 48-60. 1-EmCm, 2-Rg, 3-Hv, 4-Pr, 5-On.

140. 140. Kappelman, L.A., et al., Calculating the Cost of Year-2000 Compliance. Communications of the ACM, 1998. 41(2): p. 30-39. 1-Ot, 2-Nr, 3-Sv, 4-Pr, 5-On. (S)

141. Keil, M., et al., An Investigation of Risk Perception and Risk Propensity on the Decision to Continue a Software Development Project. Journal of Systems and Software, 2000. 53(2): p. 145-157. 1-Un, 2-Ej, 3-Ex, 4-St, 5-On.

142. Keller-Mcnulty, S., M.S. McNulty, and D.A. Gustafson, Stochastic Models for Software Science. Journal of Systems and Software, 1991. 16(1): p. 59-68. 1-Sm, 2-Th, 3-HvDm, 4-Pr, 5-On.

143. Kemerer, C.F., An Empirical Validation of Software Cost Estimation Models. Communications of the ACM, 1987. 30(5): p. 416-429. 1-EmCm, 2-RgFpTh, 3-Hv, 4-Pr, 5-Oi.

144. Kemerer, C.F., Reliability of Function Points Measurement: A Field Experiment. Communications of the ACM, 1993. 36(2): p. 85-97. 1-Sm, 2-Fp, 3-Ex, 4-Pr, 5-On. (S)

145. Kemerer, C.F. and B.S. Porter, Improving the Reliability of Function Point Measurement: An Empirical Study. IEEE Transactions on Software Engineering, 1992. 18(11): p. 1011-1024. 1-Sm, 2-Fp, 3-SvHv, 4-Pr, 5-On.

146. Kirsopp, C. and M. Sheppard, Making Inferences with Small Numbers of Training Sets. IEE Proceedings Software Engineering, 2002. 149(5): p. 123-130. 1-Ds, 2-An, 3-Si, 4-Pr, 5-DhShFi.

147. Kitchenham, B., Empirical Studies of Assumptions that Underlie Software Cost-Estimation Models. Information and Software Technology, 1992. 34(4): p. 211-218. 1-Pf, 2-RgFp, 3-Hv, 4-Pr, 5-BcBaBeDeWi-KeKmMe.

148. Kitchenham, B., A Procedure for Analyzing Unbalanced Datasets. IEEE Transactions on Software Engineering, 1998. 24(4): p. 278-301. 1-Ds, 2-Rg, 3-Hv, 4-Pr, 5-Bc.

149. Kitchenham, B. and S. Linkman, Estimates, Uncertainty, and Risk. IEEE Software, 1997. 14(3): p. 69-74. 1-OiUn, 2-Nr, 3-Ox, 4-Nr, 5-No.

150. Kitchenham, B., et al., An Empirical Study of Maintenance and Development Estimation Accuracy. Journal of Systems and Software, 2002. 64(1): p. 57-77. 1-Em, 2-RgCbFp, 3-DmHv, 4-Pr, 5-Oi.

151. Kitchenham, B. and N.R. Taylor, Software Project Development Cost Estimation. Journal of Systems and Software, 1985. 5(4): p. 267-278. 1-Em, 2-Rgth, 3-DmHv, 4-Pr, 5-Oi.

152. Kitchenham, B.A., The Question of Scale Economies in Software—Why Cannot Researchers Agree? Information and Software Technology, 2002. 44(1): p. 13-24. 1-Pf, 2-RgOt, 3-Hv, 4-Pr, 5-KeBeMe.

153. Kitchenham, B.A., et al., Modeling Software Bidding Risks. IEEE Transactions on Software Engineering, 2003. 29(6): p. 542-554. 1-EpOiUn, 2-Th, 3-Si, 4-Nr, 5-No.

154. Kitchenham, B.A., et al., What Accuracy Statistics Really Measure. IEE Proceedings Software, 2001. 148(3): p. 81-85. 1-Ep, 2-Nr, 3-Th, 4-Nr, 5-No.

155. Kitchenham, B.A. and N.R. Taylor, Software Cost Models. ICL Technical Journal, 1984. 4(1): p. 73-102. 1-Em, 2-RgTh, 3-HvDm, 4-Pr, 5-Oi.

156. Koch, S. and G. Schneider, Effort, Co-Operation and Co-Ordination in an Open Source Software Project: GNOME. Information Systems Journal, 2002. 12(1): p. 27-42. 1-Em, 2-Th, 3-DmCa, 4-Pr, 5-On.

157. Kusters, R.J., M.J.I.M. Genuchten, and F.J. Heemstra, Are Software Cost-Estimation Models Accurate? Information and Software Technology, 1990. 32(3): p. 187-190. 1-Em, 2-EjRg, 3-Hv, 4-Pr, 5-On.

158. Kusumoto, S., et al., Software Project Simulator for Effective Process Improvement. Transactions of the Information Processing Society of Japan, 2001. 42(3): p. 396-408. 1-Em, 2-OtSi, 3-DmHv, 4-Pr, 5-On.

159. Känsälä, K., Integrating Risk Assessment with Cost Estimation. IEEE Software, 1997. 14(3): p. 61-67. 1-Un, 2-Nr, 3-Rl, 4-Pr, 5-On. (S)

160. Lanza, R.B., Getting to Realistic Estimates and Project Plans: A Monte Carlo Approach. Information Strategy: The Executive's Journal, 2003. 19(4): p. 26-34. 1-EmUn, 2-Si, 3-Ot, 4-Nr, 5-No.

161. Laranjeira, L.A., Software Size Estimation of Object-Oriented Systems. IEEE Transactions on Software Engineering, 1990. 16(5): p. 510-522. 1-SmUn, 2-Rg, 3-HvDm, 4-Pr, 5-Oi.

162. Larsen, K.R.T. and P.A. Bloniarz, A Cost and Performance Model for Web Service Investment. Communications of the ACM, 2000. 43(2): p. 109-116. 1-Em, 2-Ot, 3-DmOx, 4-Nr, 5-No.

163. Lassez, J.-L., et al., A Critical Examination of Software Science. Journal of Systems and Software, 1981. 2(2): p. 105-112. 1-Em, 2-Th, 3-Hv, 4-Pr, 5-On.

164. Lavazza, L. and G. Valetto, Requirements-Based Estimation of Change Costs. Empirical Software Engineering, 2000. 5(3): p. 229-243. 1-Em, 2-Rg, 3-Hv, 4-Pr, 5-On.

165. Lederer, A.L., et al., Information System Cost Estimating: A Management Perspective. MIS Quarterly, 1990. 14(2): p. 159-176. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-On.

166. Lederer, A.L. and J. Prasad, The Validation of a Political Model of Information Systems Development Cost Estimating. ACM Transactions on Computer Personnel, 1991. 13(2): p. 47-57. 1-Oi, 2-Ej, 3-Sv, 4-Pr, 5-On.

167. Lederer, A.L. and J. Prasad, Nine Management Guidelines for Better Cost Estimating. Communications of the ACM, 1992. 35(2): p. 51-59. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-Le.

168. Lederer, A.L. and J. Prasad, Information Systems Software Cost Estimating: A Current Assessment. Journal of Information Technology, 1993. 8(1): p. 22-33. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-Le.

169. Lederer, A.L. and J. Prasad, Systems Development and Cost Estimating. Challenges and Guidelines. Information Systems Management, 1993. 10(4): p. 37-41. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-Le.

170. Lederer, A.L. and J. Prasad, Causes of Inaccurate Software Development Cost Estimates. Journal of Systems and Software, 1995. 31(2): p. 125-134. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-Le.

171. Lederer, A.L. and J. Prasad, Perceptual Congruence and Systems Development Cost Estimation. Information Resources Management Journal, 1995. 8(4): p. 16-27. 1-Oi, 2-Ej, 3-Sv, 4-Pr, 5-Le.

172. Lederer, A.L. and J. Prasad, A Causal Model for Software Cost Estimating Error. IEEE Transactions on Software Engineering, 1998. 24(2): p. 137-148. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-Le.

173. Lederer, A.L. and J. Prasad, Software Management and Cost Estimating Error. Journal of Systems and Software, 2000. 50(1): p. 33-42. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-Le.

174. Lee, A., C. Chun Hung, and J. Balakrishnan, Software Development Cost Estimation: Integrating Neural Network with Cluster Analysis. Information and Management, 1998. 34(1): p. 1-9. 1-Em, 2-AnNn, 3-DmHv, 4-Pr, 5-Bc.

175. Lee, H., A Structured Methodology for Software Development Effort Prediction Using the Analytic Hierarchy Process. Journal of Systems and Software, 1993. 21(2): p. 179-186. 1-Em, 2-EjWb, 3-Dm, 4-Nr, 5-No.

176. Leung, H.K.N., Estimating Maintenance Effort by Analogy. Empirical Software Engineering, 2002. 7(2): p. 157 - 175. 1-Em, 2-Anrg, 3-DmHv, 4-Pr, 5-Oi.

177. Lewis, J.P., Limits to Software Estimation. Software Engineering Notes, 2001. 26(4): p. 54-59. 1-Oi, 2-Nr, 3-Ot, 4-Nr, 5-No.

178. Lind, M.R. and J.M. Sulek, Undersizing Software Systems: Third versus Fourth Generation Software Development. European Journal of Information Systems, 1998. 7(4): p. 261-268. 1-EmOi, 2-Nr, 3-ReHv, 4-Pr, 5-On.

179. Lind, M.R. and J.M. Sulek, A Methodology for Forecasting Knowledge Work Projects. Computers & Operations Research, 2000. 27(11-12): p. 1153-1169. 1-Em, 2-NnRg, 3-DmHv, 4-Pr, 5-On.

180. Lo, -.B.-W.-N. and Xiangzhu-Gao, Assessing Software Cost Estimation Models: Criteria for Accuracy, Consistency and Regression. Australian Journal of Information Systems, 1997. 5(1): p. 30-44. 1-Ep, 2-Nr, 3-Rehv, 4-Pr, 5-Dh.

181. Lokan, C.J., Early Size Prediction for C and PASCAL Programs. Journal of Systems and Software, 1996. 32(1): p. 65-72. 1-Sm, 2-Rg, 3-DmHv, 4-Pr, 5-On.

182. Lokan, C.J., An Empirical Analysis of Function Point Adjustment Factors. Information and Software Technology, 2000. 42(9): p. 649-659. 1-Sm, 2-Fp, 3-Hv, 4-Pr, 5-Is.

183. Lokan, C.J. and S.D. Conte, Size Prediction for Pascal Programs. Journal of Computer and Software Engineering, 1995. 3(4): p. 377-394. 1-Sm, 2-Rg, 3-DmHv, 4-Pr, 5-Oi.

184. Londeix, B., Deploying Realistic Estimation (Field Situation Analysis). Information and Software Technology, 1995. 37(12): p. 655-670. 1-Oi, 2-Nr, 3-Ox, 4-Pr, 5-On. (S)

185. Low, G.C. and D.R. Jeffery, Function Points in the Estimation and Evaluation of the Software Process. IEEE Transactions on Software Engineering, 1990. 16(1): p. 64-71. 1-Sm, 2-Fp, 3-Ex, 4-Pr, 5-On.

186. MacDonell, S.G., Comparative Review of Functional Complexity Assessment Methods for Effort Estimation. Software Engineering Journal, 1994. 9(3): p. 107-116. 1-Em, 2-Nr, 3-Re, 4-Nr, 5-No.

187. MacDonell, S.G., Establishing Relationships between Specification Size and Software Process Effort in CASE Environments. Information and Software Technology, 1997. 39(1): p. 35-45. 1-Sm, 2-Rg, 3-HvDm, 4-Pr, 5-On.

188. MacDonell, S.G., Software Source Code Sizing Using Fuzzy Logic Modeling. Information and Software Technology, 2003. 45(7): p. 389-404. 1-Sm, 2-RgOt, 3-DmHv, 4-St, 5-On.

189. MacDonell, S.G., T. Fletcher, and B.L.W. Wong, Industry Practices in Project Management for Multimedia Information Systems. International Journal of Software Engineering and Knowledge Engineering, 1999. 9(6): p. 801-815. 1-EmEp, 2-Nr, 3-Sv, 4-Pr, 5-On.

190. MacDonell, S.G. and M.J. Shepperd, Combining Techniques to Optimize Effort Predictions in Software Project Management. Journal of Systems and Software, 2003. 66(2): p. 91-98. 1-Em, 2-CbRgEjAn, 3-DmHv, 4-Pr, 5-On.

191. Madachy, R.J., Knowledge-Based Risk Assessment and Cost Estimation. Automated Software Engineering, 1995. 2(3): p. 219-230. 1-Em, 2-RgSi, 3-DmHv, 4-Pr, 5-BcOn.

192. Madachy, R.J., Heuristic Risk Assessment Using Cost Factors. IEEE Software, 1997. 14(3): p. 51-59. 1-EmUn, 2-Rg, 3-DmHv, 4-Pr, 5-BcOn.

193. Mair, C., et al., An Investigation of Machine Learning Based Prediction Systems. Journal of Systems and Software, 2000. 53(1): p. 23-29. 1-Em, 2-RgNnAnCt, 3-Hv, 4-Pr, 5-Dh.

194. Marouane, R. and A. Mili, Economics of Software Project Management in Tunisia: Basic TUCOMO. Information and Software Technology, 1989. 31(5): p. 251-257. 1-Cm, 2-Rg, 3-Hv, 4-Pr, 5-BcOn.

195. Marshall, I.M., et al., Code-Based Analysis of the Development Effort of a Large Scale Courseware Project. Information and Software Technology, 1997. 39(8): p. 541-549. 1-Em, 2-Rg, 3-HvDm, 4-St, 5-On. (S)

196. Marshall, I.M., et al., Predicting the Development Effort of Multimedia Courseware. Information and Software Technology, 1994. 36(5): p. 251-258. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-Oi.

197. Marwane, R. and A. Mili, Building Tailor-Made Software Cost Model: Intermediate TUCOMO. Information and Software Technology, 1991. 33(3): p. 232-238. 1-CmEm, 2-Rg, 3-Hv, 4-Pr, 5-BcOn.

198. Matson, J.E., B.E. Barrett, and J.M. Mellichamp, Software Development Cost Estimation Using Function Points. IEEE Transactions on Software Engineering, 1994. 20(4): p. 275-287. 1-Em, 2-FpRg, 3-DmHv, 4-Pr, 5-OnAgKe.

199. Matson, J.E. and J.M. Mellichamp, An Object-Oriented Tool for Function Point Analysis. Expert Systems, 1993. 10(1): p. 3-14. 1-Sm, 2-Fp, 3-Dm, 4-Nr, 5-No. (S)

200. Maxwell, -.K.-D., -.L. Van-Wassenhove, and -.S. Dutta, Software Development Productivity of European Space, Military, and Industrial Applications. IEEE Transactions on Software Engineering, 1996. 22(10): p. 706-718. 1-SmOt, 2-Rg, 3-Hv, 4-Pr, 5-On.

201. Maxwell, -.K., -.L. van-Wassenhove, and -.S. Dutta, Performance Evaluation of General and Company Specific Models in Software Development Effort Estimation. Management Science, 1999. 45(6): p. 787-803. 1-Cm, 2-Rg, 3-Hv, 4-Pr, 5-Es.

202. McCulla, P., The Estimating Process (IT Project Management). International Journal of Project Management, 1989. 7(1): p. 36-38. 1-EmOi, 2-EjRgFp, 3-OxDm, 4-Nr, 5-No. (S)

203. Mendes, E., S. Counsell, and N. Mosley, Web Hypermedia Cost Estimation: Further Assessment and Comparison of Cost Estimation Modelling Techniques. New Review of Hypermedia and Multimedia, 2002. 8: p. 199-229. 1-Em, 2-RgAnCt, 3-DmHv, 4-Pr, 5-On.

204. Mendes, E., N. Mosley, and S. Counsell, Web Metrics—Estimating Design and Authoring Effort. IEEE Multimedia, 2001. 8(1): p. 50-57. 1-Em, 2-Rg, 3-Hv, 4-St, 5-On.

205. Mendes, E., N. Mosley, and S. Counsell, Comparison of Web Size Measures for Predicting Web Design and Authoring Effort. IEE Proceedings Software, 2002. 149(3): p. 86-92. 1-EmSm, 2-Rg, 3-DmHv, 4-St, 5-On.

206. Mendes, E., et al., A Comparative Study of Cost Estimation Models for Web Hypermedia Applications. Empirical Software Engineering, 2003. 8(2): p. 163-196. 1-Em, 2-RgAnCt, 3-DmHv, 4-Pr, 5-Oi.

207. Miranda, E., Improving Subjective Estimates Using Paired Comparisons. IEEE Software, 2001. 18(1): p. 87-91. 1-Sm, 2-Ej, 3-DmHv, 4-Pr, 5-On.

208. Misic, V.B. and D.N. Tesic, Estimation of Effort and Complexity: An Object-Oriented Case Study. Journal of Systems and Software, 1998. 41(2): p. 133-143. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-Oi.

209. Miyazaki, Y., et al., Method to Estimate Parameter Values in Software Prediction Models. Information and Software Technology, 1991. 33(3): p. 239-243. 1-SmEp, 2-Rg, 3-Hv, 4-Pr, 5-Oi.

210. Miyazaki, Y., et al., Robust Regression for Developing Software Estimation Models. Journal of Systems and Software, 1994. 27(1): p. 3-16. 1-SmEp, 2-Rg, 3-HvDm, 4-Pr, 5-Oi.

211. Mizuno, O., et al., Statistical Analysis of Deviation of Actual Cost from Estimated Cost Using Actual Project Data. Information and Software Technology, 2000. 42(7): p. 465-473. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-On.

212. Mohanty, S.N., Software Cost Estimation: Present and Future. Software—Practice and Experience, 1981. 11(2): p. 103-121. 1-Em, 2-RgTh, 3-Re, 4-Nr, 5-No.

213. Moores, T.T., Developing a Software Size Model for Rule-Based Systems: A Case Study. Expert Systems with Applications, 2001. 21(4): p. 229-237. 1-Em, 2-Rg, 3-Hv, 4-Pr, 5-On.

214. Morgan, M.J., Controlling Software Development Costs. Industrial Management & Data Systems, 1994. 94(1): p. 13-18. 1-Oi, 2-Nr, 3-ReDm, 4-Nr, 5-No.

215. Moser, S., B. Henderson-Sellers, and V.B. Misic, Cost Estimation Based on Business Models. Journal of Systems and Software, 1999. 49(1): p. 33-42. 1-EmSm, 2-FpRg, 3-DmHv, 4-Pr, 5-Oi.

216. Moses, J., Measuring Effort Estimation Uncertainty to Improve Client Confidence. Software Quality Journal, 2002. 10: p. 135-148. 1-Un, 2-By, 3-DmHv, 4-Pr, 5-Bc. (S)

217. Moses, J. and M. Farrow, A Procedure for Assessing the Influence of Problem Domain on Effort Estimation Consistency. Software Quality Journal., 2003. 11(4): p. 283-300. 1-Cm, 2-RgBy, 3-DmHv, 4-Pr, 5-Bc.

218. Mukhopadhyay, T. and S. Kekre, Software Effort Models for Early Estimation of Process Control Applications. IEEE Transactions on Software Engineering, 1992. 18(10): p. 915-924. 1-EmSm, 2-RgFp, 3-HvDm, 4-Pr, 5-On.

219. Mukhopadhyay, T., S.S. Vicinanza, and M.J. Prietula, Examining the Feasibility of a Case-Based Reasoning Model for Software Effort Estimation. MIS Quarterly, 1992. 16(2): p. 155-171. 1-Em, 2-EjRgAnFp, 3-DmHv, 4-Pr, 5-On.

220. Murali, C.S. and C.S. Sankar, Issues in Estimating Real-Time Data Communications Software Projects. Information and Software Technology, 1997. 39(6): p. 399-402. 1-Cm, 2-Fp, 3-HvDm, 4-Pr, 5-Oi.

221. Musilek, P., et al., Software Cost Estimation with Fuzzy Models. Applied Computing Review, 2000. 8(2): p. 24-29. 1-Un, 2-RgOt, 3-ThDm, 4-Nr, 5-No.

222. Myrtveit, I. and E. Stensrud, A Controlled Experiment to Assess the Benefits of Estimating with Analogy and Regression Models. IEEE Transactions on Software Engineering, 1999. 25(4): p. 510-525. 1-Em, 2-EjAnRg, 3-DmHvEx, 4-Pr, 5-On.

223. Myrtveit, I., E. Stensrud, and U.H. Olsson, Analyzing Data Sets with Missing Data: An Empirical Evaluation of Imputation Methods and Likelihood-Based Methods. IEEE Transactions on Software Engineering, 2001. 27(11): p. 999-1013. 1-Oi, 2-Nr, 3-Hv, 4-Pr, 5-On.

224. Navlakha, J.K., Choosing A Software Cost Estimation Model for Your Organization: A Case Study. Information and Management, 1990. 18(5): p. 255-261. 1-EpEm, 2-Rg, 3-HvDm, 4-Pr, 5-Oi.

225. Nesi, P. and T. Querci, Effort Estimation and Prediction of Object-Oriented Systems. Journal of Systems and Software, 1998. 42(1): p. 89-102. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-On.

226. O'Brien, S.J. and D.A. Jones, Function Points in SSADM. Software Quality Journal, 1993. 2(1): p. 1-11. 1-Sm, 2-Fp, 3-Ox, 4-Nr, 5-No.

227. Ohlsson, N., C. Wohlin, and B. Regnell, A Project Effort Estimation Study. Information and Software Technology, 1998. 40(14): p. 831-839. 1-Em, 2-Ej, 3-Rl, 4-St, 5-On.

228. Ooi, G. and C. Soh, Developing an Activity-Based Costing Approach for System Development and Implementation. Advances in Information Systems, 2003. 34(3): p. 54-71. 1-Em, 2-RgWb, 3-DmHv, 4-Pr, 5-On.

229. Oriogun, P.K., A Survey of Boehm's Work on the Spiral Models and COCOMO II—Towards Software Development Process Quality Improvement. Software Quality Journal, 1999. 8: p. 53-62. 1-Em, 2-Rg, 3-DmOx, 4-Pr, 5-No.

230. Orr, G. and T.E. Reeves, Function Point Counting: One Program's Experience. Journal of Systems and Software, 2000. 53(3): p. 239-244. 1-Sm, 2-Fp, 3-Ox, 4-Pr, 5-On.

231. Parr, F.N., An Alternative to the Rayleigh Curve Model for Software Development Effort. IEEE Transactions on Software Engineering, 1980. 6(3): p. 291-296. 1-Em, 2-Th, 3-Th, 4-Nr, 5-No.

232. Pendharkar, P.C. and G.H. Subramanian, Connectionist Models for Learning, Discovering, and Forecasting Software Effort: An Empirical Study. Journal of Computer Information Systems, 2002. 43(1): p. 7-14. 1-Em, 2-Nn, 3-Dm, 4-Pr, 5-On.

233. Pengelly, A., Performance of Effort Estimating Techniques in Current Development Environments. Software Engineering Journal, 1995. 10(5): p. 162-170. 1-Em, 2-RgFpTh, 3-Hv, 4-Pr, 5-On.

234. Pfleeger, S.L., Model of Software Effort and Productivity. Information and Software Technology, 1991. 33(3): p. 224-231. 1-Em, 2-Rg, 3-HvDm, 4-Pr, 5-On.

235. Pfleeger, S.L. and T.B. Bollinger, The Economics of Reuse: New Approaches to Modelling and Assessing Cost. Information and Software Technology, 1994. 36(8): p. 475-484. 1-Em, 2-Wb, 3-ReDm, 4-Nr, 5-No.

236. Pham, H. and H. Wang, A Quasi-Renewal Process for Software Reliability and Testing Costs. IEEE Transactions on Systems, Man & Cybernetics, Part A Systems & Humans, 2001. 31(6): p. 623-631. 1-Em, 2-Th, 3-Th, 4-Nr, 5-No. (S)

237. Pham, H. and X. Zhang, A Software Cost Model with Warranty and Risk Costs. IEEE Transactions on Computers, 1999. 48(1): p. 71-75. 1-Em, 2-Th, 3-DmHv, 4-Pr, 5-On.

238. Pickard, L., B. Kitchenham, and S.J. Linkman, Using Simulated Data Sets to Compare Data Analysis Techniques Used for Software Cost Modelling. IEE Proceedings Software, 2001. 148(6): p. 165-174. 1-Em, 2-RgCt, 3-Si, 4-Nr, 5-No.

239. Pillai, K. and N. Sukumaran, A Model for Software Development Effort and Cost Estimation. IEEE Transactions on Software Engineering, 1997. 23(8): p. 485-497. 1-Em, 2-Th, 3-SiDm, 4-Nr, 5-No.

240. Prechelt, L. and B. Unger, An Experiment Measuring the Effects of Personal Software Process (PSP) Training. IEEE Transactions on Software Engineering, 2000. 27(5): p. 465-472. 1-Oi, 2-Ej, 3-Rl, 4-St, 5-On.

241. Prechelt, L., et al., A Controlled Experiment on Inheritance Depth as a Cost Factor for Code Maintenance. Journal of Systems and Software, 2003. 65(2): p. 115-126. 1-Em, 2-Rg, 3-DmEx, 4-St, 5-On.

242. Prietula, M.J., S.S. Vicinanza, and T. Mukhopadhyay, Software-Effort Estimation with a Case-Based Reasoner. Journal of Experimental and Theoretical Artificial Intelligence, 1996. 8(3-4): p. 341-363. 1-Em, 2-AnFpRg, 3-DmHv, 4-Pr, 5-KeOn.

243. Prokhorov, V.V. and D.V. Smirnov, A Model of Estimation and Optimization of Expenditures in Multilevel Fuzzy Environments. Programming and Computer Software, 2001. 27(5): p. 252-259. 1-Em, 2-Ot, 3-Dm, 4-Nr, 5-No.

244. Putnam, L.H., A General Empirical Solution to the Macro Software Sizing and Estimating Problem. IEEE Transactions on Software Engineering, 1978. 4(4): p. 345-360. 1-EmPfUn, 2-Th, 3-ThDm, 4-Nr, 5-No.

245. Putnam, L.H. and A. Fitzsimmons, Estimating Software Costs. Datamation, 1979. 25(11): p. 171-178. 1-EmUn, 2-Th, 3-ThDm, 4-Nr, 5-No.

246. Putnam, L.H., D.T. Putnam, and W. Myers, Adapting Project Estimation to Advancing Technologies. American Programmer, 1996. 9(6): p. 23-29. 1-Em, 2-WbAn, 3-Ox, 4-Pr, 5-No.

247. Rask, R., P. Laamanen, and K. Lyytinen, Simulation and Comparison of Albrecht's Function Point and Demarco's Function Bang Metrics in a CASE Environment. IEEE Transactions on Software Engineering, 1993. 19(7): p. 661-671. 1-Sm, 2-Fp, 3-Si, 4-Nr, 5-On.

248. Ratcliff, B. and A.L. Rollo, Adapting Function Point Analysis to Jackson System Development. Software Engineering Journal, 1990. 5(1): p. 79-84. 1-EmSm, 2-Fp, 3-DmHv, 4-Pr, 5-On.

249. Reifer, D.J., ASSET-R: A Function Point Sizing Tool for Scientific and Real-Time Systems. Journal of Systems and Software, 1990. 11(3): p. 159-172. 1-Sm, 2-Fp, 3-HvDm, 4-Pr, 5-On.

250. Reifer, D.J., Web Development: Estimating Quick-to-Market Software. IEEE Software, 2000. 17(6): p. 57-64. 1-Em, 2-Rg, 3-DmOx, 4-Nr, 5-No.

251. Ruiz, M., I. Ramos, and M. Toro, A Simplified Model of Software Project Dynamics. Journal of Systems and Software, 2001. 59(3): p. 299-309. 1-Oi, 2-Nr, 3-Si, 4-Pr, 5-On.

252. Rush, C. and R. Roy, Expert Judgement in Cost Estimating: Modelling the Reasoning Process. Concurrent Engineering: Research and Applications, 2001. 9(4): p. 271-284. 1-Em, 2-Ej, 3-DmSv, 4-Pr, 5-On.

253. Rutherford, J., Software Estimating and Metrics—A Pragmatic Approach. GEC Journal of Research, 1995. 12(2): p. 66-75. 1-Em, 2-Rg, 3-ReOx, 4-Nr, 5-No.

254. Samson, B., D. Ellison, and P. Dugard, Software Cost Estimation Using an Albus Perceptron (CMAC). Information and Software Technology, 1997. 39(1): p. 55-60. 1-Em, 2-RgNn, 3-DmHv, 4-Pr, 5-Bc.

255. Schneider, V., Prediction of Software Effort and Project Duration-Four New Formulas. SIGPLAN Notices, 1978. 13(6): p. 49-59. 1-Em, 2-Th, 3-DmHv, 4-Pr, 5-On.

256. Schooff, R.M. and Y.Y. Haimes, Dynamic Multistage Software Estimation. IEEE Transactions on Systems, Man, and Cybernetics, 1999. 29(2): p. 272-284. 1-Em, 2-SiOt, 3-Th, 4-Nr, 5-No.

257. Sengupta, K. and T.K. Abdel-Hamid, The Impact of Unreliable Information on the Management of Software Projects: A Dynamic Decision Perspective. IEEE Transactions on Systems, Man, and Cybernetics, 1996. 26(2): p. 177-189. 1-Oi, 2-Ej, 3-Siex, 4-St, 5-On.

258. Shepperd, M. and M. Cartwright, Predicting with Sparse Data. IEEE Transactions on Software Engineering, 2001. 27(11): p. 987-998. 1-Em, 2-EjRgOt, 3-Hv, 4-Pr, 5-Oi.

259. Shepperd, M., M. Cartwright, and G. Kadoda, On Building Prediction Systems for Software Engineers. Empirical Software Engineering, 2000. 5(3): p. 175-182. 1-Ep, 2-Nr, 3-Hv, 4-Nr, 5-No.

260. Shepperd, M. and G. Kadoda, Comparing Software Prediction Techniques Using Simulation. IEEE Transactions on Software Engineering, 2001. 27(11): p. 1014-1022. 1-Em, 2-RgAnCtNn, 3-Si, 4-Nr, 5-On.

261. Shepperd, M. and C. Schofield, Estimating Software Project Effort Using Analogies. IEEE Transactions on Software Engineering, 1997. 23(11): p. 736-743. 1-Em, 2-RgAn, 3-DmHv, 4-Pr, 5-AgAtDhKeOiOn.

262. Shoval, P. and O. Feldman, A Combination of the Mk-II Function Points Software Estimation Method with the ADISSA Methodology for Systems Analysis and Design. Information and Software Technology, 1997. 39(13): p. 855-865. 1-EmSm, 2-Fp, 3-Dm, 4-Nr, 5-No.

263. Shukla, K.K., Neuro-Genetic Prediction of Software Development Effort. Information and Software Technology, 2000. 42(10): p. 701-713. 1-Em, 2-RgNnOt, 3-DmHv, 4-Pr, 5-BcKe.

264. Smith, R.K., J.E. Hale, and A.S. Parrish, An Empirical Study Using Task Assignment Patterns to Improve the Accuracy of Software Effort Estimation. IEEE Transactions on Software Engineering, 2001. 27(3): p. 264-271. 1-Em, 2-RgFp, 3-Hv, 4-Pr, 5-On.

265. Srinivasan, K. and D. Fisher, Machine Learning Approaches to Estimating Software Development Effort. IEEE Transactions on Software Engineering, 1995. 21(2): p. 126-137. 1-Em, 2-FpCtNnTh, 3-DmHv, 4-Pr, 5-BcKe.

266. Stamelos, I. and L. Angelis, Managing Uncertainty in Project Portfolio Cost Estimation. Information and Software Technology, 2001. 43(13): p. 759-768. 1-Un, 2-AnOt, 3-Hv, 4-Pr, 5-Ab.

267. Stamelos, I., et al., On the Use of Bayesian Belief Networks for the Prediction of Software Productivity. Information and Software Technology, 2003. 45(1): p. 51-60. 1-EmUn, 2-EjBy, 3-DmHv, 4-Pr, 5-Bc.

268. Stensrud, E., et al., A Further Empirical Investigation of the Relationship between MRE and Project Size. Empirical Software Engineering, 2003. 8(2): p. 139-161. 1-Ep, 2-Rg, 3-Hv, 4-Pr, 5-OnAgKeFiDh.

269. Stensrud, E., Alternative Approaches to Effort Prediction of ERP Projects. Information and Software Technology, 2001. 43(7): p. 413-423. 1-Em, 2-RgAnCtEj, 3-Re, 4-Nr, 5-No.

270. Stewart, B., Predicting Project Delivery Rates Using the Naive-Bayes Classifier. Journal of Software Maintenance and Evolution: Research and Practice, 2002. 14(3): p. 161-179. 1-Em, 2-CtNnBy, 3-DmHv, 4-Pr, 5-Is. (S)

271. Strike, K., K. El-Emam, and N. Madhavji, Software Cost Estimation with Incomplete Data. IEEE Transactions on Software Engineering, 2001. 27(10): p. 890-908. 1-Ds, 2-Nr, 3-Hv, 4-Pr, 5-On.

272. Stutzke, R.D., Software Estimating Technology: A Survey. Journal of Defense Software Engineering (Crosstalk), 1996. 9(5): p. 17-22. 1-Em, 2-Nr, 3-Re, 4-Nr, 5-No.

273. Subramanian, A. and M.C. Lacity, Determinants of Variability in Function Point Estimates. Journal of End User Computing, 1997. 9(4): p. 19-28. 1-Sm, 2-Fp, 3-Ex, 4-St, 5-On.

274. Subramanian, G.H. and S. Breslawski, Dimensionality Reduction in Software Development Effort Estimation. Journal of Systems and Software, 1993. 21(2): p. 187-196. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-Bc.

275. Subramanian, G.H. and S. Breslawski, An Empirical Analysis of Software Effort Estimate Alterations. Journal of Systems and Software, 1995. 31(2): p. 135-141. 1-Oi, 2-Nr, 3-Sv, 4-Pr, 5-On.

276. Symons, C.R., Function Point Analysis: Difficulties and Improvements. IEEE Transactions on Software Engineering, 1988. 14(1): p. 2-11. 1-EmSm, 2-Rg, 3-HvDm, 4-Pr, 5-On.

277. Taff, L.M., J.W. Borchering, and J.W.R. Hudgins, Estimeetings: Development Estimates and a Front-End Process for a Large Project. IEEE Transactions on Software Engineering, 1991. 17(8): p. 839-849. 1-Em, 2-EjCb, 3-RlDm, 4-Pr, 5-On.

278. Tate, G. and J.M. Verner, Software Sizing and Costing Models: A Survey of Empirical Validation and Comparison Studies. Journal of Information Technology, 1990. 5(1): p. 12-26. 1-EmCmSm, 2-RgThFp, 3-Re, 4-Nr, 5-No.

279. Tausworthe, R.C., The Work Breakdown Structure in Software Project Management. Journal of Systems and Software, 1980. 1(3): p. 181-186. 1-Em, 2-Wb, 3-Ox, 4-Nr, 5-No.

280. Uemure, T., S. Kusumoto, and K. Inoue, Function-Point Analysis Using Design Specifications Based on the Unified Modelling Language. Journal of Software Maintenance and Evolution: Research and Practice, 2001. 13(4): p. 223-243. 1-Sm, 2-Fp, 3-Hv, 4-Pr, 5-On.

281. Van Der Poel, K.G. and S.R. Schach, A Software Metric for Cost Estimation and Efficiency Measurement in Data Processing System Development. Journal of Systems and Software, 1983. 3(3): p. 187-191. 1-SmEm, 2-Rg, 3-HvDm, 4-Pr, 5-Oi.

282. Van Genuchten, M., Why Is Software Late? An Empirical Study of Reasons for Delay in Software Development. IEEE Transactions on Software Engineering, 1991. 17(6): p. 582-590. 1-Oi, 2-Nr, 3-Sv, 4-Nr, 5-On.

283. Verner, J. and G. Tate, A Model for Software Sizing. Journal of Systems and Software, 1987. 7(2): p. 173-177. 1-Sm, 2-Fp, 3-Th, 4-Nr, 5-No.

284. Verner, J. and G. Tate, Estimating Size and Effort in Fourth-Generation Development. IEEE Software, 1988. 5(4): p. 15-23. 1-EmSm, 2-RgFP, 3-HvDm, 4-Pr, 5-Oi.

285. Verner, J. and G. Tate, A Software Size Model. IEEE Transactions on Software Engineering, 1992. 18(4): p. 265-278. 1-Sm, 2-Fp, 3-HvDm, 4-Pr, 5-On.

286. Verner, J.M., S.P. Overmyer, and K.W. Mccain, In the 25 Years Since the Mythical Man-Month What Have We Learned about Project Management? Information and Software Technology, 1999. 41(14): p. 1021-1026. 1-Em, 2-Nr, 3-Sv, 4-Pr, 5-On.

287. Vicinanza, S.S., T. Mukhopadhyay, and M.J. Prietula, Software Effort Estimation: An Exploratory Study of Expert Performance. Information Systems Research, 1991. 2(4): p. 243-262. 1-Em, 2-EjFpRg, 3-ExHv, 4-Pr, 5-Ke.

288. Vijayakumar, S., Use of Historical Data in Software Cost Estimation. Computing and Control Engineering Journal, 1997. 8(3): p. 113-119. 1-Em, 2-Rg, 3-DmHv, 4-Pr, 5-On.

289. Walkerden, F. and D.R. Jeffery, Software Cost Estimation: A Review of Models, Process, and Practice. Advances in Computers, 1997. 44: p. 59-125. 1-EmEp, 2-RgEjFpAnTh, 3-Re, 4-Nr, 5-No.

290. Walkerden, F. and R. Jeffery, An Empirical Study of Analogy-Based Software Effort Estimation. Empirical Software Engineering, 1999. 4(2): p. 135-158. 1-Em, 2-EjAnRg, 3-DmHv, 4-St, 5-On.

291. Walston, C. and C. Felix, A Method of Programming Measurement and Estimation. IBM Systems Journal, 1977. 16(1): p. 54-73. 1-EmPf, 2-Rg, 3-DmHv, 4-Pr, 5-On.

292. Warburton, R.D.H., Managing and Predicting the Costs of Real-Time Software. IEEE Transactions on Software Engineering, 1983. 9(5): p. 562-569. 1-Em, 2-Th, 3-Hv, 4-Pr, 5-On. (S)

293. Watson, K.I., COCOMO as a Schedule Prognosis and Validation Tool: A Case Study. Software Quality Journal, 1992. 1(4): p. 193-208. 1-Cm, 2-Rg, 3-Hv, 4-Pr, 5-On.

294. Weinberg, G.M. and E.L. Schulman, Goals and Performance in Computer Programming. Human Factors, 1974. 16(1): p. 70 - 77. 1-EmOi, 2-Ej, 3-Ex, 4-St, 5-On.

295. Whang, S., Contracting for Software Development. Management Science, 1992. 38(3): p. 307-324. 1-Ot, 2-Nr, 3-Re, 4-Nr, 5-No.

296. Wittig, G. and G. Finnie, Estimating Software Development Effort with Connectionist Models. Information and Software Technology, 1997. 39(7): p. 469-476. 1-Em, 2-Nn, 3-SiDmHv, 4-Pr, 5-AsDh.

297. Wittig, G.E. and G.R. Finnie, Using Artificial Neural Networks and Function Points to Estimate 4GL Software Development Effort. Australian Journal of Information Systems, 1994. 1(2): p. 87-94. 1-Em, 2-Nn, 3-DmHv, 4-Pr, 5-Oi.

298. Wolverton, R.W., The Cost of Developing Large-Scale Software. IEEE Transactions on Software Engineering, 1974. C-23(6): p. 615-636. 1-EmOiPf, 2-RgWb, 3-DmCa, 4-Pr, 5-On.

299. Woodfield, S.N., V.Y. Shen, and H.E. Dunsmore, A Study of Several Metrics for Programming Effort. Journal of Systems and Software, 1981. 2(2): p. 97-103. 1-Em, 2-RgTh, 3-HvDm, 4-Pr, 5-Oi.

300. Wrigley, C.D. and A.S. Dexter, A Model for Measuring Information System Size. MIS Quarterly, 1991. 15(2): p. 245-257. 1-Sm, 2-Rg, 3-Hv, 4-Pr, 5-On. (S)

301. Yau, C. and L.-Y. Gan, Comparing the Top-Down and Bottom-Up Approaches of Function Point Analysis: A Case Study. Software Quality Journal, 1995. 4(3): p. 175-187. 1-EmCm, 2-FpWb, 3-DmHv, 4-Pr, 5-On.

302. Yau, C. and T. Ho Leung, Modelling the Probabilistic Behaviour of Function Point Analysis. Information and Software Technology, 1998. 40(2): p. 59-68. 1-SmUn, 2-Fp, 3-DmHv, 4-Pr, 5-On.

303. Zhang, X. and H. Pham, A Software Cost Model with Error Removal Times and Risk Costs. International Journal of Systems Science, 1998. 29(4): p. 435-442. 1-Em, 2-Th, 3-DmHv, 4-Pr, 5-On. (S)

304. Zhang, X. and H. Pham, A Software Cost Model with Warranty Cost, Error Removal Times and Risk Costs. IIE Transactions, 1998. 30(12): p. 1135-1142. 1-Em, 2-Th, 3-DmHv, 4-Pr, 5-On.

REFERENCES

[1] B. Boehm, C. Abts, and S. Chulani, "Software Development Cost Estimation Approaches—A Survey," *Annals of Software Eng.*, vol. 10, pp. 177-205, 2000.

[2] B.W. Boehm, *Software Engineering Economics*, p. 767. Prentice-Hall, 1981.

[3] L.C. Briand and I. Wieczorek, "Resource Estimation in Software Engineering," *Encyclopedia of Software Eng.*, J.J. Marcinak, ed., pp. 1160-1196, John Wiley & Sons, 2002.

[4] J.S. Edwards and T.T. Moores, "A Conflict between the Use of Estimating and Planning Tools in the Management of Information Systems," *European J. Information Systems*, vol. 3, no. 2, pp. 139-147, 1994.

[5] R.E. Fairley, "Recent Advances in Software Estimation Techniques," *Int'l Conf. Software Eng.*, pp. 382-391, 1992.

[6] T. Foss et al., "A Simulation Study of the Model Evaluation Criterion MMRE," *IEEE Trans. Software Eng.*, vol. 29, no. 11, pp. 985-995, Nov. 2003.

[7] S. Grimstad, M. Jörgensen, and K.J. Moløkken-Östvold, "Software Effort Estimation Terminology: The Tower of Babel," *Information and Software Technology*, pp. 302-310, vol. 48, no. 4, 2006.

[8] F.J. Heemstra, "Software Cost Estimation," *Information and Software Technology*, vol. 34, no. 10, pp. 627-639, 1992.

[9] M. Jørgensen, "Experience with the Accuracy of Software Maintenance Task Effort Prediction Models," *IEEE Trans. Software Eng.*, vol. 21, no. 8, pp. 674-681, Aug. 1995.

[10] M. Jørgensen and T. Gruschke, "Industrial Use of Formal Software Cost Estimation Models: Expert Estimation in Disguise?" *Proc. Conf. Evaluation and Assessment in Software Eng. (EASE '05)*, pp. 1-7, 2005.

[11] B.W.N. Lo and Xiangzhu Gao, "Assessing Software Cost Estimation Models: Criteria for Accuracy, Consistency and Regression," *Australian J. Information Systems*, vol. 5, no. 1, pp. 30-44, 1997.

[12] S.G. MacDonell and M.J. Shepperd, "Combining Techniques to Optimize Effort Predictions in Software Project Management," *J. Systems and Software*, vol. 66, no. 2, pp. 91-98, 2003.

[13] C. Mair et al., An Investigation of Machine Learning Based Prediction Systems, *J. Systems and Software*, vol. 53, no. 1, pp. 23-29, 2000.

[14] K. Moløkken and M. Jörgensen, "A Review of Software Surveys on Software Effort Estimation," *Proc. Int'l Symp. Empirical Software Eng.*, pp. 223-230, 2003.

[15] M. Shepperd, M. Cartwright, and G. Kadoda, "On Building Prediction Systems for Software Engineers," *Empirical Software Eng.*, vol. 5, no. 3, pp. 175-182, 2000.

[16] F. Walkerden and D.R. Jeffery, "Software Cost Estimation: A Review of Models, Process, and Practice, *Advances in Computers*, vol. 44, pp. 59-125, 1997.

Magne Jørgensen received the Diplom Ingeneur degree in Wirtschaftswissenschaften from the University of Karlsruhe, Germany, in 1988 and the Dr. Scient. degree in informatics from the University of Oslo, Norway, in 1994. He has about 10 years industry experience as a software developer, project leader, and manager. He is now a professor of software engineering at the University of Oslo and a member of the software engineering research group of Simula Research Laboratory in Oslo, Norway. He has supported software project estimation improvement work and been responsible for estimation courses in several software companies.

Martin Shepperd received the PhD degree in computer science from the Open University in 1991 for his work in measurement theory and its application to software engineering. He is a professor of software technology at Brunel University, London, and the director of the Brunel Software Engineering Research Centre (B-SERC). He has published more than 90 refereed papers and three books in the area of empirical software engineering, machine learning and statistics. He is the editor-in-chief of the journal *Information & Software Technology* and was an associate editor of the *IEEE Transactions on Software Engineering* (2000-2004). He has also previously worked for a number of years as a software developer for a major bank.