Looking back on previous estimation error as a method to improve the uncertainty assessment of benefits and costs of software development projects

Magne Jørgensen
Simula Metropolitan,
Center for Digital Engineering
Oslo, Norway
magnej@simula.no

***Abstract*— Knowing the uncertainty of estimates of benefits and costs is useful when planning, budgeting and pricing projects. The traditional method for assessing such uncertainty is based on prediction intervals, e.g., asking for minimum and maximum values believed to be 90% likely to include the actual outcome. Studies report that the traditional method typically results in too narrow intervals and intervals that are too symmetric around the estimated most likely outcome when compared with the actual uncertainty of outcomes. We examine whether an uncertainty assessment method based on looking back on the previous estimation error of similar projects leads to wider and less symmetric prediction intervals. Sixty software professionals, with experience from estimating software project costs and benefits, were randomly divided into a group with a traditional or a group with a looking back-based uncertainty assessment method. We found that those using the looking back-based method had much wider prediction intervals for both costs and benefits. The software professionals of both groups provided uncertainty assessment values suggesting a left-skewed distribution for benefits and a right-skewed distribution for cost, but with much more skew among those using the looking back-based method. We argue that a looking back-based method is promising for improved realism in uncertainty assessment of benefits and costs of software development projects.**

Keywords—uncertainty assessment, software benefits and cost, controlled experiment

I. INTRODUCTION

There is no shortage of studies of human judgement documenting that people will give prediction intervals that are too narrow and too symmetric when asked to use the traditional uncertainty assessment method, i.e., the method based on giving minimum and maximum values with, typically, 90% confidence in including the actual value [2, 14, 18]. A typical result demonstrating the lack of correspondence between the confidence level and rate of including the actual effort (the hit rate) in the prediction interval is the one in [11], where software professionals giving traditional minimum–maximum intervals included the actual effort only 20–40% of the time, in spite of being instructed to be, typically, 90% sure to include the actual effort. Even after extensive feedback and training, the confidence level is typically much higher than the hit rate [5]. While knowledge of the problems with the traditional method for eliciting effort uncertainty intervals is not new, in particular when used in situations with high uncertainty, it is in common use and promoted, e.g., in the context of the PERT (Program Evaluation and Review Technique) tool and as part of the PMBOK (Project Management Book of Knowledge) [12, 16]. A possible reason for the promotion and widespread use of this method, frequently known as three-point estimates, is that the statistical theory behind it is sound, there are no clear alternatives, and there is typically little on-the-job feedback to show that the judgment-based input to the method frequently is strongly biased [11]. Unfortunate consequences of too narrow and symmetric uncertainty intervals are, amongst others, unrealistic cost-benefits analyses and too low budgets.

In previous papers we suggested and empirically evaluated an alternative method, based on the assumption that the distribution of the estimation accuracy of earlier, similar, software projects can be used to predict the uncertainty of new projects [6, 8-10]. Assume, for example, that a software professional wants to assess the uncertainty of a project that has been estimated to cost around 1 million USD. He or she looks back on estimation error experience from similar projects (memory-based or based on actual estimation error measurement) and reports that only 20% of them cost less than the estimated cost and around 10% cost more than twice the estimated cost. The software professional may be asked to add more empirical error data to provide the full empirical error distribution, but even these two data points (p20 = 1 mill. USD and p90 = 2 mill. USD) are sufficient to establish an uncertainty distribution, given the selection of a proper non-symmetric distribution belonging to the location-scale family [3, 13], e.g., a log-normal or gamma distribution. Figure 1 displays the log-normal distribution based on the above two estimation error data points. The cumulative error distribution is displayed in Figure 2. Figure 2 shows, for example, that the p50 (the value it is 50% likely to overrun) is around 1.3 mill. USD.

The results when using this alternative uncertainty assessment method have been good, suggesting increased realism compared to the traditional method [6, 17].

This paper extends our previous evaluations of the outlined alternative method by adding an analysis of the uncertainty assessment of benefits and of the benefits to costs ratio (return on investment) of software projects. In addition, the paper, compared to the previous papers, has a more explicit focus on differences in the skewness of the distributions provided by the traditional and the alternative uncertainty assessment method.

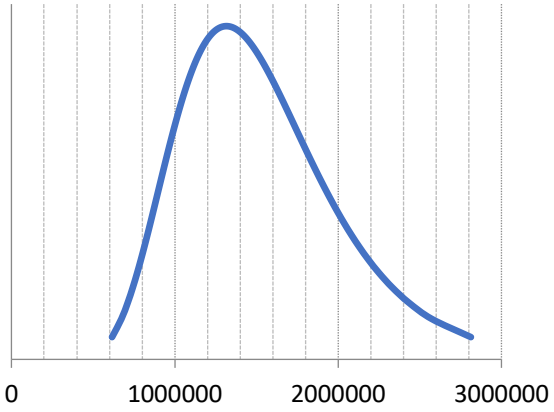

Fig. 1. Probability density cost distribution (log-normal)

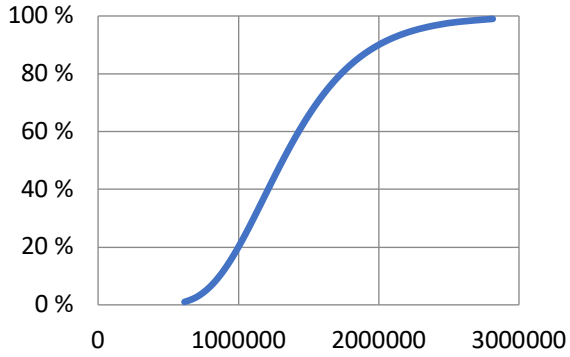

Fig. 2. Cumulative cost distribution (log-normal)

The hypotheses to be tested are the following:

H1: Those asked to look back (using the alternative method) will give wider and more *left-skewed* uncertainty intervals for software development benefits.

H2: Those asked to look back (using the alternative method) will give wider and more *right-skewed* uncertainty intervals for software development costs.

H3: The uncertainty assessment of those asked to look back (using the alternative method) will result in a lower benefits-to-costs ratio (return on investment) for software development costs.

The motivation for H1 and H2 is that we expect that software professionals instructed to look back will be reminded that much higher benefits and much lower costs than, respectively, the estimated benefits and costs are rare. Much lower benefits and much lower costs than estimated are, on the other hand, not uncommon. H3 will be true if H1 and H2 are true, but it is nevertheless interesting to examine on its own. Our expectation is that the difference in the expected benefits-to-costs ratio between those using the traditional and the alternative uncertainty assessment method is substantial.

II. Method

The participants were software professionals, mainly managers and projects leaders, attending a seminar on benefits management. The software professionals were first instructed to indicate their level of experience in benefits and costs estimation on a scale from 0 (none) to 5 (very high). Of the total of 65 responses received, from around 100 seminar participants, five had no experience with either costs or benefits management. These were removed from the analysis, leaving 60 responses for our analysis. The median experience level of the remaining participants was 3 (medium high) for benefits and 3 for costs estimation.

Following their responses on their experience level, the participants read the following description of a scenario based on a real-life software project (translated from Norwegian):

Digital solution for planning applications for construction work – uncertainty of costs and benefits

The municipal council of Oslo has decided to develop an IT system that will ease the planning applications of the non-professional and professional actors of the municipality desiring to build houses and other constructions.

The planning applications should, after secure login and selection of the relevant application type, be pre-registered with all the information the municipality already has about the person, company or property the application concerns. The field of the application where information about the construction work is to be provided should include links to the relevant regulations.

The user's application input will be checked automatically (where possible) with respect to the regulation adherence and correctness of the application type. Warnings should be provided if any non-adherences are found. The users should have the opportunity to download relevant maps, and have the functionality to put their own constructions on that map. All application information, e.g., feedback and decisions from the municipality, should be done through the software system. Status updates should in addition be given using secure email.

The estimate of the expected ***benefits*** *– given by experienced people at the municipality – is that the municipality will save 4 man-years per year when the system has been implemented. This will mainly be a consequence of the higher quality of the planning applications received and the need for fewer iterations before an application can be approved. In addition, it is expected that the users will save about 10 man-years each year through a simplified application process. Together with other quantitative benefits (summed over 10 years) the expected total benefit is expected to be around 120 million Norwegian Kroner. Non-quantitative benefits such as happier users and less illegal construction work are not included in the calculations.*

The estimate of the expected ***costs*** *– given by an external provider with relevant experience, is estimated to be around 65 million Norwegian Kroner.*

The above information is clearly not sufficient to say much about the realism of the benefits and costs estimates. Try, nevertheless, based on your own experience and other relevant knowledge with similar projects, to assess the uncertainty of the estimates of the benefits and of the costs.

The participants were randomly divided into two groups: traditional and alternative uncertainty assessment. These two groups had different instructions about the format of their uncertainty assessments.

Traditional Group (minimum–maximum values)

Uncertainty of the benefits estimate

Based on my experience with similar projects, I believe that the actual benefits (with 90% certainty) will be in the interval: _____ (minimum) and _____ (maximum) Norwegian Kroner.

Uncertainty of the costs estimate

Based on my experience with similar projects, I believe that the actual costs (with 90% certainty) will be in the interval: _____ (minimum) and _____ (maximum) Norwegian Kroner.

Alternative Group (looking back on previous benefits and costs estimation error)

Uncertainty of the benefits estimate

Input the proportion of similar (they do not have to be very similar), already completed, software projects for which you believe the benefits achieved were:

Less than half of the estimated benefits: _____% (Proportion of projects: 0%=none ... 100%=all)

Less than the estimated benefits: _____% (Proportion of projects: 0%=none ... 100%=all)

More than twice the estimated benefits: _____% (Proportion of projects: 0%=none ... 100%=all)

Uncertainty of the costs estimate

Input the proportion of similar (they do not have to be very similar), already completed, software projects for which you believe the actual costs were:

More than twice the estimated costs: _____% (Proportion of projects: 0%=none ... 100%=all)

More than the estimated costs: _____% (Proportion of projects: 0%=none ... 100%=all)

Less than half the estimated costs: _____% (Proportion of projects: 0%=none ... 100%=all)

The key difference between the two groups was that the first group used the traditional minimum–maximum uncertainty assessment method, with a given confidence level (here 90%), while the second were asked to assess the actual uncertainty (as indicated by their estimation error) of previous software projects. We suspected, as described in Section 1, that the traditional method would lead to the assessment of substantially less and more symmetric uncertainty than the uncertainty assessment method based on those looking back on the error, and implicitly the uncertainty, of similar, previously completed, software projects.

III. RESULTS

The assessments of the two groups differed greatly, both in terms of degree of uncertainty and in the amount of right- and left-skewedness of the implied uncertainty distributions. Tables I and , together with Figures 3–6 display key characteristics of the uncertainty assessments. For readability purposes, we translated the original uncertainty values into percentages of the estimates, i.e., in percent of the benefits estimate of 120 mill. Norwegian Kroner and of the costs estimate of 65 mill. Norwegian Kroner.

We have, for simplicity, assumed that a 90% confidence effort interval implies that the minimum is interpreted as the 5% level (p5), where it is only 5% likely that the actual value will be equal or less, and that the maximum is interpreted as the 95% level (p95), where it is 95% likely that the actual value will be equal or less. This is a common, although not necessary, interpretation of a 90% confidence effort prediction interval.

The *interval width* is measured as: *(p95-p5)/estimate*, where p5 and p95 are the values directly provided by the software professionals in the traditional method group, and the fitted values, assuming a PERT-distribution (using the tool @risk), for those in the alternative method group. The choice of a PERT-distribution to calculate the p5 and p95 for those in the alternative method group is based on the fact that this is a method frequently used in effort uncertainty assessment situations and that it enables us to compare the same pX-values for the two uncertainty assessment methods. The estimates are the same for the two groups, i.e., 120 mill. Norwegian Kroner for the benefits and 65 mill. Norwegian Kroner for the costs.

The *interval skew* is measured as: *distribution mean/estimate*, where the distribution mean is calculated using the PERT-formula: (Minimum + 4 x Estimate + Maximum). The minimum and maximum values are, as before, those provided by the software professionals in the traditional group and the fitted ones in the alternative uncertainty assessment group. The PERT-formula assumes that the Estimate is the mode (the most likely value). While this was not clear from the scenario description (Section 2), the intended interpretation of the benefits and costs estimate was not described, and it makes no large difference for the comparison of the two approaches. A skew-value larger than one, i.e., when the mean (expected value) of the distribution is higher than the estimate, indicates a right-skewed distribution, while a skew-value less than one indicates a left-skewed distribution. Notice that our measure of interval skew deviates from the traditional measure of distribution skew based on the difference between mode and mean.

TABLE I. TRADITIONAL UNCERTAINTY ASSESSMENT (MEDIAN VALUES)

Uncertainty assessment	***Benefits***	***Cost***
Minimum (p5)	67% of estimate	81% of estimate
Maximum (p95)	125% of estimate	154% of estimate
Interval width	0.54	0.69
Right-/left-skew	0.95 (weak left-skew)	1.08 (weak right-skew)

TABLE II. ALTERNATIVE UNCERTAINTY ASSESSMENT (MEDIAN VALUES)

Uncertainty assessment	*Benefits*	*Cost*
Probability of actual value less than half of estimate	30%	1%
Probability of actual value less than estimate	65%	25% (=100% - 75%)
Probability of actual value more than estimate	35% (=100% - 65%)	75%
Probability of actual value more than twice the estimate	5%	30%
Fitted minimum (p5)	22%	61%
Fitted maximum (p95)	200%	325%
Interval width	1.78	2.64
Right-/left-skew	0.89 (weak left-skew)	1.66 (strong right-skew)

Figures 3–6 display the benefits and costs uncertainty distributions of the groups. The uncertainty distributions are based on fitting the distribution to the three values p5, estimate (interpreted as the mode) and p95. The values are transformed so that: i) The value 1.0 is the estimated benefits in Figures 1 and 3, and the estimated costs in Figures 2 and 4, and ii) The values are in percentage of the estimate, e.g., the value 1.4 denotes a value 140% of the estimate. For each graph, the p5 (minimum) and p95 (maximum) values are indicated.

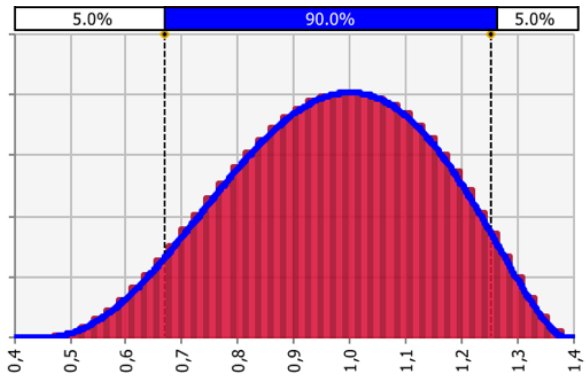

Fig. 3. Benefits distribution for traditional uncertainty assessment

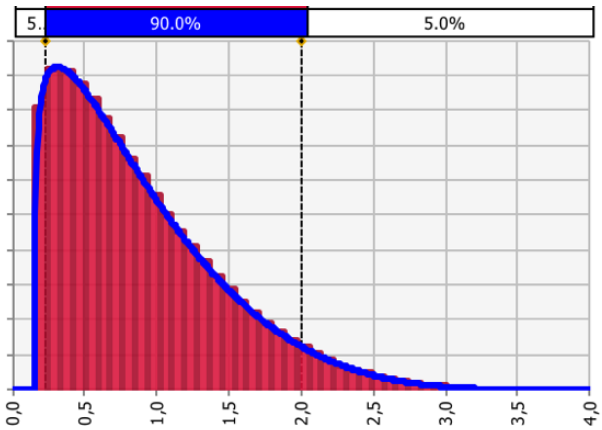

Fig. 4. Costs distribution for traditional uncertainty assessment

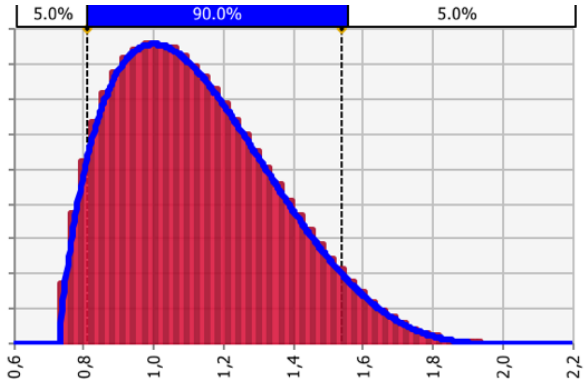

Fig. 5. Benefits distribution for alternative uncertainty assessment

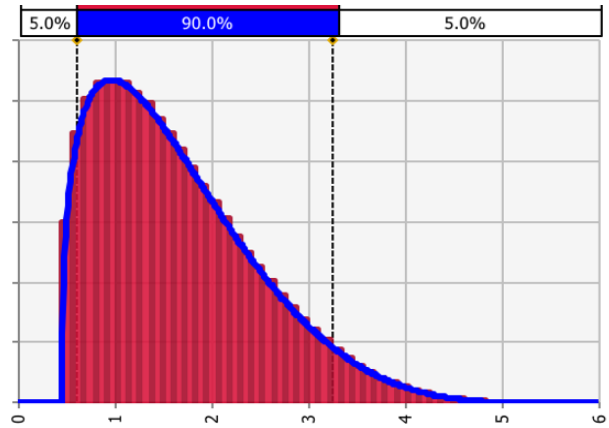

Fig. 6. Costs distribution for alternative uncertainty assessment

As can be seen from the values at the x-axes of the graphs and in the tables, there is a substantial difference in the assessed uncertainty between the traditional and the alternative method. Both H1, wider distributions, and H2, more left-skewed benefits distribution and, more right-skewed cost-distribution, when using the alternative uncertainty assessment method are consequently supported.

IV. DISCUSSION

A. *Which method led to the most realistic assessments?*

A key question is which of the uncertainty approaches led to the most realistic assessment. While, in this case, the answer to this would require that we knew the outcome of the (still on-going) project[1], we argue that there are at least two reasons to believe that the alternative approach gave the most realistic assessments:

- Looking back on previous experiences, sometimes called the use of "reference class" estimation or "analogy"-based estimation, when estimating software costs and benefits is documented to give more realism [4, 7]. While this has mainly been documented for cost estimates, we find it reasonable to assume that a

[1] In fact, we would not know the realism of the uncertainty assessments even if we knew the outcome of the project. In order to know the realism of the uncertainty assessments we would need many uncertainty assessments and actual outcomes, and to compare the confidence level or probability with the actual hit rate. After all, being 90% confident means that one will be wrong in 10% of the cases.

similar realism improvement will be present in uncertainty assessment contexts.

- Empirical data suggest that the uncertainty of a project of the type used as the case in this study is high. Software development projects in Norway were, for example, found to have an average costs overrun of 67% for projects with a public client [15]. Other surveys, for example [1], find that costs overrun distributions are strongly right-skewed, with 41% of data management projects having a costs overrun of more than 25%, and many of them 2-3 times more.

A further argument in favour of the alternative uncertainty assessment method is that the assessments were based on the respondents' actual experiences about typical estimation error and bias, e.g. how typical over-estimating the benefits and under-estimating the costs of similar projects were. The respondents were randomly divided into groups, which implies that the group using the traditional method probably is likely to have had, as a group, about the same experience regarding costs and benefit estimation error. In other words, those using the traditional method assessed the uncertainty to be much lower than what they had probably experienced in similar projects prior to this one. As far as we can judge, there was nothing in the project description that indicates a substantially lower complexity or risk of this project compared to other governmental projects of similar size and type.

B. Implications for benefits-to-costs ratio (test of H3)

An interesting implication of our results is that the benefits-to-costs ratio (return on investment) analysis including uncertainty would give very different values for the two approaches.

As a benchmark value, we start with the *non-stochastic (statistically naïve) benefits-to-cost analysis*, i.e., without taking uncertainty into consideration. Here we use the estimated benefits and costs and get a return on investment of 120 mill. Norwegian Kroner / 65 mill. Norwegian Kroner = 1.85, i.e., the expected benefits-to-costs ratio is strongly positive. Very often, as far as we have experienced, this non-stochastic value is the one used when making decisions about whether to start a software project or not.

Then we use at the uncertainty assessment of those in the traditional group, using the benefit and costs distributions based on the median assessments of p5 and p95, the PERT-distribution and a Monte Carlo simulation to simulate the ratio of benefits to costs (10,000 simulations). We then get an expected return of investment of 1.6 (see Figure 7), i.e., the expected benefits-to-costs ratio is still strongly positive, although slightly less than with the non-stochastic calculations.

Finally, we use the uncertainty assessment of those in the alternative group, using the median probability assessments (PERT-fitted p5 and p95) and simulate the benefits-to-costs ratio using Monte Carlo simulation (10,000 simulations). Now the expected benefits-to-costs ratio is as low as 1.2, see Fig8. In this case, the probability of making no profit at all is as high as 40%. In other words, using the alternative, arguably more realistic, uncertainty analysis makes it much less obvious that the project is worth starting.

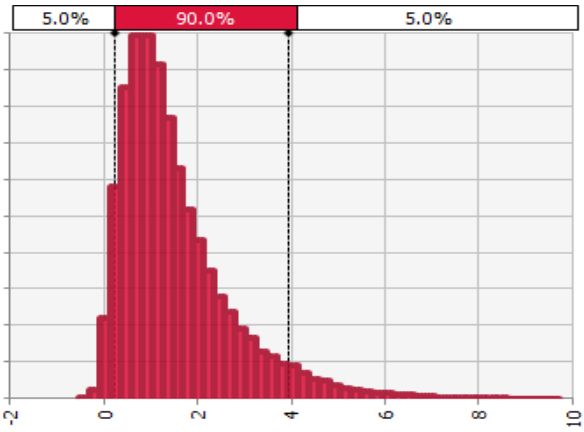

Fig. 7. Benefit / Costs – traditional uncertainty assessment

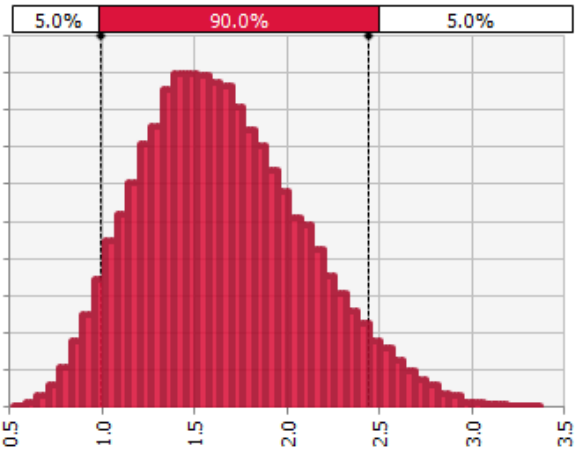

Fig. 8. Benefits / Costs – alternative uncertainty assessment

C. Limitations

There are several limitations to take into consideration when interpreting and using the results reported in this experiment. While the results are consistent with previous results on the traditional uncertainty assessment approach, i.e., that it leads to too narrow and symmetric intervals, we cannot exclude that those using the traditional intervals had the most accurate assessment of the underlying uncertainty. This can only be assessed when aggregating assessments and outcomes over many projects. What we can be confident about, however, is that those using the traditional uncertainty assessment were much more optimistic about the uncertainty than would be warranted by similar projects. We interpret this as a high likelihood of those in the traditional group being over-optimistic about the uncertainty.

The generalizability of the results to other project contexts and other software professionals is to a large extent unknown, as neither the project nor the participants were selected to represent a particular population. When looking at the roles, experience level and organizations of the participants (using the list of participants of the seminar), however, we see that they represent relevant roles. They were, with a few exceptions, software managers on the client side or project managers on the

provider side. The fact that they spent time visiting a seminar on benefits management, and had previous experience in estimating benefits and costs, indicates that they may have been more than averagely interested and, perhaps, more than average skilled in this topic.

We have assumed an underlying PERT-distribution for our analyses. The uncertainty values and results are affected by this choice. We evaluated the use of log-normal and gamma distribution, which gave similar results, i.e., there is little reason to believe that the choice of underlying uncertainty distributions had a large impact on the result.

V. Conclusions

Software professionals asked to give benefits and costs uncertainty assessments based on the estimation error they recalled having experienced on similar software projects (termed the alternative method) gave wider uncertainty intervals than those using the traditional minimum–maximum 90% confidence intervals. It also led to more left-skewed benefits distributions and more right-skewed costs distributions. The difference in assessment of benefits and costs uncertainty led to a substantial difference in the assessment of the profitability of the project, i.e., the benefits-to-costs ratios were highly favorable using the traditional method while much less so for the alternative method.

Assuming that the recalled projects were similar in terms of uncertainty to the one to be assessed, the alternative method is, we argue, likely to have led to more realistic uncertainty assessment. Previous empirical results on the use of reference-class and analogy-based, i.e., looking-back based, estimation models for software development effort, support the suggestion that looking back on previous experience-based methods leads to more realistic judgments.

We plan to conduct more studies comparing the traditional and different variants of the alternative uncertainty assessment method, where we will vary the elicitation format and, preferably, compare with the actual benefits achieved and costs spent.

References

[1] Budzier, A. and B. Flyvbjerg, *Overspend? Late? Failure? What the data say about IT project risk in the public sector.* Commonwealth Governance Handbook, 2012. **13**: p. 145-157.

[2] Connolly, T. and D. Dean, *Decomposed versus holistic estimates of effort required for software writing tasks.* Management Science, 1997. **43**(7): p. 1029-1045.

[3] Cook, J.D., *Determining distribution parameters from quantiles*. 2010, UT MD Anderson Cancer Center Department of Biostatistics Working Paper Series.

[4] Flyvbjerg, B., *Curbing optimism bias and strategic misrepresentation in planning: Reference class forecasting in practice.* European Planning Studies, 2008. **16**(1): p. 3-21.

[5] Gruschke, T.M. and M. Jorgensen, *The role of outcome feedback in improving the uncertainty assessment of software development effort estimates.* Acm Transactions on Software Engineering and Methodology, 2008. **17**(4).

[6] 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.

[7] Jørgensen, M., *Top-down and bottom-up expert estimation of software development effort.* Information and Software Technology, 2004. **46**(1): p. 3-16.

[8] Jørgensen, M., *The Ignorance of Confidence Levels in Minimum-Maximum Software Development Effort Intervals.* Lecture Notes on Software Engineering, 2014. **2**(4).

[9] 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.

[10] Jørgensen, M. and K.H. Teigen. *Uncertainty Intervals versus Interval Uncertainty: An Alternative Method for Eliciting Effort Prediction Intervals in Software Development Projects*. in *International Conference on Project Management (ProMAC)*. 2002. Singapore.

[11] 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.

[12] Kerzner, H., *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. 2003: John Wiley & Sons.

[13] Little, T., *Schedule estimation and uncertainty surrounding the cone of uncertainty.* Software, IEEE, 2006. **23**(3): p. 48-54.

[14] McKenzie, C.R.M., M. Liersch, and I. Yaniv, *Overconfidence in interval estimates: What does expertise buy you?* Organizational Behavior and Human Decision Processes, 2008. **107**: p. 179-191.

[15] Moløkken, K., M. Jørgensen, S.S. Tanilkan, H. Gallis, A.C. Lien, and S.E. Hove, *Project Estimation in the Norwegian Software Industry-A Summary*. . 2004: Simula Research Laboratory, 3.

[16] PMI, *Guide to the Project Management Body of Knowledge (PMBOK(r) Guide)-Sixth Edition*. 2017.

[17] Winman, A., P. Hanson, and P. Jusling, *Subjective probability intervals: how to reduce overconfidence by interval evaluation.* Journal of experimental psychology: learning, memory and cognition, 2004. **30**(6): p. 1167-1175.

[18] Yaniv, I. and D.P. Foster, *Precision and accuracy of judgmental estimation.* Journal of Behavioral Decision Making, 1997. **10**(1): p. 21-32.