On Prediction of Cost and Duration for Risky Software Projects Based on Risk Questionnaire

Osamu Mizuno, Takuya Adachi, Tohru Kikuno
Dept. of Informatics and Mathematical Science,
Graduate School of Engineering Science, Osaka University
{o-mizuno, adati, kikuno}@ics.es.osaka-u.ac.jp

Yasunari Takagi
Social Systems Business Company,
OMRON Corporation
taka@biwa.kusatsu.omron.co.jp

Abstract

This paper proposes a new approach that can discriminate risky software development projects from smoothly or satisfactorily going projects and give explanation for the risk. We have already developed a logistic regression model which predicts whether a project becomes risky or not. However, the model returned the decision with the calculated probability only. Additionally, a formula was constructed based on the risk questionnaire which includes 23 questions. We therefore try to improve the previous method with respect to accountability and feasibility.

In new approach, we firstly construct a new risk questionnaire including only 9 questions (or risk factors) $Q1, Q2, \cdots, Q9$, each of which concerns with the project management. We then apply multiple regression analysis to actual project data PJ1, PJ2, $\cdots$, PJ32, and clarify a set of factors which contributes essentially to estimate the relative cost error and the relative duration error, respectively. We then apply the constructed formulas to another project data PJ33, PJ34, $\cdots$, PJ40. The analysis results show that both the cost and duration of risky projects are estimated fairly well by the formulas. We can thus confirm that our new approach is applicable to software development projects in order to discriminate risky projects from appropriate projects and give reasonable explanations for the risk.

Keywords

Software risk management, Questionnaire, Cost and duration, Regression analysis

1 Introduction

Software process includes all activities and documents related to the software development. Recently, the improvement of software process, that is, software process improvement, is recognized as one of the most effective method for increasing both the quality of the products and the productivity of the projects in software development organizations (or software development companies).

Since 1993, we have been involved in software process improvement[5, 6] at a certain company (let's call it Company **A** for the sake of convenience). At Company **A**, the Software Engineering Process Group (SEPG) has been organized for eight years to promote the process improvement. We have already achieved several actual improvements[9, 10, 12, 13] in the quality and the productivity. This study is also a part of the SEPG's activities in 2000 towards the effective process management at Company **A**.

In Company **A**, many software development projects are executed every year. It was observed that some projects sometimes fall into an uncontrollable state. Such projects are called "risky projects." After repeating such uncontrollable states, risky projects occasionally go into the most dangerous state, which results in so-called "death march projects[16]." We therefore have to identify the risky projects at the early stage of the development so that they do not become death march projects.

In [10], we have proposed a logistic regression model to identify risky projects based on the questionnaire for project managers and developers. By this model, we could predict whether a project becomes risky or not, according to the probability calculated. Then, using the actual project data in Company **A**, we have empirically evaluated the effectiveness of our model. However, project managers and developers were still skeptical because no concrete proof, such as the increase of development cost and the delay of development duration of the project, is presented to them. We therefore have to establish a new method to improve the accountability for the managers and developers.

On the other hand, there were yet another problems on the questionnaire. The questionnaire used in the model was distributed to the managers of the finished projects[10]. That is, it was not considered to be applied to the on-going projects, and thus it included some kinds of questions that could not be answered by managers of the on-going project. Furthermore, there were too many questions in the questionnaire. Actually, it included 23 questions to be answered by project managers. It therefore seemed rather difficult to apply the questionnaire in [10] to an on-going project.

In order to solve the first problem (that is, the accountability to the project managers and developers), we extended our previous approach. In the new approach, we tried to estimate the relative errors between estimated and actual ones for the cost and the duration of the risky projects. Thus we obtained the relative error of the cost and the duration for a risky project as well as the probability of being risky. Relative errors of the cost and the duration are deeply related to the definition of risky projects, and thus they may explain how the project will go wrong in the future.

Next, to solve the second problem (that is, the feasibility of the questionnaire), we chose only such specific questions that are closely related to project management. As a result, we were able to reduce the number of the questions in the questionnaire, and constructed a new questionnaire which includes only 9 questions (or risk factors). We then applied the multiple regression analysis to actual project data PJ1, PJ2, $\cdots$, PJ32 and clarified a set of factors which contributes essentially to estimate the relative cost error and the relative duration error, respectively. Based on the analyses results, we actually constructed two formulas to calculate the relative cost error and the relative duration error for risky software projects.

Finally, we applied the constructed formulas to another project data PJ33, PJ34, $\cdots$, PJ40. The analysis results showed that both the cost and duration of risky projects were estimated fairly well by the formulas. We can thus confirm that our new approach can be applicable to on-going software development projects in order to discriminate risky projects and give reasonable explanations for the risk.

Now, we briefly summarize related works (especially, risk management based on the questionnaire).

For the cost estimation, Boehm originally proposed the COCOMO model[1], and then after a lot of estimation models have been proposed to obtain good software management. There were also many studies that tried to perform the risk management by the questionnaire [2, 3, 4, 7, 8, 11, 15].

For example, in the Software Engineering Institute (SEI) at Carnegie Melon University, the Software Risk Evaluation (SRE) model was developed[15, 14]. It is used to identify, analyze, communicate and mitigate the technical risks associated with the acquisition of software-intensive systems. Then the Team Risk Management (TRM) model was also developed by SEI[4]. It is a means of creating effective communication about software risk so that the risk can be abated or mitigated.

On the other hand, Jiang *et al.* tried to identify the risk factors that related to the project effectiveness by the questionnaire[7]. They tried to investigate the project effectiveness by the regression analysis based on the questionnaire. However, the project effectiveness was a subjective measure and it was also evaluated by the same person that evaluated the risks. More concrete and objective measures such as the cost, the duration and the quality should be estimated for the effective project management.

The rest of this paper is organized as follows: Section 2 shows the software development in Company **A** and background of our study. Section 3 explains the outline of our approach. The construction of the estimating formula is shown in Section 4. Section 5 shows an application of the formula to actual projects. Finally, Section 6 concludes this paper.

2 Background

2.1 Risk management in Company A

The main products of Company **A** consist of embedded software in ticket vending machines, ATMs(automated teller machines) and POS(point of sales) systems. The software process adopted in Company **A** is the waterfall model. To promote process improvement in Company **A**, the SEPG was established 7 years ago. Various activities undertaken by the SEPG, such as collecting software metrics and formalizing review activity, have been carried out. Software metrics and project data, such as productivity, quality and duration, are therefore constantly collected from all the software development projects. The SEPG recognizes and reports that the average values of both quality and productivity for all projects at Company **A** have increased steadily year by year.

However, the SEPG also observed the fact that not a few projects tend to be in a somewhat uncontrolled state for a certain period of their development, and that most of them return to a controlled state by themselves or by virtue of the senior managers' leadership. A certain proportion, however, actually collapses. While the number of such cases is quite small, they should clearly be avoided as much as possible. The necessity of the risk management thus increased in Company **A**. In this paper, we refer such projects "*risky projects*." A similar type of a project has been dubbed the *death march project*[16]. The 'death march project' describes a project which does not have sufficient project resources and thus eventually fails in the course of its development. However, as mentioned before, most risky projects to be targeted in this paper are able to be brought back under control state and finally deliver the products successfully. It is possible, however, for them to collapse temporarily at the same stage during the software development period.

2.2 Prediction of risky projects

So far, we have proposed an approach for the identification of risky projects that might cause disastrous problems during the development[10]. In order to determine key risk factors in risky projects, we designed a questionnaire with 23 questions to be distributed to project managers of the projects. Based on the responses to the questionnaire, we collected a risk assessment data and applied logistic regression analysis in order to obtain a logistic model. The model was shown to be statistically significant, and the goodness-of-fit of the model was also good. We also carried out an effectiveness analysis of the constructed model. The result showed that the constructed model can nicely predict risky projects in the new data set.

2.3 Problems to be overcome

When we try to apply the approach in [10] to on-going projects, the following problems will occur:

(1) Probability is too abstract, and no concrete explanation is given.

Using the previous model, we were able to predict whether a project becomes risky or not according to the calculated probability. However, it was difficult to explain how a project goes wrong. In other words, project managers and developers were still skeptical because no concrete explanation, such as the increase of the development cost and the delay of the duration, were presented.

(2) Too many questions make managers and developers less cooperative.

Generally speaking, a lot of questions can contribute to the accuracy of the analysis. However, the more questions are presented, the less project managers and developers are cooperative. Because they do not have enough time to answer especially during development. Actually, they often said that 23 questions are too much to answer.

Taking these problems into account, we try to establish more feasible and more accountable method, which can provide the reasonable proof with concrete measures, such as the cost and the duration of the risky project, as well as the probability of being risky.

3 Outline of Our Approach

In this section, we propose an approach to estimate the cost and the duration of software projects. Using the new approach, we can discriminate risky projects among software projects and give reasonable explanation for the risk.

3.1 New questionnaire

In order to increase feasibility of the questionnaire, we constructed a new questionnaire based on the previous one. The questionnaire used in this study is shown in Figure 1. It is intended to be applied to the on-going project management rather than the review of the project plans. So, the number of questions related to the project plans or the initial estimation is reduced so much. The reduction of questions are performed according to interviews with developers and an SEPG in the company.

The questionnaire includes 9 questions to answer. The number of questions is drastically decreased from that of the questionnaire in [10] (that was 23). Detailed description for each question is as follows:

$Q1)$ Unreasonable customers.

This item checks whether the requirements as stated by the customer are clear and consistent. It is important for the developers to understand that what the customer wants to achieve is not clear, or that the requirements as stated by the customer may not be consistent.

$Q2)$ Over-optimism in estimating the technical issues.

This item checks whether the developers have estimated technical issues with which they are not very familiar in an over-optimistic way. Usually, over-optimism regarding the estimations tends to cause underestimation. For example, the developers may not consider the need to prepare activities dealing with risks in the development.

$Q3)$ Insufficient estimations were carried out using the result of successful projects in the past.

This item checks whether the developers are estimating the current project by simply referring to similar projects from the past, which they may have developed themselves. Naturally it is advisable with regard to estimations which have provided positive experience from similar projects in the past be put to good use. But at the same time, it is very dangerous to depend on them too much with inadequate analysis.

$Q4)$ Wrong people available (lack of skills, lack of training, lack of expertise).

	Questions	Evaluation
Q1	Unreasonable customers.	
Q2	Over-optimism in estimating technical issues.	
Q3	Insufficient estimations were carried out using the results of successful projects in the past.	
Q4	Wrong people available (lack of skills, lack of training, lack of expertise).	
Q5	Unclear responsibilities and authorities.	
Q6	Low morale on the part of the developers.	
Q7	Lack of perception on the part of the managers to ensure a concerned effort.	
Q8	Requirement or specification changes were not managed sufficiently.	
Q9	Lack of progress reporting.	

Note: Please answer each question with "Strongly agree", "Agree", "Disagree" or "Strongly disagree".

Figure 1. Questionnaire for project management

This item checks whether the skill level needed for the development has been clarified and whether a sufficient number of developers with sufficient skill levels can be mustered. Needless to say, it is important to clarify the skill level needed for the development. However in some cases, such important tasks could be forgotten or neglected due to limitations of human resources.

$Q5$) Unclear responsibilities and authorities.

This item checks that the project has been systematically divided into technical activities by using the Work Breakdown Structure(WBS), and whether the responsibility for each technical activity has also been clearly specified. Unclear responsibility may prevent the early detection of serious problems concealed in the project.

$Q6$) Low morale on the part of the developers.

This item checks whether the morale of the developers is low or not. For instance, if the developer feels that someone else can deal with the delay of his/her own activity, then the developer's morale is clearly low. Low morale usually makes it difficult to detect even small problems which eventually may become very serious and cause huge delays in a project.

$Q7$) Lack of perception on the part of the managers to ensure a concerted effort.

This item checks that the developers are actually working on the assigned project. The developer may frequently be disturbed by problems deriving from the previous project on which he was engaged. The project manager should carefully manage developers so that they are not excessively distracted by problems from previous projects.

$Q8$) Requirement or specification changes were not managed sufficiently.

This item checks that the changes in requirements or specifications are appropriately managed and the developers kept informed. It is well known that uncertain changes in requirements or specifications confuse the developers and finally result in a risky project.

$Q9$) Lack of progress reporting.

This item checks whether the developers are providing the project managers with regular progress reports. In the developmental environment where a project manager criticizes an erring developer bad timing in reporting problems on the project may cause delays. In the worst case no reporting is done at all.

This questionnaire was delivered to several ongoing projects with detailed comments shown above. The managers and developers in projects recieved the questionnaire and answered them. The detailed comments were for mitigating the difference of perceptual scales among different people. Additionally, it was reported that there were less additional efforts to answer this questionnaire.

The accuracy of self-evaluation is one of the most important factors in such questionnaire based research. In the experiment to be shown in Section 4, we thought that the managers and developers answered rather honestly since the development projects had already finished. When we apply this approach to ongoing projects, we have to consider the accuracy of self-evaluation. It is one of our important future works.

3.2 Relative errors for cost and duration

Here, we introduce the following two metrics to evaluate the state of the software project:

RE_{cost}**:** It measures the relative cost error between the estimated and actual costs. This metric is formulated as follows:

$$RE_{cost} = \frac{\text{Actual cost}}{\text{Initially estimated cost}} \times 100$$

This metric is measured by percentage(%).

$RE_{duration}$**:** It measures the relative duration error between the estimated and actual durations. This metric is formulated as follows:

$RE_{duration}$ = Actual duration−Initially estimated duration

This metric is measured by month(s).

Please note that we obtain RE_{cost} and $RE_{duration}$ directly based on the data (rather than calculating them according to the formulas to be described in Section 4.).

There are several reasons for using the relative errors in this study. Firstly, these relative errors were one of the most important measures for the SEPG to judge whether a project become risky or not. Secondly, since the costs and the durations for the target projects were rather varied, it was difficult to estimate the cost and the durations themselves.

3.3 Regression models for estimation

In [10], we used the logistic regression model to predict whether a project becomes risky or not. It was formulated as follows:

$$E(Y|x_1,\cdots,x_n) = \frac{e^{b_0+b_1x_1+\cdots+b_nx_n}}{1+e^{b_0+b_1x_1+\cdots+b_nx_n}}$$

In this approach, we try to dynamically estimate the relative cost error, RE_{cost} and the relative duration error, $RE_{duration}$. For estimating RE_{cost} and $RE_{duration}$, we use the multiple regression model. It is formulated as follows:

$$Y = b_0 + b_1x_1 + b_2x_2 + \cdots + b_nx_n$$

The objective variable Y corresponds to either RE_{cost} or $RE_{duration}$. The explanatory variables $x_1,\cdots,x_n$ correspond to the risk factors $Q1,\cdots,Q9$ shown in Figure 1. (The actual values for $Q1,\cdots,Q9$ are shown in Table 1.)

On constructing regression models, we used the stepwise method to determine the coefficients in the formula. We used the actual responses from the project managers in Company **A** to determine them.

3.4 Estimation of cost and duration

Once we constructed models for the cost and the duration, we can estimate the costs and the durations for new projects. For the new projects, we also deliver the questionnaire during their development. As soon as we obtain the responses from these projects, we calculate the probabilities of being risky project and estimate RE_{cost}s and $RE_{duration}$s. Using these results, we can explain how projects go wrong with respect to the cost and the durations.

These results (that is, the estimated values, RE_{cost} and $RE_{duration}$, and the probability) will be notified to the developers as well as the project managers.

4 Experiment 1: Construction of Estimating Formula

Our proposed approach consists of two parts: construction of formula and its application to other projects. In this experiment, we show how the estimating formulas are constructed from the responses of the questionnaire.

4.1 Target projects

In this experiment, we used the actual projects from Company **A**. Forty projects conducted from 1996 to 1998 were available for this experiment[1]. These projects are for development of embedded software in ATMs(automated teller machines), ticket vending machines, and POS (Point Of Sales) systems. The size of software developed in these projects were about from 10000 LOC to 100000 LOC. Efforts needed for these projects were 10 person-months on average.

We divide these data into 2 groups: Source data group for the construction of estimating formulas, and target data group for the evaluation of applicability. For source data, we used the data of 32 projects PJ1, PJ2, $\cdots$, PJ32 in 1996 and 1997. The data of 8 projects PJ33, PJ34, $\cdots$, PJ40 in 1998 were used as the target data.

The questionnaire were delivered to project managers of these 40 projects. Table 1 shows the response of the questionnaire from 32 projects PJ1, PJ2, $\cdots$, PJ32 in 1996 and 1997. (In the following tables, the responses such as “strongly agree”, “agree”, “disagree” and “strongly disagree” are transformed into numerical values 4, 3, 2 and 1, respectively.)

The actual RE_{cost}, $RE_{duration}$ and judgment of risky project for each project are also shown in Table 1. Basically, the judgment of risky projects by the SEPG were performed using the values of RE_{cost} and $RE_{duration}$. In this study, if the RE_{cost} is larger than 120% or the $RE_{duration}$ is larger than 2 months, then we judged the project as a risky project. This condition is, however, not the only condition for the judgment. The interview with the developers and the project managers was also important for the judgment. That’s why there are some projects that were judged as no problem(risky) in spite of their extremely high(low) RE_{cost} or extremely high(low) $RE_{duration}$. For example, in some cases, the reported cost and duration are different from actual cost and duration due to the political pressure in the company.

4.2 Construction of estimating formulas

At first, we constructed a prediction formula for risky projects using the approach in [10]. By the step-wise lo-

[1]Note that these 40 projects were not all the projects in Company **A**.

Table 1. Responses from project managers

Project	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	REcost (%)	REdura. (months)	Risky / Not
PJ1	4	3	3	3	3	3	1	1	4	132.30%	2	Risky
PJ2	3	4	1	3	1	3	4	3	3	142.10%	5	Risky
PJ3	1	4	4	3	3	1	1	3	1	163.30%	10	Risky
PJ4	1	3	1	1	3	3	3	3	1	169.10%	6	NoProblem
PJ5	4	1	1	1	1	1	1	4	1	120.10%	9	Risky
PJ6	3	3	3	4	4	1	3	4	3	208.60%	1	Risky
PJ7	3	4	1	1	1	1	1	4	3	102.50%	2	Risky
PJ8	1	1	1	1	3	1	1	4	1	97.70%	0	NoProblem
PJ9	1	1	1	1	3	2	4	1	1	99.20%	1	NoProblem
PJ10	1	1	1	1	3	1	1	3	1	99.80%	-2	NoProblem
PJ11	1	1	3	1	3	1	1	1	1	100.00%	1	NoProblem
PJ12	1	1	1	1	3	1	1	3	3	100.20%	1	NoProblem
PJ13	4	3	1	3	3	1	1	4	1	117.10%	0	NoProblem
PJ14	1	4	1	1	1	1	1	4	1	109.30%	0	NoProblem
PJ15	1	1	1	1	1	1	1	1	1	70.00%	2	NoProblem
PJ16	1	3	4	3	1	1	1	1	1	98.40%	2	NoProblem
PJ17	3	4	4	4	4	3	4	3	4	108.60%	9	Risky
PJ18	3	1	1	4	3	3	1	3	3	117.50%	3	Risky
PJ19	1	1	1	4	4	1	1	1	4	64.10%	-5	Risky
PJ20	1	1	1	1	1	1	1	1	1	101.10%	0	NoProblem
PJ21	1	1	1	3	1	1	1	4	4	98.90%	0	NoProblem
PJ22	1	1	1	1	1	1	1	4	1	100.00%	0	NoProblem
PJ23	1	1	1	1	3	1	1	2	1	101.90%	0	NoProblem
PJ24	1	1	3	1	3	1	3	1	3	100.00%	0	NoProblem
PJ25	1	3	3	4	3	1	3	3	3	106.20%	0	NoProblem
PJ26	1	1	3	4	1	1	4	3	1	79.20%	0	NoProblem
PJ27	1	1	1	1	1	1	1	4	1	100.00%	0	Risky
PJ28	1	1	1	3	4	1	3	1	3	88.60%	0	NoProblem
PJ29	1	1	3	1	3	1	1	1	1	106.20%	2	NoProblem
PJ30	1	4	1	1	1	3	1	1	1	108.10%	2	NoProblem
PJ31	4	3	2	4	3	3	3	3	3	122.10%	5	NoProblem
PJ32	1	1	3	1	3	1	3	3	1	103.30%	1	NoProblem

gistic regression analysis, three risk factors were chosen to include the risky project prediction formula. Table 2 shows the selected risk factors $Q1$, $Q2$ and $Q9$, and their values of coefficients. It was confirmed that the formula is statistically significant with 0.01 level.

Table 2. Parameters for the risky project prediction

Risk factors for risky project prediction		Coefficient
	Intercept	-2.819
Q1	Unreasonable customers.	0.719
Q2	Over-optimism in estimating technical issues.	0.675
Q9	Lack of progress reporting.	0.585

Table 3 shows how the constructed formula can predict the risky projects in the source data group. It is shown that 26 ($= 18 + 8$) projects were predicted correctly.

We then constructed the estimating formula for the relative error of the development cost. Four risk factors, $Q2$, $Q3$, $Q6$ and $Q8$, shown in Table 4, were included in the estimating formula. The coefficients for these risk factors are also shown in Table 4. It was confirmed that the formula is statistically significant at 0.01 level.

Table 5 shows the relationship between the actual relative errors and the calculated relative errors for the source data group in Table 1. On the other hand, the Spearman's

Table 3. Result of self risk prediction

		Predicted	
		No problem	Risky
Actual	No problem	18	4
	Risky	2	8

* The threshold between risky and no problem is 0.3.

Table 4. Parameters for cost estimation

Risk factors for REcost		Coefficient
	Intercept	0.550
Q2	Over-optimism in estimating technical issues.	0.060
Q3	Insufficient estimations were carried out using the results of successful projects in the past.	0.059
Q6	Low morale on the part of the developers.	0.082
Q8	Requirement or specification changes were not managed sufficiently.	0.080

rank correlation coefficient was 0.744 (It was also significant at 0.01 level). This result implies that the constructed formula is suitable statistically.

Finally, we constructed the estimating formula for the relative error of the development duration. Three risk factors $Q3$, $Q6$ and $Q8$, shown in Table 6 were included in the estimating formula. The coefficients for these risk factors are also shown in Table 6. It was confirmed that the formula is statistically significant at 0.01 level.

Table 5 also shows the relationship between the actual relative errors and the calculated relative errors for the source data group. The Spearman's rank correlation coefficient was 0.596 (It was also significant at 0.01 level).

As a result, both constructed cost and duration formulas are confirmed their statistical significance. Then, we will show that these formulas can be used for prediction of the cost and duration for new projects.

5 Experiment 2: Application to Actual Projects

In the second case study, we applied the constructed formulas to the target data group, that is, 8 projects PJ33, PJ34, $\cdots$, PJ40 in 1998.

5.1 Prediction of risky projects

Table 7 shows the result of prediction for risky projects. Note that we selected the threshold of probabilities between the risky projects and no problem ones as 30%[2]. We can see that four projects, PJ33, PJ34, PJ35 and PJ40 have higher probabilities than 30%. We thus can predict these projects become risky projects.

[2]The threshold (30%) is determined based on the proportion between the number of risky and no problem projects.

Table 5. Relationships for RE_{cost} and $RE_{duration}$

	REcost			REduration		
	Calculated	Rank	Actual	Calculated	Rank	Actual
PJ1	123.3%	9	132.3%	4.71	3	2
PJ2	133.5%	3	142.1%	4.11	4	5
PJ3	134.8%	2	163.3%	3.68	7	10
PJ4	127.5%	6	169.1%	4.11	4	6
PJ5	107.1%	17	120.1%	1.16	14	9
PJ6	130.9%	5	208.6%	3.38	8	1
PJ7	125.1%	7	102.5%	1.16	14	2
PJ8	107.1%	17	97.7%	1.16	14	0
PJ9	91.3%	27	99.2%	0.61	25	1
PJ10	99.1%	22	99.8%	0.35	26	-2
PJ11	94.9%	24	100.0%	0.95	22	1
PJ12	99.1%	22	100.2%	0.35	26	1
PJ13	119.1%	11	117.1%	1.16	14	0
PJ14	125.1%	7	109.3%	1.16	14	0
PJ15	83.1%	29	70.0%	-1.27	29	2
PJ16	112.8%	14	98.4%	2.06	13	2
PJ17	151.2%	1	108.6%	7.44	1	9
PJ18	115.5%	13	117.5%	4.11	4	3
PJ19	83.1%	29	64.1%	-1.27	29	-5
PJ20	83.1%	29	101.1%	-1.27	29	0
PJ21	107.1%	17	98.9%	1.16	14	0
PJ22	107.1%	17	100.0%	1.16	14	0
PJ23	91.1%	28	101.9%	-0.46	28	0
PJ24	94.9%	24	100.0%	0.95	22	0
PJ25	122.9%	10	106.2%	2.57	9	0
PJ26	110.9%	15	79.2%	2.57	9	0
PJ27	107.1%	17	100.0%	1.16	14	0
PJ28	83.1%	29	88.6%	-1.27	29	0
PJ29	94.9%	24	106.2%	0.95	22	2
PJ30	117.5%	12	108.1%	2.49	12	2
PJ31	133.4%	4	122.1%	5.22	2	5
PJ32	110.9%	15	103.3%	2.57	9	1

Table 6. Parameters for duration estimation

	Risk factors for REduration	Coefficient
	Intercept	-5.070
Q3	Insufficient estimations were carried out using the results of successful projects in the past.	1.110
Q6	Low morale on the part of the developers.	1.880
Q8	Requirement or specification changes were not managed sufficiently.	0.810

The actual evaluations done by the SEPG for these 8 projects are also shown in Table 7. We can see that the projects PJ33, PJ34 and PJ35 were actually the risky projects. So, we can say that 7 projects out of 8 were predicted correctly by the constructed prediction formula.

5.2 Estimation of development cost

Table 8 shows the result of estimation for RE_{cost}. The values of RE_{cost}s for these projects were, however, not so close to the actual RE_{cost}s. As a whole, the estimated RE_{cost}s tend to be larger than the actual RE_{cost}s. For 4 projects (PJ33, PJ35, PJ37 and PJ39) out of 8 projects, the differences between estimated and actual RE_{cost}s were less than 10%.

Table 7. Result of risk prediction

Projects in 1998	Q1	Q2	Q9	Prediction of being risky	Actual evaluation
PJ33	3	1	3	44.7%	Risky
PJ34	3	2	1	33.1%	Risky
PJ35	4	4	4	95.8%	Risky
PJ36	1	1	1	5.6%	NoProblem
PJ37	1	3	1	18.7%	NoProblem
PJ38	3	1	1	20.1%	NoProblem
PJ39	1	1	1	5.6%	NoProblem
PJ40	3	3	1	49.2%	NoProblem

Table 8. Result of cost estimation

Projects in 1998	Q2	Q3	Q6	Q8	Calculated REcost (%)	Actual REcost (%)
PJ33	1	1	1	4	107.3%	105.0%
PJ34	2	2	4	3	135.8%	106.9%
PJ35	4	3	1	2	129.1%	119.5%
PJ36	1	1	1	2	91.3%	101.3%
PJ37	3	1	1	3	111.3%	103.9%
PJ38	1	3	1	4	119.1%	100.0%
PJ39	1	1	1	1	83.3%	83.0%
PJ40	3	3	1	1	107.1%	89.6%

5.3 Estimation of development duration

Table 9 shows the result of estimation for $RE_{duration}$. This result does not show so accurate results, either. As a whole, the estimated $RE_{duration}$s tend to be smaller than the actual $RE_{duration}$s. For 3 projects (PJ35, PJ36 and PJ37) out of 8 projects, the differences between estimated and actual $RE_{duration}$ were less than 1 month.

Table 9. Result of duration estimation

Projects in 1998	Q3	Q6	Q8	Calculated REdura. (months)	Actual REdura. (months)
PJ33	1	1	4	1.17	4.00
PJ34	2	4	3	7.10	3.00
PJ35	3	1	2	2.57	3.00
PJ36	1	1	2	-0.46	0.00
PJ37	1	1	3	0.36	1.00
PJ38	3	1	4	3.39	0.00
PJ39	1	1	1	-1.27	2.00
PJ40	3	1	1	0.95	2.00

5.4 Discussion

Table 10 shows the comparison of above prediction and estimations. In this experiment, the ranks of cost and duration estimations among 40 projects from 1996 to 1998 are meaningful to see how the cost and/or the duration will exceed from initial estimation.

For example, $PJ34$ has 2nd rank for RE_{cost} on 40 projects. This suggests that $PJ34$ may cause some problem on development cost and will exceeds its estimation with rather high degree.

Table 10. Comparison of estimations

Projects in 1998	Predicted as risky (prob. > 30%)	Calculated REcost (Rank on 40 projects)	Calculated REdura. (Rank on 40 projects)	Actually Risky
PJ33	v	21	17	v
PJ34	v	2	2	v
PJ35	v	7	11	v
PJ36		33	35	
PJ37		18	31	
PJ38		13	9	
PJ39		36	36	
PJ40	v	22	26	

Here we investigate the cost and the duration for projects, which were predicted as risky (that is, PJ33, PJ34, PJ35 and PJ40).

- For PJ33, the estimated RE_{cost} and $RE_{duration}$ were ranked as the 21st and the 17th, respectively. We cannot find any reason of being risky from the cost and the duration. However, from the project management, we should not consider that the prediction was wrong. This result implies that there may exist other risk factors which affect the cost or the duration.

- For PJ34 and PJ35, both the estimated RE_{cost} and $RE_{duration}$ were ranked rather high (the 2nd and the 2nd, respectively, for PJ34 and for PJ35, the 7th and the 11th, respectively.). We can thus have strong confidence on the prediction. We can also make reasonable explanation: how the cost and the duration exceed and which risk factors are related to, for the developers of PJ34 and PJ35.

- Finally, for PJ40, the estimated RE_{cost} and $RE_{duration}$ were the 22nd and the 26th, respectively. We cannot find any reason of being a risky project from the cost and the duration, either.

As a result, PJ33, PJ34 and PJ35 were actually risky projects due to their large $RE_{duration}$s (See Table9). We can say, at least, that estimated $RE_{duration}$s for PJ34 and PJ35 indicated clearly the excess of the durations.

Thus unfortunately, the proposed estimating approaches for the cost and the duration do not have absolutely high level of accuracy. However, we can see that the results of estimations can be used sufficiently to give approximate explanations for those risky projects.

6 Conclusion

In this paper, we have proposed a new prediction method for risky software projects, which gives a reasonable explanation and the probability of being risky. This reasonable explanation consists of the relative errors RE_{cost} and $RE_{duration}$, for the cost and the duration, respectively. The proposed method attained the feasibility of the risk questionnaire by reducing the number of questions to nine. The result of the experimental evaluation showed that the estimations of cost RE_{cost} and duration $RE_{duration}$ were useful to explain the reasons of risky projects.

At this point, we have not yet evaluated whether this method is applicable to brand-new projects. Since it is based on the statistical analysis of previous data, Applying to new projects seems to be difficult. Investigations on its limitation and application areas are still remained as future works.

We are now planning to apply the new prediction formula to more projects in Company **A** in order to increase the accuracy of the formula. Furthermore, we have to establish a new control method, by which risky projects will be improved and finally get away from becoming confused projects.

Acknowledgments

Authors would like to thank Associate Prof. Shinji Kusumoto of Osaka University for his discussions and advice to our analysis in this paper. We also would like to thank Dr. Keishi Sakamoto for his cooperation in our empirical studies.

References

[1] B. W. Boehm. *Software Engineering Economics*. Prentice-Hall, 1981.
[2] E. H. Conrow and P. S. Shishido. Implementing risk management on software intensive projects. *IEEE Software*, 14(3):83–89, 1997.
[3] R. Fairley and P. Rook. Risk management for software development. In *Software Engineering*, pages 387–400. IEEE CS Press, 1997.
[4] R. P. Higuera, A. J. Dorofee, J. A. Walker, and R. C. Williams. Team risk management: A new model for customer-supplier relationships. Technical Report CMU/SEI-94-SR-5, Software Engineering Institute, 1994.
[5] W. S. Humphrey. *Managing the Software Process*. Addison-Wesley, MA, 1989.
[6] W. S. Humphrey. *A Discipline for Software Engineering*. Addison-Wesley, MA, 1995.
[7] J. Jiang and G. Klein. Software development risks to project effectiveness. *Journal of Systems and Software*, 52:3–10, 2000.
[8] D. W. Karolak. *Software Engineering Risk Management*. IEEE CS Press, CA, 1996.
[9] O. Mizuno, T. Kikuno, K. I. Y. Takagi, and K. Sakamoto. Analyzing effects of cost estimation accuracy on quality and productivity. In *Proc. of 20th International Conference on Software Engineering*, pages 410–419, 1998.

[10] O. Mizuno, T. Kikuno, Y. Takagi, and K. Sakamoto. Characterization of risky projects based on project managers' evaluation. In *Proc. of 22nd International Conference on Software Engineering*, pages 387–395, 2000.

[11] F. J. Sisti and S. Joseph. Software risk evaluation method version 1.0. Technical Report CMU/SEI-94-TR-19, Software Engineering Institute, 1994.

[12] Y. Takagi, T. Tanaka, N. Niihara, K. Sakamoto, S. Kusumoto, and T. Kikuno. Analysis of review's effectiveness based on software metrics. In *Proc. of 5th International Symposium on Software Reliability Engineering*, pages 34–39, 1995.

[13] T. Tanaka, K. Sakamoto, S. Kusumoto, K. Matsumoto, and T. Kikuno. Improvement of software process by process description and benefit estimation. In *Proc. of 17th International Conference on Software Engineering*, pages 123–132, 1995.

[14] R. C. Williams, G. J. Pandelios, and S. G. Behrens. Software risk evaluation (sre) method description (version 2.0). Technical Report CMU/SEI-99-TR-029, Software Engineering Institute, 1999.

[15] R. C. Williams, J. A. Walker, and A. J. Dorofee. Putting risk management into practice. *IEEE Software*, 14(3):75–82, 1997.

[16] E. Yourdon. *Death March : The Complete Software Developer's Guide to Surviving 'Mission Impossible' Projects*. Prentice-Hall Computer Books, 1997.