Portfolio Management of Software Development Projects Using COCOMO II

Wiboon Jiamthubthugsin Department of Computer Engineering Faculty of Engineering Chulalongkorn University Bangkok 10330 Thailand +66-(0)2-218-6956

wiboon@gmail.com

ABSTRACT

Software development projects are subject to external and internal risks that cause delays, budget overrun and poor quality. Portfolio management can be used to alleviate this problem, as it pools resources together and allows for resource sharing among projects. Consequently, projects are more likely to succeed. However, portfolio management using only deadlines and the number of employees to improve probability of success is still confined. This paper proposes integrating portfolio management with COCOMO II that offers more management flexibility. Managers can adjust other resources, such as tools, staff capability, communication support, etc. to improve the project's success. The proposed method can also be applied despite limited historical data and expert judgment. In addition, this paper introduces time constraints into portfolio management without assuming unrealistic linearity between effort and time.

Categories and Subject Descriptors

D.2 [Software]: Software Engineering D.2.9 [Software Engineering]: Management

General Terms

Management

Keywords

Portfolio Management, COCOMO II, Software Project Risk, Software Project, Software Development

1. INTRODUCTION

Portfolio management of software development projects means management based on an overall performance of the project set. To decide whether to bid for a contract, software cost estimation can provide a condition in a portfolio management. Specifically, available resources should meet estimated resources of the bidden project.

Generally, the approaches for cost estimation can be categorized into three types: expert judgment, algorithmic models and machine learning [12, 14]. The disadvantages of expert judgment

Copyright is held by the author/owner(s). *ICSE'06*, May 20-28, 2006, Shanghai, China. ACM 1-59593-085-X/06/0005. Daricha Sutivong Department of Computer Engineering Faculty of Engineering Chulalongkorn University Bangkok 10330 Thailand +66-(0)2-218-6993

daricha.s@chula.ac.th

are unrepeatability and dependency on experts' quality [7]. Machine learning requires a lot of data for training. When lacking data and experts, most people are likely to use algorithmic models. Hence, this paper proposes using COCOMO II, which is a standard algorithmic cost estimation model that consists of many relevant factors in practice.

Since estimation is carried out in an early stage with limited data, it is difficult to forecast a single correct value. Therefore, a range or a probability distribution is normally preferred. An approach of using probability distribution to handle an estimation error is studied by Fewster [8]. The approach takes various probability distributions to represent effort uncertainty based on historical data or experts.

Portfolio management proposed by Fewster focuses on adjusting only deadlines and the number of employees. However, effective project management should focus on people, product, process and project [13], not only one. Therefore, this paper proposes adjusting other factors besides deadlines and the number of employees, e.g. supportive tool, people quality, communication support, etc. to improve the probability of project success. The approach integrates the COCOMO II cost model with portfolio management of projects in order to adjust various factors of the projects in the portfolio. This enables a project manager to manage over various parameters besides deadlines and the number of employees.

In addition, since portfolio management increases flexibility in reducing project risks, experiments are performed to confirm this assumption. The results can guide managers in adopting a portfolio viewpoint in their project management.

Section 2 summarizes background knowledge relevant to this study. Section 3 describes our proposed approach. The experiments and results are presented in section 4. Finally, section 5 offers concluding remarks.

2. BACKGROUND

This section describes principles and studies relevant to the proposed approach: COCOMO II, NOSTROMO and portfolio management by deadline planning.

2.1 COCOMO II

COCOMO II is an objective cost model for planning in software project [1, 2]. Post-architecture COCOMO, a process model, consists of 5 scale factors and 17 cost drivers. Scale factors

account for relative economies or diseconomies of scale encountered in software projects of different sizes. Cost drivers are adjustment to the estimated effort. Degrees of scale factors and cost drivers include extra low, very low, low, nominal, high, very high and extra high. With user's assessment on scale factors, cost drivers and software size, COCOMO II provides effort and time estimations, as shown in (1) and (2) respectively.

$$Effort = A \times Size^{E} \times \prod_{i=1}^{17} EM_{i}$$
⁽¹⁾

$$Time = C \times (Effort)^F$$
(2)

Where E and F can be calculated from (3) and (4).

$$E = B + 0.01 \times \sum_{j=1}^{5} SF_j$$
 (3)

$$F = D + 0.2 \times (E - B) \tag{4}$$

Size is represented by KLOC. *A*, *B*, *EM* (effort multipliers for cost drivers) and *SF* (scale factors) are calibrated with effort data of real projects. *C* and *D* are calibrated with schedule data of real projects.

2.2 NOSTROMO

NOSTROMO (National Obscurity Statistical Risk Observation Model) is the methodology using Monte Carlo simulation to model uncertainty risk in COCOMO II [11]. It models uncertainties of 31 factors in COCOMO II, such as 17 cost drivers, 5 scale factors, software size, etc.

NOSTROMO assumes that uncertainty degrees of these factors can be categorized into 4 levels — level 1 for no uncertainty, level 2 for low uncertainty, level 3 for medium uncertainty and level 4 for high uncertainty. These uncertainties can be characterized by a probability distribution – none for no uncertainty, normal for low uncertainty, triangular for medium uncertainty and uniform for high uncertainty. In addition, NOSTROMO defines a range of scale factors and cost drivers based on its uncertainty, except for a software size which is based on the development phase [3].

After defining uncertainty of factors in terms of probability distribution, Monte Carlo simulation is used to generate a set of possible estimated values, such as effort and time, from each uncertain factor.

2.3 Portfolio management method by deadline

Fewster [8] proposed a portfolio management method that uses deadlines and effort probability model. The effort estimation model provides the probability of success with the proposed deadlines and a set of new deadlines that deliver a required probability of success. The conditions used to determine portfolio success are that project 1 must be finished by deadline 1 and that project 1 and 2 must be finished by deadline 2 and so forth, given that deadline 1<deadline 2, etc. In terms of effort, if there are *E* employees, the total available effort (person-day) from start date to deadline of *d* days, is $E \times d$. Thus, the condition for *n* projects is represented as (5).

$$\begin{pmatrix} Y_{1} \\ Y_{1} + Y_{2} \\ \vdots \\ Y_{1} + Y_{2} + \dots + Y_{n} \end{pmatrix} \leq E \begin{pmatrix} d_{1} \\ d_{2} \\ \vdots \\ d_{n} \end{pmatrix}$$
(5)

 Y_i represents the required estimated effort to succeed in project *i*. If $Y_1 + ... + Y_j > Ed_j$ for any *j*, then deadline d_j has failed. Probability of success is obtained from simulation. Specifically, $Y_1, ..., Y_n$ are generated from a predetermined probability distribution. If $Y_1, ..., Y_n$ satisfy all conditions, then the set of *Y* succeeds. Probability of success of a portfolio equals to the ratio of the number of the set *Y* successes to the number of trials in the simulation.

3. APPROACH

This section describes the proposed approach for portfolio management of software projects.

3.1 Effort and time distributions

Effort and time distributions are used to realize probability of project success. This paper proposes constructing the distributions based on NOSTROMO, but only focuses on uncertainties from the main 23 factors in COCOMO II, namely 5 scale drivers, 17 cost drivers and software size.

Uncertainty of each factor is represented with a probability distribution. To define a probability distribution, its type and parameters must be identified. Firstly, a distribution type of each factor is assigned according to the uncertainty level based on NOSTROMO: normal, triangular and uniform for low, medium and high uncertainty levels respectively. Secondly, parameters of these probability distributions, namely center, min and max, are specified. These parameters are derived from the factor's rating and its error, evaluated by the user. Finally, after determining all distributions of cost drivers, scale factors and size, Monte Carlo simulation is performed to yield effort and time estimations.

As an example, assume that the factor considered is a cost driver TOOL. If uncertainty level is medium, its probability distribution type is triangular according to NOSTROMO. Assume that the user rates the TOOL level as low and error of 1. The center will be low while and min and max are very low (low-1) and nominal (low+1) respectively. The steps are repeated for all factors before the simulation begins

3.2 Portfolio management under effort constraints vs. time constraints

The equation (5) converted to person-month represents effort constraints. Since COCOMO II also provides time estimation as in (2), a condition for portfolio management in terms of time constraints can be constructed as in (6).

$$\begin{pmatrix} T_1 \\ T_1 + T_2 \\ \vdots \\ T_1 + T_2 + \dots + T_n \end{pmatrix} \leq \begin{pmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{pmatrix}$$
(6)

 T_i is an estimated development time (month) of project *i*. The time constraints in (6) are compared with effort constraints in (5). Note that the number of employees (*E*) is not in (6), as COCOMO II assumes an average number of employees [4], which is implied by (2).

3.3 Adjustment of project factors

This paper proposes adjusting other factors, besides deadlines and the number of employees, to improve the probability of project success. These factors are cost drivers and scale factors of COCOMO II, which represent resources in software projects. Since adjustment should affect probability of success, $p_1 \neq p_2$ when p_1 and p_2 are probability of success for projects with different factor ratings.

3.4 Portfolio benefits vs. uncertainty levels

Managing projects as a portfolio offers more flexibility to handle uncertainty and risks than managing each project individually. Therefore, portfolio management should be especially beneficial when uncertainties are high. Experiments are performed to illustrate the benefit in terms of improvement in probability of success. Specifically, projects with high uncertainty should gain more benefit (in terms of probability of success: p) from portfolio management than projects with low uncertainty, or *ratio_h* should be more than *ratio_l*, as shown in table 1.

Probability of success	Individual projects	Portfolio	Benefit
Low uncertainty	p_{il}	p_{pl}	$Ratio_l = p_{pl}/p_{il}$
High uncertainty	p_{ih}	p_{ph}	$Ratio_h = p_{ph}/p_{ih}$

4. EXPERIMENTS AND RESULTS

This section explains experiments in details and their results.

4.1 Experimental data

We use the data of software projects from Burapasayan [5], which employed COCOMO II to estimate software project costs for Progress Software Co.Ltd. The data consist of eight projects, which form a portfolio for our test case. Each project contains information on ratings of scale drivers, cost factors, software size (KLOC), and the start and finish dates of the projects.

4.2 Results of portfolio management under effort constraints vs. time constraints

Since the data only contain factor ratings but not the uncertainty levels of these rating, the experiment uses the default levels of uncertainty as proposed by McDonald [10]. For example, the default levels of RESL, DATA, ACAP and PCAP are high.

For the probability distribution of size, Madachy [9] proposed various distributions, such as triangular, normal, lognormal and truncated lognormal. Chulani [6] studied the size distribution and noticed that it has a positive skew and that log(*size*) is likely to be a normal distribution. Therefore, the experiment uses a lognormal distribution for software size.

After determining all probability distributions, Monte Carlo simulation is employed to generate the samples of effort (personmonth) and time (month), as shown in figure 1 and 2.



Figure 1. Probability distribution of effort (person-month).



Figure 2. Probability distribution of time (month).

Assume *E* to be three (the number of employees in each project), the results show that probability of success is 99.93% under effort constraints and 90.54% under time constraints.

Note that the probability of success under time constraints is less than that under effort constraints. This results from the linear relationship between time and effort assumed under effort constraints (effort $\leq E \times$ time in (5)), while the relationship under time constraints is not assumed to be linear, as in (2). For example, suppose that a project requires 120 person-months of effort. Under effort constraints, 10 persons must work for 12 months, while under time constraints, the project needs longer than 12 months [4]. Thus, the longer time required under time constraints leads to a lower probability of success.

4.3 Results of adjustment of project factors

TOOL is chosen for this experiment as an example of a factor that can be adjusted in practice. Two portfolios of projects are constructed: the first one with TOOL rated as very high for all projects and the second one with TOOL rated as very low for all projects. The experimental results are shown in table 2.

 Table 2. Probability of success for TOOL under effort constraints and time constraints

TOOL	Probability of success		
rating	effort constraints	time constraints	
Very low	98.80%	82.71%	
Very high	99.97%	94.84%	

When an effective tool is used in software projects (TOOL is rated very high), the probability of success is higher under both effort constraints and time constraints. Other factors in the COCOMO II model, such as tools, staff capability, communication support, etc., can also be similarly adjusted to improve probability of success. Tradeoffs among various alternative adjustments can also be analyzed to aid project manager's decision.

4.4 Results of portfolio benefits

To study benefits of portfolio management as a function of uncertainty levels, two portfolios are constructed. Probability distributions for all factors in the first and second portfolios are defined as normal and uniform respectively (for low and high uncertainties). The benefit results of low and high uncertainties under effort constraints and under time constraints are shown in table 3 and 4 respectively.

Table 3. Benefits of low and high uncertainties using effort constraints

Uncontainty laval	Probability of s	Benefit	
Uncertainty level	Individual projects	Portfolio	
Low Uncertainty	95.65%	99.72%	1.04
High Uncertainty	91.08%	99.21%	1.09

 Table 4. Benefits of low and high uncertainties using time constraints

Uncontainty laval	Probability of s	Benefit	
Uncertainty level	Individual projects	Portfolio	
Low Uncertainty	10.99%	90.34%	8.22
High Uncertainty	5.42%	81.28%	15

The results show that high-uncertainty projects gain more benefit from portfolio management than low-uncertainty ones. Therefore, portfolio management can be a useful methodology for managers to handle highly uncertain or risky software projects effectively.

5. CONCLUSION

This paper proposes an integration of COCOMO II, NOSTROMO model and portfolio management of software projects. The concept allows us to construct a probability distribution when historical data and expert judgments are unavailable. Moreover, COCOMO II allows us to introduce time constraints into portfolio management without assuming unrealistic linearity between time and effort. Since COCOMO II consists of many parameters of software projects for cost prediction, these factors, in addition to deadlines and the number of employees, can be utilized for managerial adjustment to improve probability of success. These parameters represent practical alternatives for project managers to trade off among possible resources. Finally, experimental results illustrate that portfolio management can be an effective tool in handling software projects, especially those with high uncertainty.

6. REFERENCES

- Boehm, B., Clark, B., Horowitz, E. and Westland, C. Cost Models for Future SoftwareLife Cycle Processes: COCOMO 2.0. Annals of Software Engineering, 1, 57-94.
- [2] Boehm, B.W. *Software cost estimation with Cocomo II*. Prentice Hall, Upper Saddle River, NJ, 2000.
- [3] Boehm, B.W. *Software engineering economics*. Prentice-Hall, Englewood Cliffs, N.J., 1981.
- [4] Braude, E.J. Software engineering : an object-oriented perspective. J. Wiley, New York, 2001.
- [5] Burapasayan, K. Software Cost Estimation Model for Software Development Project. Master Project, Department of Computer Engineering, Chulalongkorn University, Bangkok, 2004.
- [6] Chulani, S., Boehm, B. and Steece, B. Bayesian analysis of empirical software engineering cost models. *IEEE Transactions on Software Engineering*, 25 (4), 573-583.
- [7] Fenton, N.E. and Pfleeger, S.L. Software metrics : a rigorous and practical approach. International Thomson Computer Press; PWS Publishing Co., Boston, 1997.
- [8] Fewster, R.M. and Mendes, E., Portfolio management method for deadline planning. In *Proceedings of Ninth International Software Metrics Symposium*, 2003, 325-336.
- [9] Madachy, R. and Stutzke, R., Use of Cost Models in Risk Management. In 17th International Forum on COCOMO and Software Cost Modeling, (Los Angeles, CA, 2002).
- [10] McDonald, P., Giles, S. and Strickland, D., Extensions of Auto-Generated Code and NOSTROMO Methodologies. In 19th International Forum on COCOMO and Software Cost Modeling, (Los Angeles, CA, 2004).
- [11] McDonald, S., and Wildman, NOSTROMO: Using Monte Carlo Simulation to Model Uncertainty Risk in COCOMO II (U). In 18th International Forum on COCOMO and Software Cost Modeling, (Los Angeles, 2003).
- [12] Mendes, E., Mosley, N., and Watson, I., A Comparison of Case-based Reasoning Approaches to Web Hypermedia Project Cost Estimation. In *Proceedings 11th International World-Wide Web Conference*, (Hawaii, 2002).
- [13] Pressman, R.S. SOFTWARE ENGINEERING : A Practitioner's Approach. McGraw-Hill, 2005.
- [14] Shepperd, M., Schofield, C. and Kitchenham, B., Effort estimation using analogy. In *Proceedings of the 18th International Conference on Software Engineering*, 1996, 170-178.