



## ANALYSIS OF THE CAUSES AND LEVEL OF MAINTENANCE FOR ENTERPRISE SYSTEMS IN CONSTRUCTION COMPANIES

Chijoo LEE<sup>1\*</sup>, Chiheyon LEE<sup>2</sup>, Eul-Bum LEE<sup>3</sup>

<sup>1</sup>*Department of Architectural Engineering, Yonsei University, Seoul, South Korea*

<sup>2</sup>*ALUX Company Limited, Seoul, South Korea*

<sup>3</sup>*Graduate Institute of Ferrous Technology & Graduate School of Engineering Mastership, Pohang University of Science and Technology, Pohang, South Korea*

Received 13 May 2018; accepted 08 August 2018

**Abstract.** A construction company without a similar information technology (IT) system in the past has insufficient historical data to use for investment decision-making of IT system. An estimation of maintenance costs is especially more uncertain than the initial investment costs, and the uncertainty is greater when the IT system is used over a long period, such as an enterprise system (ES). This study proposes estimation criteria for the maintenance costs of an ES for an accurate investment decision. First, the causes of maintenance are determined, and the level of maintenance analyzed. Then, the result is compared with a general trend of maintenance incidence (bathtub curve) that is widely used as reference criteria to estimate maintenance cost. The level of maintenance was high during the early stage but steadily decreased in the middle and end stage because high-cost maintenance activities were postponed with the approach of the time in which the ES was being restructured. This trend is different from the bathtub curve that increases again during the end stage. Thus, when a maintenance contract is negotiated, the maintenance level that affects maintenance cost should be considered as well as the incidence of maintenance.

**Keywords:** maintenance cost, estimation criteria, service level agreement, fuzzy inference.

### Introduction

A company that has not implemented a similar information technology (IT) system has insufficient reference data about required costs for making investment decisions. The problem is especially prevalent in the construction industry because it has implemented IT systems later than other industries such as manufacturing industry. Thus, an investment risk for construction companies is larger if their initial investment costs are larger, such as implementation of an Enterprise System (ES). The ES such as Enterprise Resource Planning (ERP) and Customer Relationship Management (CRM) support several services, including reporting, data analytics, and analysis of information flow for planning, marketing, accounting (Romero, Vernadat 2016). To correct faults, improve speed, and adapt to environment, the ES are modified after installation, the ES entails a high maintenance cost as well as initial investment (IEEE 1219-1998; Li *et al.* 2014). The maintenance cost may exceed the initial investment because the life-cycle of the ES is long (Nguyen *et al.* 2011).

Generally, the contract specifies the criteria by which initial investment and license costs are estimated. However, it has been typically determined to be within 10–15% of the initial investment (Ministry of Knowledge Economy 2011) based on external factors, such as the reputation of an IT company, although the maintenance costs are high. In order to accurately conduct a feasibility analysis on whether to implement an ES, factors that affect maintenance costs should be considered. This study proposes estimation criteria of maintenance costs during the ES life-cycle for improving investment feasibility accuracy.

As the reference criterion, the cost of maintenance during lifecycle can be estimated using the bathtub curve on the basis of an failure incidence (Jiang 2013). This curve classifies the general trend of maintenance incidence into three stages: 1) failure is high during the initial stage; 2) low during the intermediate stage; and 3) increases during the late stages due to wear-out (Figure 1). In addition to incidences of maintenance, maintenance cost is also affected

\*Corresponding author. E-mail: [news3749@gmail.com](mailto:news3749@gmail.com)

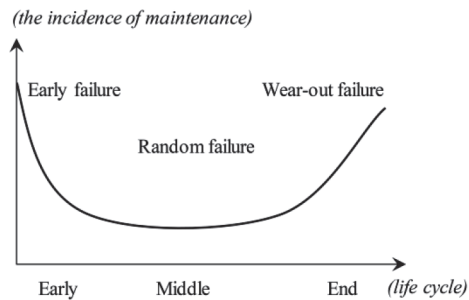


Figure 1. Bathtub curve that describes the change over time (x axis) in maintenance costs (y axis) in an ES

by the causes, as the causes affect the level of maintenance. Thus, the proposed estimation criteria of maintenance costs in this study are causes and level of maintenance.

To provide reference criteria for the estimation of maintenance costs, this study identified the causes of maintenance based on the following: (a) service level agreement (SLA); (b) previous studies (April 2010; IEEE 1219-1998; ISO/IEC 14764 2006); and (c) an interview of an IT consult with 15-years' experience with construction and ES operation. The SLA is an agreed level of maintenance between a construction company and an IT company, and the construction company pays the maintenance cost based on the SLA (Ranaldo, Zimeo 2016; Ul Haq et al. 2011).

The level of maintenance is analyzed based on urgency, difficulty, and damage determined by an interview with the IT consult. Urgency is the priority of maintenance to achieve the goal of the ES. Difficulty represents the complexity of maintenance. The maintainer's status (elementary, intermediate, and advanced) varies with difficulty. Damage represents the effort required for maintenance; e.g. that an advanced maintainer must work for 5 h. For analysis, the lifecycle of the ES was divided into early, middle and end stages (Unger et al. 2009), then this study surveyed 30 IT consultants who worked on ES operation to identify factors that affect maintenance cost. Fuzzy inference was used to mitigate the problem of ambiguous language used to express these opinions (Mendes et al. 2011; Zhang et al. 2014). The maintenance levels were analyzed and the result was compared with the bathtub curve (Figure 1).

As described earlier, the objective of this study is to propose estimation criteria for the maintenance costs of an ES that will be operating for a long time. The results of this study are applicable as reference data for cost estimation when feasibility analysis is conducted to determine whether to implement an ES. The results are useful for both construction and IT companies during negotiation of a fair maintenance contract.

## 1. Literature review

Previous studies identified factors that may affect maintenance cost (Buchmann et al. 2011; Correa et al. 2016; Li et al. 2010; Nguyen et al. 2011; Ren et al. 2011a; Tsu-

noda et al. 2015) and provided methods to estimate the development cost of software (April 2010; Matijevic et al. 2012; Ng, Gable 2010; Ren et al. 2011b; Salmeron, Lopez 2012; Sarno et al. 2015). Several previous studies have proposed a method for the successful implementation of an ES, though these are not directly related to maintenance (Hadidi et al. 2017; Niu et al. 2013, 2014; Soja et al. 2015; Wu et al. 2009; Xu 2011). However, the previous studies did not analyze the causes of maintenance or provide reference criteria for estimation of maintenance costs.

Among the previous studies that have analyzed influence factors on maintenance costs, Buchmann et al. (2011) proposed a method to estimate realistic maintenance costs based on less effort and cost required using a multilevel approach. Correa et al. (2016) derived the factors related to maintenance and supporting costs. Li et al. (2010) analyzed influence factors on corrective maintenance activities and described and analyzed the defect type that causes high maintenance costs based on interviews with IT system developers. Nguyen et al. (2011) proposed a method to analyze size and effort by maintenance types and to predict programmer effort for maintenance. Ren et al. (2011a) analyzed the impact of maintenance types and technical and nontechnical factors on software maintenance and proposed an estimation method for maintenance costs. Tsunoda et al. (2015) proposed a benchmark for the efficiency of software maintenance and analyzed the influence factors on efficiency in various organizations.

Previous studies also proposed methods to estimate maintenance costs. April (2010) analyzed the trend of software maintenance for the improvement of maintenance ability and analyzed the difference between software maintenance activities and the project department of information system development. Ng and Gable (2010) compared the government service provider (GSP) ERP maintenance process and maintenance data standard with IEEE/EIA 12207 software engineering standards based on an analysis of GSPs in Australia and proposed a guideline for participants using the difference. Salmeron and Lopez (2012) described an objective for ERP system manager maintenance and proposed predicting the risk impact of the objective.

A suitable system should be selected for reducing maintenance costs. Among the related previous studies, Hadidi et al. (2017) proposed a supporting method of decision-making for the ERP priority module implementation. The expected benefits of ERP system implementation are described and the generally used ERP module in the construction industry is outlined. Then, the ERP priority module was analyzed based on global indices, including a criteria index and module index, using the derived importance index. Niu et al. (2013) proposed a scenario method to analyze the impact of the architecture type on satisfying business requirements for the selection of suitable software architecture among the various alternatives. Niu et al. (2014) proposed a framework for a significant requirements analysis of an engineering enterprise system based on quality attribute scenarios. The requirements include customizability, flexibility, reusability, extensibility,

performance, efficiency, adaptability, autonomy, interoperability, and composability. Soja *et al.* (2015) analyzed the critical success factors and barriers during an ES lifecycle; the factors include project management, team, change management, solution, motivation, finance, top management, clear plan and goals, vendor, communication, knowledge, technical, training, and testing. Wu *et al.* (2009) proposed a component-based method to develop an industry-oriented ERP system. The objective of the method is to improve operability based on business process modeling and software reuse. Xu (2011) investigated the modeling method of a complicated ES and explained recent trends of ES.

Li *et al.* (2014) conducted research highly relevant for this study and analyzed maintenance activities during a software lifecycle using survey and statistical methods. For the analysis, maintenance trends of the information system were analyzed during each maintenance step of the lifecycle and the steps were divided into three areas: user support (training, consultative, evaluative), repair (adaptive, reductive, corrective), and enhancement (groomative, preventative, performance, enhance). The results of their research could be used as reference data for industry practices. This study analyzes the level of maintenance based on urgency, difficulty, and damages, and the integrated level of maintenance was also analyzed; estimation criteria for maintenance costs were proposed. This is the main difference between this study and previous studies. Another difference is the research subject. This study considers construction companies that have not experienced an ES in the past.

The influence factors on maintenance and the estimation methods for maintenance costs explored in these previous studies can be described (Table 1). First, the previous studies identified technical and non-technical factors

that affect maintenance cost. The technical factors, related to the characteristics of the system, include the following: complexity; code (quality, lines of code change); document quality (unified programming specification, detailed description); defect location and time of discovery; interfaces to surrounding systems; size of system (program, component, code, class, database); and system architecture (mainframe, web system, client server).

These factors have been used in a constructive cost model (COCOMO), a function point (FP), and a combination of fuzzy theory and statistical methods to estimate software development costs. The maintenance cost is the product of development cost and maintenance rate. However, a construction company who had not previously implemented a similar ES would have insufficient technical knowledge about it. This construction company would have limited information during the planning stage, and would therefore have difficulty applying the existing methods. Especially, maintenance rate is uncertain, and is typically 10–15% of initial investment (Ministry of Knowledge Economy 2011). The main contents of COCOMOs and FPs are as follows:

- COCOMO estimates a cost for system development by quantifying lines of code (LOC). The method requires information about the development size, and is therefore more suitable to obtain a detailed estimate than an approximate estimate. COCOMOs are of three types: 1) the basic type that just estimates LOC; 2) the intermediate type that also considers cost factors such as hardware and individual capacity; 3) the advanced type that further considers cost factors for analysis and design during software development. COCOMO2 was developed for large projects and to improve estimation accuracy during the planning stage.
- FPs estimate the size of system quantitatively and qualitatively based on the amount of information processing and functional complexity. A user can understand an FP easily because functions were derived based on the user’s requirements. However, development of such functions requires advanced analysis ability. Furthermore, the estimate is ambiguous when the user’s requirements were not derived.

The non-technical factors include the following: external environment (industry type, business rules, workflow); human (maintainers’ skills and experience, changed and non-changed number of maintainers); support time (multiple time zone, working time); equipment; and material (hardware). The factors can be easier to use than technical factors. However, the maintenance cost cannot be estimated based on non-technical factors because the estimate requires quantification of the level of factors to be maintained. For example, human factors such as maintainers’ skill and experience affect the maintenance cost but the relationships have not been analyzed. To obtain this information, the causes for maintenance should be identified, then the level of maintenance must be analyzed. If the construction company and the IT company can determine the

Table 1. Influence factors on maintenance cost

Classification	Factors
Technical factors	Size of system Complexity of system Number of components Unified programming specification Detailed description of the software system Percentage of online system Defect locality and discovered time Maintenance type System architecture Interfaces to surrounding systems
Non-technical factors	Experience Human capacity Number of engineer Support time Level of service level agreement External environment Resource factors User needs

level of maintenance (e.g. effort and time) that each task requires, they can estimate the expected maintenance cost and use it as a reference when negotiating an SLA.

This study proposes estimation criteria for maintenance costs during the lifecycle of a construction company that has no experience with a similar ES. For this proposal, this study analyzes the causes and levels of maintenance to be used as criteria estimations of maintenance costs for ES.

### 2. Research model

The proposed method is divided into two steps (Figure 2). First, the causes for maintenance were derived based on an SLA that is used in 70% of large companies in South Korea (Korea IT Service Industry Association 2006), and on previous studies (April 2010; IEEE 1219-1998; ISO/IEC 14764 2006). Then the causes were classified by an IT consultant who has 15 years of experience with construction and operation of ES. Next, the ES lifecycle was divided to three stages, similar to the bathtub curve (Figure 1), and the level of maintenance was analyzed based on urgency, difficulty, and damage. Then, the result was compared with bathtub curve and the differences are noted.

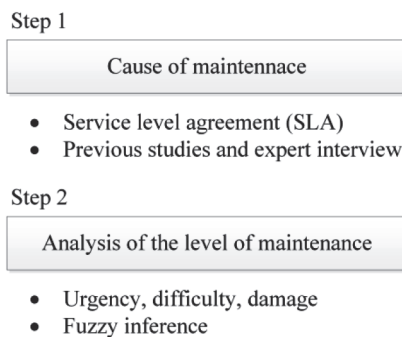


Figure 2. Research procedure

For analysis of the maintenance level, a survey was conducted using a ten-point to classify maintenance level based on difficulty, urgency, and damage. All respondents (a total of 30 persons) currently perform maintenance work on ES and have at least five years of experience. The survey was conducted from June 20 to July 1, 2016. For the integration of maintenance level, this study used fuzzy inference. A fuzzy inference system based on ‘if-then’ rules was used to analyze the integrated level of maintenance based on the difficulty and urgency of maintenance, and the level of damage. The basic rule of fuzzy inference is a multiple-input single-output system as follows:

$$\text{If } x \text{ is } A_k \text{ and } y \text{ is } B_k, \text{ then } z \text{ is } C_k. \tag{1}$$

Fuzzy inference converts information with ambiguous and uncertain boundaries into useful information. For example, ‘very low’ is not specified as 0.2, but can be anywhere from 0.1 to 0.4. This study applied triangular fuzzy numbers which are expressed as three points l, h, and m (minimum, median, and maximum value, respectively; i.e.,  $l \leq m \leq h$ ) and can be easily approached conceptually

Table 2. Triangular fuzzy numbers applied for fuzzy inference

	Low	Middle	High
Very low	0	0.2	0.4
Low	0.2	0.4	0.6
Middle	0.4	0.6	0.8
High	0.6	0.8	1.0
Very high	0.8	1.0	1.0

and can be easily analyzed mathematically (Table 2) (Akay et al. 2011; Shen et al. 2016).

In this study, for fuzzy inference, the max-min method under Mamdani’s method, and the center-of-gravity method were used; these are the most commonly used among methods for fuzzy inference (Camastra et al. 2015). Mamdani’s method uses  $\text{Min}(\mu_A(x) \wedge \mu_B(x))$ , i.e., the minimum value between A and B in Eqn (2), then  $\text{Max}(\mu_A(x) \vee \mu_B(x))$ , i.e., the maximum value between minimized A and B in Eqn (3), which are fuzzy synthesis rules. The center-of-gravity method changes the fuzzified analysis results to constant values (Eqn (4)).

$$\mu_{ck}(z) = \min\{\mu_{AB}, \mu_{ck}(z)\}; \tag{2}$$

$$\mu_{ck}(z) = \max\{\mu_{ck}(z), \mu_{ck}(z)\}; \tag{3}$$

$$x = \frac{\int x \cdot \mu_c(x) dx}{\int \mu_c(x) dy}. \tag{4}$$

### 3. Causes for of maintenance

Among the causes of maintenance (Table 3), first, functional enhancement is development of new modules for users’ continual or periodic work. This category includes extraction and confirmation of related data for user’s periodic work by coding or query. Next, adaptive maintenance revises functionalities to improve existing system screening and processing, for example by modifying simple algorithms or counting formulas as required by the user. Adaptive maintenance also includes non-functional changes, such as to the screen layout. Third, perfective maintenance improves simple functionality, quality or operability. For example, batch jobs periodically execute

Table 3. Causes of maintenance

Classification	Factors
Functional enhancement	New development
	Change of functionality
Adaptive maintenance	Change of simple program
	Change of non-function
Perfective maintenance	Request of batch job
	Environment setting
Preventive maintenance	Speed monitoring
	Data monitoring

programs such as settlement of accounts. Improvement of batch jobs for periodical use is a functional enhancement. Another perfective task to improve the user environment of a developed system for after installation of new programs, or by installing new package versions. Lastly, preventive maintenance averts possible failure of a system by monitoring its workflow to prevent decrease of system speed by such causes as conflict between programs, and accumulation of data. The purpose of data monitoring is to correct or delete erroneous data caused by program faults or input error, and to revise data when the system is modified.

#### 4. Analysis of the level of maintenance

A previous study divided the lifecycle of a system into four steps: an introduction step for user support, a growth step for repairing the increasing problem, a maturity step for reflection on additional requirements according to a changed work process and user environment, and a decline step for the implementation of a new system using new technology (Hsiang-Jui 2004). In this study, the proposed four steps are condensed into three steps (early, middle, end step) through interviews with an IT consultant who has 15 years of experience in the planning, construction, and operation of an ES system and the level of maintenance are analyzed. The early step is similar to the introduction and growth steps; the middle step is similar to the maturity step, and the end step is similar to the decline step in the previous study.

##### 4.1. The level of maintenance by causes

For functional enhancement, urgency was judged to be more important than difficulty and damage during the ES lifecycle (Figure 3). Generally, addition of a new functionality is urgent because the construction company requires it for goal achievement. The importance of difficulty of maintenance and degree of damage were similar during the early and middle stages, but difficulty increased during the final stage, because of incompatibility and conflicts with existing programs.

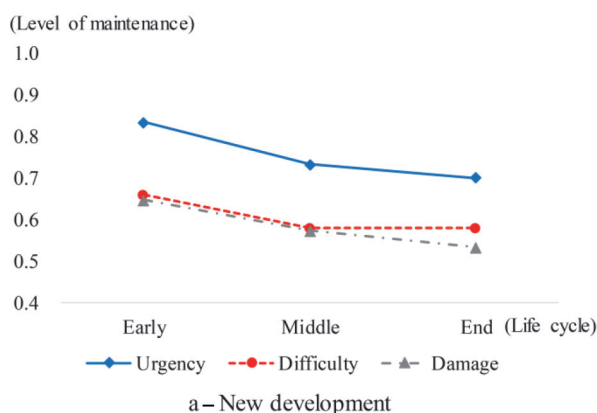


Figure 3. Level of functional enhancement: a - New development

For adaptive maintenance, the urgency of functionality change was more important than difficulty and damage during the early and middle stage, but lower during the end stage (Figure 4(a)); this change occurs because the construction company's requirements for functionality change were decreased by planned upgrades of the ES. The urgency of change to simple programs and to non-functions was judged to be more important than difficulty and damage during the lifecycle (Figure 4(b), 4(c)). However, the urgency decreases during upgrading of the ES because these are not intended to change major functionality of ES but to improve the user's convenience and business efficiency. Difficulty and damage were low because changes to simple programs and non-functions are not major tasks.

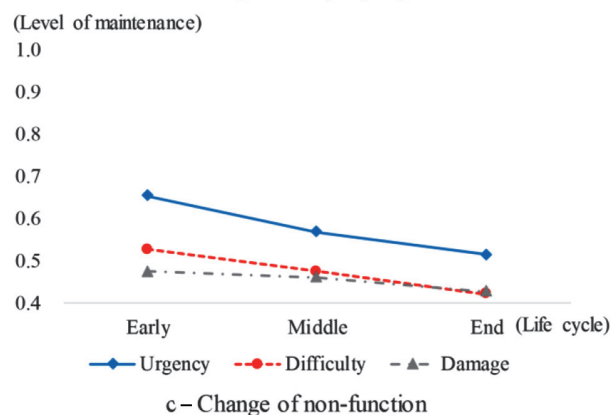
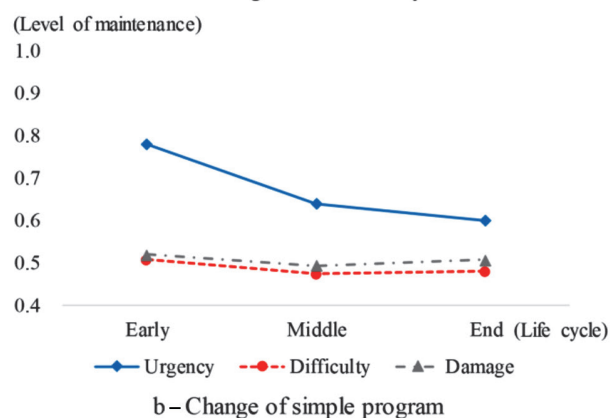
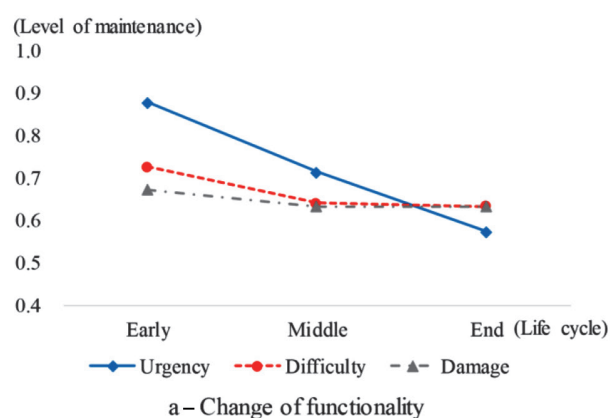


Figure 4. Level of adaptive maintenance: a - Change of functionality; b - Change of simple program; c - Change of non-function

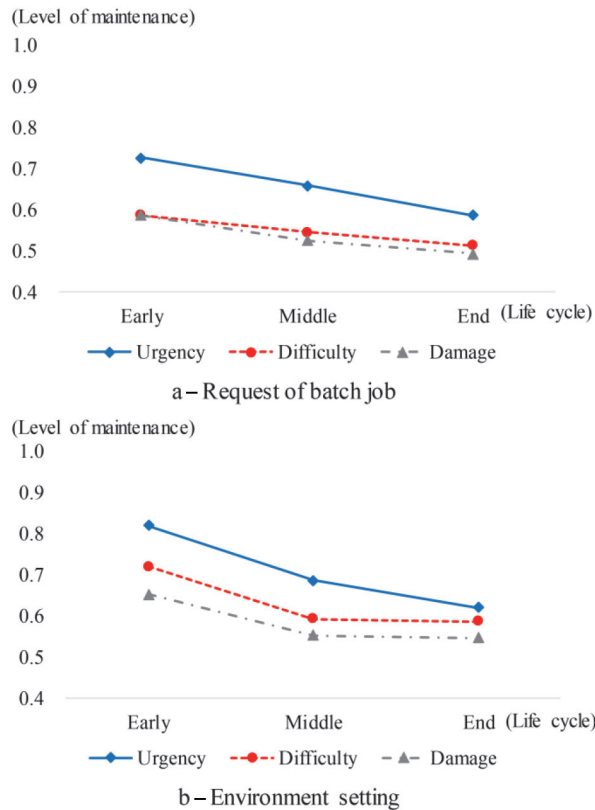


Figure 5. Level of perfective maintenance: a – Request of batch job; b – Environment setting

Overall, the importance of maintenance of functions such as change of functionality and simple programs was higher than of non-functions.

In perfective maintenance, urgency of the batch job request and environment setting were higher than difficulty and damage during life cycle (Figure 5). For environment settings, the level of urgency, difficulty, and damage were similar during the end stage because the urgency was decreased by a decrease in the requirement to install new programs. However, the level of difficulty and damage were not decreased by complicated relations among the installed programs during the product lifecycle.

In preventive maintenance, the level of urgency, difficulty, and damage of speed monitoring were higher during in middle and end stages than during the early stage (Figure 6). The cause for this trend is that the speed de-

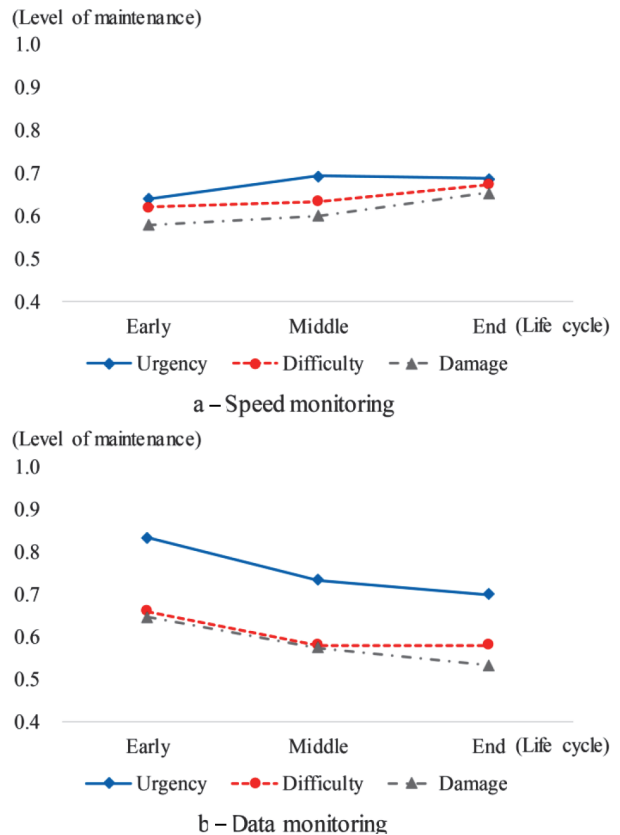


Figure 6. Level of preventive maintenance: a – Speed monitoring; b – Data monitoring

creased over time because the number of programs added and program changes increased, and data accumulated. The importance of urgency of data monitoring was higher than difficulty and damage during the lifecycle because data errors generally lead to bad decision-making. During the early stage, the urgency, difficulty, and damage were all higher than during the middle and end stages because installation of programs and input of data before ES construction caused data errors.

#### 4.2. The integrated level of maintenance derived by fuzzy inference

In this section, the analyzed level of maintenance by causes (Section 4.1) is integrated based on urgency, difficulty, and damage using fuzzy inference. For fuzzy inference, 125 ‘if-then’ rules were established (Table 4). The levels

Table 4. Fuzzy inference rules applied with the level of urgency, difficulty and damage

Rule	Difficulty		Urgency		Damage		Output
1	Very high	and	Very high	and	Very high	then	VH
2	Very high	and	Very high	and	High	then	VH
3	Very high	and	Very high	and	Middle	then	H
...							
123	Very low	and	Very low	and	Middle	then	L
124	Very low	and	Very low	and	Low	then	VL
125	Very low	and	Very low	and	Very low	then	VL

Table 5. The result of fuzzy inference

Maintenance causes		Early	Middle	End
Functional enhancement	New development	<u>0.73</u>	<u>0.64</u>	<u>0.63</u>
	Change of functionality	<u>0.72</u>	<u>0.66</u>	<u>0.64</u>
Adaptive maintenance	Change of simple program	0.57	0.49	0.51
	Change of non-function	0.52	0.48	0.45
Perfective maintenance	Request of batch job	0.58	0.54	0.51
	Environment setting	<u>0.72</u>	0.58	<u>0.59</u>
Preventive maintenance	Speed monitoring	0.63	<u>0.65</u>	<u>0.68</u>
	Data monitoring	<u>0.67</u>	0.57	0.57
Mean		0.64	0.58	0.57

of difficulty, urgency and damage were divided into ‘very high’, ‘high’, ‘medium’, ‘low’, and ‘very low’, and fed into Eqns (2)–(4). For example, if urgency and difficulty are VH and damage is M, the result of fuzzy inference is H.

In the result of integrated level of maintenance (Table 5), the level of maintenance of new development and change of functionality were all higher than the average level of maintenance during the lifecycle. These maintenance tasks occurred most frequently during the early stage because the IT companies did not satisfy the user’s requirement, or because the user’s requirement changed. The level of maintenance was also high during middle and end stage due to problems such as compatibility and conflict among programs.

During the early stage, the causes of high maintenance requirements were determined to be environment setting and data monitoring. The causes were that many environment settings were established for the programs that were used before ES implementation; that data errors occurred; and that many changes were caused by program installation, revision and data input during the early stage. For the middle and end stages, the maintenance level of speed monitoring was higher than the average maintenance level. The cause is that accumulated data and added new programs reduce the speed of the ES, so the frequency of maintenance to reduce speed slowdowns increased. Speed slowdowns were not common during the early stage. During the end stage, the level of environment setting maintenance also increased as a result of complex relationships among the accumulated programs.

The integrated level of maintenance was highest during the early stage and decreased steadily until the end stage (Figure 7). This trend is different from the bathtub curve that represents the general trend of the incidence of maintenance (Jiang 2013). The level and incidence of maintenance are similar during the early and middle stages, but during the end stage the level of maintenance decreased but the incidence increased; these opposing trends occur because the need for high-level maintenance such as new development or functionality change decreased as a result of planned ES reconstruction during the end stage. Thus, if the construction company negotiates a contract with the

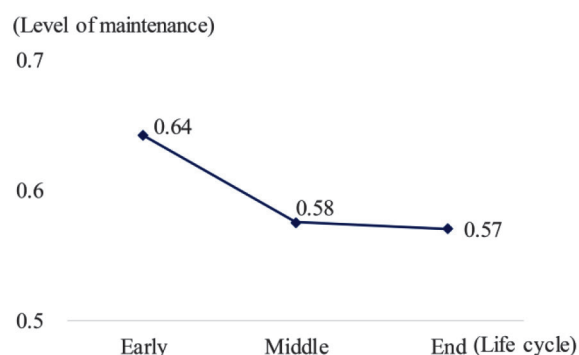


Figure 7. The integrated level of maintenance during life cycle

IT company based only on the incidence of maintenance, the contract price could differ from the actual cost of maintenance. The results of this study may be used to estimate maintenance costs based on causes and the level of maintenance. The construction company may estimate investment costs based on the actual required costs of maintenance, and the IT development company may negotiate the fair costs of maintenance separate from external factors such as reputation.

### 4.3. The economic difference between the proposed method and bathtub curve

The ES lifecycle could be not represented clearly, because the time of ES implementation is not long and information about ES is confidential in construction companies in South Korea. In this section, the lifecycle is 10 years for the early (3 years), middle (5 years), and end (2 years) steps based on an interview with the IT consult. The economic difference between the proposed method and the bathtub curve is compared. The initial investment cost is taken as million USD 11,355,000 (1 USD = 1,135.5 KRW; July 21, 2018), which is the initial implementation cost of an ERP system in a construction company in South Korea as of 2012. Another assumption is that maintenance costs are incurred based on the level of maintenance. For example, when the proposed method is applied, the maintenance cost is assumed to be 0.65 (early), 0.58 (middle),

and 0.57 (end) of the initial investment cost incurred (Figure 7). When the bathtub curve is applied, the maintenance cost is assumed to be 0.65 (early), 0.58 (middle), and 0.67 (end) of the initial investment cost incurred (Figure 1). The real discount rate is applied as 3.83%, which is the result of Eqn (5) based on the consumer price index and business loan interest rate. For the difference analysis of maintenance costs during 10 years, Eqn (6) is used to change from present value to future value:

$$i = \frac{1+in}{1+f} - 1; \quad (5)$$

$$FV = PV(1+i)^t. \quad (6)$$

The maintenance cost calculated by using the proposed method is approximately million USD 23,186,799 during 10 years, but the maintenance cost calculated by using the bathtub curve is approximately million USD 24,166,115. The difference is about million USD 979,316, and this is 8.62% of the initial investment cost. Although the economic difference includes several assumptions and relative comparisons in this section, the proposed method can contribute to a more accurate estimation of investment costs in decision-making on whether to implement the ES or not.

## Conclusions

Maintenance cost of enterprise system (ES) is large because it operates for a long time (Nguyen *et al.* 2011). Thus, the maintenance cost is a main economic factor in the decision whether to implement an ES. However, if the construction company has not implemented a similar ES in the past, they have insufficient reference criteria, so accurate estimation of the maintenance cost is difficult. This study proposed reference criteria to estimate maintenance costs based on the causes and level of maintenance. Then, the result was compared with the bathtub curve that is generally used to represent the incidence of maintenance.

First, causes for maintenance were derived based on previous studies (April 2010; IEEE 1219-1998; ISO/IEC 14764 2006), a service level agreement (SLA), and an interview with an IT consultant who has 15 years of experience with ES. Then, the causes were classified as functional enhancement (new development), adaptive maintenance (change of functionality, simple program, and non-function), perfective maintenance (request of batch job, environment setting), and preventive maintenance (speed and data monitoring).

Next, the level of maintenance was analyzed based on urgency, difficulty, and damage during the lifecycle. Among the causes of maintenance, the level of the new development and change of functionality were high during lifecycle, the environment was high during the early and end stages, and data monitoring was high during the early stage. Especially, the integrated level of maintenance was similar to the bathtub curve during the early and middle stages, but different from it during the end stage in which the bathtub curve increases but the integrated level

of maintenance decreased slightly because the amount of maintenance was decreased by reconstruction planning of new ES. Thus, the maintenance causes and levels should be considered as well as their incidence when determining whether to invest.

This study proposed reference criteria for the estimation of maintenance cost; the importance of the criteria was compared with the standard incidence of maintenance (bathtub curve). The construction company who has not implemented a similar ES can refer to the proposed reference criteria during negotiation of maintenance contracts with IT company. The IT company can charge appropriate maintenance cost based on the result of this study. However, if the level of maintenance costs is analyzed based on technical characteristics such as complexity, number of components, and system architecture, the accuracy is more improved than using the proposed method involving a survey. In the future, methods to estimate maintenance costs according to technical characteristics will be studied. Furthermore, for implementation decisions regarding an IT system, a decision method based on the main considering factors, such as economic factors (initial investment cost, maintenance cost, expected benefit) and suitability, will be studied.

## References

- Akay, D.; Kulak, O.; Henson, B. 2011. Conceptual design evaluation using interval type-2 fuzzy information axiom, *Computers in Industry* 62: 138–146. <https://doi.org/10.1016/j.compind.2010.10.007>
- April, A. 2010. Studying supply and demand of software maintenance and evolution services, in *Seventh International Conference on the Quality of Information and Communications Technology (QUATIC)*, 29 September – 2 October 2010, Oporto, Portugal, 352–357. <https://doi.org/10.1109/QUATIC.2010.65>
- Buchmann, I.; Frischbier, S.; Putz, D., 2011. Towards an estimation model for software maintenance costs, in *15th European Conference on Software Maintenance and Reengineering (CSMR)*, 1–4 March 2011, 313–316. <https://doi.org/10.1109/CSMR.2011.45>
- Camastra, F.; Ciaramella, A.; Giovannelli, V.; Lener, M.; Rastelli, V.; Staiano, A.; Staiano, G.; Starace, A. 2015. A fuzzy decision system for genetically modified plant environmental risk assessment using Mamdani inference, *Expert Systems with Applications* 42: 1710–1716. <http://doi.org/10.1016/j.eswa.2014.09.041>
- Correa, S. L. L.; Mexas, M. P.; Drumond, G. M.; Meirino, M. J. 2016. Cost elements identification for maintenance and support of ERP systems in Brazilian IFES: An approach based on TCO and ITIL, *IEEE Latin America Transactions* 14: 2372–2381. <https://doi.org/10.1109/TLA.2016.7530435>
- Hadidi, L.; Assaf, S.; Alkhiami, A. 2017. A systematic approach for ERP implementation in the construction industry, *Journal of Civil Engineering and Management* 23(5): 594–603. <https://doi.org/10.3846/13923730.2016.1215348>
- Hsiang-Jui, K. 2004. Quantitative method to determine software maintenance life cycle, in *IEEE International Conference on Software Maintenance*, 232–241. <https://doi.org/10.1109/ICSM.2004.1357807>



- IEEE 1219-1998 IEEE standard for software maintenance. IEEE, 1998.
- ISO/IEC 14764 Software engineering – Software life cycle processes – Maintenance. Geneva: International Organization for Standardization, 2006.
- Jiang, R. 2013. A new bathtub curve model with a finite support, *Reliability Engineering & System Safety* 119: 44–51. <https://doi.org/10.1016/j.res.2013.05.019>
- Korea IT Service Industry Association. 2006. *Survey of the IT outsourcing market's environment*.
- Li, J.; Stålhane, T.; Kristiansen, J. M. W.; Conradi, R. 2010. Cost drivers of software corrective maintenance: An empirical study in two companies, in *IEEE International Conference on Software Maintenance (ICSM 2010)*, 12–18 September 2010, Timișoara, Romania, 1–8. <https://doi.org/10.1109/ICSM.2010.5609538>
- Li, S.-H.; Yen, D. C.; Lu, W.-H.; Chen, T.-Y. 2014. The characteristics of information system maintenance: an empirical analysis, *Total Quality Management & Business Excellence* 25: 280–295. <https://doi.org/10.1080/14783363.2013.807679>
- Matičević, T.; Ognjanović, I.; Sendelj, R. 2012. Enhancement of software projects' Function Point Analysis based on conditional non-functional judgments, in *First Mediterranean Conference on Embedded Computing (MECO 2012)*, 19–21 June 2012, Bar, Montenegro, 283–287.
- Mendes, J.; Araújo, R.; Sousa, P.; Apóstolo, F.; Alves, L. 2011. An architecture for adaptive fuzzy control in industrial environments, *Computers in Industry* 62: 364–373. <https://doi.org/10.1016/j.compind.2010.11.001>
- Ministry of Knowledge Economy. 2011. *The standard of software cost estimation*.
- Ng, C. S.-P.; Gable, G. G. 2010. Maintaining ERP packaged software – A revelatory case study, *Journal of Information Technology* 25: 65–90. <https://doi.org/10.1057/jit.2009.8>
- Nguyen, V.; Boehm, B.; Danphitsanuphan, P. 2011. A controlled experiment in assessing and estimating software maintenance tasks, *Information and Software Technology* 53: 682–691. <https://doi.org/10.1016/j.infsof.2010.11.003>
- Niu, N.; Xu, L. D.; Bi, Z. 2013. Enterprise information systems architecture – analysis and evaluation, *IEEE Transactions on Industrial Informatics* 9: 2147–2154. <https://doi.org/10.1109/TII.2013.2238948>
- Niu, N.; Xu, L. D.; Cheng, J. R. C.; Niu, Z. 2014. Analysis of architecturally significant requirements for enterprise systems, *IEEE Systems Journal* 8: 850–857. <https://doi.org/10.1109/JSYST.2013.2249892>
- Ranaldo, N.; Zimeo, E. 2016. Capacity-driven utility model for service level agreement negotiation of cloud services, *Future Generation Computer Systems* 55: 186–199. <https://doi.org/10.1016/j.future.2015.03.007>
- Ren, Y.; Xing, T.; Chen, X.; Chai, X. 2011a. Research on software maintenance cost of influence factor analysis and estimation method, in *3<sup>rd</sup> Workshop on Intelligent Systems and Applications*, 15–18 June 2011, Chaves, Portugal, 1–4. <https://doi.org/10.1109/ISA.2011.5873461>
- Ren, Y.; Xing, T.; Qiang, Q. 2011b. Function point analysis to knowledge representation of software size measurement, in *2011 International Conference on Information Science and Technology*, 24–25 September 2011, 122–125. <https://doi.org/10.1109/ICIST.2011.5765224>
- Romero, D.; Vernadat, F. 2016. Enterprise information systems state of the art: Past, present and future trends, *Computers in Industry* 79: 3–13. <https://doi.org/10.1016/j.compind.2016.03.001>
- Salmeron, J. L.; Lopez, C. 2012. Forecasting risk impact on ERP maintenance with augmented fuzzy cognitive maps, *IEEE Transactions on Software Engineering* 38: 439–452. <https://doi.org/10.1109/TSE.2011.8>
- Sarno, R.; Sidabutar, J.; Sarwosri, 2015. Improving the accuracy of COCOMO's effort estimation based on neural networks and fuzzy logic model, in *International Conference on Information & Communication Technology and Systems (ICTS)*, 197–202. <https://doi.org/10.1109/ICTS.2015.7379898>
- Shen, Y.-C.; Chen, P.-S.; Wang, C.-H. 2016. A study of enterprise resource planning (ERP) system performance measurement using the quantitative balanced scorecard approach, *Computers in Industry* 75: 127–139. <https://doi.org/10.1016/j.compind.2015.05.006>
- Soja, P.; Themistocleous, M.; Cunha, P. R. d.; Mira da Silva, M. 2015. Determinants of enterprise system adoption across the system lifecycle: Exploring the role of economic development, *Information Systems Management* 32: 341–363. <https://doi.org/10.1080/10580530.2015.1080005>
- Tsunoda, M.; Monden, A.; Matsumoto, K.; Ohiwa, S.; Oshino, T. 2015. Benchmarking software maintenance based on working time, in *3<sup>rd</sup> Applied Computing and Information Technology/International Conference on Computational Science and Intelligence (ACIT-CIS)*, 12–16 July 2015, Okayama, Japan, 20–27.
- Ul Haq, I.; Huqqani, A. A.; Schikuta, E. 2011. Hierarchical aggregation of service level agreements, *Data & Knowledge Engineering* 70: 435–447. <https://doi.org/10.1016/j.datak.2011.01.006>
- Unger, T.; Mietzner, R.; Leymann, F. 2009. Customer-defined service level agreements for composite applications, *Enterprise Information Systems* 3: 369–391. <https://doi.org/10.1080/17517570903033431>
- Wu, S. L.; Xu, L.; He, W. 2009. Industry-oriented enterprise resource planning, *Enterprise Information Systems* 3: 409–424. <https://doi.org/10.1080/17517570903100511>
- Xu, L. D. 2011. Enterprise systems: State-of-the-art and future trends, *IEEE Transactions on Industrial Informatics* 7: 630–640. <https://doi.org/10.1109/TII.2011.2167156>
- Zhang, J.; Qin, W.; Wu, L. H.; Zhai, W. B. 2014. Fuzzy neural network-based rescheduling decision mechanism for semiconductor manufacturing, *Computers in Industry* 65: 1115–1125. <https://doi.org/10.1016/j.compind.2014.06.002>