Evaluating the relationship between process improvement and schedule deviation in software maintenance

Ho-Won Jung a,∗, Dennis R. Goldenson b,1

a Korea University Business School, Anam-dong 5Ka, Sungbuk-gu, Seoul 136-701, Republic of Korea
b Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890, USA

Abstract

A basic proposition of process assessment models is that higher process maturity is associated with improved project performance and product quality. This study provides empirical evidence to support this proposition by testing the hypothesis that higher process maturity is negatively associated with schedule deviation in software maintenance. Next, the present study investigates whether two process context factors (organizational size and geographical region) modify the relationship between process maturity and schedule deviation by using a moderator testing method. Our results show that organizational size does not influence the relationship, while geographical region is deemed to be an independent variable.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

1.1. The importance of software maintenance

The Capability Maturity Model® (CMM®) for Software (SW-CMM) [71] cites the definition of maintenance from IEEE Std 610-1990 [42] as “the process of modifying a software system or component after delivery to correct faults, improve performance or other attributes, or adapt to a changed environment”. This definition includes at least three types of software maintenance:

- Corrective maintenance: To correct processing, performance, or implementation faults in the software.
- Adaptive maintenance: To adapt the software to changes in the environment such as new hardware or the next release of an operating system. Adaptive maintenance does not change the system’s functionality.
- Perfective maintenance: To perfect the software with respect to its performance, processing efficiency, maintainability, or to accommodate the addition of new or changed user requirements.

The IEEE has estimated that the annual cost of software maintenance in the United States exceeds $70 billion [20]. Others have estimated the magnitude of software maintenance costs to range from 40% to 80% of overall software lifecycle costs [5,53]. In a recent study on the economics of software maintenance, Jones [45] states that “more than 50% of the global software population is engaged in modifying existing applications rather than writing new applications”. He also points out that the maintenance percentage among the United States software population has increased from 52% in 1995 to 76% in 2005 and is expected to steadily increase.

While the SW-CMM and CMMI [12] are intended for both development and maintenance processes, difficulties in implementing the model in maintenance-only organizations have been reported [6,19]. Swanson and Beath [85] claimed that software maintenance is fundamentally different from new system development since the maintainer must interact with an existing system. Others have criticized the SW-CMM and CMMI for not directly addressing maintenance [6,59]. One survey study, conducted in the United Kingdom, failed to find evidence that higher maturity companies manage maintenance more effectively than lower maturity companies [37]. However, the survey does not explicitly state how it defines maturity.

Niessink and van Vliet [68] investigated the difference between software maintenance and software development from a service point of view. They argued that software maintenance could be seen as providing a service, while software development is concerned with the development of products. Hence, they developed a separate information technology (IT) service CMM targeted at software maintenance organizations and other IT service providers. Similarly, Kajko-Mattsson [50] developed a problem management
maturity model for corrective maintenance. Under best practices such as the SW-CMM, however, maintenance costs actually declined over the 5 years, especially for systems software and defense applications produced by companies [45].

1.2. This study

A basic proposition of all capability maturity modeling, including both the SW-CMM and CMMI, is that higher process maturity is associated with better project performance and product quality. This implies that improving maturity is expected to subsequently improve both performance and quality. Given both the high cost of software maintenance and process improvement activities, it is important to provide objective evidence about the relationship between process maturity and performance in a maintenance context. Testing this proposition can be considered an evaluation of the predictive validity of a maturity measure [22].

This study provides empirical evidence that higher process maturity is in fact negatively associated with schedule deviation in software maintenance. The study also investigates whether two process context factors, organizational size and geographical region, systematically modify the strength and form of the relationship between process maturity and schedule deviation by using a moderator testing method. The analysis is based on 752 maintenance projects from 441 SW-CMM assessments. A negative binomial regression [10] is used to account for non-negative integer values and the existence of multiple reports of no deviations in the schedule. The results are validated using a bootstrap resampling method [21,65]. To the authors’ knowledge, this is the first study to investigate the relationship between process maturity and schedule deviation in software maintenance by using the results of full-scale SW-CMM assessments.

Although using the data set originally used in [49], the present study applies a different method. A sufficient amount of data from CMMI appraisals was not yet available for the present study. The study will be replicated when enough data are available. As the SW-CMM is an important source for CMMI, the results should be similar.

This study contains the following seven sections: in Section 2, we present the study’s research model and hypotheses, including reviews of previous studies on the process maturity–performance relationship in software process assessment. In Section 3, we address the data collection and measures. In Section 4, we describe the analysis methods, including a negative binomial regression, a procedure for testing hypothesized moderators, and a bootstrap method for examining the stability of the results. In Section 5, we present the analysis results. In Section 6, we discuss the study limitations. In Section 7, we present our final concluding remarks.

2. Research model and empirical hypotheses

2.1. Research model

The SW-CMM provides a structured path for organizing software processes into five maturity levels (MLs), or evolutionary steps, which lay successive foundations for continuous process improvement (Table 1). The SW-CMM covers practices for planning, engineering, and managing software development and maintenance. Its underlying proposition is that more mature software organizations, when following these key practices, will be better able to meet their cost, schedule, functionality, product quality, and other performance objectives [71].

A previous study demonstrated that the relationship between process maturity and performance outcomes may depend on the context in which the process improvement takes place [23]. There may be a plethora of potential moderating factors that affect the relationship between maturity and the schedule deviation under investigation. This study examines two moderating factors as specification variables, where a specification variable changes the strength or form of the relation between a dependent and independent variables [75,81]. The hypothesized moderators in this study are the geographical region and size of the organization assessed. Fig. 1 presents the relationship between process maturity and schedule deviation in software maintenance and indicates that this relationship depends on the hypothesized moderators.

The justification for the model components and formal hypotheses are addressed in the next section. In this study, process improvement is a synonym for CMM-level transition from lower to higher.

| Table 1 |
|-----------------|-----------------|-----------------|
| Maturity levels (MLs) and their key process areas [71] |
| Level 1 initial | Competent people (and heroes) |
| Level 2 repeatable | Project management processes |
| Level 3 defined | Engineering processes and organizational support |
| Level 4 managed | Product and process quality |
| Level 5 optimizing | Continuous process improvement |
| Focus | Key process areas |
| Defect prevention | – Software quality management |
| Technology change management | – Process change management |
| Training program | – Software product engineering |
| – Integrated software management |
| – Intergroup coordination |
| – Peer review |
| Requirements management |
| – Software project planning |
| – Software project tracking and oversight |
| – Software Subcontract Management |
| – Software quality assurance |
| – Software configuration management |

2.2. Empirical hypotheses

The following subsections state three research hypotheses depicted in Fig. 1. The measures used in the hypotheses are described in Section 3.

2.2.1. Process maturity and schedule deviation

All previous studies of the maturity–performance relationship in process improvement are based either implicitly or explicitly on the theoretical model depicted in Fig. 1. A survey study of individuals from SW-CMM-assessed organizations showed that higher maturity organizations tend to perform better on various subjective measures of performance, including the ability to meet schedule and budget and increase product quality, staff productivity, customer satisfaction, and staff morale [32,40]. Another survey-based study found evidence of the relationships among seven software processes and measures of project performance including schedule, budget, quality, and rework [17]. The Software Engineering Institute (SEI) presented results from 12 organizations in 2003 and 35 in 2006 demonstrating the benefits gained by means of process improvement [33].

The benefits of the SW-CMM were investigated with two measures (cost and schedule) extracted from U.S. Air Force contracts [60]. The results of that study showed that projects with more capable processes typically perform better on indices of both cost and schedule performance compared to those with lower levels of process capability. A study combining questionnaire data with existing project metrics found that the SW-CMM-based process capability was also associated characteristically with a reduction...
in delivered defects after controlling for size and personnel capability [58]. A study analyzing 161 projects concluded that development effort could be reduced by 4–11% by process improvement, where the percentages refer to the development effect after isolating for the other effects [13]. Recently, a review study of 17 papers [29] addressed benefits such as defects, productivity, rework, cycle time, schedule, error deflection effectiveness, and return on investment with respect to process improvement in CMM.

Many studies have addressed the relationship between performance and one or more elements of SW-CMM key process areas (KPAs). In particular, studies using a structural equation model included KPA elements as indicators of exogenous or endogenous latent variables. They employed process quality or product quality as performance variables [74]. Niazi et al. [67] reviewed critical success factors, including KPA elements, for process improvement in CMM. A study analyzing the development of 30 products showed that software process capability is positively associated with software quality [38].

Numerous case studies in the SW-CMM have demonstrated benefits from increased process maturity [17,18,36,43,52,57,59,72,76,88,93,94]. CMMI case studies have also shown benefits from process improvement [26,31,63,78].

The theoretical basis shown in Fig. 1 implies that schedule deviation is negatively associated with ML. Testing the hypothesis in software maintenance allows the direct relationship between maturity and schedule deviation to be evaluated:

- **Hypothesis 1 (Basis relationship):** Increasing ML reduces schedule deviation in software maintenance.

2.2.2. Effect of geographical region

The SEI’s Process Maturity Profile [80] shows the distribution of the SW-CMM and CMMI MLs by geographical regions, which are classified as US and non-US organizations. A case study comparing Siemens companies located in Germany with those in the US addresses the importance of cultural factors in process improvement [70]. As do differences in product types and the defined processes themselves, cultural factors can greatly influence the adoption of new or changed processes. Our previous analysis of the data set in this study also used geographical region as a moderator variable [49]. The differences in software engineering process practices between Sweden and elsewhere in Europe have also been reported [92].

Regional differences in capability levels have also been extensively evaluated in the SPICE Phase 2 Trials. The Trial Report rated the Australian software industry as the world’s best practice since significantly more of the Australian assessments yielded ratings of capability levels 2 and 3 than early assessments in other non-US countries. Another study based on the SPICE Phase 2 Trials revealed the low capability levels achieved by European assessments compared with those from other geographical regions, including South Asia and Pacific [41].

Hence, this study tests the influence of the region where the assessment was conducted on the base relationship (Hypothesis 1). The research hypothesis with regional factor as a moderator is formulated as follows:

- **Hypothesis 2 (Region):** The relationship between ML and schedule deviation in software maintenance differs across geographical regions.

2.2.3. Effect of organizational size

Organizational size has been one of the most popular moderators discussed in the process improvement literature. Empirical literature, however, shows inconsistent results about the effect of organizational size (e.g., confirmed as a moderator but with contradictory results, no impact on the relationship, predictor variable, or control variable).

A study based on the SW-CMM [8] demonstrated that the implementation of some processes and practices might not be as cost-effective for small organizations as for large ones. Studies based on the SPICE Phase 2 Trials [24,25] showed that in some processes, the relationship between capability level and performance was stronger in large organizations. A contradictory result [55] indicated that standardization of lifecycle processes is associated with management information system’s success in smaller organizations but not in large ones.

However, other studies have shown instances where organizational size did not influence the relationship between process maturity and performance [17,32]. In those studies, organizations with relatively few software employees appeared to benefit from higher process maturity in the same manner that larger organizations did. This result is consistent with a study on the relationship between organizational capability and requirements engineering process success [23].

A meta-analytic study on IT innovation and adoption addressed 21 empirical studies that used organizational size as an independent variable [62]. In a study of the assimilation of software process innovation, organizational size is included as a control variable because it likely serves as a proxy for other variables such as slack resources, education, specialization, and scale [28,86]. Another study about the relationship between research and development and firm performance also included organizational size as a control variable [61].

Based on previous studies, the authors of this paper hypothesize that organizational size acts as a moderator affecting the relationship between maturity and schedule deviation:

3. Data collection and measures

3.1. Data collection

SEI-authorized, lead assessors of the SW-CMM were required to provide reports to SEI for their completed assessments. Assessment data from the reports were kept in an SEI repository called the Process Appraisal Information System (PAIS\(^*\)). PAIS included information for each assessment on the company and appraised entity, KPA profiles, organization and project context, functional area representatives groups, findings, and related data. All assessment data were kept confidential and were available only to SEI employees who were associated with their research and development (R&D) activities.

Not all the assessments included KPA rating profiles because the determination of an ML or KPA rating was optional and was provided at the discretion of the assessment sponsor. The data set analyzed for this study was extracted from appraisal reports in PAIS for the period of January 1998 through December 2001.

As shown in Fig. 2, a total of 441 organizations reported their assessments for 752 maintenance projects. Among them, approximately 65% of the organizations and projects were from the US and the other 35% originated from non-US organizations. In Fig. 2, since more than one maintenance project was assessed in some organizations, the number of organizations was fewer than the number of projects.

In detail, a single maintenance project was reported in 247 (56%) of the 441 organizations assessed. Two organizations assessed six maintenance projects each, which was the highest number of maintenance processes assessed in a single assessment. The mean, median, and standard deviation of the number of maintenance projects in these assessed organizations were 1.67 (1.75), 1 (1.5), and 1.01 (0.95), respectively, where the numbers refer to US organizations and the numbers in parentheses to non-US organizations.

3.2. Measures

This section addresses the measures employed to test this study’s three hypotheses. The units of analysis in this study were projects in the maintenance phase of their lifecycles. Since the organization typically is the unit of a maturity rating in SW-CMM assessments, this study’s measure of maturity was organization-wide. If several maintenance projects were assessed in a single organization, all of the projects had the same level of maturity but had their own individual values of schedule deviation. Issues concerning this study’s measures are described later.

This study sometimes used interchangeable terms in parenthesis for independent (predictor) and dependent (criterion) variables when referring to other studies to maintain consistency with them.

3.2.1. Process maturity

In SW-CMM assessments, the assessment of a single organization covers several projects. The KPA profiles for an organization are the aggregate of assessment team judgments across those projects to produce a single ML for the entire organization within the scope of the assessment. PAIS only provides an ML from 1 to 5. The information at the detailed level of KPA goals or common features was not recorded.

Fewer than five maintenance projects at MLs 4 and 5 reported schedule deviation greater than one month in duration. Other things being equal, a lower incidence of reported schedule deviation at MLs 4 and 5 is quite consistent with the base relationship hypothesized in this paper. Regardless, this study followed the conservative statistical rule of thumb stating that there should be at least five observations to provide sufficient confidence in the analysis results [68]; hence, the ML 4 and 5 organizations were excluded from the statistical analysis.

In this study, the explanatory variable, process maturity, was coded such that ML 3 is 3, ML 2 is 2, and ML 1 is 1. This study considered the coded value as an interval scale and then employed parametric statistics\(^*\) as in previous studies [24,25,47]. When the scales have interval-level properties, it is considered appropriate to test the moderating effects [3]. A linear transformation of the variable scores, like the coding used in this study, is also known to have a very small risk of spurious interactions [51]. A previous study provided ample confidence in the internal-consistency reliability over 0.9 of that measure [48].

3.2.2. Schedule deviation

In the context of software maintenance, schedule deviation is the performance measure that is used as the dependent variable. In this study, it was defined as the absolute value (non-negative integer) of the difference between the actual schedule and the planned schedule (i.e., \( y = |(\text{Actual} - \text{Planned})| \)). Schedule deviation \( y \) was expressed in months ahead or behind schedule, with a value of zero indicating that the project was on schedule. Although being ahead of schedule is certainly less serious than being behind schedule, too few projects reported being ahead of schedule to allow a separate analysis here.

Several candidate measures of schedule deviation can be considered, including relative measures, standard deviations and relative value. The following two relative measures were considered: \( y = [(\text{Actual} - \text{Planned})/\text{Actual}] \) or \( y = [(\text{Actual} - \text{Planned})/\text{Planned}] \) [16,82]. However, such measures could not be applied to a data

\(^*\) Classical measurement theory posits that variables should be measured on at least an interval scale to permit the computation of the mean and related parametric statistics [69,83], but using only nonparametric methods on non-interval scale data would exclude much useful study [69]. Many authors argue that a useful study can be conducted even if the prescriptions are violated [7,83,89]. El Emam and Birk [24,25] provide detailed discussion of the scale-type issue in studies of process capability and maturity.

\(^*\) PAIS has been replaced. Appraisals now are reported using the SEI Appraisal System (SAS), http://www.sei.cmu.edu/appraisal-program/profile/report-faq.html
set characterized by excess zeros such as the ones in this study. If the denominator has a small value, the measure may be exaggerated and take on an unreasonably large value.

3.2.3. Geographical region

A variety of classifications of geographical region exists for the assessed organization. The classifications depend on what information is needed in the study. Use of a different classification of the assessed organizations may lead to different analysis results.

The SEI Process Maturity Profile [80] classifies regions into the US and non-US because of the typical profile difference between these two regions. This study followed the SEI classification category (i.e., US and non-US). In this nominal scale, the US was coded 0 and the non-US 1.

3.2.4. Organizational size

The IT staff size is known to be a better measure for organizational size than the firm size because the IT staff size taps into IT-specific resources and capabilities that directly affect IT innovation adoption [62,73]. A European project that provided process improvement guidance for small organizations classified organizational size into two groups based on whether the organization had a large or a small IT staff, with the cut-off point being 50 IT staff [77]. Some studies [24,25,46] in the SPICE Trials followed the definition of the SPIRE project.

To investigate a moderating effect, the sample can be divided into two groups by a median, quartile, or predefined criteria [81]. However, the median is strongly recommended [1,3], and was used in this study. For brevity’s sake, small-size organizations were coded 0, and other organizations 1.

4. Data analysis

4.1. Negative binomial regression model (NBRM)

Schedule deviation, as defined in this study, was a relatively rare occurrence in the data set. Its value was limited to non-negative integers. Those characteristics compelled the study to employ a regression rather than Pearson or Spearman correlations to investigate the association between process maturity and schedule deviation. In a model including a moderator, regression is highly recommended [1]. However, this study sometimes utilized the correlation coefficient to explain the results.

More than one regression model exists for explicitly accommodating those characteristics. So it was necessary to select an appropriate one. Considering the strengths and weaknesses of each candidate model, as well as various perceptions in the software engineering community, the negative binomial regression model (NBRM) was used for the study analysis.

The NBRM is an extension based on the Poisson regression model (PRM) assuming equidispersion (i.e., variance equals to mean). The NBRM allows for a larger value of variance than the mean. It deals with issues such as non-normality, non-constant variance, and negative expected value. In the NBRM, a random error term \( e_i \) is added to the mean of PRM in order to deal with over dispersion or under dispersion. This allows the variance to exceed (or be less than) the mean [10].

The expected schedule deviation of the NBRM is represented as [34]:

\[
\hat{\mu}_i = \exp \left( \beta_0 + x_i \beta + e_i \right) = \exp \left( \beta_0 + x_i \beta \right) \exp \left( e_i \right) = \mu_i \delta_i,
\]

where \( e_i \) is assumed to be uncorrelated with a vector \( x_i \) of independent variables and is considered either as the combined effects of unobserved variables that have been omitted from the model [34] or as another source of pure randomness [39], i.e., unobserved heterogeneity. Another method for deriving the NBRM is from contagion. This happens when higher process maturity may increase performance (i.e., reduce the schedule deviation). Both unobserved heterogeneity and contagion approaches have the same NBRM distribution.

Because of the random error term, observations with the same \( x_i \) have different values of \( \mu_i \). The probability density function of a negative binomial distribution is:

\[
f(y_i|x_i) = \Gamma \left( y + a^{-1} \right) \\
\pi(y_i) = y_i! \Gamma \left( a^{-1} \right) \exp \left( a^{-1} \left( \frac{1}{\lambda_i} - 1 \right) \right). \tag{3}
\]

where \( E(\hat{\mu}_i) = \mu_i = \exp \left( \beta_0 + x_i \beta \right) \) is the same as a Poisson mean because \( \delta_i \) in Eq. (1) is commonly assumed a gamma distribution with parameter \( a \), where \( E(\delta_i) = 1 \) and \( \text{Var}(\delta_i) = a \). The variance is \( \text{Var}(y_i|x_i) = \mu_i + a \mu_i^2 \), where \( a \) is known as the dispersion parameter since the variance increases as \( a \) increases (\( a > 0 \) for overdispersion) [10]. Refer to Long [64] for a detailed explanation.

4.2. Determination of moderator variables

Identifying the moderator variable consists of a series of two analyses: moderated multiple regression (MMR) and subgroup analysis [81]. These two analyses categorize the hypothesized moderator variable into one of the specification variables described in Fig. 3. MMR has become a wide-spread method for evaluating the moderating effects of a categorical variable and is endorsed by several professional organizations that are listed in [23].

MMR, which sometimes only refers to Eq. (4), requires a set of three estimated means of the NBRM in Eq. (1):

\[
\hat{\mu}_i = \exp \left( \beta_0 + b_1 x_i \right); \tag{2}
\]

\[
\hat{\mu}_i = \exp \left( \beta_0 + b_1 x_i + b_2 z \right); \tag{3}
\]

\[
\hat{\mu}_i = \exp \left( \beta_0 + b_1 x_i + b_2 z + b_3 y \right). \tag{4}
\]

where \( x, y, \) and \( z \) are the independent, dependent, and hypothesized moderator variables, respectively. In MMR, the independent variable is usually centered by subtracting the mean in order to avoid the collinearity risk between the independent and the interaction term [4]. However, the dummy variable is not centered [44].

MMR results determine a type of variable for a hypothesized moderator according to the guideline in Fig. 3. Fig. 3 is interpreted as follows:

- If a hypothesized moderator belongs to Quadrant 1, i.e., \( b_3 = 0 \) and \( b_2 \neq 0 \), then the following step is to identify it as intervening, exogenous, antecedent, suppressor, or independent. Rosenberg [75] provides the means for identification.
- If a hypothesized moderator fits in Quadrant 2, i.e., \( b_3 = 0 \) and \( b_2 = 0 \), then the next step is a subgroup analysis. The moderator variable modifies the strength of the relationship between independent and dependent variables.
- If a hypothesized moderator fits in Quadrant 3, i.e., \( b_3 \neq 0 \) and \( b_2 = 0 \), then it is sometimes called a quasi-moderator in the psychometric literature since \( z \) is an independent variable (i.e., related to the dependent variable).
- Quadrant 4 indicates a pure-moderator.
In the subgroup analysis, two subgroups first were formed by splitting the sample by a median, quartile, or other type of split of the hypothesized variable. However, a median split was highly recommended because of the statistical power to detect the moderator effect [1]. Then, the base relationship was tested for each of the subgroups. If the strength of the relationship differed, then the hypothesized variable was a homologizer [81]. This moderated the strength of the relationship because the model error term in some groups is higher or lower than that in other subgroups.

4.3. Model stability

The model stability (also referred to as sensitivity) denotes the extent to which the model is affected by high-influence data points. Model stability can be assessed by evaluating the changes to model parameters found when the model is derived from different partitions of the data set [54]. Results based on a non-random sample, such as in this study, must include assurances of the models’ stability [65,66].

In this study, the stability of the analysis results was evaluated by a bootstrap resampling technique that samples B times from the original observation with replacement where B is a large number such as 1000 [21,65]. A measure for evaluating stability bias is defined as [65]:

$$\text{BIAS} = \frac{\sum_{b=1}^{B} t_b^*}{B} - 0,$$

where $t_b^*$ is a value of the description at the $b$th bootstrap sample, where $b = 1, \ldots, B$, and $\hat{\theta}$ is an estimated parameter value from an original data set. Typical descriptions include measures of central tendency (e.g., means or medians), dispersion (e.g., variance or control limits), or relationship (e.g., correlation coefficients or internal consistency).

The degree of bias was evaluated against the standard error (SE) of the estimated parameter (i.e., coefficient) distribution of $B$ replicates. The SE is computed as:

$$\text{SE} = \sqrt{\frac{\sum_{b=1}^{B} (t_b^* - \bar{t})^2}{B-1}},$$

where $\bar{t} = \sum_{b=1}^{B} t_b^* / B$.

If the bias is large relative to the SE, there is an instability problem. The judgment criterion for ignoring the bias is whether the absolute value of the bias is less than one-quarter of the size of the SE [21]. Hence, a description from the original data set can be considered to be stable. The lower and upper limits of the confidence interval of each parameter, like the coefficient is determined by methods such as a normal, basic, percentile, adjusted bootstrap percentile, or others [90].

5. Results

5.1. Descriptive statistics

Table 2 shows the regional ML distribution. As noted earlier, if two or more maintenance projects existed in an assessed organization, ML was counted two or more times. The most frequent ML was 2 (Repeatable) in both regions, followed by level 3 (Defined), and level 1 (Initial). These means and standard deviations were 2.07 and 0.73 in the US, and 2.13 and 0.67 in the non-US countries, respectively.

The proportion of organizations at ML 2 was clearly not larger than that at ML 1 in software industries throughout the world [27]. More likely, as early adopters of a new technology and specifically as organizations interested in software process improvement, the organizations in this study sample were drawn from the higher end of the maturity spectrum.

Table 2 also shows the number of assessed maintenance projects at each ML. A total of 55 projects (7.3%) showed schedule deviations: 47 schedule delays and eight projects ahead of schedule. We discuss this issue in Section 6.

Figs. 4 and 5 show the arithmetic mean (left side) and variance (right side) of the schedule deviation at each ML in both geographical region and organizational size, respectively. Though the values of means and variances were subject to a lack of robustness, the performance of schedule deviation improved with increasing ML in both geographical region and organizational size, which was consistent with the claim in capability/maturity models, including the SW-CMM.
5.2. Analysis results

5.2.1. Correlation coefficients

This study first investigated the correlation coefficients among the analyzed variables in order to obtain preliminary information and identify the risk of multicollinearity. As seen in Table 3, the correlation coefficient of \(-0.13\) between schedule deviation and ML indicated an inverse relationship. However, the high correlation between ML and its interaction terms in Table 3 indicates some risk of multicollinearity in the regression model.

5.2.2. Base relationship

Table 4 shows the results of our MMR for the three study hypotheses. All study coefficients were from centered independent variables except region and size. When centered independent variables were entered into the regression equations containing interaction terms, Eq. (4), the regression coefficients \(b_0\) and \(b_2\) differed from those of the regression analysis with un-centered raw data. However, centering in the NBRM did not change coefficients \(b_1\).
and \( b_1 \) and their test statistics. More on centering can be found in Cohen et al. [15] and Aguinis [3].

In the second row, the value \(-1.038\), denoting the base relationship, was statistically significant (accept Hypothesis 1), i.e., there is an underlying relationship between process maturity and schedule deviation in software maintenance. A bootstrap simulation of 1000 replicates showed model stability with a bias value of \(-0.005\) and an SE of 0.234. The bias was less than one-quarter of the size of its SE. Therefore, this study concluded that the estimated coefficients in the final model were stable. In addition, the upper limit of the bootstrap percentile confidence interval, \(-0.667\), also supported the acceptance of Hypothesis 1 at the one-sided level of 0.05.

First, the study evaluated the goodness-of-fit of the NBRM using pseudo-\( R^2 \) and \( \chi^2 \) goodness-of-fit tests \(^6\). They are known as useful methods to demonstrate how well the fitted model explains the data \([10]\). The \( \chi^2 \) goodness-of-fit value, exhibiting the aptness of the final model, gave a \( P \)-value of 0.406 (\( \chi^2 = 4 \), degrees of freedom = 4), which implied that the estimated value from the model was not different from the actual value. The dispersion parameter \( a \) in the NBRM had a value of 19.08 (\( P \)-value < 0.0001), i.e., over dispersion. Vuong’s test \([91]\) also rejected the null hypothesis, thereby showing NBRM and PRM to be indistinguishable.

Effect size is frequently required to measure the effectiveness of a theory in explaining or predicting an empirical observation. It is different from significance tests which evaluate the probability of obtaining the sampling outcome by chance. The correlation coefficient and the coefficient of determination (\( R^2 \)) in ordinary least squares (OLS) regression belong to the effect sizes. Cohen \([14]\] classifies the effect size of \( R^2 \) into three categories: small (0.0196), medium (0.0799), and large (0.25). A summarized table for effect sizes can be found in \([56]\). In this context, the base relationship in this study had a pseudo-\( R^2 \) value of 0.082 that corresponded to a small effect size in OLS interpretation. A meta-analytic review of previous empirical studies found a corrected \( R^2 \) of 0.078 in the relationship between user participation and system success \([84]\).

The statistical hypothesis of the base relationship was refined such that the mean of the schedule deviation proceeds to zero and the variance is also reduced as the process improvement proceeds from lower maturity to higher, as depicted in \([71, p. 28]\). A bootstrap simulation with 1000 replicates was used in this study to investigate the statistical hypothesis. The box-whisker plot in Fig. 6 shows the statistically significant reductions of both mean (left side) and variance (right side) in schedule deviation in the process improvement from ML 1 to ML 2. These reductions were consistent with the result of a previous study that showed a similarly large improvement from ML 1 to 2 \([29]\). They were also consistent with Figs. 4 and 5.

\(^{6}\) In \([9]\), a variety of pseudo-\( R^2 \) exists to measure goodness-of-fit in the NBRM. The pseudo-\( R^2 \) presented from this study computes the deviance reduction from a null model with only an intercept to the base model. The formula of our pseudo-\( R^2 \) is Eq. (2.11) (p. 214) in \([9]\).

\(^{7}\) A null hypothesis for \( \chi^2 \) goodness-of-fit is that no difference exists between actual counts and estimated counts, i.e., \( \chi^2 = \sum \left(O_i - E_i \right)^2 / E_i \), where \( O_i \) and \( E_i \) are observed and estimated schedule deviation, respectively (\( E_i \geq 5 \)). Thus, a large statistic and small \( P \)-value imply a poor model fit. The \( P \)-value is a right-tail probability \([10]\).

5.2.3. Results of testing region

In Table 4, row 3 (Hypothesis 2) shows that region is a significant variable with a positive coefficient. The pseudo-\( \Delta R^2 \) had a value of zero\(^8\). This implied that the interaction term did not explain the proportion of variance in the schedule deviation. Thus, it was not a moderator variable but an intervening, extraneous, antecedent, suppressor, or additional predictor variable, as shown in Quadrant 1 in Fig. 3. Therefore, Hypothesis 2 was rejected. As seen in Table 3, geographical region was significantly correlated to schedule deviation (dependent variable) but not to ML (independent variable). Thus, geographical region was deemed to be an independent variable because it probably serves as a proxy for other variables such as slack resources, education, specialization, and scale \([28, 86]\).

5.2.4. Results of testing size

The fourth row (Hypothesis 3) in Table 4 shows that neither organizational size itself nor an interaction term are significant (\( \Delta R^2 = 0 \)). That is, size was classified into Quadrant 2 in Fig. 3. Thus, subgroup analysis was conducted to investigate whether size moderates the strength of the base relationship. In the study’s subgroup analysis, the sample was split into small and large sizes by a medium value, following a recommendation from Aguinis and Stone-Romero \([1]\). Then, the base relationship was retested for the two data sets.

In OLS, the subgroup difference can be tested by using the Chow test \([11]\) or by comparing \( R^2 \) of the two subgroups \([81]\). The present study examined whether the two subgroups differed with respect to the pseudo-\( R^2 \). The result showed that the pseudo-\( R^2 \)s did not differ across the two subgroups that had a value of 0.085 (small size) and 0.083 (large size), i.e., no heterogeneity of the form of the relationship. The strength of the relationship between ML and schedule deviation did not depend on the magnitude of the error term. Hypothesis 3 was rejected.

5.2.5. Random variations

In the preceding models, we assumed that all of the projects in an assessed organization have the same level of maturity but have their own individual values of schedule deviation. However, any two maintenance projects in an organization might be more similar to one another because they share the same maintenance process context (application domain, business purpose, development methodology, etc.). In this context, the coefficient of maturity level may be assumed to have varied randomly across organizations. This assumption was analyzed by using multilevel/hierarchical modeling \([30, 79]\).

\(^{8}\) This is the same concept as the testing of \( H_0: b_2 = 0; H_1: b_2 \neq 0 \) with \( H_0: \Delta R^2 = 0; H_1: \Delta R^2 \neq 0 \) in OLS, where \( \Delta R^2 \) is the difference in \( R^2 \) between Eqs. (4) and (3).
In our multilevel regression of the NBRM, we tested the random slope of maturity by using a statistical routine, called GLIMMIX macro in SAS 9.1.3 [79]. The results showed that the coefficient of maturity level did not vary across organizations. Thus, we concluded that the relationship between ML and schedule deviation in software maintenance does not alter across accessed organizations. For brevity’s sake, the estimations are not reported here in detail.

5.2.6. Summary

We found, as expected, that assessed ML is in fact negatively related to schedule deviation in software maintenance. The results were quite robust in spite of the data limitations. The study analysis demonstrated that the base model is supported with a small to medium effect size. The present study also revealed the geographical region to be an independent variable. As in [17,23,32], organizational size did not influence the form or strength of the relationship between process maturity and schedule deviation.

Eq. (5) shows the result, including the two moderators and ML. Its pseudo-$R^2$ is 0.14.

$$\hat{\mu} = \exp(0.115_{(0.053)} - 1.211_{(0.0001)} \text{ML} + 1.235_{(0.002)} \text{Region} + 0.153_{(0.005)} \text{Size})$$

(5)

If a curvilinear relationship between dependent and independent variables exists, then it is possible to detect the interaction effects when such effects do not exist [15]. Although this study tested nonlinearity by adding a quadratic term for ML in the analyses, none showed significance.

6. Limitations and discussion

This study had a limited range only extending to ML 3 because of the low number of assessments at MLs 4 and 5. This is known as a range restriction as defined by a sample range divided by a population range in an independent variable. The range restriction influences the statistical power for detecting a moderating effect [1]. Therefore, researchers are encouraged to exert every effort to include higher level assessments in future studies.

The measure of process maturity was limited to three levels in the present study. As an alternative, we also tried using the two-point rating scale of “Fully Satisfied” (coded 1) and “Not Satisfied” (coded 0) for each KPA. A sum of the coded ratings was then used to create an alternative variable to express process maturity. This approach seems reasonable for research purposes in the case of partial information for maturity ratings. However, the sample size would have been reduced since some assessments reported only MLs. Regardless, the results using a data set including the KPA 0/1 rating only showed a negligible difference from those of the present study.

The criterion variable of schedule deviation used in this study is a self-reported, non-negative integer measured by month where a project may be ahead, behind, or on schedule. Using such a measure raises significant accuracy issues. Time ahead or behind schedule is measured in months, which probably introduces a rounding error in the projects’ replies. Moreover, the measure does not account for variations in project size and duration. For example, a 2-month delay in a 1-month project is treated the same as a 2-month delay in a 9-month project.

In particular, a very large proportion, approximately 92% of the projects in the maintenance phase of their lifecycles, reported being on schedule, which clearly contradicts both the results of previous studies and practical experience in the development phase. Several reasons may account for this divergence. The question that was used to measure schedule deviation only asked whether or not the project was on time, but the criteria for being on time were not specified. One likely conjecture is that many projects periodically modify their baseline schedule estimates to incorporate any incurred delays and thereby reduce the number of delay reports. Another possible conjecture is that assessments often include exemplary projects. Moreover, removing the ML 4 and 5 samples might not be necessary if we had a more finely grained unit of measure of schedule deviation than months, especially if there is in fact more contagion as ML increases.

As expected, the reported schedule deviation was higher for projects that were in lifecycle phases other than maintenance. For example, more than 25% of the projects in test and integration reported being a month or more behind schedule, which implied that 92% of the projects being within plus or minus one month on schedule in the maintenance phase was not supportively biased.

7. Final remarks

Although the SEI’s PAIS database contained the largest number of assessment cases available anywhere, the data set was not a random sample, and the results cannot be generalized to all SW-CMM
maintenance assessments conducted around the world. This sometimes is referred to as a threat to external validity due to the type of sampling [87]. Interpretation of the results should be limited to SW-CMM maintenance assessments reported to PAIS by the then-current base of SW-CMM users. This study is only one inductive instance of empirical evidence to support the proposition that higher process maturity leads to better maintenance performance. No single study on maintenance can be fully definitive. Similar studies must be conducted that include assessment results that may not have been well represented in the PAIS database. Such studies should include sample surveys as well as results from mini-assessments and similar “lighter weight” appraisals conducted on organizations that are not yet ready to invest in a full and comprehensive appraisal.

Schedule deviation is one of the performance criteria worth considering. Other measures of performance such as cost, productivity, quality, and customer satisfaction should be evaluated in future analyses of predictive validity. Those studies should include a reasonable number of other moderators, especially such as application domain, cultural factors, and enablers and disinfectors of process improvement. However, incorporating many such factors into a single model may be considered unmanageable and less prudent than starting with a more parsimonious model. Finally, we will replicate this study when a sufficient amount of data is available from CMMI appraisals.

Acknowledgments

Thanks are due first and foremost to the assessors, sponsors, and others who participated in the SW-CMM assessments. This work would not have been possible without the information that they regularly provided to the SEI. Mike Zucher, Kenny Smith, and Xiaobo Zhou provided invaluable support in extracting the data on which the study is based. Special thanks go to Sheila Rosenthal for her expert support with our references and to Lauren Heinz for helping in improving the readability of the document. We also express our thanks to our SEI colleagues – Will Hayes, Mike Konrad, Steve Masters, Jim McCurry, Mark Paulk, Mike Phillips, Gene Miluk, and Dave Zubrow – for their valuable comments on this work. The SEI is a Federally Funded Research and Development Center sponsored by the U.S. Department of Defense and operated by Carnegie Mellon University. The research was supported by a Korea University Foundation Grant (2006). This support is gratefully acknowledged. The authors thank the Editor and two anonymous referees whose comments and suggestions greatly helped to improve the contents and presentation of this paper.

References

