Early Phase Software Effort and Schedule Estimation Models

Wilson Rosa
IT Estimating Division
Naval Center for Cost Analysis
wilson.rosa@navy.mil

Barry Boehm and Brad Clark
Center for Systems and Software
University of Southern California
boehm@usc.edu
brad@software-metrics.com

Ray Madachy
Department of Systems Engineering
Naval Postgraduate School
rjmadach@nps.edu

Cheryl Jones and John McGarry
US Army RDECOM-ARDEC
cheryl.l.jones128.civ@mail.mil
john.mcgarry4.civ@mail.mil

Nick Lanham
Aircraft Estimating Division
Naval Center for Cost Analysis
Nick.lanham@navy.mil

Corinne Wallshein
IT Estimating Division
Naval Center for Cost Analysis
corinne.wallshein@navy.mil

ABSTRACT

Context: Software cost estimates have the highest utility at early elaboration phase. The dilemma, however, is that most input parameters needed for estimation are not available at this stage.

Objective: This study introduces a set of effort and schedule estimation models for software development projects in the early elaboration phase. It evaluates the direct impact of functional requirements on software development effort.

Method: The first set of models examines the effect of functional requirements, peak staff, and design scope on effort. The second set predicts development duration in terms of estimated effort. The statistical models are based on data from 40 military and IT programs delivered during the time period from 2006 to 2014.

Result: Statistical results show that estimated functional requirements and estimated peak staff are significant contributors to software development effort. Results also reveal that estimated or actual effort is a valid predictor of development duration.

Conclusion: These models may also be applicable to non-military sectors, to the extent that their practices are similar to those in the military.

General Terms

Management, Measurement, Design, Economics

Keywords

COCOMO, cost estimation, schedule estimation, functional requirements, elaboration phase, early phase

1. INTRODUCTION

1.1 Problem Statement

Defensible estimates are mostly needed at the early elaboration phase of a software-intensive system's definition and development. Mainstream cost models [4, 5, 6, 8, 9, 16, 19, 22, 23] are not very helpful at the early elaboration phase as most of these require a large number of input parameters that are not available until after preliminary design review.

1.2 Purpose Statement

The purpose of this study is to provide a set of software cost estimation models for projects at the early elaboration phase. It also examines the direct effect of functional requirements on software development cost. The data analysis framework in this study builds on causal relationships established in past studies [4, 5, 6, 10, 29].

1.3 Deficiencies in Past Studies

Most recent studies [1, 2, 3, 7, 10, 12, 18, 20, 28] on early phase software effort estimation have focused on using either COSMIC [11] or Function Point Analysis [15] as the independent size variable. Using these size measures at the early elaboration phase, however, pose a number of challenges. Since, COSMIC and FPA are derived from project size, such as functional requirements or base functional components, this leads to a larger Magnitude of Relative Error due to subjectivity in their calculation. COSMIC and FPA do not provide the means for software size collection because it requires an additional staff beyond the development and acquisition program management team. In addition, there are no automated counting tools for COSMIC or FPA sizing.

Only a single study [29] has specially examined the impact of project size on software development effort. However, this study measured project size in terms of base functional components, and used data reported at the computer software configuration item level from a single company. Whether estimated functional requirement is a valid predictor for software cost estimation remains an open question.

1.4 Significance of Proposed Study

This study will remedy these limitations in 3 ways:

- Measure the direct effect of estimated functional requirements on software development effort.
- Perform statistical analysis on parameters that are made available to analysts at early elaboration phase such as

    - Estimated functional requirements
    - Estimated peak staff
- Use dataset from multiple companies reported at the total project level rather than by Computer Software Configuration Items, as requirements counts at elaboration phase are provided at the aggregate level.

1.5 Research Questions

This study will address three research questions:

Question 1: Does estimated functional requirement relate to actual development effort?

Question 2: Do estimated functional requirement and estimated peak staff relate to actual effort?

Question 3: Does estimated effort relate to actual development duration?

2. LITERATURE REVIEW

This section is divided into three. The first sub-section defines the independent variable, functional requirements. The second describes how functional requirements is collected and counted in the Department of Defense (DoD). The third sub-section examines scholarly literature on early phase software cost estimation.

2.1 Defining Functional Requirements

Functional requirements outline the detailed functions of a software system and its elements. A function is described as a set of inputs, behavior, and outputs. According to the IEEE Standard 830 [17], functional requirements should define the fundamental actions that must take place in the software in accepting and processing the inputs and in processing and generating the outputs. These are generally listed as "shall" statements starting with "The system shall…"

Functional requirements [17] should include:

- Validity checks on the inputs
- Exact sequence of operations
- Responses to abnormal situations, including
    - Overflow
    - Communication facilities
    - Error handling and recovery
- Effect of parameters
- Relationship of outputs to inputs, including
    - Input/output sequences
    - Formulas for input to output conversion

Estimating software cost using functional requirements has its advantages over the traditional Functional Size Measures (FSM) – COSMIC and Function Point Analysis (FPA). Since functional requirements are the foundation for deriving FSM, these can be counted without subjectivity. In addition, functional requirements are available much earlier than COSMIC and FPA. Despite these advantages, most cost analysts do not use functional requirements for early phase estimation as there are no publicized requirements-based estimation models.

2.2 Sizing Functional Requirements in DoD

In DoD, functional requirements are written in accordance with the Software Requirements Specification Data Item Description (SRS-DID) [14]. This standard specifies the requirements and the methods to be used to ensure that each functional requirement has been met. Developers use the SRS-DID to write and report the software requirements in the Software Requirements Specification (SRS). The list of functional requirements may be delivered in the developer's format, and may reside in a computer-aided software engineering or other automated tool such as Dynamic Object-Oriented System.

As for sizing functional requirements, there are no standard counting rules in DoD. However, software development contracts are required to report the number of functional requirements in a standard form called Software Resource Data Report (SRDR). The SRDR [13] is required for all contracts with software effort greater than $20 million. Developers submit the SRDR Initial Developer Report 60 days after contract award, and the SRDR Final Developer Report 60 days after contract completion. The initial report provides the estimated number of functional requirements whereas the final report provides the actual number of requirements. Table 1 defines functional requirements in accordance with the SRDR Initial and Final Developer Report.

Table 1 Functional Requirements as defined in the SRDR Questionnaire

Report	Software Requirements Definition
SRDR Initial Developer Report	Provide the estimated number of software requirements. The method of counting estimated number of requirements implemented by the development software will be the same as that ultimately used for counting the actual, as-built requirements (as reported in the SRDR Final Developer Report). Do not count requirements concerning external interfaces not under project control (see next item, "Total Requirements"). The SRDR Data Dictionary shall provide both a definition of what types of requirements are included in the count (i.e., functional, security, safety, other derived requirements, etc.) and the units (e.g., "shalls," "sections," paragraphs, etc.) and counting methods used.
SRDR Final Developer Report	Provide the actual number of software requirements. The method of counting actual number of requirements implemented by the development software must be the same as that used for counting estimated requirements (as reported in the SRDR Initial Development Report). Do not count requirements concerning external interfaces not under project control (see next item, "Total Requirements"). The SRDR Data Dictionary shall provide both a definition of what types of requirements are included in the count (i.e., functional, security, safety, other derived requirements, etc.) and the units (e.g., "shalls," "sections," paragraphs, etc.) and counting methods used.

2.3 Studies on Early Phase Cost Estimation

Most recent studies [1, 2, 3, 7, 10, 12, 18, 20, 28] on early phase software effort estimation have focused on using Functional Size Measurements such as COSMIC and Function Point Analysis as the predictor variable. Only a single study has examined the direct effect of project size on effort estimation:

Tunalilar and **Demirors** [29] investigated the impact of the project size on software development effort. Project size was measured in terms of the base functional components (BFC) of COSMIC measurement method. The data used in this study was provided by a single company on 40 software configuration items. The authors conducted single and multivariate regression analyses to determine whether the BFC inputs (entry, exit, read, write) should be analyzed at the aggregate level. Results showed that using the individual BFC, instead of COSMIC Functional Size Measurement, is appropriate for effort estimation at early phase. Although the focus of this paper was on assessing the impact of the BFC inputs, it proved the validity of using functional user requirements as a means for early phase effort estimation.

3. RESEARCH METHOD

3.1 Population and Sample

The sample was identified as 40 completed projects that have been implemented for the United States Department of Defense. This study focused on projects reported at the total level rather than by computer software configuration items (CSCIs), as requirements count at elaboration phase are provided at the aggregate level. The breakout according to operating environment is shown in Figure 1. Of the 40 projects, 30 are defense systems (in light red) while 10 (in light blue) are information technology systems.

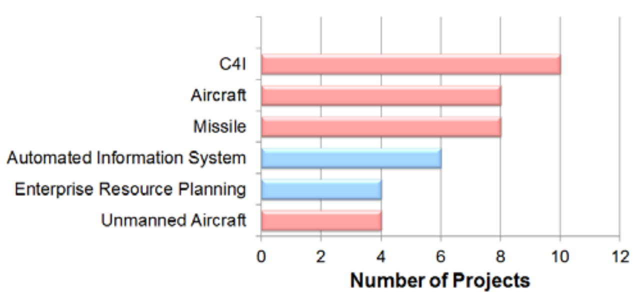

Figure 1 Number of Projects by Operating Environment

Figure 2 shows the project breakout by delivery year. The time period does not pose a threat to validity as the chart confirms that the dataset is based on very recent projects.

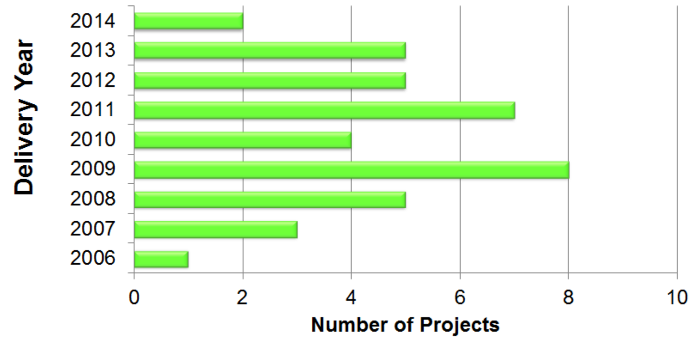

Figure 2 Number of Projects by Delivery Year

3.2 Instrumentation

Data were collected by means of a questionnaire [13] containing over 20 items. The data collection questionnaire used in the study was from an existing one; *Software Resource Data Report (SRDR)* questionnaire (DoD 5000.4 M-2, 2002). The questionnaire allowed the collection on data items such as functional requirements, source lines of code, effort, and schedule [19, 21, 24, 25, 26].

The SRDR Initial Developer Report was used to collect the estimated functional requirements, estimated peak staff and estimated effort. The SRDR Final Developer Report was used to collect the actual effort, actual development duration, actual functional requirements, and actual peak staff. The SRDR questionnaires and forms can be accessed via the following links:

http://cade.osd.mil/Files/Policy/2011-SRDRInitial.pdf
http://cade.osd.mil/Files/Policy/Initial_Developer_Report.xlsx

http://cade.osd.mil/Files/Policy/2011-SRDRFinal.pdf
http://cade.osd.mil/Files/Policy/Final_Developer_Report.xlsx

3.3 Model and Variable Selection

The effort and schedule equation forms are based on the COCOMO II Post-Architecture and schedule models [5, 6] without the effort multipliers. The variables used in the study are identified in Table 2.

Table 2: Variable Names and Definitions

Variable Name	Type	Definition
Actual Effort	Dependent	Actual effort (in Person-Month) as reported in the SRDR Final Developer Report. Captures all the associated engineering effort, by the developer for analyzing, designing, coding, testing, integrating, and managing the software development project.
Actual Duration	Dependent	The actual software development duration. The time required to complete all activities up to the point of development test & evaluation.
Estimated Effort	Independent	Estimated effort (in Person-Month) as reported in the SRDR Initial Developer Report. Captures all the associated engineering effort, by the developer for analyzing, designing, coding, testing, integrating, and managing the software development project.
Estimated Total Requirements	Independent	The estimated number of functional requirements as reported in the Initial SRDR Developer Report.
Estimated Peak Staff	Independent	The estimated peak staff measured in terms of full-time equivalents.

3.4 Model Validity Measures

The measures for validating the model are described in Table 3.

Table 3 Model Validity Measures

Measure	**Symbol**	**Description**
Coefficient of Determination	R^2	The Coefficient of Determination shows how much variation in dependent variable is explained by the regression equation.
Coefficient of Variation	CV	Percentage expression of the standard error compared to the mean of dependent variable. A relative measure allowing direct comparison among models.
Measure of Magnitude	F-test	The value of the F test is the square of the equivalent t test; the bigger it is, the smaller the probability that the difference could occur by chance.
P-value	α	Level of statistical significance established through the coefficient alpha ($p \leq \alpha$).

4. DATA ANALYSIS

4.1 Descriptive Statistics

Figure 3 compares the median number of functional requirements by mission area. The boxplot shows higher requirements for defense projects compared to information technology (IT) projects.

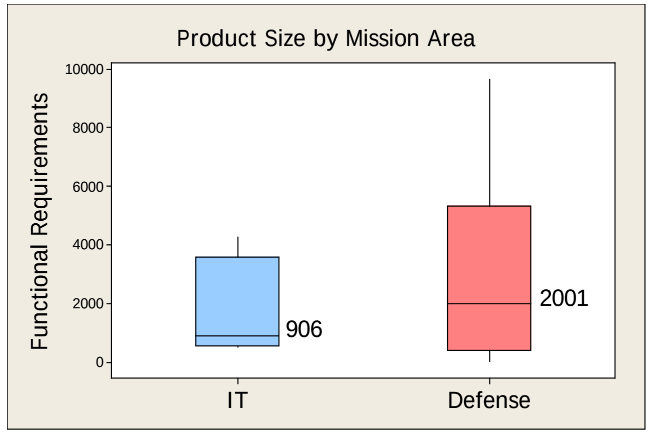

Figure 3 Product Size by Mission Area

Figure 4 compares the median software development duration by mission area. The result shows longer duration for defense systems due to interdependencies with hardware design and manufacturing schedules.

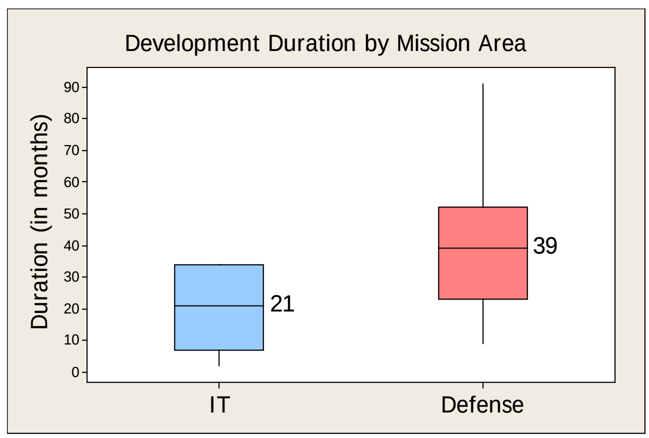

Figure 4 Project Duration by Mission Area

Figure 5 shows the median effort growth by project type. Effort growth is the percent difference between baseline and actual effort. The result shows higher effort growth for defense systems when compared to enterprise resource planning (ERP) and automated information systems (AIS). This finding is consistent with the perception that defense systems experience more overruns than IT systems due to greater complexity.

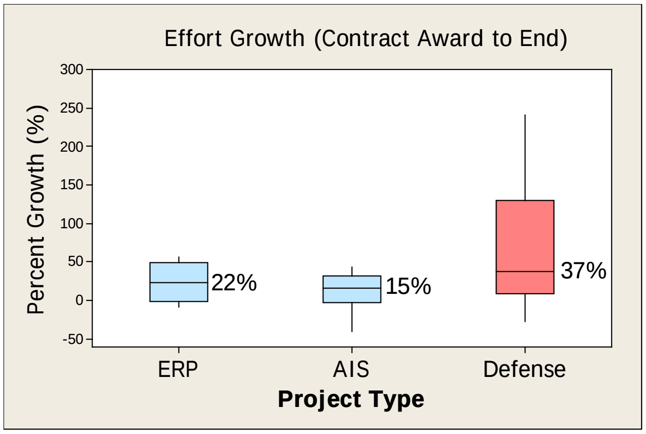

Figure 5 Effort Growth by Project Type

Figure 6 compares productivity (hours per requirement) by project scope. The result shows no significant difference between enhancement and new projects.

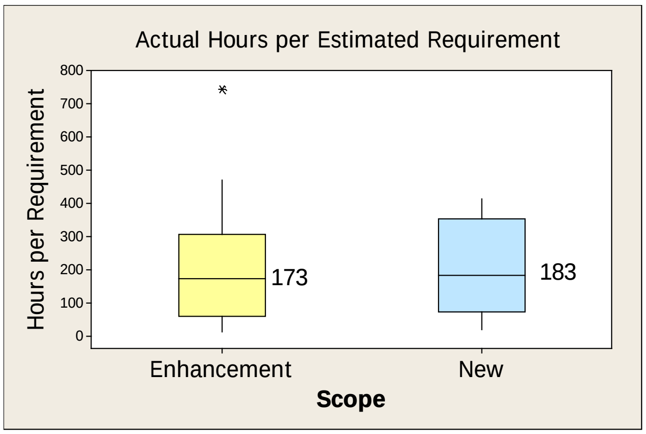

Figure 6 Productivity Comparison by Project Scope

Figure 7 compares the median productivity (hours per requirement) by project type. The boxplot shows higher "hours per requirement" for enterprise resource planning (ERP) projects as these require additional effort for customizing and configuring third party software such as SAP and Oracle Release 12. On the other hand, automated information systems (AIS) show a lower "hours per requirement" as these reside on a commercial-the-shelf hardware platform.

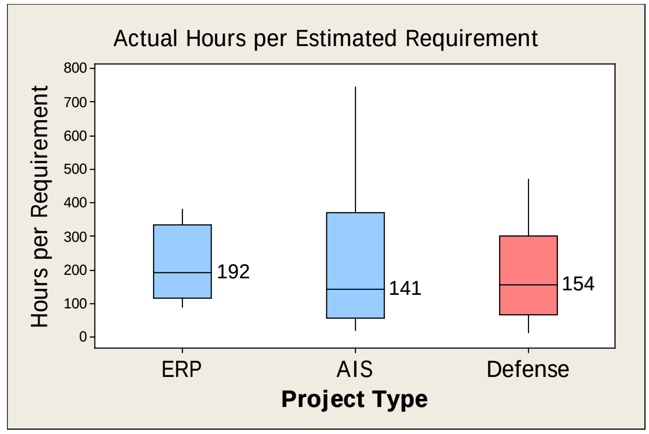

Figure 7 Productivity Comparison

4.2 Pairwise Correlation Analysis

Pairwise correlation analysis [27] was chosen over structural equation modeling (SEM) as the number of observations (40) was far below the minimum observations (200) needed to perform SEM [21].

The correlation coefficient ranges from negative (≥-1) to positive (≤+1) values. The larger the absolute value, the stronger the correlation. Table 4 shows the correlation coefficients between effort, functional requirements, and other multi-dimensional metrics. The criterion for acceptance is 0.4. The analysis results indicate the following:

- The correlation coefficient for the relationship between estimated total requirements and actual effort (0.7) is very strong, indicating that when requirements are added to the project, the effort tends to increase.
- The correlation for the relationship between estimated peak staff and actual effort (0.4) is strong, indicating that when more resources are added to the project, the effort tends to be greater.
- The correlation between estimated effort and actual duration (0.2) is weak. In contrast, the correlation between actual effort and actual duration is strong (0.6), indicating that when effort grows the project takes longer to complete.
- The correlation for the relationship between estimated total requirements and actual duration (0.4) is strong, indicating that when more requirements are added to the project, the project takes longer to complete.
- The requirements volatility and scope parameters showed a poor correlation to effort.

Table 4 Pairwise Correlation Matrix

	Actual Effort	Actual Duration	Estimated Total Requirements	Actual Total Requirements	Estimated Effort	Actual Peak Staff	Estimated Peak Staff
Actual Effort	1.0	0.6	0.7	0.7	0.6	0.4	0.4
Actual Duration	0.6	1.0	0.4	0.4	0.2	-0.2	-0.2
Estimated Total Requirement	0.7	0.4	1.0	0.9	0.6	0.2	0.2
Actual Total Requirement	0.7	0.4	0.9	1.0	0.6	0.3	0.3
Estimated Effort	0.6	0.2	0.6	0.6	1.0	0.6	0.6
Actual Peak Staff	0.4	-0.2	0.2	0.3	0.6	1.0	1.0
Estimated Peak Staff	0.4	-0.2	0.2	0.3	0.6	1.0	1.0
RVOL	0.1	0.1	0.0	0.0	0.1	0.1	0.1
Scope	0.2	-0.1	0.1	0.1	0.1	0.4	0.4

5. EFFORT MODEL RESULT

5.1 EFFORT MODEL 1

The equation below predicts software engineering effort based on the estimated total functional requirements. This model is appropriate when requirement counts are provided at the aggregate level. The regression model is applicable for projects ranging between 25 and 13900 estimated functional requirements, and different mission areas (defense and information technology).

$$PM = 22.37 \times eREQ^{0.5862} \qquad \textit{Eq. (1)}$$

Where:

PM = Engineering Labor in Person Months

eREQ = Estimated functional requirements

Figure 8 displays the actual versus predicted plot for model 1. In this plot, each point is effort measured in terms of person-months (PM). The points around the straight line indicates a strong correlation between the actual and predicted values.

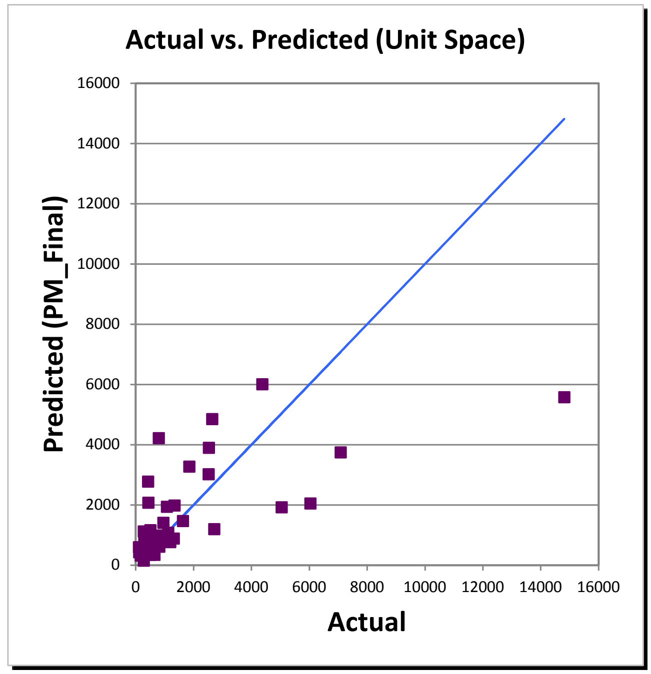

Figure 8 Actual vs Predicted Plot for Model 1

Table 5 displays the statistical significant for Model 1. The result shows that the effect of estimated functional requirements (*t-statistic is 7.3870*) on effort is "highly" significant. The intercept was not removed from the equation as the effect is marginally significant.

Table 5 Model 1 Diagnostic Test

Variable	**Coefficient**	**T stat**
Intercept	22.37	1.8262
eREQ	0.5862	7.3870

Table 6 displays the accuracy of Model 1. The R^2 value shows that 76% percent of the variation of software development effort has been explained by the regression.

Table 6 Model 1 Accuracy

Measure	**Symbol**	**Result**
Coefficient of Determination	R^2	76
Coefficient of Variation	CV	64
Mean Absolute Deviation	MAD	58
Sample Size	N	40
Mean (Effort)	Mean	1739

5.2 EFFORT MODEL 2

The equation below predicts software engineering effort based on the estimated total functional requirements and estimated peak staff. This model is appropriate when requirements and staffing levels are provided at the aggregate level. The regression model is applicable for projects ranging between 25 and 13900 estimated functional requirements, and different mission areas (defense and information technology).

PM = 11.82 x **eREQ**$^{0.4347}$ x **eStaff**$^{0.4269}$ ***Eq. (2)***

$$PM = 11.82 \times eREQ^{0.4347} \times eStaff^{0.4269} \quad \text{Eq. (2)}$$

Where:

PM = Engineering Labor in Person Months

eREQ = Estimated functional requirements

eStaff Estimated peak staff measured in terms of full-time equivalent

Figure 9 displays the actual versus predicted plot for model 2. In this plot, each point is effort measured in terms of person-months (PM). The points around the straight line indicates there is a strong correlation between the actual and predicted values. The plot also shows that Model 2 is more accurate than Model 1 as the points are closer to the regression line.

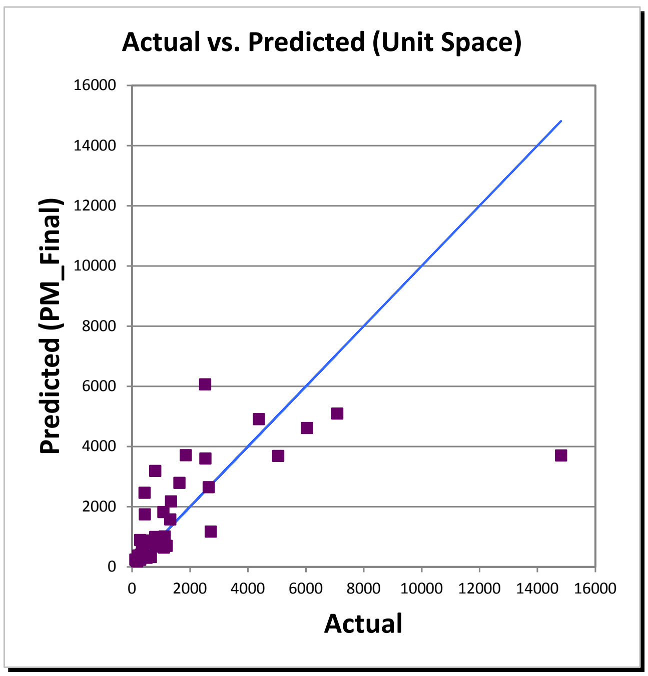

Figure 9 Actual vs Predicted Effort for Model 2

Table 7 displays the statistical significant for Model 2. The result shows that the effects of estimated functional requirements (*t-statistic is 4.7140*) and estimated peak staff (t-statistic is 3.5372) on effort are "highly" significant. The intercept was not removed from the model as the effect on effort is marginally significant.

Table 7 Model 2 Diagnostic Test

Variable	**Coefficient**	**T stat**
Intercept	11.82	1.8790
eREQ	0,4347	4.7140
eStaff	0.4269	3.5372

Table 8 displays the accuracy of Model 2. The R^2 value shows that 78% percent of the variation of software development effort has been explained by the regression.

Table 8 Model 2 Accuracy

Measure	**Symbol**	**Result**
Coefficient of Determination	R^2	78
Coefficient of Variation	CV	54
Mean Absolute Deviation	MAD	47
Sample Size	N	40
Mean (Effort)	Mean	1739

6. SCHEDULE MODEL RESULT

The equation below predicts software development duration using the estimated effort (in person-months) as the independent variable. This model is appropriate when estimated effort is reported at the aggregate level. The regression model is applicable for projects ranging between 17 and 7132 estimated person-months, and different mission areas (defense and information technology).

TDEV = e**PM**$^{0.5290}$ ***Eq. (3)***

Where:

TDEV = Software development duration in months

ePM = Estimated Effort in Person Months

Figure 10 displays the actual versus predicted plot for model 3. In this plot, each point is development duration measured in terms of months. The points around the straight line indicate there is a correlation between the actual and predicted values. The few outliers, however, were not removed as the points were verified.

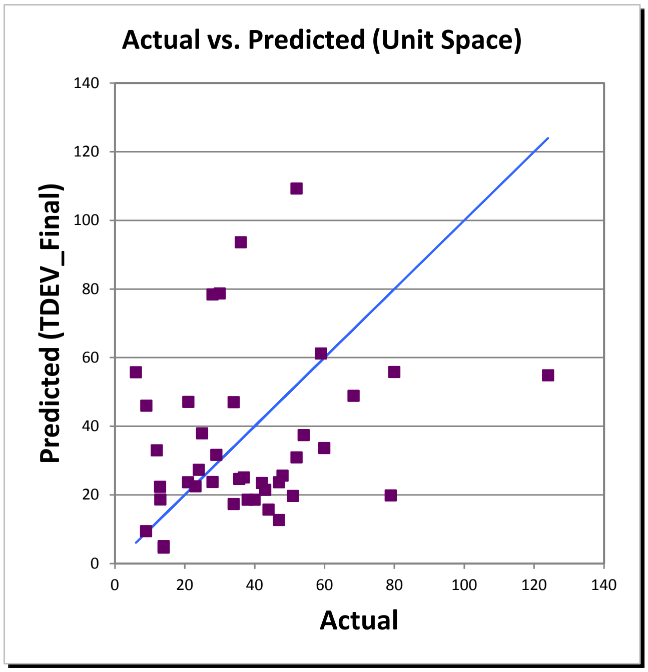

Figure 10 Actual vs Predicted Duration for Model 3

Table 9 displays the statistical significant for Model 3. The result shows that the effect of estimated effort (*t-statistic is 26.14*) on development duration is "highly" significant. The intercept, however, was removed from the model as the effect was not significant.

Table 9 Model 3 Diagnostic Test

Variable	**Coefficient**	**T stat**
ePM	0.529	26.14

Table 10 displays the accuracy of Model 3. The R^2 value shows that 94% percent of the variation of software development duration has been explained by the regression.

Table 10 Model 3 Accuracy

Measure	**Symbol**	**Result**
Coefficient of Determination	R^2	94
Coefficient of Variation	CV	60
Sample Size	N	40
Mean (Duration)	Mean	38

7. Conclusion

This study introduced a set of effort and schedule estimation models for software development projects at the early elaboration phase using data from 40 projects reported at the aggregate level.

7.1 Primary Findings

Results showed that estimated functional requirements is a significant contributor to development effort. The regression model explains 76% of the variation in software development effort. The variation in effort becomes more significant (78%) when estimated peak staff is added to the regression model. Thus, the effect of estimated functional requirements on software development effort shall be interpreted along with estimated peak staff.

Estimated effort is a significant contributor to development duration. The regression equation explains 94% of the variation in schedule.

7.2 Model Usage and Limitations

Although the model in this study is not highly precise, it has the advantage of providing information on its relative accuracy. The model may also be applicable to non-military sectors, to the extent that their practices are similar to those in the military.

- Since the data was collected at the aggregate level, the resulting estimation models are not appropriate for projects reported at the CSCI level.
- These models shall not be used for projects at inception phase when requirements are provided at a higher level.
- Do not use Model 1 or Model 2 if your input parameter is outside of the effort model range.
- Do not use Model 3 if your input parameter is outside of the schedule model range.

7.3 Future Work

Develop similar effort and schedule estimation models using project data reported at the CSCI level.

Examine the impact of functional requirements along with requirements volatility, process maturity, and percent reuse.

8. REFERENCES

[1] Attarzadeh, I., Mehranzadeh, A., Barati, A., 2012. Proposing an Enhanced Artificial Neural Network Prediction Model to Improve the Accuracy in Software Effort Estimation. Computational Intelligence, Communication Systems and Networks (CICSyN), 2012 Fourth International Conference on, Dept. of Comput. Eng., Islamic Azad Univ., Dezful, Iran, pp.167-172.

[2] Attarzadeh, I., Siew Hock Ow., 2011. Improving estimation accuracy of the COCOMO II using an adaptive fuzzy logic model. Fuzzy Systems (FUZZ), 2011 IEEE International Conference on, Dept. of Software Eng., Univ. of Malaya, Kuala Lumpur, Malaysia, pp.2458-2464.

[3] Aysolmaz, B., Demirors, O., 2014. Automated Functional Size Estimation Using Business Process Models with UPROM Method. Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2014 Joint Conference of the International Workshop on, Inf. Inst., Middle East Tech. Univ., Ankara, Turkey, pp. 114-124.

[4] Boehm B., 1981. Software Engineering Economics. Englewood Cliffs, NJ, Prentice-Hall.

[5] Boehm B., Abts C., Brown W., Chulani S., Clark B., Horowitz E., Madachy R., Reifer D., Steece B., 2000. Software Cost Estimation with COCOMO II. Prentice-Hall.

[6] Boehm, B., 2000. Safe and simple software cost analysis. *Software, IEEE*, 17(5), pp. 14–17.

[7] Braz, M.R., Vergilio, S.R., 2006. Software Effort Estimation Based on Use Cases. Computer Software and Applications Conference, 2006. COMPSAC '06. 30th Annual International, Dept. of Comput. Sci., Fed. Univ. of Parana, Curitiba, pp. 221-228.

[8] Clark, B., Devnani-Chulani, S., and Boehm, B., 1998. Calibrating the COCOMO II Post-Architecture model. *Proc. Int'l Conf. Software Eng. (ICSE '98)*, pp. 477–480.

[9] Clark, B. K., 1997. The Effects of Software Process Maturity on Software Development Effort. Technical Report No. AAT 9816016, University of Southern California, CA.

[10] Condori-Fernandez, N., Daneva, M., Buglione, L., Ormanjieva, O., 2010. Experimental Study Using Functional Size Measurement in Building Estimation Models for Software Project Size. 2010 Eighth ACIS International Conference on Software Engineering Research, Management and Applications.

[11] COSMIC., 2009. The Common Software Measurement International Consortium. *The COSMIC Functional Size Measurement Method* . Version 3.0.1. http://nesma.org/themes/sizing/cosmic-fsm, site accessed on 11 April 2015.

[12] Cuadrado-Gallego, J.J., Rodriguez-Soria, P., Gonzalez, A., Castelo, D., Hakimuddin, S., 2010. Early Functional Size Estimation with IFPUG Unit Modified. Computer and Information Science (ICIS), 2010 IEEE/ACIS 9th International Conference on, Comput. Sci. Dept., Univ. of Alcala, Alcala de Henares, Spain, pp. 729-733.

[13] Department of Defense., 2011. Software Resource Data Report. http://dcarc.cape.osd.mil/Files/Policy/2011-SRDRFinal.pdf , Site accessed on 4 April 2015.

[14] Department of Defense., 1994. Software Requirement Specification Data Item Description. http://www.abelia.com/498pdf/SRS-DID.PDF, site accessed on 4 April 2015.

[15] Furey, S., 1997. Why We Should Use Function Points [software metrics]. *Software, IEEE*, 14(2), pp. 28–30.

[16] Galorath., 2001. *SEER-SEM™ User's Manual*. Galorath, Inc., El Segundo, CA, pp. 1-4, 6-6, 15-6.

[17] IEEE Std 830., 1998. IEEE Recommended Practice for Software Requirements Specifications. http://standards.ieee.org/findstds/standard/830-1998.html, site accessed on 12 April 2015.

[18] Kaya, M., Demirors, O.,2011. E-Cosmic: A Business Process Model Based Functional Size Estimation Approach. Software Engineering and Advanced Applications (SEAA), 2011 37th EUROMICRO Conference on, Inf. Inst., Middle East Tech. Univ., Ankara, Turkey, pp. 404-410.

[19] Madachy, R., Boehm, B., Clark, B., Tan, T., Rosa, W., 2011. US DoD Application Domain Empirical Software Cost Analysis. *2011 International Symposium on Empirical Software Engineering and Measurement*, pp. 392 – 395.

[20] Malik A., 2010. Quantitative and Qualitative Analyses of Requirements Elaboration for Early Software Size Estimation. PhD Dissertation, Computer Science Department, University of Southern California.

[21] Lipkin, I., 2011. Test Software Development Project Productivity Model. Dissertation Thesis, University of Toledo, Toledo, OH.

[22] PRICE., 1998. Your Guide to PRICE S: Estimating Cost and Schedule of Software Development and Support. PRICE Systems, L.L.C, Mount Laurel, NJ, pp. 1–452.

[23] QSM., 2001. *User's Guide to SLIM-Estimate 5.0*, 2nd Edition. QSM, Inc., McLean,VA, pp. 1–360.

[24] Rosa,W., Madachy, R., Boehm, B., Clark, B., 2014. Simple empirical software effort estimation model. In *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement* (ESEM '14), ACM, New York, NY, USA, Article 43 , 4 pages.

[25] Rosa W., Packard T., Krupanand A., James Bilbro J., Hodal M., 2013. COTS integration and estimation for ERP. Journal of Systems and Software, 86(2), pp. 538-550.

[26] Tan, T., 2012. Domain-Based Effort Distribution Model for Software Cost Estimation. PhD Dissertation, Computer Science Department, University of Southern California.

[27] TECOLOTE, Inc., 2014. Automated Cost Estimating Integrated Tools: CO$TAT. http://ww.aceit.com/Pages/Products/ProductPage.aspx?id=f638a6d8-60e9-414a-9970-7fed249b9d25, site accessed on 1 July 2014.

[28] Tsunoda, M.,Kamei, Y., Toda, K., Nagappan, M., Fushida, K., Ubayashi, N., 2013. Revisiting software development effort estimation based on early phase development activities. Mining Software Repositories (MSR), 2013 10th IEEE Working Conference on, Toyo Univ., Saitama, Japan, pp. 429-438.

[29] Tunalilar, S., Demirors, O., 2012. EFES: An Effort Estimation Methodology. 2012 Joint Conference of the 22nd International Workshop on Software Measurement and the 2012 Seventh International Conference on Software Process and Product Measurement, pp. 205-214.