Used SLIM Model to Estimate Software Cost

Abeer Salim Jamil
Computer Science& Information Systems Department
Mansour University Collage (MUC)

ABSTRACT

Software cost estimation is the process of predicting the amount of effort required to build a software system . Models provides one or more mathematical algorithms that compute cost as a function of a number of variables, size is primary cost factor in most models and can be measuring using lines of code (**LOC**). Models used to estimate cost of the software, the SLIM is used to estimate the constraint model and SLIM suitable to for large projects and has fewer parameters needed to generate an estimate.

In this paper implement **SLIM** Model equation over the two different projects, we make the analysis of these two projects and conclusion relation between effort and duration and this relation is increase with increase the size of project.

Key words: SLIM Model , Norden – Rayleigh Curve ,Software Equation,
Manpower –Buildup Equation

1. Introduction

Software Cost estimation is the process of predicting the mount of effort required to build a software system. cost estimates are needed throughout the software lifecycle. Preliminary estimates are required to determine the feasibility of a project. Detailed estimates are needed to assist project planning. The actual effort for individual tasks is compared with estimated and planned values, enabling project managers to reallocate resources when necessary.**[1,2,3]**

Most of the work in the cost estimation field has focused on algorithmic cost modeling. In this process cost are analyzed using mathematical formulas linking costs or inputs with metrics to produce an estimated output. The formulas used in a formal model arise from the analysis of data.**[1,3]**

Software Life – Cycle Model(SLIM) is based on Putnam's analysis of the life-cycle in terms of a so called Rayleigh distribution of project personal level versus time. It makes use of a so called Rayleigh curve to estimate project effort, schedule and defect rate. In Software Life Cycle Management (SLIM), productivity is used to link the basic Rayleigh manpower distribution model to the software development characteristics o f size and technology factors,
productivity, P, is the ratio of software product size , S, and development effort E that is :-**[4,8]**

$$P = \frac{S}{E} \qquad \dots (1)$$

The Rayleigh curve used to define the distribution of effort is modeled by the differential equation :-**[5]**

$$dy = 2Kate^{-at\wedge 2} \quad \ldots (2)$$

2. SLIM Model (Software Life cycle Management)

Putnam's SLIM is one of the first algorithmic model. It is based on the Norden / Rayleigh function and generally known as a macro estimation model (It is for large project). SLIM can record analyze data from privously completed projects which are then used to calibrate then a set of questions can be answered to get values of manpower buildup existing database. **[7,8]**

SLIM enables a software cost estimator to perform the following functions:- **[8,9]**

1. Calibration fine tuning the model to represent the local software development environment by interpreting a historical database of past projects.

2. Build an information model of the software system , collection software characteristics, personal attributes, computer attributes.

3. Software sizing SLIM uses an automated version of the lines of code (LOC) costing technique.

The Algorithm used is **:-[9]**

$$K=(LOC / (C* t^{4/3}))*3 \quad .. \quad (3)$$

K is the total life cycle effort in working years, **t** is development and the **C** is the technology constant, combining the effort of
using tools, languages, methodology and quality assurance (QA) time in years. The value of technology constant varies from 610 to 57314. for easy, experienced projects technology constant is high.

Putman developed a constraint model called SLIM to be applied to project exceeding 70,000 lines of code. Putman's model assumes that effort for software projects is distributed similarly to a collection of Rayleigh curves. Putman suggests that staffing rises smoothly during the project and then drops sharply during acceptance testing. **[11,12]**

The SLIM model is expressed as two equations describing relation between the development effort and the schedule , the first equation, called the software equation, states that development effort is proportional to the cube of the size and inversely proportional to the fourth power of the development time. The second equation , the manpower-buildup equation, states that the effort is proportional to the development time. **[10,11]**

3. The Norden – Rayleigh Curve [10,12]

The Norden / Rayleigh Curve represents manpower as a function of time. Norden observed that the Rayleigh distribution provides a good approximation of the manpower curve for various hardware development process.

SLIM uses separate Rayleigh curve for design and code, test and validation, maintenance , and management . A Rayleigh curve is show in Figure (**1**).

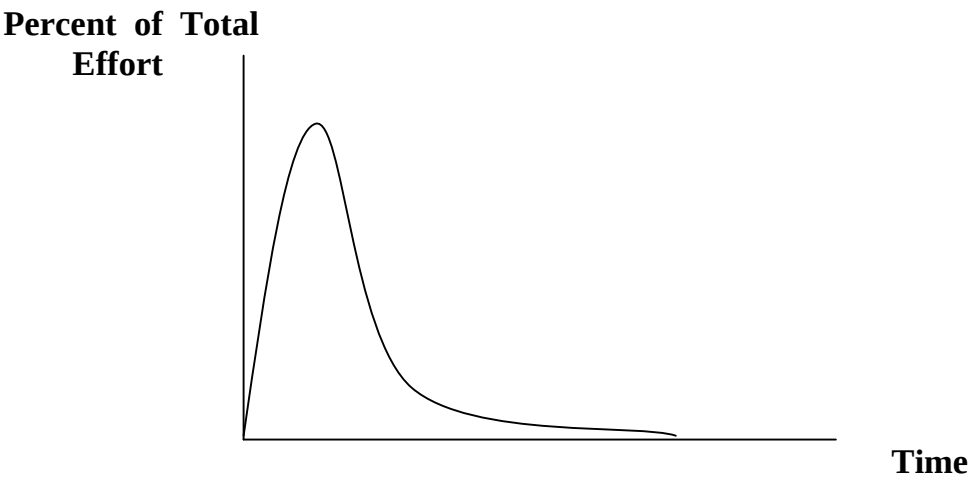

Figure (1) Rayleigh Curve

Development effort is assumed to represent only 40 percent of the total life cycle cost. Requirements specification is not included in the model. Estimation using SLIM is not expected to take place until design and coding.

4. The Software Equation [7,9,12]

Putman used some empirical observations a bout productivity levels to derive the software equation from the basic Rayliegh curve formula . The software equation is expressed as :- **[6]**

Size= CE^1/3(t^4/3) …..(4)

Where

C is a technology factor, **E** is the total project effort in person years, and **t** is the elapsed time to delivery in years.

The technology factor is a composite cost driver involving 14 components. It primarily reflects :-

1. Overall process maturity and management practices.
2. The extent to which good software engineering practices are used.
3. The level of programming languages used.
4. The state of the software environment.
5. The skills and experience of the software team.
6. The complexity of the application.

The software equation includes a fourth power and there has strong implications for resource allocation on large projects . Relatively small extensions in delivery a late can result in substantial reductions in effort .

5. The Manpower – Buildup Equation

The allow effort estimation, Putman introduced the manpower-buildup equation:-[**7**]

D=E/t^ 3 … (5)

Where

D is a constant called manpower acceleration, **E** is total project effort in years , and **t** is the elapsed time to delivery in years.

The manpower acceleration is 12.3 for new software with many interfaces and interaction with other systems, 15 for standalone systems, and 27 for reimplementation of existing systems. **[7,8]**

Using the software and manpower – buildup equations, we can solve for effort **:-[12]**

E=(S/C) ^9/7(d^4/7) … (6)

This equation is interesting because it shows that effort is proportional to size to the power 9/7 or ~ 1.286, which is similar to Boehm's which ranges from 1.05 to 1.20.

6. Criteria for Evaluating a model [9,10]

1. Constructiveness :- can a user tell why the model gives the estimates it
   does ? dose it help the user understand the software job to be done ?

2. Stability: - Do small differences in inputs produce small differences in
   output cost estimates?

3. Scope: - does the model cover the class of software projects whose costs
you need to estimate?

4. Ease of use: - Are the model inputs and options easy to understand and
specify ?

5. Prospectiveness: - Does the model a void the use of information that will
not be well known until the project is complete ?

6. Parsimony: - Does the model a void the use of highly redundant factors ,
or factors which make no appreciable contribution to the results?

7. Proposal using SLIM Model

This paper suggests to use SLIM model to estimate the cost of software
by apply the Putnam's SLIM, and it is based on the Norden / Rayleigh
function. This is explaining in the general algorithm:-

Step 1 : Input Software development .

Step 2 : Compute the value of **C** , **E, t** and **D** .

Step 3 : Compute the **effort** by apply the **Software equation** is :-

Size= CE^1/3(t^4/3)

Step 4 : Compute the **Manpower – Buildup Equation** to find the **total of**

time Is needed to development software By:-

D=E/t^ 3

Step 5 : Analysis the result of step **4** and step**5** And show the relation

between the development effort and the schedule of time .

Step 6 : Display the result of software cost estimate by draw the

Rayleigh Curve .

Step 7 : End .

An important advantage of applying the SLIM model to estimate the constraint model of project then the result is a good guide to show the relation between the effort and schedule time.

8. Experimental Result and Discussion

Results of applying SLIM model equation over two development software , the first project with all normal cost drivers and consist of 70,000 lines of code, after complete the operation of software equation and Manpower –Building equation estimate, the results are represented in **figure (2).**

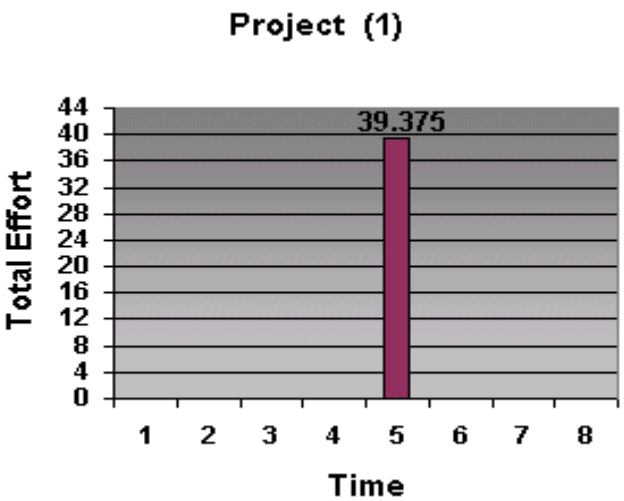

Figure (2) Represented the relation between Time and Effort

But the second development project is low for language and tools expression, very complexity and consists of 100,000 lines of code, after complete the operation of software
equation and Manpower –Building equation estimate, the results are represented in **figure (3)**.

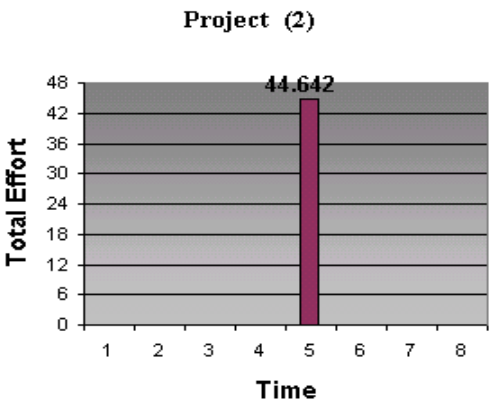

Fig.(3) relation between Time and Effort

From the figure (**2**) and figure (**3**) can be analysis the result of estimation by show the degree of effort and schedule time are increase with increase the size of project , cost driver and scale , this relation is represented in figure (**4**) .

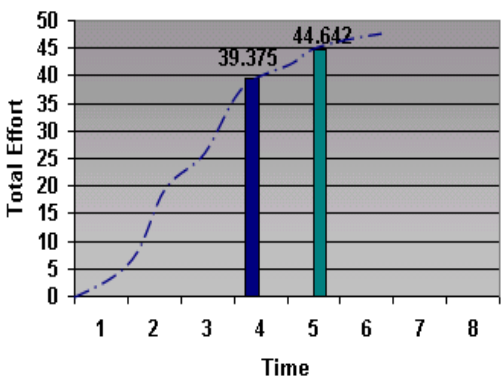

Fig. (4) Rayleigh Curve - relation between Time and Effort

9. Conclusions

In this work, we used SLIM model to estimate the cost of software to found the volume of the efforts and time of new software design, when we make an application of SLIM model on two software , we found these result :-

1. One of the main benefits of applying SLIM model is to estimate the constraint model .

2. The SLIM model had needed a fewer parameters to generate an estimation the cost of software.

3. The degree of effort and duration are increase with increase the size of project, cost drivers and scale drivers, then the average staffing is increase linearly.

4. The SLIM model is a good guide to estimate the requirements and maintenance of software.

10. Reference

[1] Pressman, Rogers, "Software Engineering ", McGraw-Hill, Singapore, 1997.

[2] Sunsmore, H.E., Conte, S.D, "Software Engineering Metrics and Models ", the Benjamin/Cummings,1986.

[3] Shooman , Martin L. ," Software Engineering Design,Reliability, and Management", McGraw-Hill, Singapore, 1988.

[4] Perrly , William e. , "Effective Methods For Software Testing ", john Wiley and Sons, Canada, 2000.

[5] Sommerville , Lan , " Software Engineering " , Addison - Wesley, Harlow, 2001.

[6] Pfleeger , Shari lawernce , " Software Engineering theory and Practice ", Second Edition , Preice- Halll , Inc, New Jersey,2001.

[7] Pressman, Rogers, "Software Engineering a Practitioner's Approach ",5th Edition , McGraw-Hill, Singapore, 2000.

[8] Robert Pressman, "Software Engineering Testing ",2000.
Available at :
www.mhhe.com/engcs/compsci/pressman

[9] Jonas Verlodt , "Software Cost Estimation Using model ",
2004.
Available at :
http://www.jsc.nasa.gov/bu2

[10] Stephen R. Schach, "Software Engineering Estimation ",2000.
Available at :

www.mhhe.com/engcs/compsci/schach

[11] T. Koomen, M. Pol," Test Process Improvement :A Practical
Step by Step Guide to Structured Testing ", Addison-Wesley,
Harlow ,2000.

[12] Perrly , William e. , "Effective Methods For Software Testing ",
john Wiley and Sons, Canada,2000.

Used SLIM Model to Estimate Software Cost

عبير سالم جميل
قسم علم الحاسبات ونظم المعلومات
كلية المنصور الجامعة

المستخلص

ان الهدف من تخمين كلفة البرمجيات هو التنبؤ في حساب كمية الجهد المطلوب لبناء برامج جديدة , حيث ان النماذج (Models) يستخدم اكثر من معادلة رياضية لحساب كلفة البرمجيات في مرحلة التصميم وذلك من خلال حساب عدد المتغيرات وحجم البرنامج (Software) والذي يعتبر عامل اساسي في حساب كلفة البرنامج والذي يحسب من خلال حساب عدد الخطوات التي يتكون منها البرنامج ومن هذه النماذج هونموذج الـــ SLIM الذي يستخدم لتخمين كلفة البرمجيات في مرحلة التصميم والذي يلائم البرامج ذات الحجم الكبير وايضا يحتاج الى معادلات رياضية وقت للحساب اقل من بقية النماذج الاخرى التي تستخدم لحساب كلفة البرمجيات .

في هذا البحث تم تطبيق معادلات الـــ SLIM Model على نوعين مختلفين من البرامج ومن ثم تم القيام بعملية التحليل والمقارنة ما بين هذه النتائج وبالتالي تم الوصول الى العلاقة ما بين حجم الجهد والوقت المطلوب لبناء البرامج حيث ان هذه العلاقة تزداد بزيادة الحجم البرنامج .