Time Estimation for Project Management Life Cycle: A Simulation Approach

P. K. Suri[1], Bharat Bhushan[2], Ashish Jolly[3]

[1]Department of Computer Science & Applications, Kurukshetra University, Kurukshetra (Haryana), India.
[2]Department of Computer Science & Applications, Guru Nanak Khalsa College, Yamuna Nagar (Haryana), India.
[3]Department of Computer Science & Applications, Shri Atmanand Jain Institute of Management & Technology, Ambala City (Haryana), India.

Summary
The Project Life Cycle refers to a logical sequence of activities to accomplish the project's goals or objectives. Regardless of scope or complexity, any project goes through a series of stages during its life. Project development has increased importance in business due to stiff competition and a fast changing business environment. Its cycle time is critical as it decides the success of a business. If the project is late to market by four months in a life cycle of five years, it loses one third of its profit. Keeping in view an attempt has been made to design a simulator for time estimation of project management process using Erlang-4 distribution. The input for the simulator has been derived by using an algorithm for generating pseudo random numbers which follows Erlang-4 distribution. This simulator will be an asset to affordably keep track of the time of phases during the process of project management.

Keywords:
Simulation, Phases, Time estimation, Project Management Process, Effort Estimation, Erlang Distribution

1. Introduction

An important aspect of Project Management is scheduling time accurately. This is a critical component of Project planning as this will decide the deadline for the completion of a project - whether small, medium or mega. Closely linked to this is our credibility since we have to meet the deadlines we have committed to. Another crucial impact of Project time scheduling is that when deadlines are not met and the timeline gets extended, project costs escalate accordingly. This will impact on the profitability of our project work. Additionally, it may lead to unnecessary stress and work load in the execution of the projects. More importantly, our credibility takes a beating and can trigger a whole range of other avoidable problems.

Therefore assessing the time period aptly for varied tasks assumes great importance in Project Planning. The usual problem is that most people underestimate the time needed to execute different types of jobs. This is especially so when the person is not very familiar with certain type of tasks. When we are ignorant about certain areas of work involved, it leads to improper judgment of time schedules [1].

Regardless of scope or complexity, any project goes through a series of stages during its life. There is first an Initiation or Birth phase, in which the outputs and critical success factors are defined, followed by a Planning phase, characterized by breaking down the project into smaller parts/tasks, an Execution phase, in which the project plan is executed, and lastly a Closure Project activities must be grouped into phases or Exit phase, that marks the completion of the project. It is of great importance to organize project phases into industry-specific project cycles because each industry sector involves specific requirements, tasks, and procedures when it comes to projects [2].

The Project Management Life Cycle is shown in the diagram below.

Fig. The Project Management Life Cycle

Oddur Benediktsson et al. [3] motivated for this work is derived from the current interest in speeding up development schedules. A key implication of the shift to more rapid development methods is the growing emphasis on fixed time and fixed effort delivered during such projects. However there appears to be little work that addresses the impacts of dealing with bound effort levels. The result of binding time and effort is to deprive project managers of the normal parameters that are used in tradeoffs. The paper attempts to introduce a quantitative analytical framework for modeling effort-boxed development in order to uncover the effects on the overall development effort and the potential leverage that can be derived from incremental delivery in such projects. Models that predict product size as an exponential function of the development effort are used in the paper to explore the relationships between effort and the number of increments, thereby providing new insights into the economic impact of incremental approaches to effort-boxed software projects.

Manuscript received May 5, 2009
Manuscript revised May 20, 2009

Jenkins et al. [4] conducted a large empirical investigation in the beginning of the 1980s. The study focused on the early stages of system development. It included development aspects, such as user satisfaction, development time, and cost overruns. The study included projects that were considered small, medium and large relative to the organizations standards. The survey measured three success factors; user satisfaction, being "on-time" and being "on-budget".

S. Kumanan O.V. Krishnaiah Chetty [5] described and exemplified by a case study the application of Petri nets to determine product development time. Researchers recommend the project management approach but the existing planning tools are limited in application necessitating improved tools.

Naveen Aggarwal et al. [6] in his paper propose a Content Management System Effort Estimation model (CMSEEM) for the current technologies using which a piece of work can be estimated more accurately. Data available were collected from twelve completed projects taken from industry and seventy different projects completed by the students. These projects are categorized based on their size and total/build effort ratio. The size of the project is estimated by using the modified object point analysis approach. The complexity of the project is determined by using a set of questionnaire which has to be filled by the project managers after completing the initial requirement analysis. The nominal size estimated after object point analysis is finalized using the adjustment factors which are calculated by considering the different characteristics of the system such as production and general system characteristics. The estimated effort is further phase wise distributed for better scheduling of the project. The proposed model is refined using the linear regression approach. Finally the model is validated using another questionnaire which has to be filled after completing the project. The proposed model shows a great improvement as compared to the earlier models used in effort estimation of Web CMS projects.

Phan et al. [7, 8] tried to assess to what extent, and for what reasons, software development projects encountered cost and schedule overruns.

Heemstra and Kusters [9, 10] conducted a survey of cost estimation in Dutch organizations. The goal was to provide an overview of the state of the art of estimation and controlling software development costs. Estimation methods, original project estimates and actual effort were analyzed.

Bergeron and St-Arnaud [11] performed a study to identify estimation methods, and to what extent they were used. They also investigated how choice of method, and underlying factors and variables, influenced estimation accuracy.

Moores and Edwards [12] sought to investigate why there was an apparent lack of use of software cost estimating tools.

Wydenbach and Paynter [13] investigated the estimation practices in New Zealand on basis of a previous survey [9].

Addison and Vallabh [14] investigated the perceptions of project managers on software project risks and controls. A "snowball sample" was used to identify 70 managers, of whom 36 returned the questionnaire. Although not a study with a focus on estimation, it reports on aspects related to budgets and plans.

2. Proposed Model

Since time estimation is stochastic in nature so here simulator [15] has been designed to evaluate time estimation for various stages of project management process. There are four major activities for project management process, requires specific time period for their completion. Each process is assumed to be mutually independent and each of these times may reasonably be assumed to follow a negative exponential distribution. Further if the average of each of these four times is the same, the sum is said to be Erlanged-4 distributed. The erlang distribution is a continuous distribution, which has a positive value for all real numbers greater than zero, and is given by two parameters: the shape *k*, which is a non-negative integer, and the rate λ, which is a non-negative real number. The distribution is sometimes defined using the inverse of the rate parameter, the scale θ. When the shape parameter *k* equals 1, the distribution simplifies to the exponential distribution.

An Erlang-m distributed random variable is the sum of m-independent and identically distributed exponential distribution and its probability density function is

$$f(x;k,\lambda) = \frac{\lambda^k x^{k-1} e^{-\lambda x}}{(k-1)!} \quad \text{for } x, \lambda \geq 0.$$

where *e* is the base of the natural logarithm and ! is the factorial function. The parameter *k* is called the shape parameter and the parameter λ is called the rate parameter. Because of the factorial function in the denominator, the Erlang distribution is only defined when the parameter *k* is a positive integer. In fact, this distribution is sometimes called the Erlang-k distribution (e.g., an Erlang-4 distribution is an Erlang distribution with *k=4*).

To generate Erlang variants with a mean time of mβ units, we have to generate m-random observations from an exponential distribution, with a mean time of β, and add them [16].

Simulation of Time Estimation for Project Management Process

M: no. of constituent phases of project management process
BETA: average value of completion time of each of the M constituent phases
RUNS: no. of times the simulation process is repeated
RANDOM: random number
RNDY_PROC: a procedure to generate random number
PROD: product of random numbers
ERLANG TIME: composite time for project completion

Algorithm to compute composite time estimation for project completion

1) Read BETA, M
2) Read RUNS
3) Compute RUNS erlang and their products by repeating steps a and b
   a) [Compute the product of m pseudo random numbers generated thru random number generation method)]
   b) [generate erlang variate for jth interval]
4) Compute composite time estimation for project management process
5) Stop

3. Results

Case 1: It is shown in the form of table regarding composite time for different values of time units (β) and four different phases of project management process on various simulation runs.

Simulation Runs	Composite Time for m=4, β=10	Composite Time for m=4, β=20	Composite Time for m=4, β=30	Composite Time for m=4, β=40
1000	10.61666	21.23333	31.84999	42.46666
2000	35.69118	71.38237	107.0736	142.7647
3000	32.36687	64.73374	97.10062	129.4675
4000	28.98682	57.97364	86.96046	115.9473
5000	29.31788	58.63575	87.95363	117.2715
6000	32.91326	65.82652	98.73978	131.653
7000	29.47346	58.94693	88.42039	117.8939
8000	44.76616	89.53231	134.2985	179.0646
9000	29.39613	58.79226	88.18839	117.5845
10000	72.28013	144.5603	216.8404	289.1205
11000	61.05927	122.1185	183.1778	244.2371
12000	33.27243	66.54487	99.8173	133.0897
13000	27.4834	54.96679	82.45019	109.9336
14000	59.59254	119.1851	178.7776	238.3702
15000	67.77319	135.5464	203.3196	271.0928
16000	64.9194	129.8388	194.7582	259.6776
17000	39.53636	79.07272	118.6091	158.1454
18000	90.56575	181.1315	271.6972	362.263
19000	28.04161	56.08322	84.12483	112.1664
20000	21.41306	42.82612	64.23918	85.65224

Table 1: Simulation Runs and Composite Time for Different β (m = 4)

Case 2: Here in Table 2 it is assume that each phase has two levels of execution and each of the levels have equal average completion time, thus no. of constituent phases becomes eight.

Simulation Runs	Composite Time for m=8, β=10	Composite Time for m=8, β=20	Composite Time for m=8, β=30	Composite Time for m=8, β=40
1000	81.48563	162.9713	244.4569	325.9425
2000	38.44621	76.89243	115.3386	153.7849
3000	44.59477	89.18954	133.7843	178.3791
4000	80.49877	160.9975	241.4963	321.9951
5000	92.27612	184.5522	276.8284	369.1045
6000	49.68892	99.37784	149.0668	198.7557
7000	77.34292	154.6858	232.0287	309.3717
8000	98.25718	196.5144	294.7715	393.0287
9000	102.9134	205.8268	308.7402	411.6536
10000	56.11856	112.2371	168.3557	224.4743
11000	105.987	211.974	317.961	423.9479
12000	49.38674	98.77348	148.1602	197.547
13000	76.51794	153.0359	229.5538	306.0717
14000	95.26999	190.54	285.81	381.08
15000	74.79996	149.5999	224.3999	299.1998
16000	54.22913	108.4583	162.6874	216.9165
17000	81.11955	162.2391	243.3587	324.4782
18000	53.21814	106.4363	159.6544	212.8725
19000	112.8652	225.7305	338.5957	451.4609
20000	91.75166	183.5033	275.255	367.0067

Table 2: Simulation Runs and Composite Time for Different β (m = 8)

Case 3: Here in Table 3 it is assume that each phase has three levels of execution and each of the levels have equal average completion time, thus no. of constituent phases becomes twelve.

Simulation Runs	Composite Time for m=12, β=10	Composite Time for m=12, β=20	Composite Time for m=12, β=30	Composite Time for m=12, β=40
1000	96.05199	192.104	288.156	384.2079
2000	136.574	273.1481	409.7221	546.2961
3000	106.8605	213.7209	320.5814	427.4419
4000	103.4778	206.9557	310.4335	413.9113
5000	129.8786	259.7572	389.6358	519.5144
6000	155.4849	310.9698	466.4547	621.9396
7000	93.65559	187.3112	280.9668	374.6224
8000	71.85863	143.7173	215.5759	287.4345
9000	86.48109	172.9622	259.4433	345.9244
10000	105.9995	211.9989	317.9984	423.9978
11000	110.359	220.7179	331.0769	441.4358
12000	90.07283	180.1457	270.2185	360.2913
13000	152.8464	305.6928	458.5392	611.3856
14000	108.7768	217.5536	326.3305	435.1073
15000	140.8167	281.6334	422.4501	563.2668
16000	119.4943	238.9886	358.4828	477.9771
17000	130.9	261.8	392.7	523.5999
18000	147.7323	295.4647	443.197	590.9293
19000	151.839	303.678	455.517	607.356
20000	139.0828	278.1656	417.2484	556.3311

Table 3: Simulation Runs and Composite Time for Different β (m = 12)

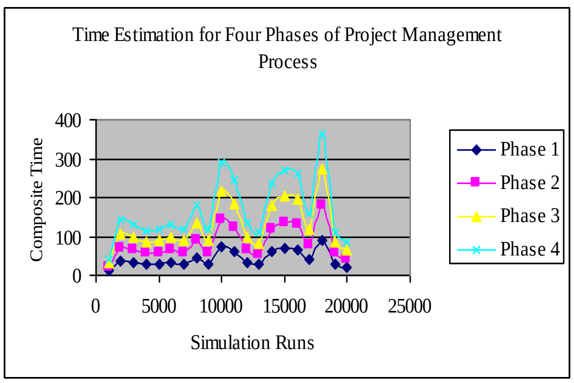
4. Discussion and Conclusion

Currently we know about many ways to estimate time for project management process and still the estimation in general is considered to be "not that scientific" because estimation is in fact performed using regression models which only approximate the influence of various parameters on time. Obviously, those approximations do not take the nature of development into consideration.

The motivation for incorporating simulation into project management process for time estimation with simulated data lead to useful information for the completion of the project well in time.

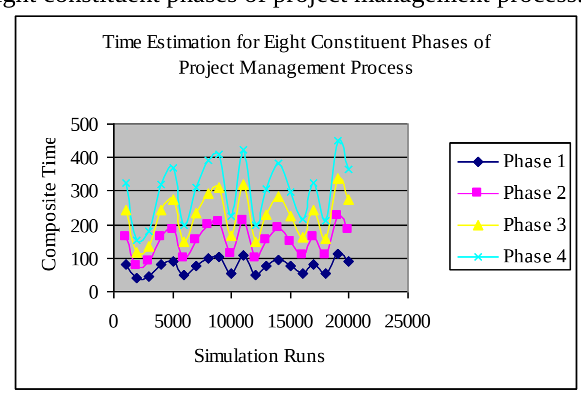
The above simulator was validated with the help of pseudo random numbers for the time to be taken to complete the four activities of project management process. The result shows that the composite time for the completion of project is very-2 close to the estimated values.

The graph 1 depicts the relationship between no. of simulation runs and composite time estimation for the four phases of project management process. It is found that the total composite time is found to be the sum of the time of each phase.



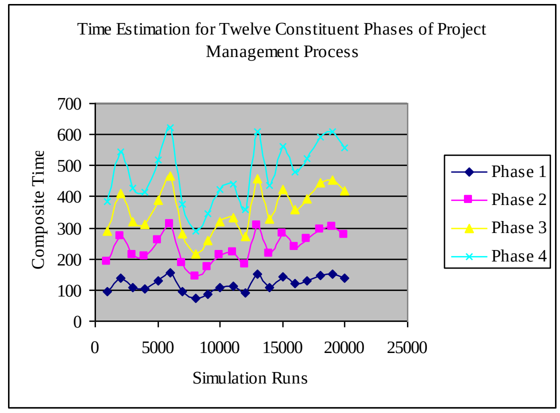
Graph No. 1

The graph 2 depicts the relationship between no. of simulation runs and composite time estimation for the eight constituent phases of project management process.



Graph No. 2

The graph 3 depicts the relationship between no. of simulation runs and composite time estimation for the twelve constituent phases of project management process.



Graph No. 3

References

[1] Walker Royce, "*Software Project Management – A Unified Framework"*, Addison Wesley, 1998.

[2] T.A. Flynn, P.E, "*Integration of the Project Management Life Cycle (PMLC) and the Systems Development Life Cycle (SDLC) in Accelerated Project Efforts, Adapting Project Management Best Practices to Unreasonable Requests*", Advanced Management Services, Inc. (AMS) © 1994 – 2007.

[3] Oddur Benediktsson and Darren Dalcher, "*Project Effort Estimation: Or, When Size Makes a Difference*", Springer Berlin / Heidelberg 2004.

[4] Jenkins, A.M., J.D. Naumann, and J.C. Wetherbe, "*Empirical Investigation of Systems Development Practices and Results*", Information & Management 7 (1984), pp. 73 - 82.

[5] S. Kumanan and O.V. Krishnaiah Chetty, "*Estimating product development cycle time using Petri nets*", Springer-Verlag London Limited, 2005.

[6] Naveen Aggarwal, Nupur Prakash, and Sanjeev Sofat, "*Content Management System Effort Estimation, Model based on Object Point Analysis*", International Journal of Computer Science and Engineering 2:4 (2008).

[7] Phan, D., D. Vogel, and Nunamaker, "*The Search for Perfect Project Management*, in *Computerworld*" (1988), pp. 95 - 100.

[8] Phan, D., "*Information Systems Project Management: An Integrated Resource Planning Perspective Model*, in *Department of Management and Information Systems*" (1990), Arizona: Tucson.

[9] Heemstra, F.J., Kusters, R.J., "*Controlling software development costs: a field study. In: International Conference on Organisation and Information Systems*" (1989) Bled, Yugoslavia.

[10] Heemstra, F. J., "*Software cost estimation*", Information and Software Technology, 1992. 34(10) pp. 627 - 639.

[11] Bergeron, F. and J.-Y. St-Arnaud, "*Estimation of Information Systems Development Efforts: A Pilot Study",* Information & Management, 1992. 22: pp. 239 - 254.
[12] Moores, T.T. and J.S. Edwards, "*Could Large UK Corporations and Computing Companies Use Software Cost Estimating Tools? - A Survey*", European Journal of Information Systems, 1992. 1(5): pp. 311 - 319.
[13] Wydenbach, G. and J. Paynter, "*Software Project Estimation: A Survey of Practices in New Zealand*", New Zealand Journal of Computing, 1995. 6(1B): pp. 317 - 327.
[14] Addison, T. and S. Vallabh, "*Controlling Software Project Risks - An Empirical Study of Methods used by Experienced Project Manager*", SAICSIT 2002, Port Elizabeth, South Africa
[15] A.M. Law, W.D. Kelton, "*Simulation Modeling and Analysis*", McGraw Hill Int. Series, New York, 2000.
[16] P. K. Suri and Bharat Bhushan, "*Simulator for Time Estimation of Software Development Process*", IJCSNS, Vol 7 No. 7, July 2007.

Dr. P.K. Suri received his Ph.D degree from Faculty of Engineering, Kurukshetra University, Kurukshetra, India and master's degree from Indian Institute of Technology, Roorkee (formerly known as Roorkee University), India. He is working as Professor in the Department of Computer Science & Applications, Kurukshetra University, Kurukshetra - 136119 (Haryana), India since Oct. 1993. He has earlier worked as Reader, Computer Sc. & Applications, at Bhopal University, Bhopal from 1985-90. He has supervised six Ph.D.'s in Computer Science and thirteen students are working under his supervision. He has more than 100 publications in International / National Journals and Conferences. He is recipient of 'THE GEORGE OOMAN MEMORIAL PRIZE' for the year 1991-92 and a RESEARCH AWARD –"The Certificate of Merit – 2000" for the paper entitled ESMD – An Expert System for Medical Diagnosis from INSTITUTION OF ENGINEERS, INDIA. His teaching and research activities include Simulation and Modeling, SQA, Software Reliability, Software testing & Software Engineering processes , Temporal Databases, Ad hoc Networks, Grid Computing , and Biomechanics.

Dr. Bharat Bhushan received his Ph.D degree from Department of Computer Science & Applications, Kurukshetra University, Kurukshetra, M.Sc (Physics) from Punjab Univ. Chandigarh and M.Sc (Comp. Sc.), MCA degrees from Guru Jambeshwar University in 2001 respectively. Presently working as Head, Department of Computer Science and Applications, Guru Nanak Khalsa College, Yamuna Nagar (affiliated to Kurukshetra University, Kurukshetra- Haryana, India) and senior most teacher of computer science in Haryana since 1984. He is a member of Board of Studies of Computer Science, Kurukshetra University. His research interest includes Software engineering, Digital electronics and Simulation Experiments.

Jolly received his MCA degree from University of Madras, Chennai in the year 1999. He is working as a Asstt Professor and Head in the Department of Computer Science & Applications, Shri Atmanand Jain Institute of Management & Technology (affiliated to Kurukshetra University, Kurukshetra), Ambala City, Haryana, India. His research area includes Simulation, Software Engineering and Software Project Management.