Volume 5, Issue 5, MAY 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Design and Development of SCRAS (Schedule/Cost Risk Analysis Simulator)

[1]Harpreet Kaur*, [2]Arpna Dhingra, [3]Er.Isha Sharma, [4]Arjinder Singh
[1, 2, 4] PG Student, [3]Assistant Professor
[1, 2, 3, 4] Department of Computer Science & Engineering,. Chandigarh University,
Gharuan, Mohali, Chandigarh, India

Abstract- In software designing and development there are a lot of resources required. These resources consume some cost. The cost or budget of any software is directly depends on the schedule. If the schedule slip then it directly effect on the budget of the software. This is the first duty of every project manager to maintain the right balance between the budget and the schedule of the software. To prevent the budget and schedule risk in software development Monte Carlo Simulation is used with the concept of Program Evaluation and Review Technique (PERT). This method is named as SCRAS (Schedule/Cost Risk Analysis Simulator). With the help of this method the budget and schedule risk will predetermined so the project manager can execute some mitigation technique and the risk will be remove or prevented.

Keywords- Scheduling; Risk analysis; Beta Distribution; Monte Carlo Simulation; SCRAS; PERT; .

I. INTRODUCTION

Management is must in every type of project whether it's Software project or Hardware project. The whole work of management is done by the project manager. When we talk about the software project there are two most important factor i.e budget and schedule. To achieve the success in software projects it's necessary the project should be completed before the deadline and within budget. Most of times if the schedule slip then budget of the project also increased. The achievement depends upon the nature of planning, scheduling and controlling the different activities of software project.. Program Evaluation and Review Technique (PERT) is a guide to assist and control the assets to meet the calendar date of the activities which include the high level of instability. it is acknowledged that venture calendar assumes a key part in task administration. To control the instability in planning procedure we apply the probabilistic Monty Carlo Simulation. Monty Carlo Simulation is concept that gives a numerical estimation of the stochastic highlight of the framework reaction. A simulation based software project risk management tool helps manager to identify high risk areas of software process and software product, and plan, organize and manage the software development project in a cautious way.

Traditional Probability Analysis Method

Duration Risk means the likelihood and loss of in culmination in the aggregate specified duration utmost. As per the meaning of span hazard, the numerical articulation of duration risk can be characterized as takes after:

$$SR = (P_f, U)$$

Where SR is schedule risk, p_f is probability of incompletion in the overall specified time limit, and U is loss for in completion,

When the overall specified time limit is H_s, it's probability of completion can be given by formula.

$$P(H \leq H_s) = \int_{-\infty}^{H_s} f(t)dt$$

Where f(t) give the probability density function of project.

The probability density function will be as follow for Beta Distribution:

$$f(t) = \frac{\tau(K_1+K_2)(t-a)^{k_1-1}(b-t)^{k_2-1}}{\tau(k_1)\tau(k_2)(b-a)^{k_1+k_2-1}}$$

$$a \leq 1 \leq b; a, b > 0$$

$$\text{In which } \tau(k) = \int_0^{\infty} z^{k-1}e^{-Z}dz$$

For each activity *J* , we have to estimate optimistic duration/cost ad_j , pessimistic duration/cost bdj_j , and most likely duration/cost md_j . Then we can find the mean μ_i the variance $\sigma_i^2$of duration/cost by the following formula:

$$\mu_j = \frac{ad_j+4md_j+b_j}{6}, \qquad \sigma_j^{\,2} = \frac{(bd_j-ad_j)^2}{36}$$

Then we can determine p (completion probability) for the total stipulated duration limit *Hs* and consulting standard normal distribution.

SCRAS (Schedule/Cost Risk Analysis Simulator)

SCRAS **is** based on one of the two popular concepts i.e. CPM and PERT . A simulation based software project scheduling tool where duration of each activity in software project is created stochastically assuming it to be belonging to beta distribution. Simulations are mainly done by technique, known as Monte Carlo . In a simulation, the model is computed many iteration, with the input values randomized from a probability distribution function chosen for each iteration from the probability distributions of each variable.

The method, used is called SCRAS, has several steps:

I. Each iteration begins by selecting duration for each risky activity at random from its range and distribution

II. The total key milestone completion dates and budget for that iteration are calculated using CPM for that particular configuration of durations and cost. Those are only possible dates for completion of the project and its milestones.

III. To determine the entire pattern of possible completion dates for the project and it is important the milestones, the risk analyst iterates the work many times. At the end of each and every iteration, the completion dates for the total project and for any important milestone are collected . The program also records which activities were on the critical path for that iteration.

II. NETWORK REPRESENTATION

With respect to each of the software project activity three parameters namely optimistic, pessimistic and most likely is established. It can be cost or duration for the activity.

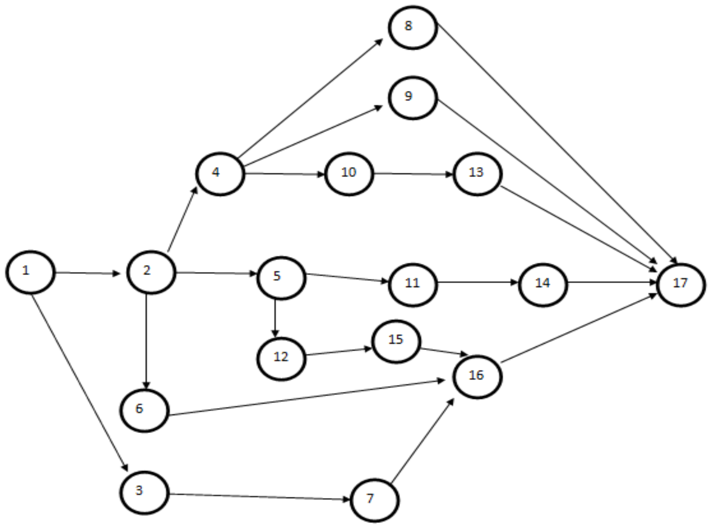

Figure 1.1

These three time values provide a measure of uncertainty associated with each activity.

Optimistic time/cost(A_k) is the lowest possible time/cost in which the activity can be finished.

Most Likely time/cost(B_k) is the estimate of the normal time/cost the activity would take.

Pessimistic time/cost(M_k) represents the longest time and highest cost the activity could take if everything goes wrong.

Table: 1.1(Three time estimation for activities)

Activity	Start Node	Finish Node	Optimistic time/cost(A_k)	Most Likely time/cost(B_k)	Pessimistic time/cost(M_k)
1	1	2	7	15	35
2	1	3	2	5	8
3	2	4	4	7	15
4	2	5	3	7	15
5	2	6	7	15	30
6	3	7	5	10	25
7	4	8	1	3	10
8	4	9	1	2	3

9	4	10	2	3	3
10	5	11	1	2	5
11	5	12	10	14	40
12	6	16	0	0	0
13	7	16	0	0	0
14	10	13	1	3	10
15	11	14	10	15	30
16	12	15	1	4	10
17	14	17	1	2	5
18	15	16	7	20	40
19	16	17	1	3	5
20	8	17	15	30	100
21	9	17	10	20	50
22	13	17	7	10	15

Notations and Definitions: The following notations and terms are used:
starting node START[X],FINISH[X],ES[Y] earliest starting ,EF[Y] earliest finishing, LS[Y] latest starting, LF[Y] latest finishing, TIME[Y] expected, A[Y] optimistic estimate, M[Y] most likely, B[Y] most pessimistic estimate.

Algorithm: SCRAS_INFORMAL

Step-1: Allocates random time samples for all the activities. Store them in array TIME.
Step-2: Traverse the network in forward direction to find the length of the critical path.
Step-3: Trace the network in backward direction to identify the activities lying on critical path.
Step-4: Repeat steps 2 and 3 for a number of times to calculate the criticality index for each activity.

1. Time samples of activities: The activity durations are decided by generating the time samples for activities using beta distribution as follows:
Generate pseudorandom numbers r1 and r2 and calculate g (A_X+ (B_X-A_X)*r1):
g (A_X+(B_X-A_X)*r1)=(r1) $^{k1-1}$*(1-r1) $^{k2-1}$/ (B_X-A_X) where k1=4 and k2=4.
Calculate g (M_X) = (M_X-A_X) $^{k1-1}$*(B_X-M_X) $^{k2-1}$/ (B_X-A_X) $^{(k1+k2-1)}$
Compare g (A_X+ (B_X-A_X)*r1)/ g (M_X) and r2. If g (A_X+ (B_X-A_X)*r1)/ g (M_X) >=r2 then TIME [X] = (A_X+ (B_X-A_X)*r1) else repeat the procedure for X=1, 2...N

2. Forward Pass: Following equations are used during forward pass:
   a) EF [X] = ES [X] + TIME [X] where X=1, 2...N (1)
   b) EN [Z] = MAX {EF [all activities terminating in Z]} where Z=1, 2...M (2)
   c) ES [Z] = EN [START_NODE [Z]] where Z = 1, 2...N (3)

The forward pass starts with EN [X] = 0. Using equation (3) we get EN[X] = ES[X] = 0. Using equation (1) we get EF[X] = TIME[X]. Proceeding this way, by using equations (2), (3) and (1) respectively, we reach the last node of the network and calculate the length of the critical path i.e. T_{min} (total completion time of the project).

3. Backward Pass: Following equations are used during backward pass:
   a) LS [X] = LF [X] - TIME [X] where X = 1, 2...N (4)
   b) LN [Z] = MIN {Ls [all activities originating in Z]} where Z = 1, 2...M (5)
   c) LF [every activity terminating in node Z] =LN [Z] where Z = 1, 2...M (6)

The backward pass starts with LN [X] = T_{min}. Using equation (6), we get LN[X] = LF[X] = Tmin. Using equation (4), we get LS [X] = T_{min} – TIME [X]. Proceeding this way, by using equations (5), (6) and (4) respectively, we reach the first node of the network and identify critical path for which LS[X] - ES[X] <= ERROR holds, where ERROR = 0.001 (say).

4. Criticality Index: Following condition is satisfied for an activity to be a critical activity
   LS [X] - ES [X] = LF [X] - EF [X] (latitude) is ZERO

III. RESULT

Table 1.2 Results of each activity

Activity(X)	Start node	Final node	A_k	B_k	M_k	μ_i	σ_i^2	σ	Risk Index
1	1	2	7	15	35	21	21.7	4.66	1
2	1	3	2	5	8	5	1.00	1.00	0
3	2	4	4	7	15	9.5	3.36	1.83	0.148
4	2	5	3	7	15	9	4.00	2.00	0.876
5	2	6	7	15	30	18.5	14.69	3.83	0
6	3	7	5	10	25	15	11.11	3.33	0
7	4	8	1	3	10	5.5	2.25	1.50	0.133
8	4	9	1	2	3	2	0.11	0.33	0.016
9	4	10	2	3	3	2.5	0.02	0.16	0.003
10	5	11	1	2	5	3	0.44	0.66	0.019
11	5	12	10	14	40	25	25	5	0.859
12	6	16	0	0	0	0	0	0	0
13	7	16	0	0	0	0	0	0	0
14	10	13	1	3	10	5.5	2.25	1.50	0.003
15	11	14	10	15	30	20	11.11	3.33	0.019
16	12	15	1	4	10	5.5	2.25	1.50	0.859
17	14	17	1	2	5	3	0.44	0.66	0.019
18	15	16	7	20	40	25.5	30.25	5.50	0.859
19	16	17	1	3	5	3	0.44	0.66	0.859
20	8	17	15	30	100	57.5	200.69	14.16	0.133
21	9	17	10	20	50	30	44.44	6.66	0.016
22	13	17	7	10	15	11	1.77	1.33	0.003

- Activity 1 became risky during all the simulation runs.
- Activity 4 became risky during86.6 percent of simulation runs.
- Activity 11, 16, 18 and 19 became risky during 85.9 percent of simulation runs

Figure 1.2 presents the result of Table 4.3 in graphical form. The criticality index measures how often one specific task was on the critical path during the simulation

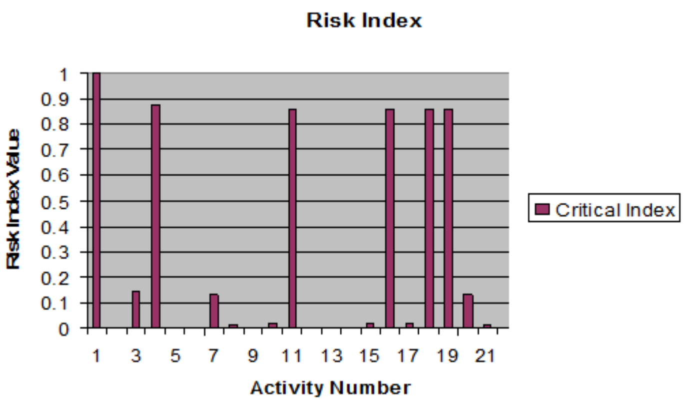

Figure:1.2

Figure 1.3 shows the simulated frequency distribution chart for the software project being simulated.

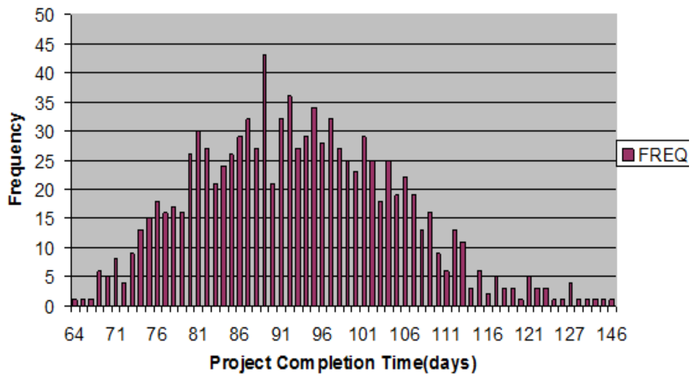

Figure 1.3

IV. CONCLUSION

In this worldwide competition, project management is paid higher attention. In software managing risks has been recognized as a very important process in order to achieve project objectives in terms of time and cost. On the basis of calculating the accumulative probability distributed data of duration and cost for each activity in project, Monte Carlo method is applied to simulate the duration and cost for each activity and the overall project to accurately determine the completion probability of the project under considering of the changeability and randomness of duration for each activity. This method can give scientific quantitative basis and information for project management. To some extent, it can also helps to project manager in decision makers easily make control with duration risk and cost risk of project and make fair decision.. The encouragement for wanting to incorporate simulation into Schedule and cost risk Analysis is clear as finding critical activities with simulated data yields fruitful information such as estimation of software project/activity completion time and cost. Simulation offers the likelihood for speaking to the unpredictability that is vital for sensible thinking around a product undertaking, including the intrinsic instability.

REFERENCES

[1] HUANG Jian-wen and WANG Xing-xia, 2009."*Risk Analysis of Construction Schedule Based on PERT and MC Simulation*" International Conference on Information Management, Innovation Management and Industrial Engineering(IEEE)

[2] Amnon Gonen, 2008. "*Simulation Results on Project Completion Time*" , Cape Town, South Africa (c) 2008 PICMET

[3] Kelly D. Moses, Stephan F. Hooker.2005. "*Development of Quantitative Cost and Schedule Risk Analysis Process*" Proceedings of the 18th International Conference on Systems Engineering (ISCEng'05)

[5] Dezhi Jin, Zhuofu Wang, Jianming Yang, Meigui Han,2010." *Construction Project Schedule Risk Analysis and Assessment Using Monte Carlo Simulation Method*" IEEE

[6] Amnon, Gonen, "*Estimating Project Completion Times – Simulation and Analytic Approach"*, Proceedings of the 2007 IEEE IEEM, 2007

[7] Boehm, B. W.,"*Software Risk Management: Principles and Practice"s*, IEEE Software, Vol. 8, No. 1, 1991

[8] Boehm, B. W.,"*Risk Control",* American Programmer, 1992, vol. 5, no. September

[9] Charette, R.N., "*Why software fails?"* , IEEE Spectrum, 2005

[10] David, T. Hulett*, "Integrated Cost / Schedule Risk Analysis"*, PMI Paper, 2004.

[11] Simmons L.F., "*Project Management-Critical Path Method (CPM) and PERT Simulated with Process Model"*, Proceedings of the 2002 Winter Simulation Conference, 2002

