European Journal of Business and Management
ISSN 2222-1905 (Paper) ISSN 2222-2839 (Online)
Vol.9, No.11, 2017

Simulation of Automated Warehouse System: Using CPM and PERT Method

Dr.K.M.Salah Uddin[1*] Afroza Akter Jinia[2]
1.Associate Professor, Dept of Management Information Systems, University of Dhaka, Dhaka
2.Department of Management Information Systems, University of Dhaka

Abstract
PERT and CPM are two most widely used method for project management. PERT use probabilistic time estimation for project completion, on the other hand CPM employs deterministic time for crashing project. Specific deadline can be achieved by crashing which is basically a time-cost trade-off. The traditional method of crashing only considers average activity times for the calculation of the critical path, ignoring the stochastic nature of activity time. Here a stochastic simulation is done to evaluate the PERT. And frequencies of project completion time in different simulation length are presented. A linear program is developed for crashing the project to minimize the required cost while attaining a specified completion time.
Keywords: PERT, Stochastic simulation, Crashing, Project completion time (PCT), Simulation length (run)

1. Introduction

A project is a combination of interrelated activities that must be performed in a certain order for its completion. Completing a project within time and budget is not an easy task. The project scheduling phase plays a central role in predicting both the time and cost aspects of a project. More precisely it determines a timetable in order to able to predict the expected time and cost of each individual activity. The conventional analysis sometimes leads to erroneous result. The alternate method of handling such situation is simulation technique. Variation in critical activities can cause variation in the project completion time. Variation in non critical activities ordinarily has no effect on the project completion time because of the slack time associated with these activities. However if non critical activities are delayed long enough to expand its slack time or the critical activities could be crashed to the slack of previous non critical activities, it become part of a new critical path and may affect the project completion time. In 1957 the Critical Path Method (CPM) was developed as a network model for project management. CPM is a deterministic method that uses a fixed time estimate for each activity. While CPM is easy to understand and use, it does not consider the time variations that can have a great impact on the completion time of a complex project. The *Program Evaluation and Review Technique* (PERT) is a network model that allows for randomness in activity completion times. PERT was developed in the late 1950's for the U.S. Navy's Polaris project having thousands of contractors. It has the potential to reduce both the time and cost required to complete a project.

A straightforward implementation of the stochastic simulation is described. Here an AOA (activity on arc) network diagram is considered; each activity has an associated probability distribution function. Mean and standard deviation are used to calculate the random variables in normal distribution. Here in this study we use normal or bell shaped or Gaussian distribution to calculate the possible project completion time in simulation.

Crashing refers to a particular variety of project schedule compression which is performed for the purposes of decreasing total period of time (also known as the total project schedule duration). The diminishing of the project duration typically take place after a careful and thorough analysis of all possible project duration minimization alternatives in which any and all methods to attain the maximum schedule duration for the least additional cost The objective of crashing a network is to determine the optimum project schedule. An optimum minimum cost project schedule implies lowest possible cost and the associated time for the software project management.

1.1 Literature Review

Van Slyke (1963) was the first of many researchers to apply Monte Carlo simulations to study PERT. Van Slyke demonstrated several advantages of using simulation including more accurate estimates of the true project length, flexibility in selecting any distribution for activity times, and the ability to calculate "criticality indexes" which are the probability of various activities being on the critical path. Van Slyke demonstrated several advantages of applying simulation techniques to PERT, including more accurate estimates of the true project length, flexibility in selecting any distribution for activity times and the ability to calculate “criticality indexes”, which are the probability of various activities being on the critical path. Steve and Dessouky (1977) described a procedure for solving the project time/cost tradeoff problem of reducing project duration at a minimum cost. The solution to the time & cost problem is achieved by locating a minimal cut in a flow network derived from the original project network. This minimal cut is then utilized to identify the project activities which should experience a duration modification in order to achieve the total project reduction. Moore et al. (1978) and Hannan (1978) reformulate the problem using goal programming. Goal programming is a modification of linear programming which can solve problems with multiple objectives. This allows goals in addition to cost minimization to be added to the problem. Because of conflicting objectives, not all goals are achieved completely. The first approach to address the problem of crashing under stochastic conditions was made by Coskun (1984). Coskun formulated the problem as a chance constrained linear programming (CCLP) problem. CCLP is a method of attempting to convert a probabilistic mathematical programming formulation into an equivalent deterministic formulation. Coskun's formulation ignored the assumed beta distribution of activity times. Instead activity times were assumed to be normally distributed with the mean and standard deviation of each known. The formulation allows a desired probability of completion within a target date to be entered. Coskun concluded that "While the solution of the CLLP formulations of the optimal PERT compression problem provides a wealth of information with significant managerial implications, the computational efforts necessary to solve the CCLP are no greater than those necessary to solve the deterministic compression problem." Ramini (1986) proposed an algorithm for crashing PERT networks with the use of criticality indices. Apparently he did not implement the algorithm, as no results were ever reported. His method does not allow for bottlenecks. Bottlenecks traditionally have multiple feeds into a very narrow path that is critical to the project's completion. Bottlenecks are the favored locations for project managers to build time buffers into their estimates, yet late projects still abound because of deviation from timetables and budgets. Ameen (1987) developed Computer Assisted PERT Simulation (CAPERTSIM), a simulation program developed as a teaching tool to teach project management techniques. Students used the program to evaluate decision making under uncertainty and cost-time relationships and trade-offs. Ameen reported students reacted very favorably to participating in the computer-assisted PERT simulation project. Johnson and Schou (1990) used simulation to compare three rules for crashing stochastic networks: They concluded Rule 3 which is "Select the least cost/day activity first. This rule is a combination of the first two rules. It reflects the idea of selecting the least cost expected value. The procedure follows Rule 2 in calculation a criticality index. The criticality index would then be multiplied by the number of days the expected time of an activity can be reduced. This yields an expected number of days that the critical path can be shortened. This expected value is then divided into the total incremental cost of expediting the activity. Theoretically, the criticality index should be regenerated and the computations repeated at each step in crashing the project" provided the lowest cost of crashing the network. Although the differences in cost were small in the examples, the authors argued that "the greater size of 'real life' problems and the likelihood of multiple critical paths would likely lead to larger differentials in the expected cost of different rules." Badiru (1991) reported development of another simulation program for project management called STARC. STARC allows the user to calculate the probability of completing the project by a specified deadline. It also allows the user to enter "duration risk coverage factor". This is a percentage over which the time ranges of activities are extended. This allows some probability of generating activity times above the pessimistic time and below the optimistic time. Foldes and Sourmis (1993) present a reformulation of crashing networks when the cost-time tradeoff is represented by a non-linear, non-differentiable convex function. Feng *et al*. (2000) presented a hybrid approach that combines simulation techniques with a genetic algorithm to solve the time-cost trade-off problem under uncertainty. Jorgensen (2003) emphasized that the simulation approach can be used for management of any project but he time estimates for project management of information systems are still less accurate than any other estimates in the project management cycle. Haga and Marold (2004), propose a simulation-based method that deals with the time-cost trade-off involved with crashing a project. The method that they proposed is a two steps approach. The first step is to apply the traditional PERT method to crash the project, and the second step consists in testing each activity that had not been crashed to the upper crashing limit to determine if crashing that activity further reduces the average total cost of the project. Haga and Marold (2005) developed a method to monitor and control a project. The output of this method is a list of crashing points at which the project should be reviewed to decide if activities need to be crashed. Crashing points are determined by a backward run through the project network. The crashing points are established at the beginning of the project and they remain fixed during the entire project life. Additional authors which have studied various PERT problems via simulation include Klingel (1966), Gray (1969), Burt (1971), Herbert (1979), Schonberger (1981), and Dodin (1984), and Kidd (1986).

1.2 Objective

The study is conducted with the objective of evaluating the "Automated Warehouse System" of "R.C.Colemarr", drawing a network and finds the critical path and predicts the time required for the project based on the PERT using Simulation.

The Second objective of the study is to crash the activities to complete the project with less time. Find the cost expansion related with time reduction of the project, and understanding the use of computer program to find the crashing schedule for the project in hand.

1.3 Methodology

This is a practical Research, which reveal the program evaluation and review technique and its application by observing project duration schedule of "Automated Warehouse System" by "R.C.Colemarr" and the most important aspect of the research is the computer simulation of the PERT and CPM technique through stochastic simulation. For simulation the program is written in C language, compiled by GCC compiler and Turbo C++ 4.0 IDE is used. The crashing of activities is done by linear programming. The other related calculation is done through the use of Microsoft Office Excel.

2. Test data

R.C.Colemarr distributes a lot of food products including vegetables, fishes, meets, eggs, seeds and so on. It also distributes a variety of food products that are sold through its own outlets and some other outlets and super market stores. The firm receives orders directly from individual outlets, with a typical order requesting the delivery of several cases of 20 to 50 different products from anywhere. Under its current warehouse operation the warehouse clerks dispatch order-picking personnel to fill each order and have the goods to moved to the warehouse shipping area. Because of low productivity of hand order picking and cost of rented warehouse, management has decided to build the automated warehouse operation by installing a computer controlled order-picking system, along with a conveyor system for moving goods from storage area to shipping area.

R.C.Colemarr's Director of material management has been named the project manager in charge of the "Automated warehouse system". After consulting with members of engineering staff and warehouse management personnel, the director compiled a list of activities associated with the project. The list of the activities compressing the project and the predecessors of each activity along with their respective optimistic, pessimistic and most probable time assumptions are reported in Table 1. The director also compiled the crashing information about the "Automated Warehouse System", to determine the activity crashing decisions and to revise activity schedule for the warehouse expansion project.

Table 1: The activity time statement with crashing information

Activity	Immediate Predecessor	Time (in weeks)				cost(in thousands)	
		Optimistic	Most Probable	Pessimistic	Crash	Normal	Crash
A	…………	4	6	8	4	1000	1900
B	…………	6	8	16	7	1000	1800
C	A,B	2	4	6	2	1500	2700
D	C	8	10	24	8	2000	3200
E	C	7	10	13	7	5000	8000
F	E	4	6	8	4	3000	4000
G	C	4	6	20	5	8000	10250
H	D,F,G	4	6	8	4	5000	6400
I	D,F	4	6	14	4	10000	12400
J	H	3	4	5	3	4000	4400
K	I,J	2	4	6	3	5000	5500

The Table 1 reports the crash time and cost of each activities of the project along with its predecessors.

3. Findings of manual calculation

From collected data where each activity should have starting node and a finish node, using the immediate predecessor information in Table 1 we can construct a topological order of activities which is shown in Table 2.

Table 2: Topological order of activity

Activity	Start Node	End Node	Mean Time (weeks)	Sigma
A	1	2	6	0.666667
B	1	3	9	1.666667
Dummy 1	2	3	0	0
C	3	4	4	0.666667
D	4	6	12	2.666667
E	4	5	10	1
F	5	7	6	0.666667
G	4	8	8	2.666667
Dummy2	6	7	0	0
H	8	9	6	0.666667
Dummy 3	7	8	0	0
I	7	10	7	1.666667
J	9	10	4	0.333333
K	10	11	4	0.666667

We can construct a graphical representation of the project or the project network from the Table 2. Figure 1 shows the AOA network for the "Automated Warehouse system" project.

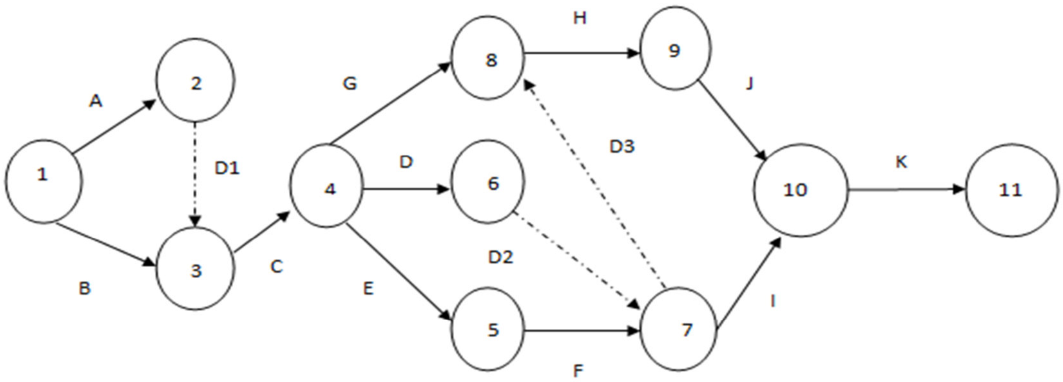

Figure 1: AOA network diagram (dummy activities are shown in dotted lines).

Table 1 describes the three time assumption about the activities of the project. The following table shows the mean time and standard deviation for each activity of the project calculated using the formulas

Mean (μ) = (Optimistic + 4 x Most likely + Pessimistic) / 6

Standard Deviation (σ) = [(Pessimistic - Optimistic) / 6]

Standard deviation is determined taking square root of the variance. For each activity expected mean time and sigma is tabulated in Table 2 where left most column present the activities the second right column shows the mean time and the right one shows the standard deviations. The mean time is estimated in weeks and sigma is taken as six decimal point precision.

Using PERT traditional calculation method with mean duration the project completion time is 43 weeks and the critical path is found as 1->3->4->5->7->8->9->10->11 and the critical activities are B, C, E, F, DUMMY 3, H, J, K. the expected mean time to complete the project is found 43 weeks equal to the longest path. In following figure critical path is shown in dotted line.

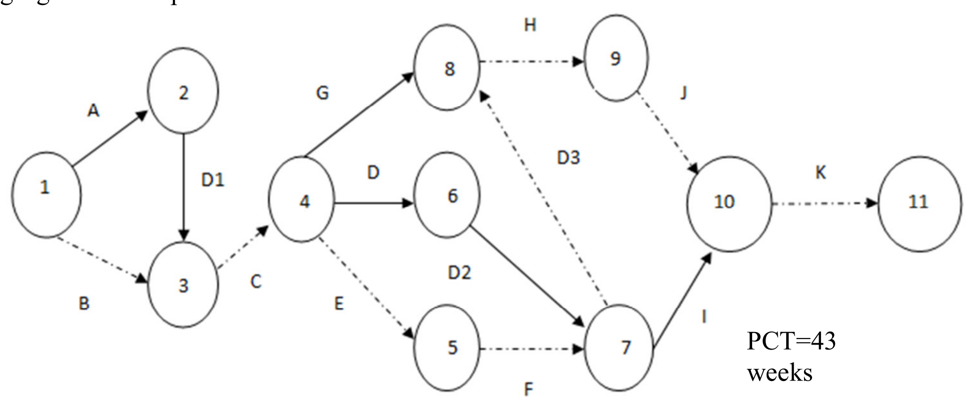

Figure 2: Critical path of the AOA network (dotted line)

The original developers of CPM provided the project manager with the option of adding resources to selected activities to reduce project completion time. Added recourses generally increases project costs, so the decisions to reduce activity times must take into consideration the additional cost involved. In effect the project manager must make a decision that involves trading reduced activity time for additional project cost. Our project's critical path is 43 weeks in length. Realizing that meeting desired Project Completion Time (PCT) is impossible unless we crash the schedule. This shortening of activity time is called crashing.

To determine just where and how much to crash activity times, information on how much each activity can be crashed and how much the crashing process costs. The normal and crash activity data is calculated from table 1 presented previously. Table 3 illustrates the cost –crash slope for each activity.

Table 3: Cost slope for crashing

Activity	Time (in weeks)		Cost(in thousands)		Maximum reduction in time	Crash cost per week
	Mean	Crash	Normal	Crash		
A	6	4	1000	1900	2	450
B	9	7	1000	1800	2	400
C	4	2	1500	2700	2	600
D	12	8	2000	3200	4	300
E	10	7	5000	8000	3	1000
F	6	4	3000	4000	2	500
G	8	5	8000	10250	3	750
H	6	4	5000	6400	2	700
I	7	4	10000	12400	3	800
J	4	3	4000	4400	1	400
K	4	3	5000	5500	1	500

The crash cost per day is calculated by using the formula of crash slop described in earlier chapter. Table 3 illustrates the cost increase of the project completion if activities are crashed one unit (here unit is 1 week), such as the activity "B" could be crashed by maximum 2 weeks and for each week crashing it will increase the cost of project completion by 4000.000.

How much and which activities should be crashed , first reaction to this question may be to consider the critical activities- B,C,E,F,H,J,K. Activity B have the lowest cost per week of the others, and crashing this activity by 2 weeks will reduce the PCT to 41 weeks with cost of 46300,000 BDT.

Table 4: Crashing using linear pro.

Iteration number	Activity with crash time	PCT in weeks	Critical Path	Total Cost in thousands
0	Normal activity time	43	B-C-E-F-H-J-K	44500
1	Decrease B by 2 Weeks	41	B-C-E-F-H-J-K	46300
2	Decrease J by 1 week	42	B-C-E-F-H-J-K	45900
3	Decrease K by 1 week	42	B-C-E-F-H-J-K	46000
4	Decrease C by 2 weeks	41	B-C-E-F-H-J-K	46700
5	Decrease E by 3 weeks	40	B-C-E-F-H-J-K	48500
6	Decrease F by 2 weeks	41	B-C-E-F-H-J-K	46000
7	Decrease H by 2 weeks	41	B-C-E-F-H-J-K	46400
8	B=7 weeks and J=3 weeks	40	B-C-E-F-H-J-K	46700
9	B=7 weeks F=4 weeks J=3 weeks	38	B-C-E-F-H-J-K, B-C-D-H-J-K	47700
10	B=7 weeks F=4 weeks J=3 weeks K=3week	37	B-C-E-F-H-J-K, B-C-D-H-J-K	48200
11	B=7 weeks C=2 weeks F=4 weeks J=3 weeks K=3weeks	35	B-C-E-F-H-J-K	49400
12	B=7 weeks C=2 weeks F=4 weeks H=4 weeks J=3 weeks K=3weeks	33	B-C-E-F-H-J-K, B-C-E-F-I-K	50800
13	B=7 weeks C=2 weeks E=8 weeks F=4 weeks H=4 weeks J=3 weeks K=3weeks	31	B-C-E-F-H-J-K	52800
14	B=7 weeks C=2 weeks D=11 weeks E=7 weeks F=4 weeks H=4 weeks J=3 weeks K=3weeks	30	B-C-D-H-J-K, B-C-E-F-H-J-K, B-C-E-F-I-K, B-C-D-I-K	54100

The table 4 shows the resulting crash schedule from a trial and error approach. The Table represents different crashed time for different project completion time and the corresponding cost with each period of project completion.

4. Findings of Computer Simulation

The program is written using C language and complied by GCC compiler. For the development of the program the Turbo C++ 4.0 integrated development environment (IDE) is used. Another IDE may be used such as Dev C, Visual C and MS Dos. The output for the program of critical path calculation is given below.

```
Project Complition Time= 43 Weeks
  I  S[i]  F[i]  T[i]  EST[i]  EFT[i]  ECT[i]  LCT[i]  CRT[i]
  1   1     2     6      0       6       3       9       0
  2   1     3     9      0       9       0       9       1
  3   2     3     0      6       6       9       9       0
  4   3     4     4      9      13       9      13       1
  5   4     6    12     13      25      17      29       0
  6   4     5    10     13      23      13      23       1
  7   5     7     6     23      29      23      29       1
  8   4     8     8     13      21      21      29       0
  9   6     7     0     25      25      29      29       0
 10   7    10     7     29      36      32      39       0
 11   7     8     0     29      29      29      29       1
 12   8     9     6     29      35      29      35       1
 13   9    10     4     35      39      35      39       1
 14  10    11     4     39      43      39      43       1
Critital Activities Are:  2  4  6  7 11 12 13 14
Slack: N 1  2  3  4  5  6  7  8  9 10 11
       S 3  0  3  0  4  0  0  8  4  3  0_
```

Output 1: Critical Path calculation

The output of the complete simulation for 500 trials is given in next snapshot, the expected time of each activity is randomly generated from the mean and sigma presented in previous chapter and each time critical path is calculated and critical activities are also identified. The frequency of an activity being critical for the simulation length (Run) =500 and criticality index for each activities are also calculated. The probable completion time of the project in each run has been calculated. The certainty of project completion within given period is also shown. To increase the reliability of the simulation result the length of RUN is increased by Run=1000, Run=2000 and Run=4000.the snapshots of respective Run s are given below

```
N=14 M=11
   0  77  0  77  13  119  193  1  21  15  222 223 414 484
Critical Indexes of Activities 1 to n
ct 1=0.00 ct 2=0.15 ct 3=0.00 ct 4=0.15 ct 5=0.03 ct 6=0.24 ct 7=0.39
ct 8=0.00 ct 9=0.04 ct 10=0.03 ct 11=0.44 ct 12=0.45 ct 13=0.83 ct 14=0.97
    Range        Frequency
32.00 to 34.00       0
34.00 to 36.00       0
36.00 to 38.00      10
38.00 to 40.00      31
40.00 to 42.00     124
42.00 to 44.00     160
44.00 to 46.00     105
46.00 to 48.00      57
48.00 to 50.00      11
50.00 to 52.00       2
52.00 to 54.00       0
54.00 to 56.00       0
```

Output 2: Simulated output for 500 run


```
N=14 M=11
   16  733  16  749  88  1123  1676  3  183  98  1964 1967 3377 3840
Critical Indexes of Activities 1 to n
ct 1=0.00 ct 2=0.18 ct 3=0.00 ct 4=0.19 ct 5=0.02 ct 6=0.28 ct 7=0.42
ct 8=0.00 ct 9=0.05 ct 10=0.02 ct 11=0.49 ct 12=0.49 ct 13=0.84 ct 14=0.96
    Range        Frequency
32.00 to 34.00       0
34.00 to 36.00       0
36.00 to 38.00      36
38.00 to 40.00     301
40.00 to 42.00     885
42.00 to 44.00    1365
44.00 to 46.00     934
46.00 to 48.00     390
48.00 to 50.00      81
50.00 to 52.00       8
52.00 to 54.00       0
54.00 to 56.00       0
```

Output 3: Simulated output for 4000 run

Differences between these outputs of Different RUN length are visible from the snapshots. The snapshot shows the change in criticality index of each activity. Such as criticality index for activity 10 decreases as the simulation 'Run' length increases. The differences between PCT range and frequency will be described graphically in next section of this chapter. The crashing of the project is calculated by MS Excel using linear programming. The slope of crash cost is calculated by C program and the output is presented in next snapshot.

Table 5: Crash time with cost

Activity	Project Completion Time	Critical Path	Total Cost(in thousands)
B=7 weeks C=2 weeks D=11 weeks E=7 weeks F=4 weeks H=4 weeks J=3 weeks K=3weeks	30Weeks	B-C-D-H-J-K, B-C-E-F-H-J-K, B-C-E-F-I-K, B-C-D-I-K	54100

The crash schedule and time of project completion is presented with total cost of project in table 5. The Table present the result of crashing based on linear programming done through MS Excel. The left most column presents the time of activities after crashing, second column shoes the project completion time and the third column shows the critical paths and the last column presents the total cost for completing the project.

```
N=14 M=11
   11  41  11  52  2  75  143  0  6  130  22  22 235  431
Critical Indexes of Activities 1 to n
ct 1=0.02 ct 2=0.06 ct 3=0.02 ct 4=0.10 ct 5=0.00 ct 6=0.15 ct 7=0.29
ct 8=0.00 ct 9=0.01 ct 10=0.26 ct 11=0.04 ct 12=0.04 ct 13=0.47 ct 14=0.86
    Range        Frequency
24.00 to 26.00       0
26.00 to 28.00       7
28.00 to 30.00      38
30.00 to 32.00      61
32.00 to 34.00      68
34.00 to 36.00      48
36.00 to 38.00      26
38.00 to 40.00       2
40.00 to 42.00       1
42.00 to 44.00       3
44.00 to 46.00       0
46.00 to 48.00       1
48.00 to 50.00       0
50.00 to 52.00       0
52.00 to 54.00       0
```

Output 4: Simulation output for 500 run after crashing

```
ct 8=0.00 ct 9=0.01 ct 10=0.26 ct 11=0.04 ct 12=0.04 ct 13=0.49 ct 14=0.88
   Range        Frequency
24.00 to 26.00      0
26.00 to 28.00      1
28.00 to 30.00      4
30.00 to 32.00     34
32.00 to 34.00    151
34.00 to 36.00    248
36.00 to 38.00    265
38.00 to 40.00    202
40.00 to 42.00     80
42.00 to 44.00     16
44.00 to 46.00      2
46.00 to 48.00      7
48.00 to 50.00      0
50.00 to 52.00      2
52.00 to 54.00      3
54.00 to 56.00      2
```

Output 5: Simulation output for 2000 run after crashing

5. Analysis of Result

According to our manual calculation the mean project completion time is 43 weeks, but according to the output 2 many more project completion time is present and the range and respective frequency of PCT is presented in tabular and graphical form here. The criticality index of each activity in different Run length is also given below.

Table 6: Criticality index of activities in different simulation length

Activity	500 run		1000run		4000run	
	Time of activity being critical	Criticality index	Time of activity being critical	Criticality index	Time of activity being critical	Criticality index
1	0	0.00	1	0.00	16	0.00
2	77	0.15	167	0.17	733	0.18
3	0	0.00	1	0.00	16	0.00
4	77	0.15	168	0.17	749	0.19
5	13	0.03	18	0.02	88	0.02
6	119	0.24	268	0.27	1123	0.28
7	193	0.39	421	0.42	1676	0.42
8	1	0.00	1	0.00	3	0.00
9	21	0.04	38	0.04	183	0.05
10	15	0.03	31	0.03	98	0.02
11	222	0.44	489	0.49	1964	0.49
12	223	0.45	490	0.49	1967	0.49
13	414	0.83	832	0.83	3377	0.84
14	484	0.97	966	0.97	3840	0.96

The above Table shows different criticality indexes for each activity in different Simulation run length. For activity 6, it has a criticality index of 0.24 in 500 run length, 0.27 and 0.28 in 1000 and 4000 run respectively. But the frequency of their being critical is different from each and other run respectively. Activity 14 is the mostly occurred critical activity.

Table 7: Frequency of Project completion time in different simulation length

Serial	PCT Range in weeks	Frequency in 500 run	Frequency in 1000 run	Frequency in 2000 run	Frequency in 4000 run
1	32 to34	0	0	0	0
2	34 to36	0	0	0	0
3	36 to38	10	16	24	36
4	38 to40	31	60	135	301
5	40 to42	124	242	460	885
6	42 to44	160	330	659	4365
7	44 to46	105	221	467	934
8	46 to48	57	110	211	390
9	48 to 50	11	19	40	81
10	50 to52	2	2	4	8
11	52 to54	0	0	0	0

Table 7 present different Project completion time (in weeks) range and the respective frequency of each range during each length of PERT simulation.

Figure 3 depicts the above findings of table 7. The vertical axis represents the frequency of Project Completion Time in different simulation iteration. The horizontal axis represents the project completion time in weeks.

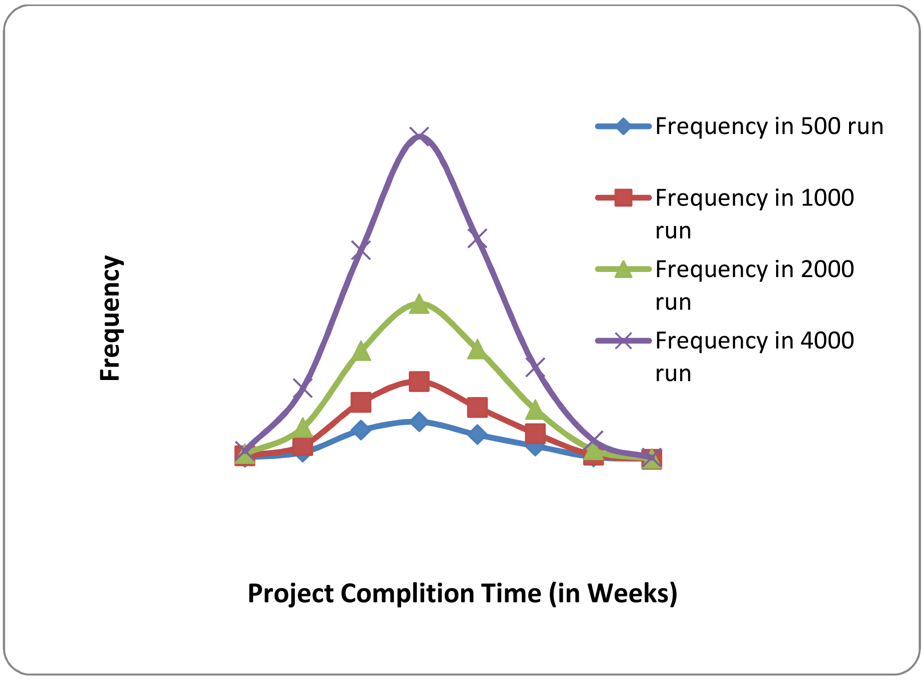

Figure 3: Frequency of PCT range in 500, 1000, 2000, 4000 Simulation run.

According to the figure 3 PCT ranges of 42 to 44 weeks have much certainty to complete the project in every simulation length; the next one is 44 to 46 week in 4000 and 2000 run. The most uncertain project duration is 50 to 52 weeks and also 36 to 38 weeks as they both of these ranges have low frequency in every run.

The following table, Table 8, presents the probability, cumulative probability to complete the project in each PCT range for 500and 4000 length of PERT simulation respectively. Figure of the respective simulation length are also presented here in figure 4.

Table 8: Frequency, probability and pct in 500 and 4000 run

	Simulation length 500		Simulation length 4000	
PCT Range in weeks	probability in %	Cumulative probability	Probability in %	cumulative probability
32 to34	0	0	0	0
34 to36	0	0	0	0
36 to38	2	2	0.9	0.9
38 to40	6.2	8.2	7.53	8.425
40 to42	24.8	33	22.1	30.55
42 to44	32	65	34.1	64.68
44 to46	21	86	23.4	88.03
46 to48	11.4	97.4	9.75	97.78
48 to50	2.2	99.6	2.03	99.8
50 to52	0.4	100	0.2	100

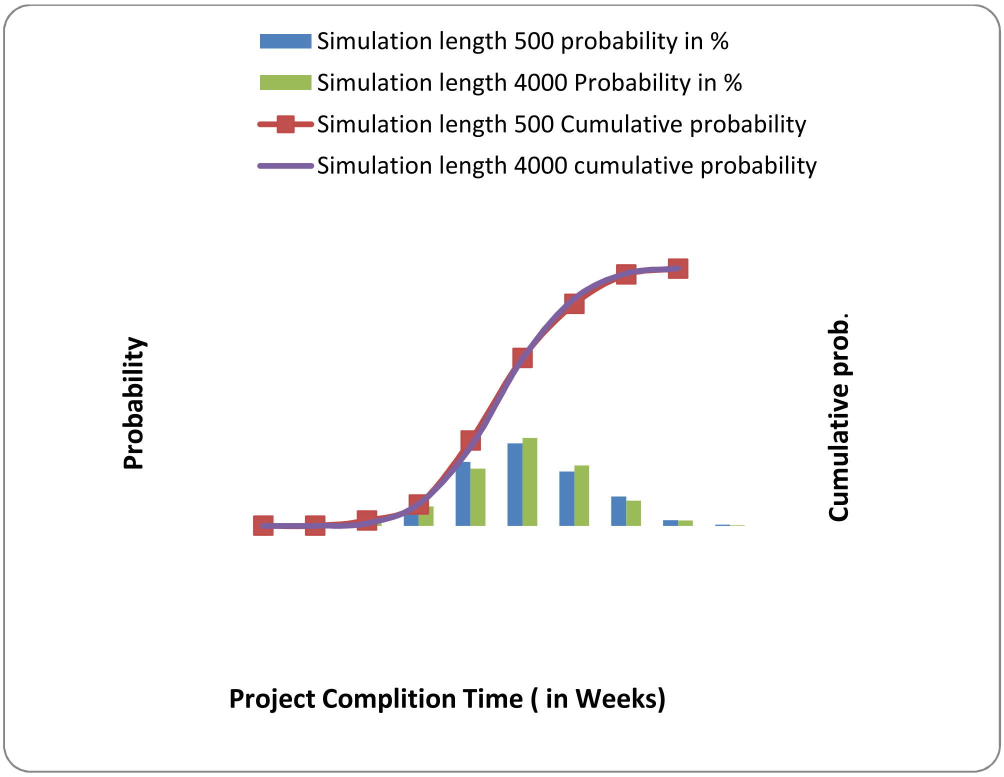

Figure 4: Probability of PCT occurrence in 500 and 4000 run

From the table it is clear that the result will change from iteration to the next as the random activity time changes. As can be seen all paths are critical for some random samples, but they are not equally likely to be critical. Interestingly the CPM critical path 1->3->4->5->7->8->9->10->11 occurs only about 1/3 of the time (as the probability is only around 33%). Hence something other than 1->3->4->5->7->8->9->10->11is critical about the other 2/3 of the time.

From the figures it could be said that the probability of project duration is the highest in 42 to 44 weeks in every simulation length. Figure 4 all proves the above statement. And also proves that completing the project less than 36 weeks is almost impossible (without crashing) in every simulation length. Based on these figures it is 100% possible to complete the project within 50 weeks and more than that. The probability of completing the project within or less than 40 weeks is around 10%.

According to the analysis of manual crashing process it is possible to reduce the duration of project completion to 30 weeks the linear program also present the same conclusion.

Table 9: Probability of PCT after crashing

PCT Range (in Weeks)	Simulation Length 500		Simulation Length 2000 run	
	Frequency	probability %	Frequency	probability %
26 to 28	7	1.4	1	0.05
28 to 30	38	7.6	4	0.2
30 to 32	61	12.2	34	1.7
32 to 34	68	13.6	151	7.55
34 to 36	48	9.6	248	12.4
36 to 38	26	5.2	265	13.25
38 to 40	2	0.4	202	10.1
40 to 42	1	0.2	80	4
42 to 44	3	0.6	16	0.8
44 to 46	0	0	2	0.1
46 to 48	1	0.2	7	0.35
50 to 52	0	0	2	0.1
52 to 54	0	0	3	0.15
54 to 56	0	0	2	0.1

The table 9 shows that completing project in 30 weeks have 12% and 2% probability in 500 and 2000 run of simulation. The highest and acceptable probability of completing the project is within 36 to 38 weeks having a probability of 13.25% on 2000 simulation.

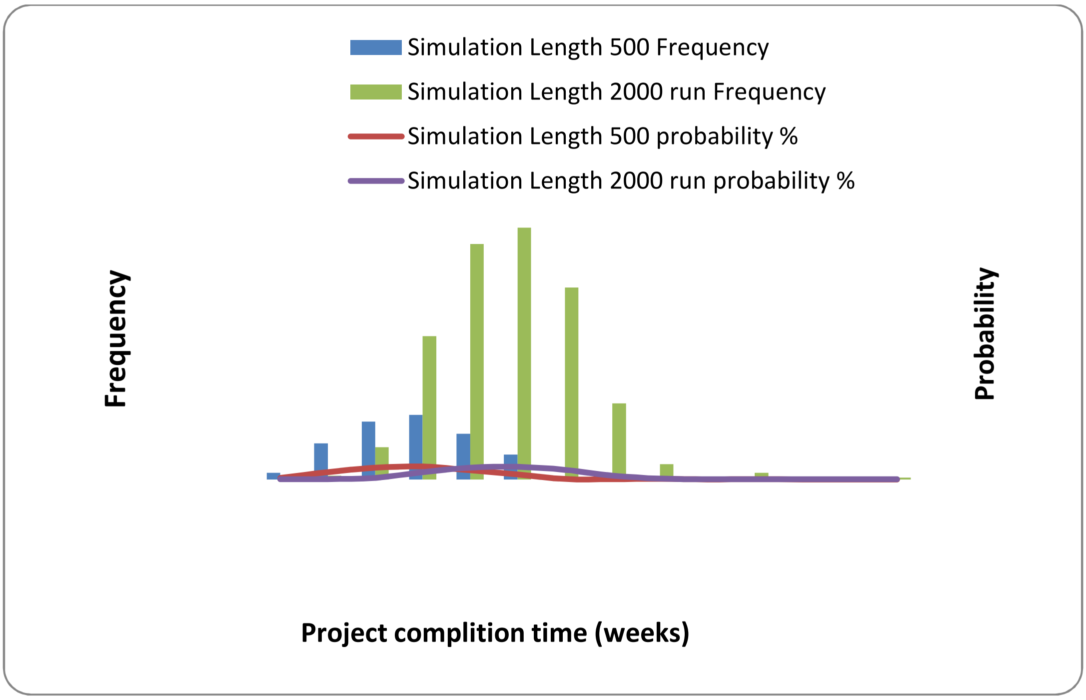

Figure 5: Probability of completing project in Crash schedule for 2000 run

Figure 5 shows in 500 run PCT range 32 to 34 Weeks have the highest probability, but in 2000 run probability of the range is 7.5%, refers this duration is somewhat risky.

6. Conclusion

A stochastic simulation model is developed to realize the time duration for the project completion based on PERT. PERT analysis implicitly assumes that all activities that are not critical could be ignored by setting the time to average values. In stochastic condition every non critical path has the chance to become critical. It is also assumed that activity duration is independent variables, but in reality they could be dependent. All this assumptions result in erroneous results. The simulation of PERT is used to determine the durations possible for the project completion and the certainty on other words the probabilities to meet those durations.

From the study it is assumed that the certainty of project completion in 43 weeks or before it is 65%, but in traditional method 43 weeks is the only certain duration to complete the project in hand “Automated Warehouse system”.

References

Van Slyke, R. M. (1963), ‘Monte Carlo methods and the PERT problem’, *Operations Research*, 13, 141-143.

Steve, P. Jr. and M.I. Dessouky, (1977), ‘Solving the project time/cost tradeoff problem using the minimal cut concept’*, Management Science*, 24: 393-400.

Moore, L. J. et al., (1978), ‘Analysis of a multi-criteria project crashing model’, *AIIE Transactions*, 10(2), 163-169.

Coskun, O. (1984), ‘Optimal probabilistic compression of PERT networks’, *J. Construct. Eng.Manage*, 110:437:446. Retrieved from http://cedb.asce.org/cgi/WWWdisplay.cgi?840265.

Ramini, S. (1986), ‘A simulation approach to time cost trade-off in project network’, *Proceedings of the Conference modeling and simulation on microcomputers*. pp: 115-120.

Ameen, D.A. (1987), ‘A computer assisted PERT simulation’, *J. Syst. Manage*., 38: 6-9. Retrieved from http://portal.acm.org/citation.cfm?id=34619.34620.

Johnson, G.A. and C.D. Schou,(1990), ‘Expediting projects in PERT with stochastic time estimates’, *Project Manage. Journai.*, 21: 29- 32.

Badiru, A.B., (1991), ‘A simulation approach to Network analyses, *Simulation*, 57: 245-255.

Foldes, S. & Sourmis F. (1993), ‘PERT and crashing revisited: Mathematical generalizations’, *European Journal of Operational Research*, 64, 286-294.

Feng, C.W., L. Liu and S.A. Burns, (2000), ‘ Stochastic construction time-cost tradeoff analysis’, *Journal of Computer Civil Eng*., 14: 117- 126.

Jorgensen, M., (2003), ‘Situational and task characteristics systematically associated with accuracy of software development effort estimates’, Proceedings of the Information Resources Management Association

International Conference, Philadelphia, PA.
Haga, W. A., and K. A. Marold. (2004), 'A simulation approach to the PERT-CPM time-cost trade-off problem', *Project Management Journal*, 35(2): 31-37.
Haga, W. A., and K. A. Marold. (2005), 'Monitoring and control of PERT networks', *The Business Review*, 3(2): 240-245.
David R. Anderson, Dennis J.Sweeney, Thomas A. Williams, 'An introduction to management science quantitative approaches to decision making'; 12th edition; south-western
D.H Hira; System simulation; S.Chand & company ltd.
Jack R. Meredith, Samuel J. Menntel,Jr. Project management a managerial approach; 17th edition; New York: Wiley.