Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 13, Number 9 (2017), pp. 6737-6754

http://www.ripublication.com

Study on Modified Unit Time Method

[1]S.Shanmugasundaram and [2]V.Mathan Kumar

[1]Assistant Professor, Department of Mathematics, Government Arts College (A), Salem. (TN), India.

[2]Assistant Professor, Department of Mathematics, Sri Ganesh College of Arts & Science, Salem, India.

Abstract

Reducing both construction project cost and time is critical in today's market driven economy. The relationship between the construction project cost and time is also known as time-cost trade off decisions. Crashing is used reducing project time by expending additional resources crashing the network is the contracting or compressing the network that means to reduce project duration at minimum cost with minimum project duration. In this paper we presents the time management by crashing technique. We give a modified unit time method algorithm to obtain optimum project cost and duration. An illustration is given to demonstrate the performance of the proposed algorithm.

Keywords: crashing, time-cost trade off, cost slope value, unit time method

1. INTRODUTION

The cost of the network activities had been optimized for various overall durations. The optimum crash of time against cost had been made. This approach was an acceptable tool of management and proving to be not only superior method planning, scheduling and controlling project progress, but also was very real and valuable assets to contractors in convincing the owner of their potentials and abilities [1].

Time-cost tradeoff problems from the late 1950s mostly concentrated on shortening overall project duration by crashing the time required to complete individual activities. Researches in this area include linear programming modes [2].

The critical path method (CPM) is used for all types of projects, such as construction, engineering, facility maintenance, software development, and many more. The CPM can be used to determine the time–cost tradeoff for projects that meet a given completion times at minimum cost and is useful when there are similar experiences from previous projects [3].

Unit time Method is the powerful method for crashing. If project requires crashing of double figure time (that is what generally required in industries); it is challenging [4]. In this paper, we present the modified algorithm of unit time method is comparatively better than existing algorithm to obtain the optimum cost and duration of the project.

Maximum flow-Minimal cut concept is another manual optimization method for construction schedules [5]. This method states that "in the network from start to end the maximal flow is equal to the minimum cut set capacity". This method is advancement over the earlier classical optimization techniques as it is possible to solve it through computer program and is helpful when there are multiple critical paths with numerous interrelated activities that result in a large number of compression sets to review.

The time-cost optimization techniques are based on the critical path method (CPM) and have been used in the construction industry over the last several years. An important advantage of CPM is that it helps project managers to identify the activities that are critical for the timely completion of project [6].

The optimization of time and cost process technique could be incorporated as a standard procedure for every project was concluded, the time spent on the actual crashing was minimal and the project management schedule could be reduced to minimum optimum level to save time and money [9].

Project costs are generally classified into two categories: the direct costs related to individual activities and the indirect costs related to overhead items. The problem we explore in this paper focuses on individual activities under the assumption that the time-cost tradeoffs for project activities are linear (Swink et al., 2006) [10].

2. NOTATION

i	Start event 'i' for an activity
j	End event 'j' for an activity
A_{ij}	Activity having start event 'i' and end event 'j'
T_{ij}	Time or duration for A_{ij}
NT_{ij}	Normal time for A_{ij}
CT_{ij}	Crash time for A_{ij}
Δ_{ij}	$NT_{ij} - CT_{ij}$
Δ_K	$NT_K - CT_K$
NC_{ij}	Normal cost for A_{ij}
CC_{ij}	Crash cost for A_{ij}
CS_{ij}	Cost slope for $A_{ij} = \dfrac{CC_{ij} - NC_{ij}}{NT_{ij} - CT_{ij}}$
CS_K	Cost slope for Activity K
TCS_{ij}	$=\Sigma\ CS_{ij}$ for least cost slope activities from different CP_s = CS_{ij} for least cost slope activity from critical path
CL_{ij}	Crash length for activity $_{ij} = NT_{ij} - CT_{ij}$
CP_K	Project path "K" which happens to be critical
CPL_K	Length for critical path 'K', $CP_K = \Sigma T_{ij}$ for $A_{ij} \in K$
NP_K	The path of the project just shorter(Next) to critical path 'K'
NNP_K	The path of the project just shorter (Next) to NP_K path 'K'
NPL_K	Length (duration) for the path of the project just shorter(next) to critical path 'K' $CP_{NK} = \Sigma T_{ij}$ for $A_{ij} \in N$
$NNPL_K$	Length (duration) for the path of the NNP_K
F_K	Critical path(K) Float limit or difference between critical path length and length of next to critical path = $CPL_K - NPL_K$
NF_K	Critical path(K) Float limit or difference between critical path length and length of next to next to critical path = $CPL_K - NNPL_K$
CTL_{ij}	Crash-Time-Limit or maximum limit activity ' i-j ' could be crashed in one stretch = $\min\{ F_K, (NT_{ij} - CT_{ij})\}$
$NCTL_{ij}$	Crash-Time-Limit or maximum limit activity ' i-j ' could be crashed in one stretch when crash activity is common to CP and NP = $\min\{ NF_K, (NT_{ij} - CT_{ij})\} = \min\{ NF_K, \Delta_{ij} \}$

3. NETWORK AND CRITICAL PATH

We use Activity On Arc (AOA) however AON network could be used. Crashing could be done without network preparation by path table.

Critical Path Float Limit: Projects have many paths which are evident from the network. These paths are denoted by number, 1, 2, 3…, K,… (Table 3.1) Each path has length. Critical path/s is/are the path/s with longest duration. This length is denoted by CPL_K. The path which is just shorter than critical path (K) is denoted by NP_K and its length by NPL_K.

Definition: The difference in these two paths is defined as Critical Path Float Limit,

$F_K = CPL_K - NPL_K$ Activities on critical path have no floats. All floats for critical activities are zero (Total Float, Free float, Safety float, Independent float, and Interfering float). The new definition is not activity float but path float.

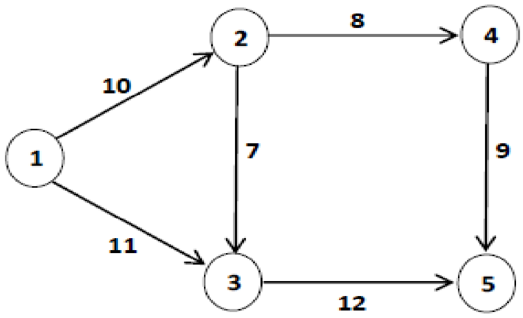

Figure 3.1

Consider AOA network showing activity durations in Figure 3.1. Three different paths are evident as listed in Table 3.1.

Table 3.1- Path Table

	Path	Length
1	1-2-4-5	10+8+9=27
2	1-2-3-5	10+7+12=29
3	1-3-5	11+12=23

The network have three paths namely path 1, 2 and 3. Path 2 is longest and hence critical path; CP_2 have length CPL_2 = 29 days. Next path just short to critical path is path 1 and its length is 27 days. This path is relative to critical path 2; NP_2 have length NPL_2 = 27 days.The difference between these paths is 2. So with notations, it is

Critical Path Float Limit for *critical path 2*: $F_2 = CPL_2 – NPL_2$ = 29 - 27 = 2 days. When activity on CP is crashed, the length of CP would be reduced by crashed time. Its effect on NP is observed:

- ❖ Case 1: No change in NPL
- ❖ Case 2: NPL reduction

E.g. If the activity 2-3 is crashed by 1 day, then

CPL_2 = 29 – 1 = 28 days.

NPL_2 = 27 days NPL unchanged.

F_2 = 28 - 27 = 1 day.

Instead of crashing 2-3, if common activity to CP and NP; 1-2 is crashed by 1 day then

CPL_2 = 9 + 7 + 12 = 28 days.

NPL_2 = 9 + 8 + 9 = 26 days.

NPL_2 is reduced from 28 to 26 days.

F_2 = 28 - 26 = 2 days.

***Theorem 1**:* If activity on CP is crashed by unit time; NPL might be unchanged or would also reduce by unit time.

Theorem 2: If activity on CP is crashed by unit time; Critical Path Float Limit might be reduced by unit time or unchanged. In Case 2: Critical Path Float Limit is unchanged, and have no effect on criticality (CP is CP and NP is NP). This would continue till common activity which is being crashed; could not be crashed any further. But consider case 1; here for crashing one time unit, CPL reduces by one unit but NPL unchanged. So the crashing could continue for one more time unit, so on. A time would come when CPL and NPL are equal or both are CP. The situation changes now, crashing cannot be done with logic of one CP. This point is when CP is crashed by the difference in (CPL – NPL) or Critical Path Float limit (F_K). This is the worst case situation.

Theorem 3: Activity on CP could be crashed by Critical Path Float Limit without affecting Criticality of NP(CP is CP and NP is NP).

Crash-Time-Limit for an Activity

Theorem 4: Any activity on CP could be crashed maximum to crash period, Δ_{ij} or (NT_{ij} - CT_{ij}). This is technological constraint on the activity and could not be violated.

Theorem 5: Any activity on CP could be crashed to Crash-Time-Limit in one stage such that Crash- Time-Limit; CTL_{ij} = min $\{F_K; \Delta_{ij}\}$ = min $\{F_K; \Delta_K\}$ Theorem 5 is evident because out of two possibilities (bounds) only least bound could be explored in one stage and it will not violate criticality. Our addition of theorem 3, 4 and 5 are implemented in Crashing. Instead of crashing one unit time, activity could be crashed by CTL, without any problem.

Crash-Time-Limit for an Activity when it is common on CP and NP

When activity 'K' is common to CP and NP; both paths would be shortened after the crashing. Hence F_K does not put any constraint on Crash limit. Under such case NNP_K should be taken for consideration. for calculation. $NF_K = CPL_K - NNPL_K$ Crash activity is common to CP, NP and NNP; then further path just shorter than $NNPL_{KN}$ should be considered. Such cases are rare but cannot be neglected. NCTL for crashing when activity is common in CP and NP, $NCTL_K$ = min $\{NF_K; \Delta_{ij}\}$ = min $\{NF_K; \Delta_K\}$

4. COST SHEET

In literature no common standards are followed for crashing. We use cost sheet format from Stevens (1996).

Cost Sheet 4.1				
Activity	Crash Time	Cost Slope	Time Shortened	
i – j	Δ_{ij}	CS_{ij}		
Time cut (crash)		-	-	-
Project duration		-	-	-
Incremental cost		-	-	-
Direct cost		-	-	-
Indirect cost		-	-	-
Total cost		-	-	-

Where with notation

Time cut (crash) = CTL_{ij}

Project duration = CPL

Incremental cost (IncC) = $CTL_{ij} * TCS_{ij}$

Direct cost (DC) = $\sum NC_{ij}$ + InC

Indirect cost (IC) = CPL * indirect cost

Total cost (TC) = DC + IC

MODIFIED ALGORITHM USING CRASH-TIME-LIMIT (CTL) FOR CRASHING

We propose CTL algorithm as follow:

Step 1: Prepare table showing activity, Immediate Predecessors, Normal time, Crash time, Normal cost and Crash cost for each activity Tabulate cost slope (CS_{ij}) or incremental cost per unit time for each activity $CS_{ij} = (CC_{ij} - NC_{ij}) / (NT_{ij} - CT_{ij})$

Step 2: Prepare AON or AOA diagram.

Show NT on network. Prepare a Path-table showing different paths and find lengths of the each path. Note CP, NP, CPL, NPL and F_K from Path table. (*No need to run Forward/backward passes*)

Step 3: Prepare Cost-sheet with CS and CTL, Direct and Indirect cost, Incremental cost and Total cost of the project.

Step 4: If no activity from any one CP could not be crashed, Then Stop, It is crash limit. If more than one critical path(CP), then go to Step 5, Else go to step 7

Step 5: If parallel path exists(more than one CP).Add the crashing activity cost slope value and parallel path/paths least cost slope activity value is compare to next least cost slope value of the original critical path.[CPCCS + PPLCS < CPNCS exists, crash the corresponding activity]Go to step 7. Otherwise go to step 6

Step 6: If CPCCS + PPLCS > CPNCS ,crash the next least cost slope activity.In this step check before least cost slope activity crash times whether completed or not.which cost slope satisfies above step 5&6 conditions,crash the same.Go to step 7.

Step 7: Note the activity (K) with least cost slope which could be ***crashed* from CP*.**

If crash activity (K) is common to CP and NP; Then crash it by $NCTL_K$; Else crash by CTL_{ij}. *Go to step 8.*

Step 8: 1. Update network 2. Update path table

3. Update Cost sheet for CPL (Total incremental cost, DC, IC, TC)

Step 9: If Total cost of the project is increased Then stop; least total cost is the optimum period; solution; Else go to Step 4.

6. ILLUSTRATION

Problem data is given in the Table 6.1 and AOA network diagram 6.1

Table 6.1: Data

Activity		IP	Duration (days)		Cost(Rs.)	
-	event		Normal	Crash	Normal	Crash
-	' i –j'		$NT_{ij}j$	CT_{ij}	NC_{ij}	CC_{ij}
A	1-2	-	9	6	12000	18000
B	2-4	A	14	4	14000	24000
C	2-3	A	4	3	2000	2400
D	3-4	C	6	4	44000	56000
E	1-5	-	14	13	16000	18000
F	5-6	E	6	6	4000	4000
G	4-6	B,D	5	3	4000	4800
H	6-7	F,G	2	1	12000	14000

Indirect cost of the project is Rs.2000 per day.

Step 1: Prepare cost slope Table (Table 6.2)

Table 6.2: Cost Slope Table

Activity		IP	Duration (days)		Cost(Rs.)		time	Cost Slope
-	event		Normal	Crash	Normal	Crash	NT_{ij} - CT_{ij}	CS_{ij}
-	' i –j'		NT_{ij}	CT_{ij}	NC_{ij}	CC_{ij}		
A	1-2	-	9	6	12000	18000	3	2000
B	2-4	A	14	4	14000	24000	10	1000
C	2-3	A	4	3	2000	2400	1	400
D	3-4	C	6	4	44000	56000	2	6000
E	1-5	-	14	13	16000	18000	1	2000
F	5-6	E	6	6	4000	4000	0	-
G	4-6	B,D	5	3	4000	4800	2	400
H	6-7	F,G	2	1	12000	14000	1	2000
TOTAL					108000			

Step 2: AOA network is prepared (Figure 6.1)

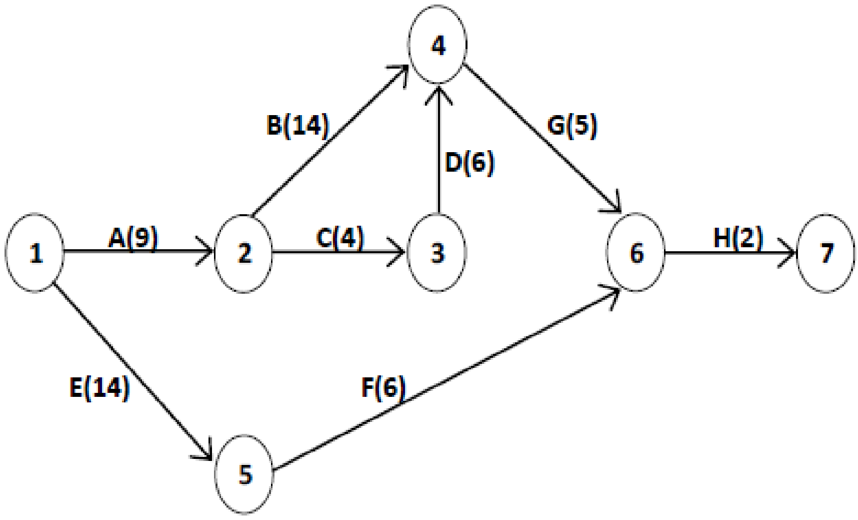

Figure-6.1

1. Prepare Path table (table 6.3)

Table 6.3: Path Table			
Path	I	A-B-G-H	9+14+5+2 = 30*
	II	A-C-D-G-H	9+4+6+5+2=26
	II	E-F-H	14+6+2=22
Critical path, CP			I
Critical path duration, CPL			30
Next to CP length, NPL			26
Critical path float limit, F			4
NCP Float limit, NF			4
Column			I

Step 3:

Table 6.4: Cost Sheet(initial)				
Activity	Time (Days)	CS_{ij}	Days Shortened	
A	3	2000	-	-
B	10	1000	-	-
C	1	400	-	-
D	2	6000	-	-
E	1	2000	-	-
F	0	-	-	-
G	2	400	-	-
H	1	2000	-	-
Days cut		////////////////	-	-
Project duration(CPL)		30	-	-
Incremental cost(IncC)		////////////////	-	-
Direct cost(DC)		108000	-	-
Indirect cost(IC)		60000	-	-
Total cost(TC)		168000	-	-

Step 4: CP is path 1 only one go to step 7.

Step 7: Path 1 is critical path; critical activities are A,B,G&H (from the path Table 6.3).

From cost sheet CS_A =2000; CS_B = 1000; CS_G = 400** and CS_H = 2000.

Least cost slope activity is G from CP; It is common to CP(I) and NP(II) but not common to NNP(III).

$NF_I = CPL_I – NNPL_{III} = 30 – 22 = 8$ days

$NCTL_I = \min\{NF_I, \Delta_G\} = \min\{8,2\} = 2$ days

Activity G to be crashed by 2 days.

Step 8: Update the network, path & cost sheet

Activity G is crashed by 2 days; Δ_G is reduced by 2 days from 2 to 0.

Days cut (CTL) = 2 days

Incremental cost (IncC) = 2 * CS_G = 2*400 = 800

Direct cost (DC) = 108000 + 800 = 108800

Indirect cost (IC) = 28*2000 = 56000

Total cost (TC) = DC + IC = 164800

Step 9: TC(Previous) = 168000

TC (After crash) = 164800

As total cost is reduced, go to step 4.

Step 4: critical path is path I; only one. Go to step 7.

Step 7: Path 1 is CP; Critical activities which can be crashed A,B and H.

From cost sheet CS_A = 2000; CS_B = 1000** and CS_H = 2000.

Least cost slope activity is B from CP; It is not common to NCP(II).

$NF_1 = CPL_1 – NPL_{II} = 28 – 24 = 4$ days

$NCTL_1 = \min\{NF_1, \Delta_B\} = \min\{4,10\} = 4$ days.

Activity B to be crashed by 4 days.

Step 8: Update the network, path & cost sheet

Activity B is crashed by 4 days; Δ_B is reduced by 4 days from 10 to 6.

Days cut (CTL) = 4 days

Incremental cost (IncC) = 4 * CS_B = 4*1000 = 4000

Direct cost (DC) = 108800 + 4000 = 112800

Indirect cost (IC) = 24*2000 = 48000

Total cost (TC) = DC + IC = 160800

Step 9: TC (Previous) = 164800

TC (After crash) = 160800

As total cost is reduced, go to step 4.

Step 4: critical path is path I ,II ; more than one CP exists. Go to step 5.

Step 5:

To check CPCCS + PPLCS < CPNCS(i.e CS_B +CSc < CS_H)

1000 + 400 < 2000

So crash the same activity B. Δ_B is reduced by 1 day from 6 to 5 and activity C is reduced by 1 day alternative (parallel).

Step 8: Update network, path & cost sheet.

Activity B is crashed by 1 day; and

Activity C is crashed by 1 day.

Here Δ_B is reduced by 1 day from 6 to 5 and

Δ_C is reduced by 1 day from 1 to 0 (alternative).

Days cut (CTL) = 1 day

Incremental cost (IncC) = 1 * CS_B + 1* CS_C = 1000+400 = 1400

Direct cost (DC) = 112800 + 1400 = 114200

Indirect cost (IC) = 23*2000 = 46000

Total cost (TC) = DC + IC = 160200

Step 9: TC (Previous) = 160800

TC (After crash) = 160200

As total cost is reduced, go to step 4.

Step 4: Parallel path exists.(More than one)CP is path I,II.Go to step 5

Step 5:

To check CPCCS + PPLCS < CPNCS (i.e CS_B +CS_D < CS_H)

1000 + 6000 < 2000(Condition not satisfied) .Go to step 6

Step 6:

To check CPCCS + PPLCS >CPNCS (i.e CS_B +CS_D > CS_H)

crash the activity H by one day. Go to step 7

Step 7:

From cost sheet CS_H = 2000.

Least cost slope activity is H form CP; H is common to all. Δ_H = 1 day.

$NF_1 = CPL_1 – NPL_{III} = 23 – 22 = 1$ day

$NCTL_1 = \min\{NF_1, \Delta_H\} = \min\{1,1\} = 1$ day.

Activity H to be crashed by 1 day.

Step 8: Update network, path & cost sheet.

Activity H is crashed by 1 day.

Here Δ_H is reduced by 1 day from 1 to 0.

Days cut (CTL)	=	1 day	
Incremental cost (IncC)	=	1* CS_H = 1*2000 = 2000	
Direct cost (DC)	=	114200 + 2000	= 116200
Indirect cost (IC)	=	22*2000	= 44000
Total cost (TC)	=	DC + IC	= 160200

Step 9: TC (Previous) = 160200

TC (After crash) = 160200

As total cost is same, go to step 4.

Step 4: CP is I,II; more than one. Go to step 5.

Step 5:

To check CPCCS + PPLCS < CPNCS (i.e CS_B +CS_D < CS_A)

1000 + 6000 < 2000(Condition not satisfied) .Go to step 6

Step 6:

To check CPCCS + PPLCS >CPNCS (i.e CS_B +CS_D > CS_A)

crash the same activity A. Go to step 7.

Step 7: Critical activity is A.

From cost sheet CS_A = 2000.

Least cost slope activity is A form CP; A is not common to NP_{III}. Δ_A= 3 days.

$NF_1 = CPL_1 – NPL_{III} = 22 – 21 = 1$ day

$NCTL_1 = \min\{NF_1, \Delta_A\} = \min\{1,3\} = 1$ day.

Activity A to be crashed by 1 day.

Step 8: Update network, path & cost sheet.

Activity A is crashed by 1 day.

Here Δ_A is reduced by 1 day from 3 to 2.

Days cut (CTL)	=	1 day	
Incremental cost (IncC)	=	$1^* CS_A = 1^*2000 = 2000$	
Direct cost (DC)	=	116200 + 2000	= 118200
Indirect cost (IC)	=	21*2000	= 42000
Total cost (TC)	=	DC + IC	= 160200

Step 9: TC (Previous) = 160200

TC (After crash) = 160200

As total cost is same, go to step 4.

Step 4: CP is path I,II,III; more than one; Go to step 5.

Step 5: To check CS_B +CS_D Value is compare to CS_A +CS_E.

CS_B +CS_D =1000+6000=7000. CS_A +CS_E=2000+2000=4000 .So crash the activity A.Go to step7.

Step 7:From cost sheet CS_A=2000, Δ_E =1 day and Δ_A= 2 days.

There is no float difference so select activity A crashed by 1 day and activity E crashed by 1 day alternative.

Step 8: Update network, path & cost sheet.

Activity A is crashed by 1 day and activity E is reduced by 1 day alternative;

Here Δ_A is reduced by 1 day from 2 to 1 and Δ_E is reduced by 1 day from 1 to 0.

Days cut (CTL)	=	1 day
Incremental cost (IncC)	=	$1^* CS_A + 1^* CS_E$ = 2000 + 2000 = 4000

Direct cost (DC) = 118200 + 4000 = 122200

Indirect cost (IC) = 20*2000 = 40000

Total cost (TC) = DC + IC = 162200

Step 9: TC (Previous) = 160200

TC (After crash) = 162200

As total cost is increased so stop.

At end path table is as shown below.

Table 6.5: Path table - Final

Path	I	A-B-G-H	9+14+5+2 = 30*	28	24	23	22	21
	II	A-C-D-G-H	9+4+6+5+2=26	24	24	23	22	21
	II	E-F-H	14+6+2=22	22	22	22	21	21
Critical path, CP			I	I	I,II	I,II	I,II	I,II,III
Critical path duration, CPL			30	28	24 22	23	22	21
Next to CP length, NPL			26	24	22	22	21	-
Critical path float limit, F			4	4	2	1	1	-
NCP Float limit, NF			4	2	-	-	-	-
Column			i	ii	iii	iv	v	vi

At end cost sheet is as shown below.

Table 6.6: Cost Sheet - Final

Activity	time	CS_{ij}	Days shortened					
	days							
A	3	2000					**1***	**1***
B	10	1000		**4***	**1***			
C	1	400			1			
D	2	6000						
E	1	2000						1
F	0	-						
G	2	400	**2***					
H	1	2000				**1***		
Days cut		////////////////	2	4	1	1	1	1
Project duration(CPL)		30	28	24	23	22	21	20
Incremental cost(IncC)		////////////////	800	4000	1400	2000	2000	4000
Direct cost(DC)		108000	108800	112800	114200	116200	118200	122200
Indirect cost(IC)		60000	56000	48000	46000	44000	42000	40000
Total cost(TC)		**168000**	**164800**	**160800**	**160200**	**160200**	**160200**	**162200**

SOLUTION:

The above table indicates, the normal duration of the project is 30 days and the associated cost is Rs.168000/-,

the optimum duration of the project is 21 days and the associated cost is Rs.160200/-,

the minimum duration of the project is 20 days and the associated cost is Rs.162200/-

CONCLUSION

Several researchers have investigated the critical path analysis in the project network with deterministic times. In our research, modified unit time algorithm is also one of the methods handled for crashing and shows the optimum solution for the construction project. This algorithm works for many managers in the construction to take decisions. If the parallel path does not exists in the network, this algorithm will give less iteration to obtain the solution.

REFERENCES

[1] Archibald, R. D. and Villoria, R. L., 1966, "*Network-based management systems (PERT/CPM)*." Wiley, New York.

[2] Elmaghraby, S.E., Salem, A., 1982, "*Optimal project compression under quadric cost functions*." Applications of Management Science 2, 1-39.

[3] Hillier, F.S., Lieberman, G.J., 1995, "*Introduction to mathematical programming*." McGraw-Hill Inc, New York.

[4] B.R.Kharde,G.J.Vikhe Pati "*An Efficient Algorithm for Crashing.*" International Journal of Scientific & Technology Research, Volume1, issue4, May2012, ISSN2277-8616.

[5] Liu, J., and Rahbar, F. (2004), "*Project time-cost tradeoff optimization by maximal flow theory.*" Journal of Construction Engineering and Management, 130(4), 607-609.

[6] Moussourakis, j., and Haksever, C. (2007), "*Models for accurate computation of earliest and latest start times and optimal compression in project networks*." Journal of Construction Engineering and Management, 133(8), 600-608.

[7] Senouci A.B. and Eldin N.N., 1996, "*A time-cost trade-off algorithm for nonserial linear projects.*" Canadian J. of Engineering, vol. 23, pp. 134-149.

[8] Stevens, J.D. (1996), "*Techniques for construction Network Scheduling*."

[9] Suri, P.K., and Bhushan, B., 2008, "*Simulator for Otimization of Software Project Cost and Schedule.*" Journal of Computer Science, vol. 12, pp. 1030-1035.

[10] Swink, M., Talluri, S., Pandejpong, T., 2006, "*Faster, better, cheaper: a study of NPD project efficiency and performance tradeoffs.*" Journal of Operation Management, 24, 542-562.