Project Networks With Mixed-Time Constraints

L Caccetta and B Wattananon

Western Australian Centre of Excellence in Industrial Optimisation (WACEIO), Curtin University of Technology, GPO Box U1987, Perth, Western Australia, 6845.

Abstract : A Project can be viewed as a well-defined collection of tasks, jobs or activities that must be performed in some order and when all tasks are completed the desired project objective is achieved. For example, in the construction of a building there are many tasks such as: site preparation; pouring of footing and slab; the construction of walls; and roofing; which must be completed in some order. A fundamental problem in project management is that of determining the minimum project duration time and the schedule, which achieves this. The Critical Path Method (CPM) and Project Evaluation and Review Technique (PERT) are fundamental tools for determining this schedule. In this paper , we concentrate on time constraints that specify when each activity can be in progress. More precisely the Time Constraint Critical Path Problem can be stated as:

> **Given a project network G in which each activity has a specified duration time and a time constraint specifying when the activity can be carried out, find the minimum total project duration time and the schedule which achieves this.**

The time constraints can be specified as: time-windows; time-schedule; or normal time. Chen et al. (1997) considered these and presented a two-phase algorithm for determining the critical path. We develop a new mixed integer linear program (MILP) formulation for this problem. We also consider the time-cost trade-off problem for networks with mixed-time constraints. The method for traditional networks extends to this problem. In addition, we develop a MILP model for determining the minimum cost schedule.

Keywords **:** *Project network; scheduling; critical paths.*

1. INTRODUCTION

Project Management is concerned with projects, such as the construction of a building, the planning and launching of a new product, the installation of a new manufacturer’s facility, the implementation of periodic maintenance in a plant facility, etc. The basic characteristic of a project consists of a well-defined collection of **tasks** (or **activities**) that must be performed in some technological sequence. Within the specified sequence the tasks may be started and stopped independently of each other. When all the tasks are completed the project is completed.

A project can be represented as a network in which the arcs represent activities and the nodes represent events (points where a group of activities are accomplished and where a new set of activities can be initiated). This yields the activity- on-arc model. An alternative network model is the activity-on-node model in which the arcs represent the predecessor restrictions and the nodes represent the activity. In this paper we consider only the activity-on-arc model.

In the early days, the scheduling of a project was done only with limited planning. The tool used for solving this problem was the “Gantt chart”, which specifies the start and finish time for each activity on a horizontal scale. In 1950’s the critical path method (CPM) and the project evaluation and review technique (PERT) were developed. CPM was first developed by E.I. du Pont de Neumours & Company as an application to construction projects and was later extended to a more advanced status by Mauchy Associates; PERT was developed for the U.S. Navy for scheduling the research and development activities for the Polaris missile program. PERT and CPM are two-phase labelling methods that are computationally very efficient.

In project management there is usually a due date for the project completion. Therefore, in some situations, a project could be completed in a shorter time than the normal program. The method of reducing the project duration by shortening the activity time at a cost is called crashing. Another cost associated with projects is the indirect cost. When both the cost components are considered we have the important time-cost trade-off problem. This can be modelled as a mathematical program. By assuming that the direct cost of an activity varies linearly with time the problem can be expressed as a linear program. The solution of this linear program simultaneously determines the optimal duration of a project and the appropriate time of each activity in the network, so that project cost is minimized.

The early work on the time constraint project management problem was carried out by Fulkerson (1961) and Kelley (1961). The time-cost tradeoff problem has more recently been studied by many authors including : Phillips and Dessouky (1977), De et al. (1995), Sunde and Lichtenberg (1995), Demeulemeester et al. (1996), Baker (1997) , and Demeulemeester et al. (1988). Solution methods for this problem include : a minimal cut approach; Dynamic Programming; a heuristic cost-time tradeoff Linear Programming (LP); and the Branch and Bound procedure.

With regard to the Critical path problem, recently Chen and Tang (1997) presented a mixed-time constraint model. These time constraints consist of two types, namely time-window (interval) constraint and time-schedule (list of start times). They presented an efficient linear time algorithm for determining the critical path that is similar to the traditional CPM. In this paper we will develop mixed integer linear programming models for determining the critical path and the minimum cost schedule in mixed-time constrained networks.

2 THE TWO-PHASE METHOD

In this section we present the essential ingredients of the two-phase method developed by Chen et al. (1979) written in the style of the standard algorithm. We adopt the following basic notation.

- s : source node.
- d : destination node.
- A_1 : set of activities or arcs having normal times.
- A_2 : set of activities or arcs having a prescribed schedule of start times (time-schedule).
- A_3 : set of activities or arcs having a prescribed interval of start times (time-window).
- ts_{ij} : schedule departure times for activity (i,j).
- tw_{ij} : time window departure time for activity (i,j) ; usually written in $[L_{ij}, U_{ij}]$.
- t_{ij} : time duration of activity (i,j).
- E_i : earliest occurrence time event i.
- ES_{ij} : earliest start time on activity (i,j).
- EF_{ij} : earliest finish time on activity (i,j).

Phase I : Forward-pass Procedure
(Calculates the early start and finish times)

Step1 Set $E_i = 0$.

Step2 For each arc (i,j) directed into node j:

(i) If (i,j) is a normal-time arc, then $ES_{ij} = E_i$ and $EF_{ij} = ES_{ij} + t_{ij}$.

(ii) If (i,j) is a time-schedule arc with times $ts_1, ts_2, \dots, ts_k, \dots, ts_v$

then

$ES_{ij} = ts_k\,(i,j)$; where $ts_{k-1} < E_i \le ts_k$.

and $EF_{ij} = ES_{ij} + t_{ij}$.

(iii) If (i,j) is time-window arc with time $[L_{ij}, U_{ij}]$, then

$$ES_{ij} = \begin{cases} L_{ij}, & \text{if } E_i \le L_{ij} \\ E_i, & \text{if } L_{ij} \le E_i \le U_{ij} \\ \infty, & \text{if } E_i > U_{ij}. \end{cases}$$

and $EF_{ij} = ES_{ij} + t_{ij}$.

Step3 For node j, if EF_{ij} has been calculated for all i, then $E_j = \max_i \{EF_{ij}\}$.

Step4 (Stopping rule)
If i = d stop, $EF_{ij} = E_d$. Otherwise go to step 2. The longest path is found.

Phase II : Backward-Pass Procedure
(Calculates the late start and finish times)

Step 1 Set $LF_j = ES_j$.

Step 2 For each arc(i,j) directed into node j :

(i) If (i, j) is normal time arc with times $ts_1, ts_2, \dots, ts_k, \dots, ts_v$, then $LS_{ij} = LF_{ij} - t_{ij}$.

(ii) If (i , j) is time-schedule arc, then

$$LS_{ij} = \begin{cases} ts_k, & LF_{ij} \ge ts_k + t_{ij} \\ \infty, & \text{otherwise.} \end{cases}$$

(iii)If (i , j) is time- window arc $[L_{ij}, U_{ij}]$, then

$$LS_{ij} = \begin{cases} U_{ij} & , \quad LF_{ij} \geq U_{ij} + t_{ij} \\ LF_{ij} & , \quad L_{ij} \leq LF_{ij} \leq U_{ij} \\ \infty & , \quad otherwise. \end{cases}$$

Step 3 For node i, if LS_{ij} has been calculated for all j, then

$$L_i = \min_j \{LS_{ij}\}.$$

Step 4 Stopping rule:
If j = s stop , $LF_{ij} = L_S$, otherwise go to Step 2.

3 MILP FORMULATIONS

In this section, the critical path problem with mixed-time constraints will be formulated as a MILP (Mixed Integer Linear Program). We start with a new MILP for determining the earliest project completion time and follow this with an MILP to determine the latest allowable occurrence time.

Determining the earliest project completion time

We begin with some further notation. Let:

$D_{ij} = \{ d^1_{ij}, d^2_{ij}, \ldots, d^k_{ij} \}$ for $(i,j) \in A_2$.

[Start time of activity (i,j)]

$$y^k_{ij} = \begin{cases} 1 & , \text{ if activity (i, j) starts at time } d^k_{ij} \\ 0 & , \text{ else.} \end{cases}$$

k : occurrence times on time-schedule ;
k = 1,2,…,K (last number of time-schedule).
x_{ij} : earliest start time of activity (i,j).

Formulation:

Minimize E_d (1)
subject to

$$E_S = 0 \quad (2)$$

[The above restriction ensures a start time is zero]

$$E_j - E_i \geq t_{ij} \quad , \quad \text{for } (i,j) \in A_1. \quad (3)$$

[The above restriction ensures that normal time constraints are satisfied]

$$E_j - x_{ij} \geq t_{ij} \quad , \quad \text{for } (i,j) \in A_2. \quad (4)$$

$$x_{ij} - \sum_k d^k_{ij} y^k_{ij} = 0 \quad , \quad \text{for (i, j) } \in A_2 . (5)$$

$$\sum_k y^k_{ij} = 1 \quad , \quad \text{for (i, j) } \in A_2 . \quad (6)$$

$y^k_{i,j}$ is binary $\forall$ i,j,k. (7)

[The restrictions (4) – (7) ensure that the time-schedule constraints are satisfied]

For $(i,j) \in A_3$

$$E_j - x_{ij} \geq t_{ij} \quad , \text{for } (i,j) \in A_3. \quad (8)$$

$$L_{ij} \leq x_{ij} \leq U_{ij} \quad , \text{for } (i,j) \in A_3. \quad (9)$$

[The two above restrictions ensure that the time-window constraints are satisfied]

Determining the latest allowable occurrence time

We have the following additional notation

T : Project Completion date (due date).

LS_{ij} : Latest start time of activity (i,j).

LSS_{ij}: Latest start time of activity $(i,j) \in A_2 \cup A_3$;

$LSS_{ij} = LF_{ij} - t_{ij}$; and $LF_{ij} \geq ts_k - t_{ij}$.

N^+_i = Neighbor set of i (with arcs directed out of i).

Formulation:

Maximize L_s (10)
subject to

$$L_d = T \quad (11)$$

[The above restrictions ensures a completion time of T]

$$L_j - L_i \geq t_{ij} \quad , \quad \text{for } (i,j) \in A_1. \quad (12)$$

[The above restrictions ensure that the normal time constraints are satisfied]

For each i and $j \in N^+_i$; $(i,j) \in A$,

$$L_i \leq LS_{ij} , \quad (13)$$

and $$LS_{ij} - L_i + m_{ij} z_{ij} \leq m_{ij}, \quad (14)$$

[m_{ij} is large positive integer]

$$\sum_j z_{ij} = 1 , \quad (15)$$

z_{ij} is binary. (16)

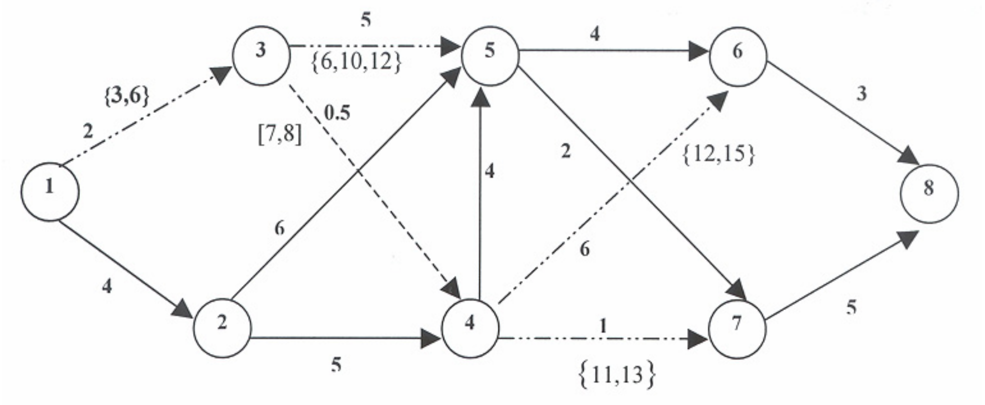

Figure 1 : Start and Finish Times.

[The restrictions (13)-(16) ensure that the event j gives the smallest latest start(i,j)]

For $(i,j) \in A_1$, $LS_{ij} = L_j - t_{ij}$. (17)

[The above restrictions ensure that the normal time constraints are satisfied]

For $(i,j) \in A_2$,

$$LSS_{ij} = L_j - t_{ij}, \quad (18)$$

$$LS_{ij} \le LSS_{ij}, \quad (19)$$

$$LS_{ij} - \sum_k d_{ij}^k y_{ij}^k = 0, \quad (20)$$

$$\sum_k y_{ij}^k = 1, \quad (21)$$

y^k_{ij} is binary for all i,j,k. (22)

[The restrictions (18)-(22) ensure that time schedule constraints are satisfied]

For $(i,j) \in A_3$,

$$LSS_{ij} = L_j - t_{ij}, \quad (23)$$

$$LS_{ij} \le LSS_{ij}, \quad (24)$$

$$L_{ij} \le LS_{ij} \le U_{ij}. \quad (25)$$

[The restrictions (23)-(25) ensure that time window constraints are satisfied]

4 EXAMPLE

Consider the 8-node network displayed in Figure 1 with

$A_1 = \{(1,2),(2,4),(2,5),(4,5),(5,6),(6,3),(6,8),(7,8)\}$

$A_2 = \{(1,3),(3,5),(4,6),(4,7)\}$ and $A_3 = \{(3,4),\}$.

Applying the two-phase method or using the MILP's described in Section 3 yields :

Activity	*t(i,j)*	E_i	$ES_{i,j}$	$EF_{i,j}$	LF_{ij}	$LS_{i,j}$
(1-2)	4	0	0	4	5	1
(1-3)	2	0	3	5	6	3
(2-4)	5	4	4	9	10	5
(2-5)	6	4	4	10	14	8
(3-4)	0.5	5	7	7.5	10	8
(3-5)	5	5	6	11	14	6
(4-5)	4	10	10	14	14	10
(4-6)	6	10	12	18	18	12
(4-7)	1	10	11	12	16	13
(5-6)	4	13	13	17	18	14
(5-7)	2	13	13	15	16	14
(6-8)	3	18	18	21	21	18
(7-8)	5	15	15	20	21	16

Table 1 : Start and Finish Times

The critical path of length 21 is : {(1,2), (2,4), (4,6), (6,8)}.

5 TIME COST TRADE-OFF

The modified two-phase method of Section 2 can be used in the traditional way to resolve the time-cost trade-off problem for networks with mixed-time constraints. That is

Step 1: Generate a preliminary schedule using normal resources (a modified two-phase method with mix-time constraints).

Step 2: Find the job along the critical path with the least cost slope. This is the job that can be crashed with least expense. If the cost of shortening the schedule by one period is less than the fixed indirect cost for one period, then the job is expedited up to the point where no further shortening is possible (either because the job duration cannot be reduced further or because some other job has become critical along a parallel path).

Step 3: Repeat Step 2 until no further shortening of critical jobs is uneconomical. (i.e. reduce the savings that would result).

We can also use the following MILP formulation.

Consider a project network with n nodes labelled 1,2, . . . , n, where node 1 represents the start of the project and node n the end of it. In addition to the earlier notation we let

x_{ij}: duration of activity (i,j).

E_i: realization of event i.

a_{ij} : cost slope of activity(i,j).

l_{ij}: Lower bound on the duration of activity (i,j).

u_{ij}: Upper bound on the duration of activity(i,j).

$D_{ij}= \{d_{ij}^1, d_{ij}^2, ..., d_{ij}^k\}$ for $(i,j) \in A_2$.

$TW_{ij} = [L_{ij}, U_{ij}]$, for $(i,j) \in A_3$.

$$y_{ij} = \begin{cases} 1 & , \text{ if activity(i, j) start to leave node i to node j} \\ 0 & , \text{ else} \end{cases}$$

f : fixed cost (per unit time).

We assume a linear cost-duration for each activity. So we can write the cost of activity(i,j) as

$$c_{ij} = b_{ij} + a_{ij}x_{ij}.$$

The MILP formulation is:

$$\text{Minimize } [\sum a_{ij}x_{ij} + f(E_n) + \sum b_{ij}] \quad (26)$$

subject to

$$E_1 = 0. \quad (27)$$

$$E_n - E_1 \leq T. \quad (28)$$

For $(i,j) \in A_1$ (normal time arcs).

$$E_j = ES_{ij} + x_{ij}. \quad (29)$$

$$l_{ij} \leq x_{ij} \leq u_{ij}. \quad (30)$$

$$E_j - ES_{ij} \geq 0. \quad (31)$$

[The above two restrictions ensure that normal time constraints are satisfied].

For $(i,j) \in A_2$ (time schedule arcs) :

$$E_j = ES_{ij}+x_{ij}. \quad (32)$$

$$l_{ij} \leq x_{ij} \leq u_{ij}. \quad (33)$$

$$E_j - ES_{ij} \geq 0. \quad (34)$$

$$ES_{ij} - \sum_k d_{ij}^k y_{ij}^k = 0. \quad (35)$$

$$\sum_k y_{ij}^k = 1. \quad (36)$$

$$y_{ij} \text{ is binary.} \quad (37)$$

[the restrictions (32) – (37) ensure that the time-schedule constraints are satisfied]

For $(i,j) \in A_3$ (time windows arcs).

$$E_j - ES_{ij} \geq x_{ij}. \quad (38)$$

$$l_{ij} \leq x_{ij} \leq u_{ij}. \quad (39)$$

$$L_{ij} \leq ES_{ij} \leq U_{ij}. \quad (40)$$

[the two above restrictions ensure that the time window constraints are satisfied].

The above MILP is easily solved by a package such as CPLEX.

Example : Consider the project network of Figure 1 with normal and crash cost data :

Activity (i,j)	Normal		Crash		Cost Slope
	Time t_{ij}	Cost c_{ij}	Time t_{ij}	Cost c_{ij}	
(1,2)	4	300	2	400	50
(1,3)	2	200	1	250	50
(2,4)	5	400	3	450	25
(2,5)	6	350	4	420	35
(3,4)	0.5	500	0.5	500	-
(3,5)	5	450	3	520	35
(4,5)	4	480	2	540	30
(4,6)	6	300	4	350	25
(4,7)	1	280	1	280	-
(5,6)	4	250	3	300	50
(5,7)	2	150	1	230	80
(6,8)	3	200	2	280	80
(7,8)	5	400	3	460	30
Total		4260		4980	

Table 2 : Normal and Crash Cost Data

Suppose the indirect cost associated with the project is \$100 per day. Then the application of the above method yields :

Total time	Direction Cost	Indirect Cost	Total Cost	Crash activities
21	4260	2100	6360	(4-6)
20	4285	2000	6285	(4-6)
19	4335	1900	6235*	(4-6),(2-4)
18	4445	1800	6245	(4-5),(6-8)

Table 3

The optimal solution is for a 19-day schedule.

6 CONCLUSION

This paper addresses project management problems in networks with mixed-time constraints. We allow activities to be restricted to time-window and time-schedule constraints. We present mixed integer linear programming models that can be efficiently solved by available commercial software such as CPLEX.

7 REFERENCES

Baker, B.M. 1997, 'Cost/time trade-off analysis for the critical path method: a deviation of the network flow approach', Journal of the Operational Research Society, vol. 48, pp1241-1244.

Chen, Y-L., Rinks, D. and Tang, K. 1997, 'Critical path in an activity network with time constraints', European Journal of Operational Research 100, pp122-133.

De, M., Dunne, E.J.,Ghosh, J.B. and Wells,C.E. 1995, 'The discrete time-cost tradeoff problem revisited', European Journal of Operational Research 81, pp225-238.

Demeulemeester, E.L., Herroelen, W.S. and Elmaghraby, S.E. 1996, 'Optimal procedures for the discrete time/cost tradeoff problem in project network', European Journal of Operational Research 88, pp50-68.

Demeulemeester, E., Reyck, B.D., Foubert,B., Herroelen,W. and Vanhoucke, M.1998, 'New computational results on the discrete time/cost trade-off problem in project networks', Journal of the Operational Research Society, vol. 49, pp1153-1163.

Fulkerson, D. 1961, 'A network flow computation for project cost curves', Management Science 7, pp. 167-178.

ILOG, Inc. CPLEX Division, 1997, *Using the CPLEX® Callable Library*,Version5.5.

Kelley, J. 1961, 'Critical-path planning and scheduling: Mathematical basis', Operations Research, vol. 9, no. 3, pp296-320.

Phillips, J.P. and Dessouky, M.I. 1977, 'Solving the project time/cost tradeoff problem using the minimal cut concept', Management Science, vol. 13, no. 6, ppB359-B377.

Sunde, L. and Lichtenberg, S. 1995, 'Net-present value cost/time tradeoff', International Journal of Project Management, vol. 13, no. 1, pp45-49.