TECHNICAL PAPER

JOURNAL OF THE SOUTH AFRICAN INSTITUTION OF CIVIL ENGINEERING

Vol 55 No 2, August 2013, Pages 12–20, Paper 886

A procedure for critical path method-based scheduling in linear construction projects

Ö Ökmen

DR ÖNDER ÖKMEN is an engineer at the General Directorate of State Hydraulic Works in Turkey. He obtained his BSc degree in Civil Engineering from the Middle East Technical University, and his Masters and PhD degrees in Construction Management (Civil Engineering) from the Gaziantep University in Turkey. He did post-doctoral research at the Epoka University in Albania. Currently he is busy with irrigation and drainage projects. His research interests include project management, risk analysis, scheduling, cost estimation, irrigation channels and pipe networks.

Contact details:
General Directorate of State Hydraulic Works
Devlet Su İşleri Genel Müdürlüğü
Proje ve İnşaat Dairesi Başkanlığı
Devlet Mah
İnönü Bulvarı
No: 16, 06100 Çankaya
Ankara
Turkey
T: +90 532 288 13 22
F: +90 312 418 33 95
E: onderokmen@hotmail.com
onderok@dsi.gov.tr

Keywords: construction management, critical path method, scheduling, network analysis, linear scheduling

Construction projects having identical or similar units, in which activities repeat from one unit to another, require schedules that ensure the continuous usage of resources from one unit to the next and the maintenance of the network logic between activities at the same time. This paper presents a procedure for critical path method-based (CPM) scheduling in linear construction projects that have repeating activities. The procedure concurrently provides the utilisation of resources without interruption and preservation of network relationships between successive units. With this procedure it is possible to represent activities by variable production rates through consecutive units and assign any kind of relationship type with lag time. The proposed procedure determines the critical units in two categories: logical and resource critical units. A spreadsheet has been developed on a table processor to computerise the procedure. Details of the procedure are described and an example application is presented.

INTRODUCTION

Construction planners widely use Bar Charts (or Gantt Charts), Line of Balance (Linear or Repetitive Scheduling) and Critical Path Method (CPM) to schedule the construction activities. CPM, which was first developed in the USA, shows the dependency relations between activities, detects the critical activities, reveals the activity float times and computes the minimum project completion time in accordance with network relations. For this reason CPM is the most convenient means of scheduling, analysing and controlling activity networks (Griffis & Farr 2000; Halphin & Woodhead 1998; Oberlender 2000).

Multi-storey building construction, highway construction and pipeline construction projects are some examples of linear or repetitive construction projects. Such projects require the implementation of a set of identical or similar activities from one unit to another as in the case of multi-storey building construction, or from one location to another as in the case of highway construction. Scheduling of linear construction projects requires uninterrupted usage of resources between similar units and enabling timely movement of crews from one unit to the next (El-Rayes & Moselhi 1998). Maintenance of work continuity leads to advantages such as maximum learning for each crew member, minimum idle time and minimum off-on movement of crew members on a project once work has begun (Ashley 1980).

Bar Chart and CPM scheduling have received criticism in the literature for their inability to maintain work continuity in linear projects (Selinger 1980; Reda 1990; Russell & Wong 1993; Hegazy *et al* 1993). Furthermore, such scheduling techniques assume the unlimited availability of resources in the initial development of a schedule, necessitating later adjustment according to the resource limitations (El-Rayes & Moselhi 1998).

The most convenient method of scheduling and controlling projects that have repetitive units is the Line of Balance or Linear Scheduling method. Unlike Bar Chart and CPM, Linear Scheduling provides work continuity between the same activities of successive units in accordance with the resource availability. It leads to the smooth and efficient flow of resources, and requires less preparation time. This is one of the main advantages of Linear Scheduling compared to CPM (Arditi & Albulak 1986).

Linear Scheduling originated in the early 1940s, and was further developed by the US Navy in 1952 and in the 1960s (Yang & Ioannou 2004). Manufacturing and production control were the areas of its application, with the objective of evaluating the production rate of finished products in a production line (Johnston 1981). The idea of repetitive scheduling originated with the use of mass production lines where the manufacturing process consists of series of workstations requiring the same resources (Huang & Sun 2005). Although it was developed mainly for manufacturing projects, Lumsden (1968) applied Linear Scheduling to construction projects. The National Building Agency of the UK used it for repetitive housing units for the first time (Yang & Ioannou 2004). Variations of Linear Scheduling in the literature include Line of Balance (Lumsden 1968), Vertical Production Method (O'Brien 1975), Linear Scheduling Method (Johnston 1981), Time Space Scheduling (Stradal & Cacha 1982), and Time Changes Charts (Mawdesley *et al* 1989).

The most important feature of CPM is its ability to specify critical path(s). Critical path(s) identify the activities which cause delay in timely project completion. For Linear Scheduling to be accepted as a valuable tool, it should also be able to determine the critical activities (Ammar & Elbeltagi 2001). Scheduling techniques applicable to linear projects must be able to provide a synonymous set of critical activities as those calculated by CPM. This ability would provide an analytical or engineering foundation on which a full range of functionality, such as float identification, resource and cost allocation, and schedule updating could be built (Harmelink & Rowings 1998). Determining the critical path in CPM, or controlling the activity path in a linear schedule, is crucial. It helps in controlling and updating the original schedule. Resource levelling of a linear schedule requires the critical segments as input (Mattila & Abraham 1998). Harmelik & Rowings (1998) introduced a method – the Linear Scheduling Model – that identifies the controlling activity path through a linear schedule based on the time and distance relationships of activities. The controlling activity path is similar to the critical path of CPM. Harris & Ioannou (1998) used a similar approach – the Repetitive Scheduling Method – to identify controlling activities of a linear schedule. However, these methods are mainly graphic-based techniques. Ammar & Elbeltagi (2001) introduced an algorithm for determining the controlling path considering resource continuity. In this algorithm, the production rate of each activity is compared with that of its successors in order to specify the start-to-start or finish-to-finish relationships between two consecutive units. However, Ammar & Elbeltagi's (2001) method assumes constant production rates and only finish-to-start relationship types between the activities within a unit. This paper presents a different procedure to apply the CPM to linear construction projects that have repeating activities. The procedure concurrently provides the utilisation of resources without interruption and the maintenance of network logic through successive units. With this procedure, it is possible to represent the activities by variable production rates from one unit to another and assign any kind of relationship type with lag time. The proposed procedure computes the project completion time, the early start and finish times, the late start and finish times, and the float times, and determines the logic and resource critical activities through a unit by unit approach. The paper contains the detailed description of the procedure, example application, discussion of advantages and limitations, and some recommendations for future work. Furthermore, a spreadsheet has been developed by using a table processor in order to computerise the procedure.

CPM-BASED SCHEDULING PROCEDURE FOR LINEAR CONSTRUCTION PROJECTS

The main aim of executing CPM is to find out the critical activities and to compute the shortest duration of project completion given the logic and resource availability constraints. In linear scheduling, however, not only logic and resource availability but also the continuity of resource usage should be considered. For this reason, the proposed CPM-based Linear Scheduling procedure executes CPM's forward and backward pass calculations through units with maintaining network logic and resource continuity.

The procedure allows the scheduler to represent the repeating activities with variable production rates along successive units. If $ud_{i,n}$ and $r_{i,n}$ denote unit duration and production rate of activity i for a particular unit n, respectively, then

$$r_{i,n} = 1/\,ud_{i,n} \tag{1}$$

The procedure requires the following data:

- Precedence relations between activities
- Activity durations along units
- The number of successive units

The following sections describe the procedure on a fictitious activity network containing activities denoted by $Act_{i = 1, 2 \dots i \dots k}$

Step 1 – Forward pass calculations

1st Activity (Act_1)

Set 0 to the Early Start of Act_1 for the 1st unit ($ES_{1,1}$). Then add the unit durations ($ud_{1,1}$, $ud_{1,2} \dots ud_{1,n}$) to 0 to find the Early Finish for the 1st unit ($EF_{1,1}$) and Early Start/Early Finish values for the succeeding units ($ES_{1,2} \dots ES_{1,n}$; $EF_{1,2} \dots EF_{1,n}$). This approach will ensure that Act_1 will be performed uninterruptedly along units while maintaining the resource continuity.

	Unit 1	Unit 2	...	Unit n
Act_1 :	$(ES_{1,1}, EF_{1,1})$	$(ES_{1,2}, EF_{1,2})$	...	$(ES_{1,n}, EF_{1,n})$
	$(0,\ 0 + ud_{1,1})$	$(ud_{1,1},\ ud_{1,1} + ud_{1,2})$	...	$(ud_{1,1} + ud_{1,2} + \dots + ud_{1,n-1},\ ud_{1,1} + ud_{1,2} + \dots + ud_{1,n-1} + ud_{1,n})$ (2)

2nd Activity (Act_2)

Act_2 is the successor of Act_1.

- **Stage – 1**
  - If the relationship between Act_1 and Act_2 is Finish-to-Start (FS) and there is a time lag between ($Lg_{1,2}$), then set Early Start of Act_2 for the 1st unit ($ES_{2,1}$) equal to the Early Finish of Act_1 for the 1st unit ($EF_{1,1}$) plus the time lag. Then add the unit durations ($ud_{2,1}$, $ud_{2,2} \dots ud_{2,n}$) to $ES_{2,1}$ to find the Early Start/Early Finish values for the succeeding units ($ES_{2,2} \dots ES_{2,n}$; $EF_{2,1} \dots EF_{2,n}$). This approach will ensure that Act_2 is performed uninterruptedly along units while maintaining the resource continuity.

	Unit 1	Unit 2	...	Unit n
Act_2 :	$(ES_{2,1}, EF_{2,1})$	$(ES_{2,2}, EF_{2,2})$	...	$(ES_{2,n}, EF_{2,n})$
	$(EF_{1,1} + Lg_{1,2},\ EF_{1,1} + Lg_{1,2} + ud_{2,1})$	$(EF_{1,1} + Lg_{1,2} + ud_{2,1},\ EF_{1,1} + Lg_{1,2} + ud_{2,1} + ud_{2,2})$	...	$(EF_{1,1} + Lg_{1,2} + ud_{2,1} + \dots + ud_{2,n-1},\ EF_{1,1} + Lg_{1,2} + ud_{2,1} + \dots + ud_{2,n-1} + ud_{2,n})$ (3)

  - If the relationship between Act_1 and Act_2 is Start-to-Start (SS) and there is a time lag time between ($Lg_{1,2}$), then set Early Start of Act_2 for the 1st unit ($ES_{2,1}$) equal to the Early Start of Act_1 for the 1st unit ($ES_{1,1}$) plus the time lag. Then add the unit durations ($ud_{2,1}$, $ud_{2,2} \dots ud_{2,n}$) to $ES_{2,1}$ to find the Early Start/Early Finish values for the succeeding units ($ES_{2,2} \dots ES_{2,n}$; $EF_{2,1} \dots EF_{2,n}$).

	Unit 1	Unit 2	...	Unit n
Act_2 :	$(ES_{2,1}, EF_{2,1})$	$(ES_{2,2}, EF_{2,2})$	...	$(ES_{2,n}, EF_{2,n})$
	$(ES_{1,1} + Lg_{1,2},\ ES_{1,1} + Lg_{1,2} + ud_{2,1})$	$(ES_{1,1} + Lg_{1,2} + ud_{2,1},\ ES_{1,1} + Lg_{1,2} + ud_{2,1} + ud_{2,2})$	...	$(ES_{1,1} + Lg_{1,2} + ud_{2,1} + \dots + ud_{2,n-1},\ ES_{1,1} + Lg_{1,2} + ud_{2,1} + \dots + ud_{2,n-1} + ud_{2,n})$ (4)

  - If the relationship between Act_1 and Act_2 is Finish-to-Finish (FF) and there is a time lag between ($Lg_{1,2}$), then set Early

Finish of Act_2 for the 1st unit ($EF_{2,1}$) equal to the Early Finish of Act_1 for the 1st unit ($EF_{1,1}$) plus the time lag. Then apply the unit durations ($ud_{2,1}$, $ud_{2,2}$ … $ud_{2,n}$) to $EF_{2,1}$ to find the Early Start/Early Finish values for the succeeding units ($ES_{2,1}$ … $ES_{2,n}$; $EF_{2,2}$ ... $EF_{2,n}$).

	Unit 1	Unit 2	…	Unit n	
Act_2 :	$(ES_{2,1}, EF_{2,1})$	$(ES_{2,2}, EF_{2,2})$	…	$(ES_{2,n}, EF_{2,n})$	
	$(EF_{1,1} + Lg_{1,2} - ud_{2,1}, EF_{1,1} + Lg_{1,2})$	$(EF_{1,1} + Lg_{1,2}, EF_{1,1} + Lg_{1,2} + ud_{2,2})$	…	$(EF_{1,1} + Lg_{1,2} + ud_{2,2} + ... + ud_{2,n-1}, EF_{1,1} + Lg_{1,2} + ud_{2,2} + ... + ud_{2,n-1} + ud_{2,n})$	(5)

- If the relationship between Act_1 and Act_2 is Start-to-Finish (SF) and there is a time lag between ($Lg_{1,2}$), then set Early Finish of Act_2 for the 1st unit ($EF_{2,1}$) equal to the Early Start of Act_1 for the 1st unit ($ES_{1,1}$) plus the time lag. Then apply the unit durations ($ud_{2,1}$, $ud_{2,2}$ … $ud_{2,n}$) to $EF_{1,1}$ to find the Early Start/Early Finish values for the succeeding units ($ES_{2,1}$ … $ES_{2,n}$; $EF_{2,2}$ ... $EF_{2,n}$).

	Unit 1	Unit 2	…	Unit n	
Act_2 :	$(ES_{2,1}, EF_{2,1})$	$(ES_{2,2}, EF_{2,2})$	…	$(ES_{2,n}, EF_{2,n})$	
	$(ES_{1,1} + Lg_{1,2} - ud_{2,1}, ES_{1,1} + Lg_{1,2})$	$(ES_{1,1} + Lg_{1,2}, ES_{1,1} + Lg_{1,2} + ud_{2,2})$	…	$(ES_{1,1} + Lg_{1,2} + ud_{2,2} + ... + ud_{2,n-1}, ES_{1,1} + Lg_{1,2} + ud_{2,2} + ... + ud_{2,n-1} + ud_{2,n})$	(6)

- **Stage – 2**

  Then, check the logic. The rules are:

$$For\ FS\ case,\ ES_{2,n} \geq EF_{1,n} + Lg_{1,2}\ ;\ for\ n{:}1,2...\ n \quad (7)$$

$$For\ SS\ case,\ ES_{2,n} \geq ES_{1,n} + Lg_{1,2}\ ;\ for\ n{:}1,2...\ n \quad (8)$$

$$For\ FF\ case,\ EF_{2,n} \geq EF_{1,n} + Lg_{1,2}\ ;\ for\ n{:}1,2...\ n \quad (9)$$

$$For\ SF\ case,\ EF_{2,n} \geq ES_{1,n} + Lg_{1,2}\ ;\ for\ n{:}1,2...\ n \quad (10)$$

If the corresponding rule is not satisfied:

$$For\ FS\ case,\ ES^*_{2,n} = ES_{2,n} + MAX\ (EF_{1,1} + Lg_{1,2} - ES_{2,1}\ , \ldots , EF_{1,n} + Lg_{1,2} - ES_{2,n}) \quad (11)$$

$$For\ SS\ case,\ ES^*_{2,n} = ES_{2,n} + MAX\ (ES_{1,1} + Lg_{1,2} - ES_{2,1}\ , \ldots , ES_{1,n} + Lg_{1,2} - ES_{2,n}) \quad (12)$$

$$For\ FF\ case,\ EF^*_{2,n} = EF_{2,n} + MAX\ (EF_{1,1} + Lg_{1,2} - EF_{2,1}\ , \ldots , EF_{1,n} + Lg_{1,2} - EF_{2,n}) \quad (13)$$

$$For\ SF\ case,\ EF^*_{2,n} = EF_{2,n} + MAX\ (EF_{1,1} + Lg_{1,2} - ES_{2,1}\ , \ldots , EF_{1,n} + Lg_{1,2} - ES_{2,n}) \quad (14)$$

where n: 1,2, … n and * denotes the new correct value.

For example, let us take Equation (3) from the Stage-1 of forward pass calculations of Act_2 and Equation (7) from the Stage-2 of forward pass calculations, which is the corresponding rule for the FS case. Furthermore, let us assume that:

- Four identical units will be constructed.
- One day of time lag exists between Act_1 and Act_2.
- Unit durations of Act_1 are 5, 5, 4, and 4 days for the units 1, 2, 3, and 4, respectively.
- Unit durations of Act_2 are 3, 3, 2, and two days for the units 1, 2, 3 and 4, respectively.
- Act_1 has the following ES and EF times for the consecutive units:

Act_1: (0, 5), (5, 10), (10, 14), (14, 18)
↓ ↓ ↓ ↓
5 days 5 days 4 days 4 days (unit durations of Act_1)

In accordance with the assumptions given above, Equation (3) is applied for Act_2:

Act_2 : (5+1, 5+1+3), (5+1+3, 5+1+3+3), (5+1+3+3, 5+1+3+3+2), (5+1+3+3+2, 5+1+3+3+2+2)
: (6, 9), (9, 12), (12, 14), (14, 16)
: ↓ ↓ ↓ ↓
: 3 days 3 days 2 days 2 days (unit durations of Act_2)

Then Equation (7) is applied to check the rule for the FS case:

- for the first unit, 6 ≥ 5 + 1 OK √
- for the second unit, 9 < 10 + 1, then apply Equation (11), which is the corresponding equation for FS.

  $ES^*_{2,1} = ES_{2,1} + MAX\ (EF_{1,1} + Lg_{1,2} - ES_{2,1}\ , \ldots , EF_{1,4} + Lg_{1,2} - ES_{2,4})$
  $ES^*_{2,1} = 6 + MAX\ (5 + 1 - 6, 10 + 1 - 9, 14 + 1 - 12, 18 + 1 - 14)$
  $ES^*_{2,1} = 6 + 5$
  $ES^*_{2,1} = 11$

  Similarly:

  $ES^*_{2,2} = 9 + 5 = 14$
  $ES^*_{2,3} = 12 + 5 = 17$
  $ES^*_{2,3} = 14 + 5 = 19$

  Finally, the unit-based early start and early finish times of Act_2 are:

  Act_2: (11, 14), (14, 17), (17, 19), (19, 21)

i^{th} Activity (Act_i)

An arbitrary activity in the network, Act_i, may be dependent on more than one predecessor activity (p_1, p_2 … p_m) with different relationships among FS, SS, FF and SF. In this general case, execute the required methods among the FS, SS, FF and SF cases presented through Equations (3) to (6) to obtain Early Start values (ES_1, ES_2 … ES_m), as many as the predecessors (m) for the activity in question (Act_i). Then, in accordance with the CPM's forward pass calculation logic, apply the maximisation process on ES_1, ES_2 … ES_m to find the true ES value ($ES_{i,1}$) for the first unit of Act_i.

$$ES_{i,1} = Max\ (ES_1, ES_2 \ldots ES_m) \quad (15)$$

Then, find ES and EF values of the other units of Act_i ($ES_{i,2}$ … $ES_{i,n}$; $EF_{i,1}$ … $EF_{i,n}$). Subsequently, check the logic. The rules are:

$$For\ FS\ case,\ ES_{i,n} \geq EF_{p,n} + Lg_{p,i} \quad (16)$$

$$For\ SS\ case,\ ES_{i,n} \geq ES_{p,n} + Lg_{p,i} \quad (17)$$

$$For\ FF\ case,\ ES_{i,n} + ud_{i,n} \geq EF_{p,n} + Lg_{p,i} \quad (18)$$

$$For\ SF\ case,\ ES_{i,n} + ud_{i,n} \geq ES_{p,n} + Lg_{p,i} \quad (19)$$

If the corresponding rule is not satisfied:

$$For\ FS\ case,\ ES^*_{i,n} = ES_{i,n} + MAX\ (EF_{p,1} + Lg_{p,i} - ES_{i,1}\ , \ldots , EF_{p,n} + Lg_{p,i} - ES_{i,n}) \quad (20)$$

For SS case, $ES^{*}_{i,n} = ES_{i,n} + MAX\,(ES_{p,1} + Lg_{p,i} - ES_{i,1}, \ldots, ES_{p,n} + Lg_{p,i} - ES_{i,n})$ (21)

For FF case, $EF^{*}_{i,n} = ES_{i,n} + ud_{i,n} + MAX[EF_{p,1} + Lg_{p,i} - (ES_{i,1} + ud_{i,1}), \ldots, EF_{p,n} + Lg_{p,i} - (ES_{i,n} + ud_{i,n})]$ (22)

For SF case, $EF^{*}_{i,n} = ES_{i,n} + ud_{i,n} + MAX[ES_{p,1} + Lg_{p,i} - (ES_{i,1} + ud_{i,1}), \ldots, ES_{p,n} + Lg_{p,i} - (ES_{i,n} + ud_{i,n})]$ (23)

where n: 1,2, … n; p: p_1, p_2 … p_m and * denotes the new correct value.

Step 2 – Backward pass calculations

Last activity (Act_k)

Set the Late Finish of the last (k^{th}) Activity for the n^{th} unit ($LF_{k,n}$) equal to its own Early Finish ($EF_{k,n}$). Then subtract the unit durations ($ud_{k,1}$, $ud_{k,2}$ … $ud_{k,n}$) from $LF_{k,n}$ to find the Late Start for the n^{th} unit ($LS_{k,n}$) and Early Start/Early Finish values for the predecessor units ($LS_{k,1}$ … $LS_{k,n-1}$; $LF_{k,1}$ ... $LF_{k,n-1}$). This approach will ensure that Act_k is performed uninterruptedly along units while maintaining the resource continuity.

$$\begin{array}{lllll} & \text{Unit 1} & \ldots & \text{Unit } n-1 & \text{Unit } n \\ Act_k: & (LS_{k,1}, LF_{k,1}) & \ldots & (LS_{k,n-1}, LF_{k,n-1}) & (LS_{k,n}, LF_{k,n}) \\ & (EF_{k,n} - ud_{k,n} - ud_{k,n-1} - \ldots - ud_{k,2} - ud_{k,1}, EF_{k,n} - ud_{k,n} - ud_{k,n-1} - \ldots - ud_{k,2}) & \ldots & (EF_{k,n} - ud_{k,n} - ud_{k,n-1}, EF_{k,n} - ud_{k,n}) & (EF_{k,n} - ud_{k,n}, = EF_{k,n}) \end{array} \quad (24)$$

k-1th Activity (Act_{k-1})

Act_{k-1} is the predecessor of Act_k.

- **Stage – 1**
  - If the relationship is FS and there is a time lag between ($Lg_{k,k-1}$), then set the Late Finish of Act_{k-1} for the n^{th} unit ($LF_{k-1,n}$) equal to the Late Start of the Act_k for the n^{th} unit ($LS_{k,n}$) minus the time lag. Then, apply the unit durations ($ud_{k-1,1}$, $ud_{k-1,2}$ … $ud_{k-1,n}$) to $LF_{k-1,n}$ to find the Early Start/Early Finish values for the preceding units ($LS_{k-1,1}$ … $LS_{k-1,n}$; $LF_{k-1,1}$ ... $LF_{k-1,n-1}$). This will provide resource continuity for Act_{k-1}.

$$\begin{array}{lllll} & \text{Unit 1} & \ldots & \text{Unit } n-1 & \text{Unit } n \\ Act_{k-1}: & (LS_{k-1,1}, LF_{k-1,1}) & \ldots & (LS_{k-1,n-1}, LF_{k-1,n-1}) & (LS_{k-1,n}, LF_{k-1,n}) \\ & (EF_{k,n} - ud_{k,n} - Lg_{k,k-1} - ud_{k-1,n} - ud_{k-1,n-1} - \ldots - ud_{k-1,2} - ud_{k-1,1}, EF_{k,n} - ud_{k,n} - Lg_{k,k-1} - ud_{k-1,n} - ud_{k-1,n-1} - \ldots - ud_{k-1,2}) & \ldots & (EF_{k,n} - ud_{k,n} - Lg_{k,k-1} - ud_{k-1,n} - ud_{k-1,n-1}, EF_{k,n} - ud_{k,n} - Lg_{k,k-1} - ud_{k-1,n}) & (EF_{k,n} - ud_{k,n} - Lg_{k,k-1} - ud_{k-1,n}, EF_{k,n} - ud_{k,n} - Lg_{k,k-1}) \end{array} \quad (25)$$

  - If the relationship is SS and there is a time lag between ($Lg_{k,k-1}$), then set Late Start of Act_{k-1} for the n^{th} unit ($LS_{k-1,n}$) equal to Late Start of Act_k for the n^{th} unit ($LS_{k,n}$) minus the time lag. Then apply the unit durations ($ud_{k-1,1}$, $ud_{k-1,2}$ … $ud_{k-1,n}$) to $LS_{k-1,n}$ to find the Early Start/Early Finish values for the preceding units ($LS_{k-1,1}$ … $LS_{k-1,n-1}$; $LF_{k-1,1}$ ... $LF_{k-1,n}$). This will provide resource continuity for Act_{k-1}.

$$\begin{array}{lllll} & \text{Unit 1} & \ldots & \text{Unit } n-1 & \text{Unit } n \\ Act_{k-1}: & (LS_{k-1,1}, LF_{k-1,1}) & \ldots & (LS_{k-1,n-1}, LF_{k-1,n-1}) & (LS_{k-1,n}, LF_{k-1,n}) \\ & (EF_{k,n} - ud_{k,n} - Lg_{k,k-1} - ud_{k-1,n-1} - \ldots - ud_{k-1,2} - ud_{k-1,1}, EF_{k,n} - ud_{k,n} - Lg_{k,k-1} - ud_{k-1,n-1} - \ldots - ud_{k-1,2}) & \ldots & (EF_{k,n} - ud_{k,n} - Lg_{k,k-1} - ud_{k-1,n-1}, EF_{k,n} - ud_{k,n} - Lg_{k,k-1}) & (EF_{k,n} - ud_{k,n} - Lg_{k,k-1}, EF_{k,n} - ud_{k,n} - Lg_{k,k-1} + ud_{k-1,n}) \end{array} \quad (26)$$

  - If the relationship is FF and there is a time lag between ($Lg_{k,k-1}$), then set Late Finish of Act_{k-1} for the n^{th} unit ($LF_{k-1,n}$) equal to Late Finish of Act_k for the n^{th} unit ($LF_{k,n}$) minus the time lag. Then apply the unit durations ($ud_{k-1,1}$, $ud_{k-1,2}$ … $ud_{k-1,n}$) to $LF_{k-1,n}$ to find the Early Start/Early Finish values for the preceding units ($LS_{k-1,1}$ … $LS_{k-1,n}$; $LF_{k-1,1}$ ... $LF_{k-1,n-1}$). This will provide resource continuity for Act_{k-1}.

$$\begin{array}{lllll} & \text{Unit 1} & \ldots & \text{Unit } n-1 & \text{Unit } n \\ Act_{k-1}: & (LS_{k-1,1}, LF_{k-1,1}) & \ldots & (LS_{k-1,n-1}, LF_{k-1,n-1}) & (LS_{k-1,n}, LF_{k-1,n}) \\ & (EF_{k,n} - ud_{k-1,n} - Lg_{k,k-1} - ud_{k-1,n-1} - \ldots - ud_{k-1,2} - ud_{k-1,1}, EF_{k,n} - ud_{k-1,n} - Lg_{k,k-1} - ud_{k-1,n-1} - \ldots - ud_{k-1,2}) & \ldots & (EF_{k,n} - ud_{k-1,n} - Lg_{k,k-1} - ud_{k-1,n-1}, EF_{k,n} - ud_{k-1,n} - Lg_{k,k-1}) & (EF_{k,n} - ud_{k-1,n} - Lg_{k,k-1}, EF_{k,n} - Lg_{k,k-1}) \end{array} \quad (27)$$

  - If the relationship is SF and there is a time lag between ($Lg_{k,k-1}$), then set Late Start of Act_{k-1} for the n^{th} unit ($LS_{k-1,n}$) equal to Late Finish of Act_k for the n^{th} unit ($LF_{k,n}$) minus the time lag. Then apply the unit durations ($ud_{k-1,1}$, $ud_{k-1,2}$ … $ud_{k-1,n}$) to $LS_{k-1,n}$ to find the Early Start/Early Finish values for the preceding units ($LS_{k-1,1}$ … $LS_{k-1,n-1}$; $LF_{k-1,1}$ ... $LF_{k-1,n}$). This will provide resource continuity for the Act_{k-1}.

$$\begin{array}{lllll} & \text{Unit 1} & \ldots & \text{Unit } n-1 & \text{Unit } n \\ Act_{k-1}: & (LS_{k-1,1}, LF_{k-1,1}) & \ldots & (LS_{k-1,n-1}, LF_{k-1,n-1}) & (LS_{k-1,n}, LF_{k-1,n}) \\ & (EF_{k,n} - Lg_{k,k-1} - ud_{k-1,n-1} - \ldots - ud_{k-1,2} - ud_{k-1,1}, EF_{k,n} - Lg_{k,k-1} - ud_{k-1,n-1} - \ldots - ud_{k-1,2}) & \ldots & (EF_{k,n} - Lg_{k,k-1} - ud_{k-1,n-1}, EF_{k,n} - Lg_{k,k-1}) & (EF_{k,n} - Lg_{k,k-1}, EF_{k,n} + ud_{k-1,n} - Lg_{k,k-1}) \end{array} \quad (28)$$

- **Stage – 2**

  Then, check the logic. The rules are:

For FS case, $LF_{k-1,n} \le LS_{k,n} - Lg_{k,k-1}$ (29)

For SS case, $LS_{k-1,n} \le LS_{k,n} - Lg_{k,k-1}$ (30)

For FF case, $LF_{k-1,n} \le LF_{k,n} - Lg_{k,k-1}$ (31)

For SF case, $LS_{k-1,n} \le LF_{k,n} - Lg_{k,k-1}$ (32)

If the corresponding rule is not satisfied:

For FS case, $LF^*_{k-1,n} = LF_{k-1,n} - MAX\,[LF_{k-1,n} - (LS_{k,n} - Lg_{k,k-1}), \ldots, LF_{k-1,1} - (LS_{k,1} - Lg_{k,k-1})]$ (33)

For SS case, $LS^*_{k-1,n} = LS_{k-1,n} - MAX\,[LS_{k-1,n} - (LS_{k,n} - Lg_{k,k-1}), \ldots, LS_{k-1,1} - (LS_{k,1} - Lg_{k,k-1})]$ (34)

For FF case, $LF^*_{k-1,n} = LF_{k-1,n} + MAX\,[LF_{k-1,n} - (LF_{k,n} - Lg_{k,k-1}), \ldots, LF_{k-1,1} - (LF_{k,1} - Lg_{k,k-1})]$ (35)

For SF case, $LS^*_{k-1,n} = LS_{k-1,n} + MAX\,[LS_{k-1,n} - (LF_{k,n} - Lg_{k,k-1}), \ldots, LS_{k-1,1} - (LF_{k,1} - Lg_{k,k-1})]$ (36)

where n: 1,2, … n and * denotes the new correct value.

i^{th} Activity (Act_i)

An arbitrary activity in the network, Act_i, may be dependent on more than one successor activity (s_1, s_2 … s_m) with different relationships among FS, SS, FF and SF. In this general case, execute the required methods among the FS, SS, FF and SF cases presented through Equations (25) – (28) to obtain Late Finish values (LF_1, LF_2 … LF_m), as many as the successors (m) for the activity in question (Act_i). Then, in accordance with the CPM's backward pass calculation logic, apply the minimisation process on LF_1, LF_2 … LF_m to find the true LF value ($LF_{i,n}$) for the last unit of Act_i.

$$LF_{i,n} = Min\,(LF_1, LF_2 \ldots LF_m) \quad (37)$$

Then find LS and LF values of the other units of Act_i ($LS_{i,1}$ … $LS_{i,n}$; $LF_{i,1}$ … $LF_{i,n-1}$). Subsequently, check the logic. The rules are:

For FS case, $LF_{i,n} \leq LS_{s,n} - Lg_{s,i}$ (38)

For SS case, $LF_{i,n} - ud_{i,n} \leq LS_{s,n} - Lg_{s,i}$ (39)

For FF case, $LF_{i,n} \leq LF_{s,n} - Lg_{s,i}$ (40)

For SF case, $LF_{i,n} - ud_{i,n} \leq LF_{s,n} - Lg_{s,i}$ (41)

If the corresponding rule is not satisfied:

For FS case, $LF^*_{i,n} = LF_{i,n} - MAX\,[LF_{i,n} - (LS_{s,n} - Lg_{s,i}), \ldots, LF_{i,1} - (LS_{s,1} - Lg_{s,i})]$ (42)

For SS case, $LS^*_{i,n} = LF_{i,n} - ud_{i,n} + MAX\,[LF_{i,n} - ud_{i,n} - (LS_{s,n} - Lg_{s,i}), \ldots, LF_{i,1} - ud_{i,1} - (LS_{s,1} - Lg_{s,i})]$ (43)

For FF case, $LF^*_{i,n} = LF_{i,n} + MAX\,[LF_{i,n} - (LF_{s,n} - Lg_{s,i}), \ldots, LF_{i,1} - (LF_{s,1} - Lg_{s,i})]$ (44)

For SF case, $LS^*_{i,n} = LF_{i,n} - ud_{i,n} + [LF_{i,n} - ud_{i,n} - (LF_{s,n} - Lg_{s,i}), \ldots, LF_{i,1} - ud_{i,1} - (LF_{s,1} - Lg_{s,i})]$ (45)

where n: 1,2, … n; s: s_1, s_2 … s_m and * denotes the new correct value.

Determination of critical units and project completion time

The activities that have no float time are called critical activities in CPM. CPM firstly satisfies logical dependencies between network activities, and subsequently enables revising the schedule in accordance with the resource availability constraints. However, resource continuity cannot be maintained with CPM. Linear projects require continuous usage of resources. This requirement leads to two kinds of criticality in linear projects: logic criticality (like the criticality defined by CPM) and resource criticality. If an activity is logically critical, any delay in the completion of that activity delays the project. If the activity is resource critical, part of the critical path (controlling path), and consequently the project, will be delayed. Therefore, some activities become critical in linear scheduling although they are non-critical according to CPM.

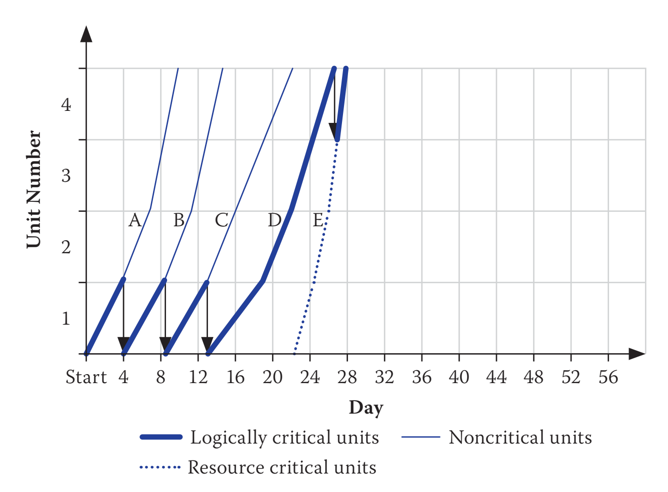

Figure 1 An example line representation of a linear schedule

One of the methods of showing the linear schedules is to use lines with constant or varying slopes drawn on graphs with the axes being units versus time or stations (or locations) versus time. The slope of an activity line represents its production rate. The proposed CPM-based scheduling procedure for linear construction projects does not include a rule-based algorithm to determine the logically and resource critical activities. Instead, the user determines the logically and resource critical activities along units on the line graph created through the results obtained in Steps 1 and 2 by using the criticality definitions proposed by linear scheduling. Early finish or late finish date of the last activity for the last unit, which is determined through forward and backward pass calculations of the procedure (Steps 1–2), is the project completion time.

As an example, unit-based criticalities of the activities of a linear schedule given in Figure 1 are as follows:

- A1, B1, C1, D1–D4 and E4 are logically critical due to logical dependency.
- E1–E3 is resource critical because E1, E2 and E3 are scheduled to provide resource continuity. If any of them is delayed, E4, which is part of the critical path (controlling path), will be delayed.
- A2–A4, B2–B4, and C2–C4 are noncritical. A delay in any of these activities will not delay the completion date of the project. However, it leads to interruption of resource usage from unit to unit.

EXAMPLE APPLICATION

The purpose of this section is to demonstrate the application of the procedure on an example construction project. The project consists of the construction of four warehouse units. Each unit requires the completion of 15 activities and each activity repeats four times through four units. Table 1 presents the data of the example application (Newitt 2005); the activity names, relationship types, lag times, and unit-based activity durations. FS, SS and FF relationships are used. It is assumed that each of the units consists of the same 15 activities. However, the quantities of production for each activity are assumed to vary from unit to unit due to the learning effect. For this reason, unit durations of activities vary from one unit to another as shown in Table 1.

Table 1 Data of example application

Id.	Activity	Predecessor & Relationship	Duration (Unit 1)	Duration (Unit 2)	Duration (Unit 3)	Duration (Unit 4)
A	Excavate	–	5	5	4	4
B	Form slab	A (FS)	3	3	2	2
C	Pour slab	B (FS)	3	3	2	2
D	Frame exterior walls	C (FS+3)	8	8	6	6
E	Frame roof	D (FS)	4	4	3	3
F	Brick sides	D (SS+3)	9	9	7	7
G	Rough electrical	E (FS) F (FS)	4	4	3	3
H	Shingle roof	E (FS)	4	4	3	3
I	Windows and doors	F (FF+2)	5	5	4	4
J	Insulate	I (FS) G (FS)	3	3	2	2
K	Install drywall	H (FS) J (FS)	5	5	4	4
L	Paint interior	I (FS) K (FF)	5	5	4	4
M	Paint exterior	F (FS)	6	6	5	5
N	Finish electrical	M (FS) L (FS)	2	2	2	2
O	Close out	N (FS)	1	1	1	1

Table 2 Forward pass solution of example application

Id.	Activity	Unit 1	Unit 2	Unit 3	Unit 4
A	Excavate	(0 – 5)	(5 – 10)	(10 – 14)	(14 – 18)
B	Form slab	~~(5 – 8)~~ (10 – 13)	~~(8 – 11)~~ (13 – 16)	~~(11 – 13)~~ (16 – 18)	~~(13 – 15)~~ (18 – 20)
C	Pour slab	(13 – 16)	(16 – 19)	(19 – 21)	(21 – 23)
D	Frame exterior walls	(19 – 27)	(27 – 35)	(35 – 41)	(41 – 47)
E	Frame roof	~~(27 – 31)~~ (36 – 40)	~~(31 – 35)~~ (40 – 44)	~~(35 – 38)~~ (44 – 47)	~~(38 – 41)~~ (47 – 50)
F	Brick sides	(22 – 31)	(31 – 40)	(40 – 47)	(47 – 54)
G	Rough electrical	~~(40 – 44)~~ (43 – 47)	~~(44 – 48)~~ (47 – 51)	~~(48 – 51)~~ (51 – 54)	~~(51 – 54)~~ (54 – 57)
H	Shingle roof	(40 – 44)	(44 – 48)	(48 – 51)	(51 – 54)
I	Windows and doors	~~(28 – 33)~~ (38 – 43)	~~(33 – 38)~~ (43 – 48)	~~(38 – 42)~~ (48 – 52)	~~(42 – 46)~~ (52 – 56)
J	Insulate	~~(47 – 50)~~ (49 – 52)	~~(50 – 53)~~ (52 – 55)	~~(53 – 55)~~ (55 – 57)	~~(55 – 57)~~ (57 – 59)
K	Install drywall	(52 – 57)	(57 – 62)	(62 – 66)	(66 – 70)
L	Paint interior	(52 – 57)	(57 – 62)	(62 – 66)	(66 – 70)
M	Paint exterior	~~(31 – 37)~~ (37 – 43)	~~(37 – 43)~~ (43 – 49)	~~(43 – 48)~~ (49 – 54)	~~(48 – 53)~~ (54 – 59)
N	Finish electrical	~~(57 – 59)~~ (64 – 66)	~~(59 – 61)~~ (66 – 68)	~~(61 – 63)~~ (68 – 70)	~~(63 – 65)~~ (70 – 72)
O	Close out	~~(66 – 67)~~ (69 – 70)	~~(67 – 68)~~ (70 – 71)	~~(68 – 69)~~ (71 – 72)	~~(69 – 70)~~ (72 – 73)

Table 2 shows the forward pass solution. Forward pass requires the use of directions given through Equations (2) – (23). It was applied manually by using the data in Table 1. In Table 2 the crossed-out rows show the eliminated values during the execution of Equations (7) – (14) and Equations (15) – (23). The values in unlined brackets represent the final and true ES and EF time values of the activities for each unit separately. For example, the activity "M – Paint Exterior" starts on the 37th day and continues consuming the resources during the next 22 days until the 59th day without interruption through the four consecutive warehouse units.

According to the forward pass solution in Table 2, linear construction of four repetitive warehouse units ends after 73 days from the beginning (EF value of the last activity for the last unit).

Table 3 shows the backward pass solution. Backward pass requires the usage of directions given through Equations (24) – (45). Backward pass was applied manually by using the data in Table 1 and the data of the forward pass solution given in Table 2. In Table 3, the crossed-out rows show the eliminated values during the execution of Equations (29) – (36) and Equations (37) – (45). The values in unlined brackets represent the final and true LS and LF time values of the activities for each unit separately. For example, the activity "J – Insulate" starts on the 49th day and continues consuming the resources during the next ten days until the 59th day without interruption through the four consecutive warehouse units.

Figures 2 and 3 show the "linear schedule graph" and the "linear schedule graph that indicates unit-based criticalities", respectively. Results of forward and backward pass solutions given in Tables 2 and 3 have been used to build these graphs. Unit-based criticalities shown in Figure 3 are discussed below. The descriptions should be followed concurrently with Tables 1 – 3.

- A1 – A4: All the subsequent units of the activity A (A1, A2, A3, A4) are logically critical. It is the first activity and has no predecessor. In

Table 3 Backward pass solution of example application

Id.	Activity	Unit 1	Unit 2	Unit 3	Unit 4
A	Excavate	(0 – 5)	(5 – 10)	(10 – 14)	(14 – 18)
B	Form slab	~~(11– 14)~~ (10 – 13)	~~(14 – 17)~~ (13 – 16)	~~(17 – 19)~~ (16 – 18)	~~(19 – 21)~~ (18 – 20)
C	Pour slab	~~(28 – 31)~~ (13 – 16)	~~(31 – 34)~~ (16 – 19)	~~(34 – 36)~~ (19 – 21)	~~(36 – 38)~~ (21 – 23)
D	Frame exterior walls	~~(22 – 30)~~ (19 – 27)	~~(30 – 38)~~ (27 – 35)	~~(38 – 44)~~ (35 – 41)	~~(44 – 50)~~ (41 – 47)
E	Frame roof	~~(40 – 44)~~ (39 – 43)	~~(44 – 48)~~ (43 – 47)	~~(48 – 51)~~ (47 – 50)	~~(51 – 54)~~ (50 – 53)
F	Brick sides	(22 – 31)	(31 – 40)	(40 – 47)	(47 – 54)
G	Rough electrical	(43 – 47)	(47 – 51)	(51 – 54)	(54 – 57)
H	Shingle roof	~~(52 – 56)~~ (48 – 52)	~~(56 – 60)~~ (52 – 56)	~~(60 – 63)~~ (56 – 59)	~~(63 – 66)~~ (59 – 62)
I	Windows and doors	(39 – 44)	(44 – 49)	(49 – 53)	(53 – 57)
J	Insulate	~~(56 – 59)~~ (49 – 52)	~~(59 – 62)~~ (52 – 55)	~~(62 – 64)~~ (55 – 57)	~~(64 – 66)~~ (57 – 59)
K	Install drywall	(52 – 57)	(57 – 62)	(62 – 66)	(66 – 70)
L	Paint interior	(52 – 57)	(57 – 62)	(62 – 66)	(66 – 70)
M	Paint exterior	(48 – 54)	(54 – 60)	(60 – 65)	(65 – 70)
N	Finish electrical	(64 – 66)	(66 – 68)	(68 – 70)	(70 – 72)
O	Close out	(69 – 70)	(70 – 71)	(71 – 72)	(72 – 73)

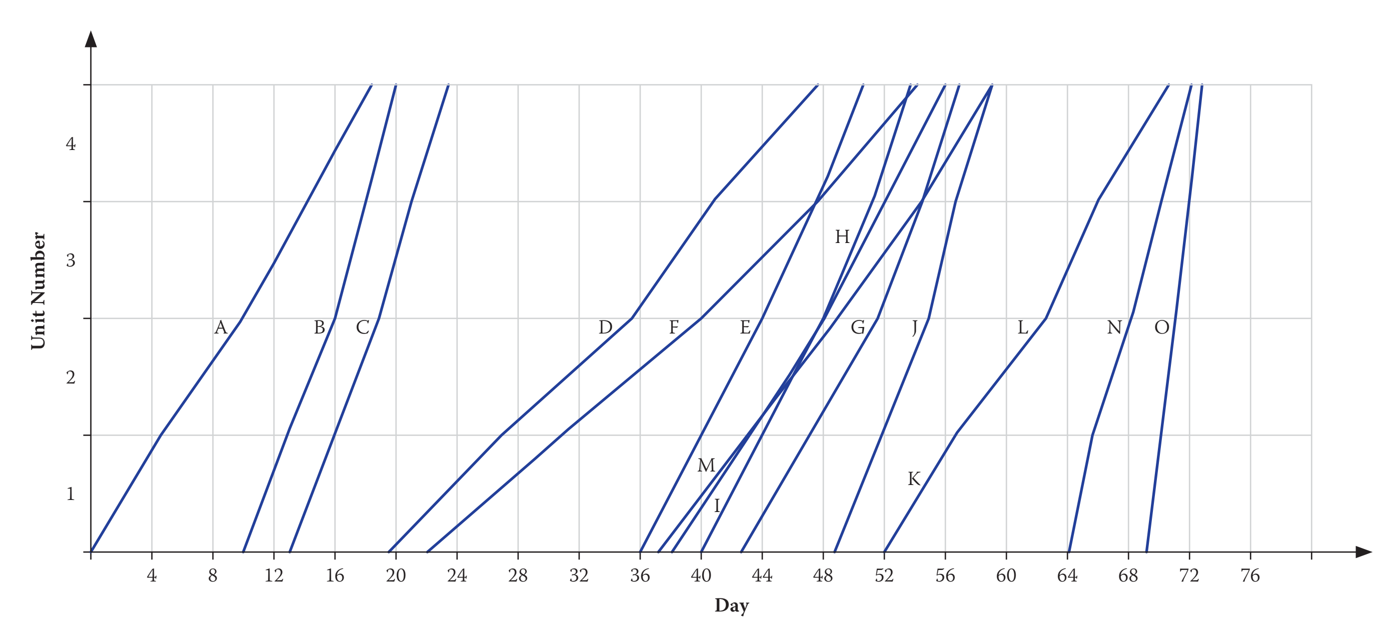

Figure 2 Linear schedule graph of example application

case a delay occurs in activity A, activity B, which is the successor activity, extends. This would also cause a delay in the project.

- B4: B4 starts after A4 finishes. They are dependent with FS relationship. However, B4 is noncritical. A delay in B4 within float time limits leads to a delay in C4, but this will not cause a delay in the project. The float time between B4 and C4 is one day. While one day of extension has no effect on the project completion time, an extension of two days would, for example, result in a delay.
- B2 – B3: B2 and B3 are logically critical. A delay in B2–B3 will cause time extension in B1. This would affect the project.
- B1: B1 maintains resource continuity of activity B and it is the predecessor of C1. Delay in B1 causes delay in the start of activity C. This would also lead to a delay in the project. Therefore it is a resource critical unit.
- C1: C1 is dependent on B1 with FS relationship. It is logically critical.
- C2 – C4: Float time exists between the C2 – C4 and their corresponding successors, D2 – D4. Therefore, C2, C3 and C4 are noncritical. The float times are 5, 6 and 10 for the C2, C3 and C4, respectively.
- D1: D1 is dependent on C1 with an FS+3 relationship. Start time of D1 is effective on the start time of activity F. Therefore, start time of D1 is considered as logically critical. However, finish time of D1 has float time although its start time does not. One day of float time is possible for D1's finish time. If D1 was extended by one day, start of F1 would not be delayed. Finally, start of D1 is a critical benchmark and D1 is noncritical.
- D2 – D4: D2, D3 and D4, which are the successors of D1, are non-critical due to the possible float times between the starts of D2, D3, D4 and starts of F2, F3, F4, respectively.
- E1 – E4: E1 – E4 are noncritical. A delay in the project is not possible in the event that E1 – E4 extend within float time limits.
- F1 – F4: Start time of activity F is dependent on start time of activity D, and start time of activity G4, which is the successor of activity F, is dependent on the finish time of F4. This makes F1 – F4 logically critical.

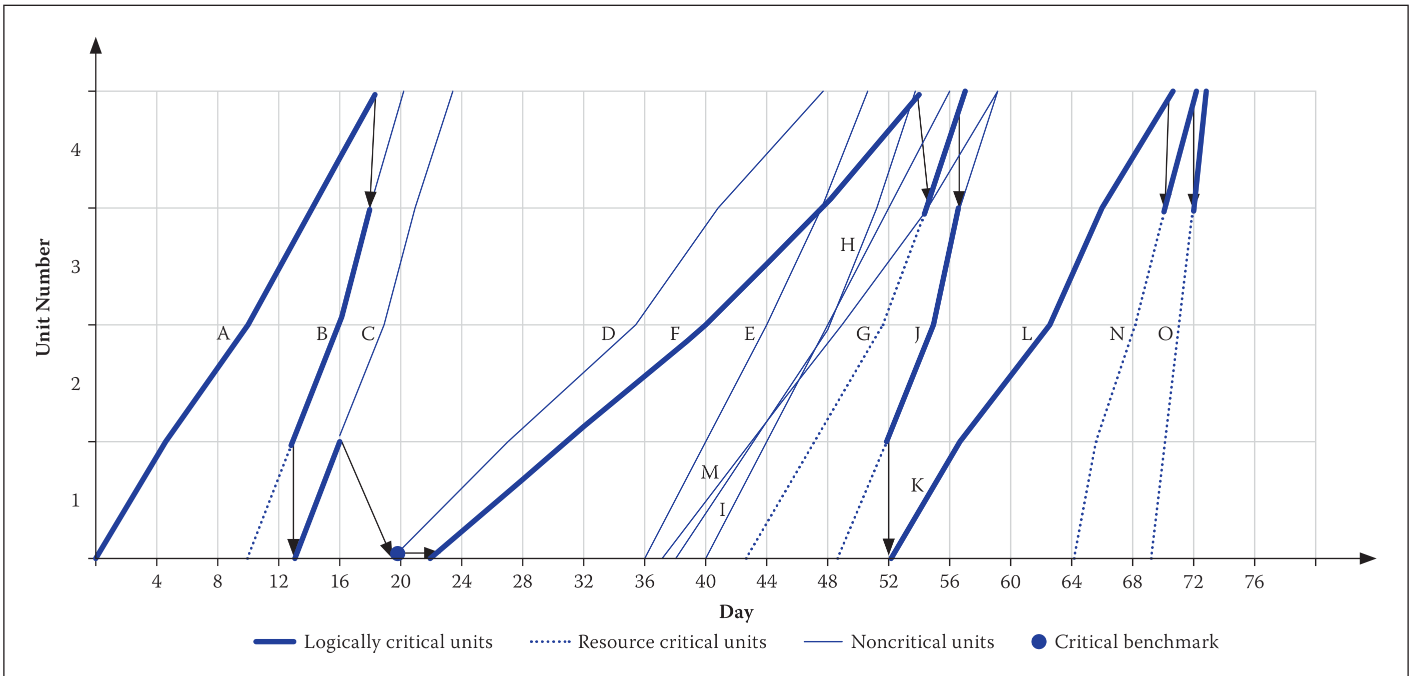

Figure 3 Linear schedule graph of example application showing unit-based criticality of activities

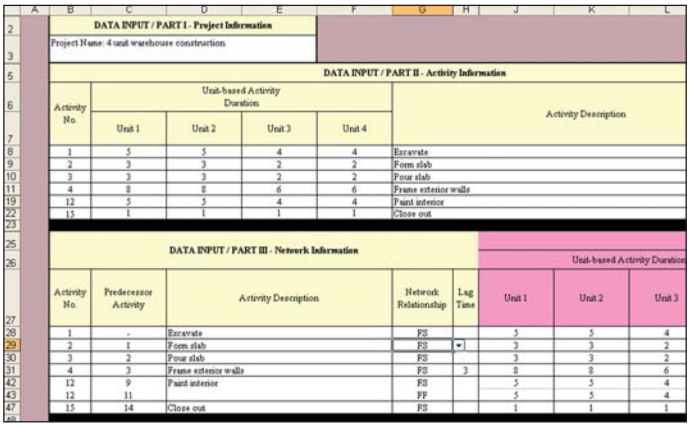

Figure 4 Data input parts of MS Excel spreadsheet

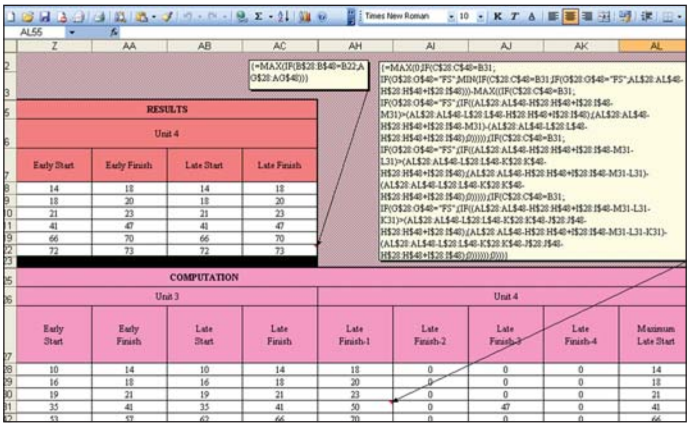

Figure 5 Results and computation parts of MS Excel spreadsheet

- G4: G4 depends on F4 with FS relationship. A delay in G4 causes delay in the project. Therefore, it is logically critical.
- G1 – G3: G1 – G3 are the predecessors of G4 and they maintain resource continuity. Any delay in G1 – G3 leads to extension in G4, which is logically critical, and in turn leads to extension in the project. Therefore, G1, G2 and G3 are resource critical units.
- J4: J4 is dependent on G4, which is a logically critical unit, with FS relationship. However, a possible delay in J4 does not cause delay in its successors, if the delay is within float time limits. Therefore, it is noncritical.
- J2 – J4: A delay in J2 – J4 will cause extension in J1, which is the successor of K1 and L1. Since activities K and L are logically critical, a delay in J2 – J4 will cause extension in J1 and, depending on this, the project will be delayed.
- J1: J1 maintains resource continuity of activity J and it is the predecessor of K1 and L1. A delay in J1 causes delay in activities K and L's start times. Since activities K and L are logically critical, the project will extend in such a case. Therefore, J1 is a resource critical unit.
- K1 – K4: Start time of the activity K is dependent on the finish time of J1, and start time of activity N4, which is the successor of K4, is dependent on the finish time of K4. This makes K1 – K4 logically critical.
- L1 – L4: Activity L is dependent on the activities I and K with FS and FF relationships. However, activity K is determinant on activity L. Since activity K is logically critical, activity L, which has the same early and later times with activity K, is also logically critical.
- N4: N4 depends on L4 and M4 with FS relationships. However, activity L is determinant on activity N. A delay in N4 causes delay in the project. Therefore, it is logically critical.
- N1 – N3: N1 – N3 are the predecessors of N4 and they maintain resource continuity. They are resource critical units. Any delay in N1 – N3 leads to extension in O4, which is logically critical, and accordingly this causes extension of the project.
- O4: O4 depends on N4 with FS relationship. A delay in O4, which determines the project completion time, causes delay in the project. Therefore, it is logically critical.
- O1 – O3: O1 – O3 are the predecessors of O4 and they maintain resource continuity. They are resource critical units. Any delay in O1 – O3 leads to extension in O4, which is logically critical and accordingly causes extension of the project.
- H1 – H4, I1 – I4, M1 – M4: These are noncritical. They can be extended within float time limits without causing any delay of the project.

PROCEDURE COMPUTERISATION

The table processor software, MS Excel, was used to computerise the procedure. First of all, CPM forward/backward pass calculations were achieved by using MS Excel formulae on an MS Excel spreadsheet. This step is important because the procedure is based on CPM calculations. The procedure was then developed on the same spreadsheet through designing the input cells, first, and setting up the procedure's algorithm with MS Excel formulae, second. Figures 4 and 5 show the developed spreadsheets. Three parts exist on the spreadsheet: "data input", "computation" and "results". The user only enters the required data into the input parts, while the solution is presented in the "results" part. In Figure 5 the formulae used in some cells are shown as an example.

DISCUSSION AND RECOMMENDATIONS

The CPM-based Linear Scheduling procedure performs well in compliance with the expectancies from a linear schedule, as the example application has shown. The conduction of further applications also provided similar satisfying results. The advantageous features of the procedure are as follows:

- Scheduling of linear construction projects containing repetitive units with identical activities is possible without ignoring resource continuity and network relationships between activities.
- The amount of data required to execute the procedure is not much more than the amount required for CPM application. The proposed procedure further requires only the number of units and the varying activity durations (or the production rates) for each unit.
- The procedure allows the scheduler to use any relationship type among start-to-start, finish-to-start, finish-to-finish and start-to-finish. Moreover, it enables assigning lag times between activities.
- The procedure enables the identification of critical activities and float times of noncritical activities. It clarifies the unit-based criticality of activities.
- The procedure determines unit-based ES, LS, EF and LF dates of activities. One can easily produce a Bar Chart schedule of units by using these values.
- The procedure enables the analysis of the float times of noncritical activities in detail.

In spite of its advantageous features, the proposed procedure also has some limitations, as discussed below together with some recommendations for future work.

- One should be eligible both in CPM and Linear Scheduling in order to use the procedure. This increases the user's work. The computerised form of the procedure is beneficial for its practical use.
- The procedure is open to improvement regarding the resource levelling and crashing issues.
- The procedure assumes that the number of repetitive units remains constant throughout all activities, and it uses the same

group of activities for each unit. However, some units may not contain a number of activities common to the other units. Elimination of this limitation would improve the procedure.

- The procedure assumes single crew members per activity. This is not practical for repetitive projects, and can be improved by considering multiple crew usage and resource availability constraints.

SUMMARY AND CONCLUSIONS

This paper presents a procedure to apply the Critical Path Method (CPM) to linear construction projects that have repeating activities. The main feature of the proposed procedure is its ability to perform forward and backward pass calculations of CPM on a linear project with simultaneously accounting network logic and maintaining the continuous resource usage, and to identify critical paths through successive units. It allows the usage of any kind of dependency relationship among finish-to-start, start-to-start, finish-to-finish and start-to-finish, and lag times between activities. Furthermore, it enables the usage of variable unit durations or production rates for activities from one unit to another. The proposed procedure computes the project completion time, early start and finish times, late start and finish times, and total float times; and determines the critical activities through a unit by unit approach. The paper contains the detailed description of the procedure, an example application, discussion of advantages and limitations, and some recommendations for future work. Furthermore, a spreadsheet has been developed by using a table processor in order to computerise the procedure.

REFERENCES

Ammar, M A & Elbeltagi, E 2001. Algorithm for determining controlling path considering resource continuity. *Journal of Computing in Civil Engineering*, 15(4): 292–298.

Arditi, D & Albulak, M Z 1986. Line-of-balance scheduling in pavement construction. *Journal of Construction Engineering and Management,* 112(3): 411–424.

Ashley, D B 1980. Simulation of repetitive-unit construction. *Journal of Construction Division,* 106 (2): 185–194.

El-Rayes, K & Moselhi, O 1998. Resource-driven scheduling of repetitive projects. *Journal of Construction Management and Economics,* 16(4): 433–446.

Griffis, F H & Farr, J V 2000. *Construction planning for engineers*. Singapore: McGraw-Hill.

Halphin, D W & Woodhead, R W 1998. *Construction management*. New York: Wiley.

Harmelink, D J & Rowings, J E 1998. Linear scheduling model: Development of controlling activity path. *Journal of Construction Engineering and Management,* 124(4): 263–268.

Harris, R B & Ioannou, P G 1998. Scheduling projects with repeating activities. *Journal of Construction Engineering and Management,* 124(4): 269–273.

Hegazy, T, Moselhi, O & Fazio, P 1993. BAL: An algorithm for scheduling and control of linear projects. *AACE Transactions*, Morgantown W. VA, pp 8.1–8.14.

Huang, R & Sun, K S 2005. System development for non-unit based repetitive project scheduling. *Journal of Automation in Construction*, 14(5): 650–655.

Johnston, D W 1981. Linear scheduling method for highway construction. *Journal of Construction Engineering and Management,* 107(2): 247–261.

Lumsden, P 1968. *The line of balance method*. London: Pergamon Press.

Mattila, K G & Abraham, D M 1998. Resource leveling of linear schedules using integer linear programming. *Journal of Construction Engineering and Management,* 124(3): 232–244.

Mawdesley, M J, Askew, W H, Lees, J, Taylor, J & Stevens, C M 1989. Time change charts for scheduling linear projects. *Proceedings,* 6th ASCE Conference on Computing in Civil Engineering, Atlanta, pp 613–620.

Newitt, J S 2005. *Construction scheduling principles and practices*. New Jersey: Prentice Hall.

Oberlender, G D 2000. *Project management for engineering and construction.* Boston: McGraw-Hill.

O'Brien, J J 1975. VPM scheduling for high-rise buildings. *Journal of Construction Division,* 101(4): 895–905.

Reda, R M 1990. RPM: Repetitive project modeling. *Journal of Construction Engineering and Management,* 116(2): 316–330.

Russell, A D & Wong, W C M 1993. New generation of planning structures. *Journal of Construction Engineering and Management,* 119(2): 196–214.

Selinger, S 1980. Construction planning for linear projects. *Journal of Construction Division*, 106(2): 195–205.

Stradal, O & Cacha, J 1982. Time space scheduling method. *Journal of Construction Division,* 108(3): 445–457.

Yang, I & Ioannou, P G 2004. Scheduling system with focus on practical concerns in repetitive projects. *Journal of Construction Management and Economics*, 22(6): 619–630.