

FLEXIBLE MODELING OF LINEAR SCHEDULES FOR INTEGRATED MATHEMATICAL ANALYSIS

Gunnar Lucko

Department of Civil Engineering
The Catholic University of America
620 Michigan Avenue NE
Washington, DC 20064, U.S.A.

ABSTRACT

Developing and analyzing schedules is essential for successfully controlling the time aspect of construction projects. The critical path method of scheduling is by far the most widely used scheduling technique in the construction industry. However, several problems related to its concept and use have been identified in the literature. The lesser known linear scheduling method has much potential for handling the rich information that project managers handle on a daily basis. While being well-suited to linear and repetitive projects, it has been a predominantly graphical method without a comprehensive underlying mathematical model. This paper introduces a new method based on singularity functions using Macaulay bracket notation. It makes beneficial use of the strengths of these functions, including their flexibility and clarity, while remaining intuitive and requiring only basic geometry and algebra knowledge from users. An example from the literature is re-analyzed step-by-step and compared successfully with the critical path method.

1 INTRODUCTION

Construction management examines the rich variety of data and their interrelationships related to the time, cost, scope, and other dimensions of projects. Scheduling is a major discipline within construction management that concerns itself primarily with analysis and control of the time aspect. The most commonly known and used scheduling method in construction management is the *critical path method* (CPM), which was conceived through efforts of Kelley and Walker (1959) based on earlier work in linear programming analysis of construction operations (Kelley 1957). The straightforward algorithm of CPM using only two mathematical operators (addition and maximum), its intuitive graphical visualization as directional network diagrams, and its implementation into commercial software tools have contributed to the vast success of

CPM in the construction industry (Galloway 2006; Anton et al. 2007).

1.1 Problems of Critical Path Method

CPM focuses strongly on the time aspect of construction projects, making it effectively *one-dimensional* (1D), contrary to the need to analyze the manifold project dimensions in an integrated manner. Criticism specific to the graphical representation has been published (Glenwright 2004, p. 5), leading to the conclusions “[t]he state of construction scheduling is in dire need of improvement.” Its most common diagrams using the precedence diagramming method (PDM) are not even time-scaled, while the time-scaled bar charts are by default shown without the necessary relationship arrows in the software. PDM only consider whole activities. Partially completed ones, as typically found whenever a schedule is updated, require splitting to be reflected graphically and do not translate well into the numerical CPM analysis based on days.

Durations in scheduling are an outcome of performing an estimating quantity take-off of the plan drawings, determining the required resources, their availability, cost, and finally their productivity. The relationships between activities are determined by laws of nature, technological, and managerial factors, such as e.g. sequencing structures from foundations to the roof and performing intermediate inspections at certain milestones. Additionally, the geometric layout of the site itself may require certain means and methods. Such spatial analysis is not part of CPM. It is thus a rather basic tool considering the richness of information that project managers need to record, analyze, and control every day. Birrell (1980) notes the singular focus of CPM on durations to be alien to construction. Jaafari (1984, p. 225) listed several conceptual detriments, including its origins on non-construction projects, unrealistic assumptions of fixed times and discrete elements, the need for a broader consideration of project dimensions in planning, and the need for continuous resource utilization.

(Reda 1990, p. 317-318) names three disadvantages of CPM: “First, it requires a large number of activities to represent the project. (...) This large number of activities makes it extremely difficult to visualize the project. (...) Second,... [it] does not guarantee maintaining the continuity of work. That is, a crew with a fast production rate may be idle while waiting for preceding crews with slower production rates to finish their work. This is because CPM schedules the start of each activity as soon as all its preceding activities are finished. Third, to minimize the project cost, CPM techniques shorten only critical activities by increasing resources allocated to them and thus increase their production rates. This may result in varying production rates for similar activities at different stages, where some of these activities may be critical.”

1.2 Advantages of Linear Scheduling Method

The *linear scheduling method* (LSM) is a versatile project management tool. Its methodology (Al-Sarraj 1990) and graphical foundation (Vorster et al. 1992) is described in detail in the literature. Under LSM, each activity is represented as a line whose slope is proportional to its productivity to link the duration with the output that is achieved in such time. Seasoned project managers know that controlling productivity is of importance to achieve project success. LSM is ideally suited for construction projects that are characterized primarily by their longitudinal spatial nature or by their repetitiveness. Such projects may be highway and utility projects (Stradal and Cacha 1982), high-rise buildings (Thabet and Beliveau (1994), and repetitive operations where crews cycle through identical work stations (El-Rayes and Moselhi 1998).

The graphical representation of an LSM schedule is a *two-dimensional* (2D) diagram with a time and location axis. Time and location buffers may be considered as distance constraints that must be adhered to between activities. It is possible to detect physical interferences by finding points where lines touch or cross (Handa and Barcia 1986). Learning effects would cause the lines to curve upward (Arditi et al. 2001). While this intuitive approach well for manually examining small schedules, existing approaches have either been purely graphical (Harmelink and Rowings 1998) or very limited to constant activities (Arditi et al. 2002) and no formal mathematical analysis method exists beyond a vector analysis method requiring advanced mathematics (Russell and Caselton 1988). The actual use of LSM accordingly has been insignificant (Arditi and Albulak 1986; Mubarak 2005) and several authors pointed at the continuing need for computerization (Arditi et al. 2002; Mattila and Park 2003). Existing software applications for linear schedules, including *Graphisoft® Control 2005* and *TILOS* do not provide an algorithm. “[B]ar charts and networks sometimes generate inaccurate and misleading information in repetitive situations. Hence, there is a need for more powerful methods

of scheduling that will allow the user to make optimum use of time and resources” (Arditi et al. 2002, p. 555).

2 SINGULARITY FUNCTIONS

A typical structural engineering problem is that a horizontal beam is loaded by different types of loads at different locations or ranges across its length. It is divided into several segments where the behavior of the load and cross-section remain regular. Loads can be single or distributed. Distributed loads and the cross-section may be constant over the length of the beam or change with a higher order. Each segment is then virtually cut free from its neighbors, boundary conditions are written to mathematically bridge these interfaces with adequate internal forces, moments, displacements, and deformations, and the entire system of equations is solved for the unknown variables. However, a more elegant method exists to perform such analysis.

Singularity functions, also called discontinuity functions, were introduced concurrently by A. O. Föppl (1854-1924) in Germany (Föppl 1927) and by W. H. Macaulay (1853-1936) in Great Britain (Macaulay 1919). Their application is still taught in some structural engineering courses, yet has largely been replaced by more advanced computerized matrix analysis. Their multiple advantages and suitability for linear schedule analysis are explained in the following sections of this paper and are demonstrated with an example of a linear schedule.

2.1 Advantages

Singularity functions are a family of functions with several highly desirable properties for mathematical analysis.

- They are based on simple geometry;
- They can still be evaluated manually;
- They keep the components of interest separate;
- They directly capture any changes;
- They can contain infinitely many segments;
- They are continuous for all arguments;
- They can be simply differentiated and integrated;
- They can be unit-scaled by multiplication;
- They can be added and subtracted provided their cutoff value a and order n are identical.

2.2 Mathematical Basis

Macaulay (1919, p. 129) published a paper describing “a function of x that is zero when x is less than a and equal to $f(x)$ when x is equal to or greater than a ” as shown in (1), using the pointed brackets introduced by Wittrick (1965).

$$\langle x-a \rangle^n = \begin{cases} 0 & \text{for } x < a \\ (x-a)^n & \text{for } x \geq a \end{cases} \quad (1)$$

where x is a variable, a is the upper boundary of the current segment, i.e. its length, and n is the order of the phenomenon. The Macaulay brackets are treated like regular curved brackets for $x \geq a$ and for $x < a$ are set to zero. Their differentiation and integration as per (2) and (3) is identical to normal mathematical functions.

$$\frac{d}{dx} \langle x-a \rangle^n = n \cdot \langle x-a \rangle^{n-1} \quad (2)$$

$$\int \langle x-a \rangle^n dx = \frac{1}{n+1} \cdot \langle x-a \rangle^{n+1} + C \quad (3)$$

Here C is an integration constant. Note that beyond the definitions of (1) through (3) the following rules apply: Terms of two singularity functions can be added and subtracted if their segment length a and exponent n are identical. Singularity function can be multiplied by any factor to scale them to the desired unit. The Macaulay brackets thus capture three pieces of information:

- The value of the phenomenon as the multiplicative factor before the brackets;
- The segment boundary as part of the variable term inside the brackets;
- The order of the phenomenon as the exponent after the brackets.

For example, $n = 0$ indicates a constant phenomenon where the factor before the brackets is the intercept and $n = 1$ indicates a linear phenomenon where the factor before the brackets is the slope. “It should however be noticed that by the use of suitable notation the equations can be greatly simplified; for it becomes possible, in all cases, to write down a single differential equation for the whole length of the beam, and to obtain a single solution of it, introducing only two constants of integration” (Macaulay 1919, p. 129). The equation may contain any number of separate terms, each of which describes one segment. After the initial term, only terms that describe changes from current to following segment need to be included.

3 SCHEDULE ANALYSIS

The following sections re-examine an example of a linear schedule from the literature, first with the traditional CPM approach for both continuous and split activities and then with the new method using singularity functions.

3.1 Critical Path Method Analysis

The following is a description of the original example: “The example project is the relocation of 5 miles (18 km) [sic, correct is 8 km] of natural gas pipeline. This example is adopted from Clough and Sears (1979), with some

modification to illustrate the RPM [repetitive project modeling] features better. The repetitive segments of the pipeline relocation have been identified as locate and clear, excavate, string pipe, lay pipe, test, and backfill. The project is divided into five typical stages; each represents 1 mile (1.6 km) of the pipeline. (...) From the typical network, it can be seen that all the activities are sequential, except for the excavate and string pipe activities, which are done concurrently. In this example, the time buffer is assumed only to be equal to the time duration of the preceding activity. The equipment crews working on the excavate and string pipe activities must have a minimum stage buffer of one stage” (Reda 1990, p. 322-323).

The published approach develops a complete linear programming model that contains the linear schedule and its constraints and solves it while maintaining work continuity and constant productivities (Reda 1990). However, the linear programming model for the mere six activities necessarily consists of a disconnected set of 25 equations that each describe only one activity duration, one relationship between predecessor and successor including their unique buffers, or the objective function. Stage buffers are converted into equivalent time buffers. To correctly express constraints for multiple successor activities an additional assumption is needed that places activities that are executed concurrently into an order. This large set of equations can only be solved with the help of a computer.

The standard CPM analysis for this example places activities into a dependency structure as shown in the precedence diagram of Figure 1, which is also called an activity-on-node diagram after its graphical symbols. Table 1 lists the input for the CPM calculation, which includes the names of the activities, a description, their successors (or alternatively their predecessors), their duration in days, and their time buffers, which are called lead times in CPM and are attributed to links between activities, not to the predecessor activity itself. Since CPM does not contain any location information, the stage buffers cannot be modeled but must be checked manually in the end.

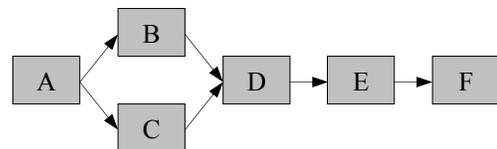


Figure 1: Activity-on-node network

The following CPM calculations are initially performed without the lead times to gain a comparison for the impact of the lead times themselves. The standard CPM formulas for earliest and latest starts and finishes and floats are used as follows. Note that this paper uses an end-of-day labeling convention for all points in time, whereas computer software uses start-of-day notation for starts and end-of-day notation for finishes (Lucko 2006).

Table 1: Activity list for critical path method

Name	Description	Successor	Duration	Lead
A	Locate & clear	B, C	5	1, 1
B	Excavate	D	20	4
C	String pipe	D	10	2
D	Lay pipe	E	25	5
E	Test	F	5	1
F	Backfill	-	10	-

Forward Pass:

$$ES_i = \max(EF_h)$$

$$EF_i = ES_i + D_i$$

Backward Pass:

$$LF = \min(LS_j)$$

$$LS_i = LF_i - D_i$$

Float Analysis:

$$TF_i = LS_i - ES_i = LF_i - EF_i$$

$$FF_i = (\min(LF_j)) - EF_i$$

where *ES*, *EF*, *LS*, and *LF* are the earliest start, earliest finish, latest start, and latest finish, respectively, *D* is the duration, *TF* is the total float, *FF* is the free float, and activities *h*, *i*, and *j* are a predecessor, a current activity, and a successor activity, respectively. Table 2 summarizes the CPM results. The total project duration is 65 days.

Table 2: Critical path method results

Name	Description	ES	EF	LS	LF	TF	FF
A	Locate & clear	0	5	0	5	0	0
B	Excavate	5	25	5	25	0	0
C	String pipe	5	15	15	25	10	10
D	Lay pipe	25	50	25	50	0	0
E	Test	50	55	50	55	0	0
F	Backfill	55	65	55	65	0	0

* Boldface activities are on the critical path.

If the lead times are considered, the CPM results increase to the values listed in Table 3. The total project duration is extended by the longest path of leads to 76 days.

Table 3: Critical path method results with leads

Name	Description	ES	EF	LS	LF	TF	FF
A	Locate & clear	0	5	0	5	0	0
B	Excavate	6	26	6	26	0	0
C	String pipe	6	16	18	28	12	12
D	Lay pipe	30	55	30	55	0	0
E	Test	60	65	60	65	0	0
F	Backfill	66	76	66	76	0	0

* Boldface activities are on the critical path.

This initial CPM analysis, however, assumes that all activities can only occur after their predecessors have been completed, i.e. the activities are stacking up with the maximum finish of the predecessors becoming the start of the successor. It is certainly more realistic to assume that activities can occur parallel to each other as long as they do not occupy the same location at the same time. The initial precedence diagram of Figure 1 is therefore extended into Figure 2, where the lead times are included after each start and finish of the respective predecessor activity. The CPM calculation now uses start-to-start and finish-to-finish relationships. Overall, four relationship types are possible, including also the already familiar finish-to-start and additionally the start-to-finish. The CPM formulas can easily be adjusted to include these different relationship types and accommodate their respective lead times.

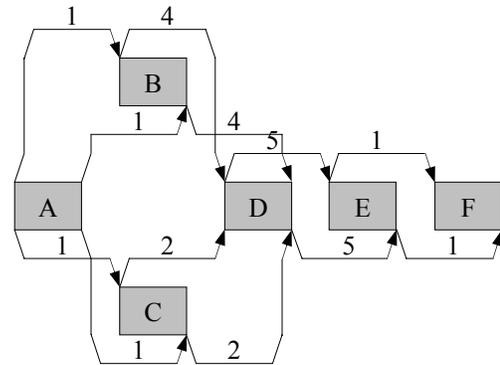


Figure 2: Activity-on-node network

Table 4 lists the extended input for the CPM calculation from Table 1 that also includes the relationship type.

Table 4: Extended activity list for critical path method

Name	Description	Successor (Type)	Duration	Lead
A	Locate & clear	B (SS, FF), C (SS, FF)	5	1, 1
B	Excavate	D (SS, FF)	20	4, 4
C	String pipe	D (SS, FF)	10	2, 2
D	Lay pipe	E (SS, FF)	25	5, 5
E	Test	F (SS, FF)	5	1, 1
F	Backfill	-	10	-

Table 5 summarizes the CPM results for this more realistic scenario of activities overlapping while maintaining their sequence. The total project duration is 41 days. Comparing this value with the previous durations of 65 days for CPM with finish-to-start relationships only and 76 days with consideration of lead times finds a reduction in duration of 24 days or 35 days, respectively, or 58.5% and 85.4% longer than the shortest possible duration.

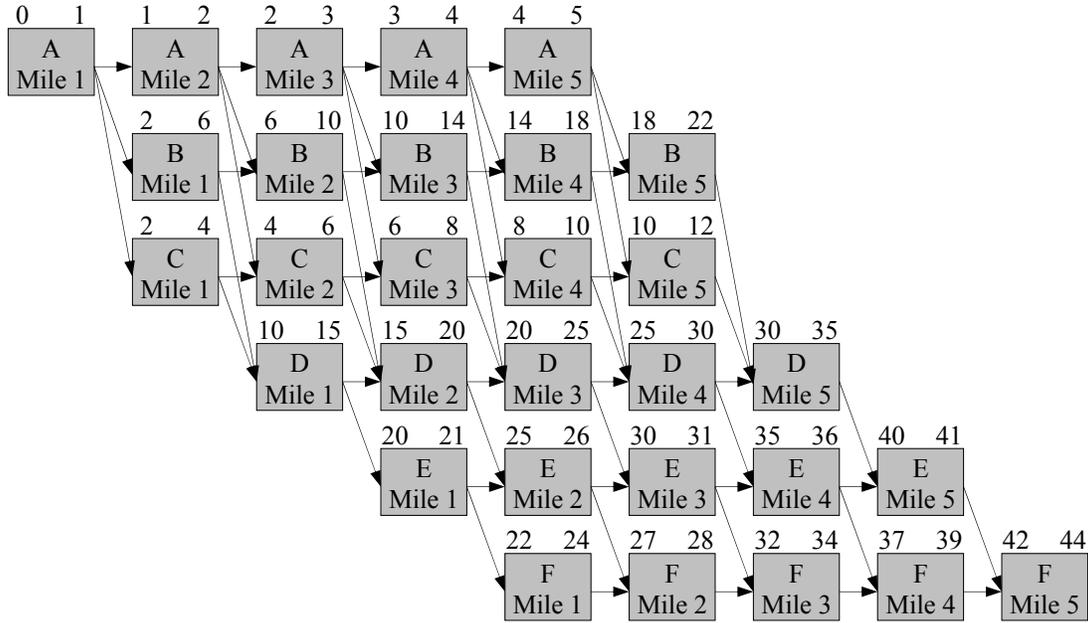


Figure 3: Repetitive activity-on-node network

Table 5: Critical path method results with leads and different relationship types

Name	Description	ES	EF	LS	LF	TF	FF
A	Locate & clear	0	5	0	5	0	0
B	Excavate	1	21	1	21	0	0
C	String pipe	1	11	18	28	17	17
D	Lay pipe	5	30	5	30	0	0
E	Test	30	35	30	35	0	0
F	Backfill	31	41	31	41	0	0

* Boldface activities are on the critical path.

However, this method still does not allow any examination of the location aspect of the schedule, whose underlying roadway project consists of five stages. Modeling the entire project as a more detailed schedule than the one shown in Figures 1 and 2 requires that all activities are split into the individual stages as shown in Figure 3 as originally presented by Reda (1990). The earliest start and finish times are listed above the activities. Considering the lead times for each activity as per Table 1 and the unit durations per stage, the total project duration is calculated to be 44 days, while the original analysis by Reda (1990) had omitted these time buffers and determined the project duration to be only 33 days. “However, using the CPM calculations the continuity of work for the crews... [is] not maintained” (Reda 1990, p. 329). Activity *E* is suffering four interruptions of four days each and activity *F* is suffering four interruptions of three days each. The stage buffers for activities *B* and *C* of one stage each have been converted based on their productivities into the previously included lead times of four and two days, respectively.

3.2 Linear Schedule Analysis

The following sections contain the analysis of this example with the new method of linear scheduling using singularity functions. This author suggests using the name *Productivity Scheduling Method* (PSM) for the new approach due to its strong reliance on describing the productivity of each activity mathematically with Macaulay brackets. Table 6 lists the activities in the new convention, where *DT* is the distance across time, *DA* is the distance across the amount (the unit is stages or miles in this case), *BT* is the time buffer, and *BA* is the amount buffer.

Table 6: Activity list for linear scheduling

Name	Description	Successor	DT	DA	BT	BA
A	Locate & clear	B, C	5	5	1	-
B	Excavate	D	20	5	4	1
C	String pipe	D	10	5	2	1
D	Lay pipe	E	25	5	5	-
E	Test	F	5	5	1	-
F	Backfill	-	10	5	-	-

3.2.1 Activities

The following (4) through (9) contain the initial set of equations describing each activity before considering any time buffers or the dependency structure, thus all activities start at zero days. The constant value before the Macaulay bracket of order zero is the intercept for the start time for an activity. The fraction before the Macau-

lay bracket of order one is the slope of the activity, i.e. its productivity, expressed as rise over run, i.e. the days y per stages x . The cutoff value inside the Macaulay bracket expresses when the particular term becomes valid; in this case all activities are starting at zero stages.

$$y(x)_A = 0d \cdot \langle x - 0st \rangle^0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^1 \quad (4)$$

$$y(x)_B = 0d \cdot \langle x - 0st \rangle^0 + \frac{4d}{1st} \cdot \langle x - 0st \rangle^1 \quad (5)$$

$$y(x)_C = 0d \cdot \langle x - 0st \rangle^0 + \frac{2d}{1st} \cdot \langle x - 0st \rangle^1 \quad (6)$$

$$y(x)_D = 0d \cdot \langle x - 0st \rangle^0 + \frac{5d}{1st} \cdot \langle x - 0st \rangle^1 \quad (7)$$

$$y(x)_E = 0d \cdot \langle x - 0st \rangle^0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^1 \quad (8)$$

$$y(x)_F = 0d \cdot \langle x - 0st \rangle^0 + \frac{2d}{1st} \cdot \langle x - 0st \rangle^1 \quad (9)$$

The advantage of this new method is that start times, productivities, and time buffers are kept separate, yet are contained in a very clear manner in one integrated system of equations for further analysis. Finish times are obtained by evaluating the equations for $x = 5$. With the flexible singularity functions using the Macaulay bracket notation, even more complicated activities can be easily expressed. If such activity contains several segments of different productivities, even of higher order behaviors, its singularity function would simply consist of the adequate number of additive terms preceded by their respective slope changes.

3.2.2 Time Buffers

The next step examines the time buffers for each activity. For brevity, the equations of the time buffer lines are omitted here, as they are identical to (4) through (8) plus the respective valued of the time buffer from Table 6 in the intercept before the Macaulay bracket of order zero.

3.2.3 Initial Configuration

The next step is to tentatively stack up all activities in their order as given in the activity list of Table 1. Stacking up means that the finish of a predecessor will become the start of a successor, which is identical to the CPM algorithm, yet is carried out on the singularity functions. Stacking up ensures that activities are positioned for the following step of collapsing the schedule to the minimum possible project duration. Graphically this intermediate step would resemble the popular computer game Tetris, where geometric shapes are stacked upon each other.

$$y(x)_A = 0d \cdot \langle x - 0st \rangle^0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^1 \quad (10)$$

Activity A in (10) remains unchanged from (4), as it is the first activity in the schedule. Its value for $x = 5$ plus the time buffer of 1 is $y(x = 5)_{A \text{ time buffer}} = 6$. This value is the new start time of its successor activities B and C .

$$y(x)_B = 6d \cdot \langle x - 0st \rangle^0 + \frac{4d}{1st} \cdot \langle x - 0st \rangle^1 \quad (11)$$

$$y(x)_C = 6d \cdot \langle x - 0st \rangle^0 + \frac{2d}{1st} \cdot \langle x - 0st \rangle^1 \quad (12)$$

Activities B and C are stacked in (11) and (12). Their values for $x = 5$ plus time buffers of 4 or 2, respectively, are $y(x = 5)_{B \text{ time buffer}} = 30$ and $y(x = 5)_{C \text{ time buffer}} = 18$. The maximum of these two values, in analogy to CPM, is used as the start time of their successor activity D .

$$y(x)_D = 30d \cdot \langle x - 0st \rangle^0 + \frac{5d}{1st} \cdot \langle x - 0st \rangle^1 \quad (13)$$

Activity D is stacked in (13). Its value for $x = 5$ plus time buffer of 5 is $y(x = 5)_{D \text{ time buffer}} = 60$. This value is used as the start time of its successor activity E .

$$y(x)_E = 60d \cdot \langle x - 0st \rangle^0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^1 \quad (14)$$

Activity E is stacked in (14). Its value for $x = 5$ plus time buffer of 1 is $y(x = 5)_{E \text{ time buffer}} = 66$. This value is used as the start time of its successor activity F .

$$y(x)_F = 66d \cdot \langle x - 0st \rangle^0 + \frac{2d}{1st} \cdot \langle x - 0st \rangle^1 \quad (15)$$

Activity F is stacked in (15). Its value for $x = 5$ is $y(x = 5)_{F \text{ time buffer}} = 76$, which is the tentative project duration. These results are identical to the CPM calculations of Table 3, as they were obtained using the same assumptions. However, the new analysis method now continues with the optimization as explained in the following section.

3.2.4 Duration Minimization

The next step is to take the pairwise differences along each link in the schedule network between the time buffers that follow predecessor activities and the successor activities in (16) through (21). It is assumed that time buffers belong to the predecessor; the opposite case could also be considered equivalently. Note that terms in singularity functions can only be added or subtracted if the order and cutoff value of their Macaulay bracket are identical.

$$y(x)_{B - A \text{ time buffer}} = 5d \cdot \langle x - 0st \rangle^0 + \frac{3d}{1st} \cdot \langle x - 0st \rangle^1 \quad (16)$$

$$y(x)_{C - A \text{ time buffer}} = 5d \cdot \langle x - 0st \rangle^0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^1 \quad (17)$$

$$y(x)_{D-B \text{ time buffer}} = 26d \cdot \langle x - 0st \rangle^0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^1 \quad (18)$$

$$y(x)_{D-C \text{ time buffer}} = 28d \cdot \langle x - 0st \rangle^0 + \frac{3d}{1st} \cdot \langle x - 0st \rangle^1 \quad (19)$$

$$y(x)_{E-D \text{ time buffer}} = 55d \cdot \langle x - 0st \rangle^0 - \frac{4d}{1st} \cdot \langle x - 0st \rangle^1 \quad (20)$$

$$y(x)_{F-E \text{ time buffer}} = 65d \cdot \langle x - 0st \rangle^0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^1 \quad (21)$$

The next step is to differentiate the distances in (16) through (21) to get the derivatives (22) through (27). The derivatives are then evaluated across x to find where $y(x)$ yields its minimum value. In other words, the location where two activities are closest to each other (via the time buffer, if existing) is calculated. Note that this approach will also treat more complicated activities correctly.

$$y(x)'_{B-A \text{ time buffer}} = 0 + \frac{3d}{1st} \cdot \langle x - 0st \rangle^0 \quad (22)$$

Evaluating (22) yields the minimum distance across time between B and the A time buffer to occur at $x = 0$.

$$y(x)'_{C-A \text{ time buffer}} = 0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^0 \quad (23)$$

Evaluating (23) yields the minimum distance across time between C and the A time buffer to occur at $x = 0$.

$$y(x)'_{D-B \text{ time buffer}} = 0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^0 \quad (24)$$

Evaluating (24) yields the minimum distance across time between D and the B time buffer to occur at $x = 0$.

$$y(x)'_{D-C \text{ time buffer}} = 0 + \frac{3d}{1st} \cdot \langle x - 0st \rangle^0 \quad (25)$$

Evaluating (25) yields the minimum distance across time between D and the C time buffer to occur at $x = 0$.

$$y(x)'_{E-D \text{ time buffer}} = 0 - \frac{4d}{1st} \cdot \langle x - 0st \rangle^0 \quad (26)$$

Evaluating (26) yields the minimum distance across time between E and the D time buffer to occur at $x = 5$.

$$y(x)'_{F-E \text{ time buffer}} = 0 + \frac{1d}{1st} \cdot \langle x - 0 \rangle^0 \quad (27)$$

Evaluating (27) yields the minimum distance across time between F and the E time buffer to occur at $x = 0$. These points are important points for the analysis, as they determine the equivalent of a critical path through the linear schedule. Since the location of these points is actually calculated rather than implicated graphically as for the vertices of Harmelink and Rowings (1998), they shall be called critical points. In the case of activities of constant productivity, they can only be starts or finishes. For more complicated activities any convex peak or concave dip in

the activity line could also be a potential critical point. The start, finish, or intermediate value of a predecessor activity is passed on at critical points (via the time buffer, if existing) to become the corresponding value of the successor activity. The final set of singularity functions for the linear schedule is given by (28) through (33).

$$y(x)_A = 0d \cdot \langle x - 0st \rangle^0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^1 \quad (28)$$

The A time buffer yields the starts of (29) and (30).

$$y(x)_B = 1d \cdot \langle x - 0st \rangle^0 + \frac{4d}{1st} \cdot \langle x - 0st \rangle^1 \quad (29)$$

$$y(x)_C = 1d \cdot \langle x - 0st \rangle^0 + \frac{2d}{1st} \cdot \langle x - 0st \rangle^1 \quad (30)$$

The B and C amount buffers of Table 6 are fulfilled.

$$y(x)_D = 5d \cdot \langle x - 0st \rangle^0 + \frac{5d}{1st} \cdot \langle x - 0st \rangle^1 \quad (31)$$

The maximum value of B and C at $x = 0$ plus their respective time buffers is used as the start for activity D .

$$y(x)_E = 30d \cdot \langle x - 0st \rangle^0 + \frac{1d}{1st} \cdot \langle x - 0st \rangle^1 \quad (32)$$

The maximum value of D at $x = 5$ plus its time buffer of $y(x = 5)_{D \text{ time buffer}} = 35$ is used as the finish for activity E . The start of D is obtained by deducting its duration.

$$y(x)_F = 31d \cdot \langle x - 0st \rangle^0 + \frac{2d}{1st} \cdot \langle x - 0st \rangle^1 \quad (33)$$

The maximum value of E at $x = 0$ plus its time buffers is used as the start for activity F . Table 7 lists all activity and buffer starts and finishes as shown in Figure 4, where P is the productivity as amount divided by time, ST is the start time, FT is the finish time, SA is the start amount, and FA is the finish amount. Note that the B and C time buffers overlap for these concurrent activities.

Table 7: Calculation results

Name	Description	P	ST	FT	SA	FA
A	Locate & clear	1.00	0	5	0	5
A	Time buffer	1.00	1	6	0	5
B	Excavate	0.25	1	21	0	5
B	Time buffer	0.25	5	25	0	5
C	String pipe	0.50	1	11	0	5
C	Time buffer	0.50	3	13	0	5
D	Lay pipe	0.20	5	30	0	5
D	Time buffer	0.20	10	35	0	5
E	Test	1.00	30	35	0	5
E	Time buffer	1.00	31	36	0	5
F	Backfill	0.50	31	41	0	5

The total project duration is calculated as 41 days using singularity functions, while the CPM analysis of Fig-

ure 1 gives a minimum of 76 days, Figure 2 gives a minimum of 41 days, and Figure 3 gives a minimum of 44 days. The linear schedule offers the major advantage of not having to split activities for more detailed calculations at the expense of causing them to become interruptible, as the singularity function for the activity of interest can be evaluated for any value of x . The linear schedule only comprises of continuous activities that may occur concurrently without any interruptions, which is desirable for a smooth workflow. On the other hand, the CPM calculations had yielded a schedule with a very unsteady execution of the last two activities E and F , who each incurred several significant interruptions due to the policy of “start activities as early as possible based on their starts” under CPM. The computational effort of the detailed CPM analysis for the complete network with repetitive activities at all stages was higher, as it requires analysis of the starts and finishes for 30 activities and their multiple dependencies instead of a system of only six singularity functions that span across the range of locations. Figure 4 shows the final linear schedule with its critical path.

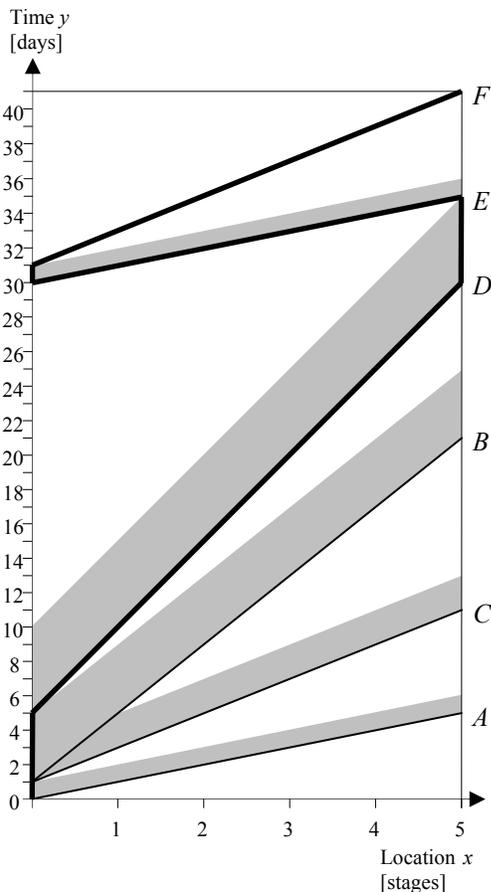


Figure 4: Linear schedule

The initial CPM analysis had indicated activities A , B , D , E , and F all to be critical. The more in-depth analysis shows that in fact only the start of activity A is critical, connecting through the A time buffer to the starts of B and C , continuing through D entirely, via the D time buffer through E , and finally via the E time buffer through F .

The absolute minimum total project duration that is theoretically possible would be achieved if resources were added to bring slower activities to the productivity of the fastest activities A and E , so that all activity lines in Figure 4 would be parallel. In that case the project would have a duration of only 16 days, which would be equal to the duration of the first activity A plus the time buffers along the longest path through the schedule network. Note that simpler than CPM, the activity stacking under the new method described in this paper requires only to specify predecessors and successors, while the specific types of relationships between their starts and finished would be created automatically in the duration minimization step.

4 CONCLUSIONS

This paper has demonstrated the application of a new analysis method for linear schedules using singularity functions. It is intuitively based on standard geometry and algebra and can be performed manually. Its mathematical formulation is flexible and performs as well as traditional CPM, yet it allows a deeper 2D analysis of time and location with fewer computations, can accommodate activities of any complexity, and enhances the existing graphical linear schedules. Future research will expand the analytical capabilities of the new method toward float and resource leveling, and will fully computerize the method.

ACKNOWLEDGEMENT

The support of the National Science Foundation (Grant CMMI-0654318) for portions of the work presented here is gratefully acknowledged. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- Al Sarraj, Z. M. 1990. Formal development of line-of-balance technique. *Journal of Construction Engineering and Management* 116(4): 689-704.
- Anton, G. A., G. Lucko, and A. K. Duzkale. 2007. Optimizing the graphical arrangement of network construction schedules. To appear in *Proceedings of the Construction Research Congress*, eds. P. S. Chinowsky, A. D. Songer, P. M. Carrillo. Reston, Virginia: American Society of Civil Engineers.
- Arditi, D., and M. Z. Albulak. 1986. Line-of-balance scheduling in pavement construction. *Journal of*

- Construction Engineering and Management* 112(3): 411-424.
- Arditi, D., O. B. Tokdemir, and K. Suh. 2001. Effect of learning on line-of-balance scheduling. *International Journal of Project Management* 19(5): 265-277.
- Arditi, D., O. B. Tokdemir, and K. Suh. 2002. Challenges in line-of-balance scheduling. *Journal of Construction Engineering and Management* 128(6): 545-556.
- Birrell, G. S. 1980. Construction planning – beyond the critical path. *Journal of the Construction Division, Proceedings of the American Society of Civil Engineers* 106(CO3): 389-407.
- Chrzanowski, E. N., and D. W. Johnston. 1986. Application of linear scheduling. *Journal of Construction Engineering and Management* 112(4): 476-491.
- Clough, R. H., and G. A. Sears. 1979. *Construction project management*. 2nd ed., New York, New York: John Wiley & Sons.
- El-Rayes, K., and O. Moselhi. 1998. Resource-driven scheduling of repetitive activities. *Construction Management and Economics* 16(4): 433-446.
- Föppl, A. O. 1927. *Vorlesungen über Technische Mechanik. Dritter Band: Festigkeitslehre*. 10th ed., Leipzig, Germany : B. G. Teubner.
- Galloway, P. D. 2006. Survey of the Construction Industry Relative to the Use of CPM Scheduling for Construction Projects. *Journal of Construction Engineering and Management* 132(7): 697-711.
- Glenwright, E. T. 2004. Let's scrap the precedence diagramming method. *Transactions of the Association for the Advancement of Cost Engineering Annual Meeting*: PS.08.1-PS.08.6.
- Handa, V. K., and R. M. Barcia. 1986. Linear scheduling using optimal control theory. *Journal of Construction Engineering and Management* 112(3): 387-393.
- Harmelink, D. J., and J. E. Rowings. 1998. Linear scheduling model: Development of controlling activity path. *Journal of Construction Engineering and Management* 124(4): 263-268.
- Harris, R. B., and P. G. Ioannou. 1998. Scheduling projects with repeating activities. *Journal of Construction Engineering and Management* 124(4): 269-278.
- Jaafari, A. 1984. Criticism of CPM for project planning analysis. *Journal of Construction Engineering and Management* 110(2): 222-233.
- Kelley, J. E. 1957. Computers and operations research in road building. *Operations Research* January/February:58-63.
- Kelley, J. E., and M. R. Walker. 1959. Critical path planning and scheduling. *Proceedings of the Eastern Joint Computer Conference* 16:160-173. New York, New York: National Joint Computer Committee, Association for Computing Machinery.
- Lucko, G. 2006. A Reconciliation between Two Competing Methods of CPM Calculations. *Proceedings of the 3rd Project Management Institute College of Scheduling Annual Conference*, Newtown Square, Pennsylvania: Project Management Institute.
- Macaulay, W. H. 1919. Note on the deflection of beams. *Messenger of Mathematics* 48(9): 129-130.
- Mattila, K. G., and A. Park. 2003. Comparison of linear scheduling model and repetitive scheduling method. *Journal of Construction Engineering and Management* 129(1): 56-64.
- Mubarak, S. 2005. *Construction project scheduling and control*. Upper Saddle River, New Jersey: Pearson Education / Prentice Hall.
- Reda, R. M. 1990. RPM: Repetitive project modeling. *Journal of Construction Engineering and Management* 116(2): 316-330.
- Russell, A. D., and W. F. Caselton. 1988. Extensions to linear scheduling optimization. *Journal of Construction Engineering and Management* 114(1): 36-52.
- Selinger, S. 1980. Construction planning for linear projects. *Journal of the Construction Division, Proceedings of the American Society of Civil Engineers* 106(CO2): 195-205.
- Stradal, O., and J. Cacha. 1982. Time space scheduling method. *Journal of Construction Engineering and Management* 108(CO3): 445-457.
- Thabet, W. Y., and Y. J. Beliveau. 1994. HVLS: Horizontal and vertical logic scheduling for multistory projects. *Journal of Construction Engineering and Management* 120(4): 875-892.
- Vorster, M. C., Y. C. Beliveau, and T. Bafna. 1992. Linear scheduling and visualization. *Transportation Research Record* 1351: 32-39.
- Wittrick, W. H. 1965. A generalization of Macaulay's method with applications in structural mechanics. *AIAA Journal* 32: 326-330.

AUTHOR BIOGRAPHY

GUNNAR LUCKO is Assistant Professor of Civil Engineering and Director of the Construction Engineering and Management Program in the Department of Civil Engineering at The Catholic University of America. He holds a German Diploma in Civil and Environmental Engineering from Hamburg University of Technology and an M.S. and Ph.D. from Virginia Polytechnic Institute and State University (Virginia Tech). His research interests include mathematical representation, simulation, and analysis of schedule networks, construction equipment operations and economics, optimization methods, constructability analysis, and engineering education. He has studied statistical equipment valuation models and has participated in research for the Construction Industry Institute, and is currently working on research funded by the National Science Foundation. He is a member of ASCE and INFORMS. His e-mail address is <lucko@cua.edu> and his web address is <<http://engineering.cua.edu/faculty/profiles/lucko.cfm>>.