MATHEMATICAL ANALYSIS OF LINEAR SCHEDULES

Gunnar Lucko

ABSTRACT
The interplay of various managerial dimensions of construction projects requires careful control. Some criticisms have been raised regarding the suitability of the critical path method to certain types of projects and the depth of analysis possible with its informational content. Its graphical representation suffers from one-dimensionality and from its semi-time scaled nature. The linear scheduling method (LSM) is ideally suited to projects with one predominant geometrical dimension, e.g. road construction, high-rise buildings, and repetitive operations. Slopes in its graphical representation are proportional to productivity. However, LSM is largely graphical and lacks a solid underlying mathematical model. This paper introduces a novel method that uses singularity functions similar to structural engineering applications. It is developed to provide a complete analysis with capabilities akin to and beyond traditional CPM. The new method is independent of its graphical representation, easily programmable, and remains intuitive enough for a human user to verify its results.

KEYWORDS: Scheduling, critical path method, definitions, symbols, network analysis, geometry, linear analysis, time dependence, location

INTRODUCTION
Construction project managers are tasked with controlling and steering projects across their manifold dimensions. Important dimensions of projects are time, cost, and their 3D scope and the operations to achieve it. A significant body of knowledge exists with respect to tools that allow project managers to model and control the time dimension of their projects, which in its entirety is called scheduling.

Among the graphical representations of schedules are bar charts with logic links, including fenced bar charts (Melin and Whiteaker 1981); time-scaled network diagrams, also known as line schedules; activity-on-the-node (AON) network diagrams (also called precedence diagramming method) and their nowadays less common counterpart activity-on-the-arrow diagrams (also called arrow diagramming method or i-j diagrams for naming the nodes); and finally linear scheduling method (LSM) diagrams derived from the line-of-balance method of production control.

Analyzing these types shows that except for LSM they feature only one predominant type of information, time, on one major diagram axis. Only linear schedules by definition have two equally important dimensions of information, time and amount.

LITERATURE REVIEW
Considering this increased inherent richness of information, the full potential of LSM over CPM has been recognized: “[CPM networks] represent the most sophisticated technique, with floats, deadlines, use of resources, and computer programs with a broad choice of outputs, allowing some optimization” (Stradal and Cacha 1982, p446). “An awareness that the traditional
network is not the most adequate tool for the planning of linear projects had led to a surge of techniques to handle such jobs in the recent past” (Handa and Barcia 1986, p387).

Linear schedules can be applied to any project that has a predominant geometric dimension. Examples in the literature apply LSM to horizontally longitudinal projects, e.g. roads, tunnels, and pipelines whose diagrams often feature a vertical time axis over a horizontal amount axis and profile (Stradal and Cacha 1982, Arditi et al. 2002). Second, examples are found for horizontally longitudinal projects, e.g. high-rise buildings (O’Brien 1975, Thabet and Beliveau 1994). Third, examples of linear schedules exist for repetitive construction operations, e.g. work crews painting numerous identical apartments, where balancing crew assignments and learning effects are possible and would be visible as parallel or curved lines in the LSM diagram, respectively (Handa and Barcia 1986, Hegazy et al. 1993, El-Rayes and Moselhi 1998, Arditi et al. 2001). The graphical representation of LSM displays amount and time, i.e. each line shows productivity for one activity – a steep slope indicates high and a gentle slope indicates low productivity. Slicing across time or amount shows when the workface is located where on the project.

NEED FOR ANALYTICAL METHODS FOR LINEAR SCHEDULES

A case study showed that LSM gives results of similar quality as CPM, but its authors concluded to use it only as an addition to CPM: “Finally, LSM is essentially graphical; it cannot be adapted to numerical computerization as readily as network methods. However, CAD drafting systems could be programmed to facilitate preparation and revision” (Chrzansowski and Johnston 1986, p. 490). Others concur: “Disadvantages include difficulties in scale on the y-axis when the segments have different measures (e.g. apartment house and sewage construction in one plan) and it is difficult for computer calculations” (Stradal and Cacha (1982, p. 446). Several pointed at the need for computerization (Arditi et al. 2002, Mattila and Park 2003).

Considering the inherent versatility and richness of information that LSM can communicate, it is also astonishing that typically very little time is spent on teaching it in higher education in construction engineering and management. Many textbooks focus primarily on network schedules, either omitting linear schedules entirely (Clough and Sears 1991, Halpin and Woodhead 1998, Gould and Joyce 2003, Patrick 2004, Buttelwerth 2005) or spending significantly less text describing them (Callahan et al. 1992, Gould 2002, Schexnayder and Mayo 2004, Mubarak 2005), typically only very few pages but at most one chapter in the entire book (Hegazy 2002). Discussions with construction managers support the view of linear schedules being treated second-rate in practice. Major project management software packages currently used in the construction industry offer only bar charts (with logic links not displayed by default) and AON network diagrams as the graphical representation of schedules.

MACAULAY BRACKET NOTATION OF SINGULARITY FUNCTIONS

A typical problem in structural engineering analysis is a horizontal beam with different types of loads at different locations or ranges across its length. An elegant and flexible method to analyze such system, rather than mathematically cutting each segment of the beam free and solving it via a system of equations with boundary conditions, was developed by Macaulay (1919) using singularity functions. Their classic definition, symbolically written with pointed brackets that were introduced by Wittrick (1965) in a generalization of Macaulay’s approach, is given by Equation 1.
\[ (x-a)^n = \begin{cases} 0 & \text{for } x < a \\ (x-a)^n & \text{for } x \geq a \end{cases} \quad \text{Eq. 1} \]

where \( x \) is the variable under consideration plotted along an axis, \( a \) is the upper boundary of the current segment measured along said axis, i.e. its length, and \( n \) is the order of the phenomenon that changes after the end of the current segment. It is possible to differentiate and integrate the brackets using standard mathematical rules.

\[ \frac{d}{dx}(x-a)^n = n \cdot (x-a)^{n-1} \quad \text{Eq. 2} \]

\[ \int (x-a)^n \, dx = \frac{1}{n+1} \cdot (x-a)^{n+1} + C \quad \text{Eq. 3} \]

where \( C \) is an integration constant. Singularity functions can be added and subtracted as long as the particular segment of length \( a \) and the exponent \( n \) of the expression are identical. This author proposes to call the use of singularity functions for analysis of linear schedules the productivity scheduling method (PSM).

LINEAR SCHEDULE EXAMPLE

Consider now a small linear schedule as shown in Figure 1, where two activities \( A \) and \( B \) are performed in the sequence \{\( A, B \}\}. Activity \( A \) takes 3 days to complete the 6 units and activity \( B \) takes 3 days. The amount that needs to be accomplished is plotted along the x-axis and the time is plotted along the y-axis. Activity \( A \) consists of the segments \( A_1 \) and \( A_2 \), between which there is a change in slope from a higher to a lower productivity. Activity \( B \) consists of the segments \( B_1, B_2, \) and \( B_3 \), between which there is a change in slope from a lower to a higher and back to a lower productivity. The exact numeric parameters for this linear schedule are provided in Table 1.

<table>
<thead>
<tr>
<th>Name</th>
<th>Sequence</th>
<th>Segment</th>
<th>DT</th>
<th>DA</th>
<th>P</th>
<th>BT</th>
<th>BA</th>
<th>ST</th>
<th>FT</th>
<th>SA</th>
<th>FA</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>A_1</td>
<td>1</td>
<td>3</td>
<td>3.0</td>
<td>1.67</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A_2</td>
<td>2</td>
<td>3</td>
<td>1.5</td>
<td>1.67</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
<td>B_1</td>
<td>1</td>
<td>2</td>
<td>2.0</td>
<td>N/A</td>
<td>N/A</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>B_2</td>
<td>1</td>
<td>3</td>
<td>3.0</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>B_3</td>
<td>1</td>
<td>1</td>
<td>1.0</td>
<td>N/A</td>
<td>N/A</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>

Table 1: Activity List

The activity sequence describes the activities and their succession without inflicting any particular intricate schedule logic that would be necessary for network schedules under CPM. The number of segments of each activity is listed as a reminder. \( DT \) is the time distance in days and \( DA \) is the amount distance in units that each segment spans. \( P \) is the activity slope, i.e. the productivity calculated by dividing the amount by the time. \( BT \) and \( BA \) describe any time or amount buffers following the execution of each activity or segment thereof that must be completed before the successor activity or any segment thereof can commence. Missing from Figure 1 is the buffer that causes activities \( A \) and \( B \) to not touch directly, but rather to be apart from each other. This buffer may be either a time or an amount buffer; the first one of which is examined in the following sections, the latter one can be derived analogously. Buffers shall be shown by gray shaded areas above the activities to which they are assigned.
Activity Stacking across Time by Activity Sequence

Figure 2 includes the time buffer and shows how the activities stack across time. It is assumed that the time buffer constraint $BT_A = 1.667$ days is assigned to activity $A$. Other constraints, e.g. desired start or finish dates, can be expressed as mathematical conditions using the Macaulay bracket notation. The mechanistic activity stacking that is similar to the game Tetris must now be converted into a mathematical description. First, all activities are written as singularity functions with Macaulay brackets without considering the activity sequence or any time and amount buffers. In other words, the origin of the time and amount axes in the coordinate system is located in their overall start point. The time coordinate of the overall finish point of the first activity in the activity sequence is determined and the time buffer is added to it. This time coordinate is taken as the overall start point for the next activity in the activity sequence. This yields an initial zigzag stacked configuration. The difference is taken between the equations for the two activities (including the buffer time of the predecessor) and is differentiated to find the location of the minimum time difference. The point where it occurs determines how far the next activity could be mechanically pushed down along the time axis until it touches the buffer time above the first activity. Finally, the equations are rewritten with their final location and their start and finish coordinates are recorded in the activity list of Table 1.

Note that this description of activity stacking assumes that the time and amount buffers are constant. For varying time and amount buffers the procedure would have to be modified to also express them as Macaulay brackets, add them to the initial activity, take the differences with the neighboring activity, and differentiate these differences to find the respective minimum time and amount differences. The two neighboring activities $A$ and $B$ are now described in Equations 4 and 5. Note that for clarity the following conventions are used beyond the original notation:

- Brackets are sorted by ascending boundaries and by ascending exponents;
- slopes and points are written as fractions and numbers before the brackets;
- extra terms at the end of the equations, sorted by descending exponents, bring their curve back to zero to keep the calculation correct at the end of the activity.

$$y(x) = 0 \cdot \langle x - 0 \rangle^0 + \frac{1}{3} \cdot \langle x - 0 \rangle^1 + \frac{1}{3} \cdot \langle x - 3 \rangle^1 - \frac{2}{3} \cdot \langle x - 6 \rangle^1 - 3 \cdot \langle x - 6 \rangle^0$$  \hspace{1cm} \text{Eq. 4}
The terms with an exponent of 0 are the intercepts of the activities with the time axis and the terms with an exponent of 1 indicate activity segments with a non-zero slope. Note that in this diagram of time over amount, positive terms decrease the activity slope from the previous to the current segment and negative terms increase the slope, as productivity is defined by the inverse of the displayed axes, amount over time. While this configuration is often used in the literature, this author recommends to use the configuration showing amount over time to remain consistent with the intuitive definition of productivity. The example in this paper will be continued with time over amount. Results from either graphical configuration are mathematically equivalent.

**Minimum Time Distance Analysis**

Now the time buffer $BT_A = 1.667$ is considered. It is added to Equation 4 to yield the initial Equation 6 and is reflected in its superscript. This is the upper boundary of the gray shaded area above activity $A$, which represents the earliest possible times from which activity $B$ could commence. Evaluating Equation 6 for the highest value of $x = 6$ yields $y(x = 6) = 3$. Adding the time buffer $BT_A = 1.667$ yields a tentative overall start time coordinate of 4.667 for the activity $B$ in Equation 7.

$$y(x)_A = 0 \cdot (x-0)^0 + \frac{1}{2} \cdot (x-0)^1 - \frac{1}{6} \cdot (x-2)^1 + \frac{2}{3} \cdot (x-5)^1 - \frac{3}{3} \cdot (x-6)^1 - 3 \cdot (x-6)^0 \quad \text{Eq. 5}$$

$$y(x)_B^{BT} = 1.667 \cdot (x-0)^0 + \frac{1}{3} \cdot (x-0)^1 - \frac{1}{3} \cdot (x-3)^1 - \frac{2}{3} \cdot (x-6)^1 - 4.667 \cdot (x-6)^0 \quad \text{Eq. 6}$$

$$y(x)_B^{BT} = 4.667 \cdot (x-0)^0 + \frac{1}{2} \cdot (x-0)^1 - \frac{1}{6} \cdot (x-2)^1 + \frac{2}{3} \cdot (x-5)^1 - \frac{3}{3} \cdot (x-6)^1 - 7.667 \cdot (x-6)^0 \quad \text{Eq. 7}$$

This approach is always on the safe side, as it keeps activity $B$ floating high above activity $A$ without interference. It is now of interest where the two activity curves are closest to each other in time under consideration of the $BT_A$ constraint. Notation of singularity functions with Macaulay brackets makes determining the exact coordinates of such point, called vertex (Harmelink and Rowings 1998), an easy subtraction of Equation 7 minus Equation 6 to yield Equation 8. Note that the function operator for taking the difference is marked with an asterisk (*). Figure 3 shows the difference $B-A$ with the vertex clearly marked by a small circle.

$$y^*(x)_B - y^*(x)_A = y(x)_B^{BT} - y(x)_A^{BT} \quad \text{Eq. 8}$$

Finding where this equation has a minimum requires differentiation by applying Equation 2 to Equation 8 to yield Equation 9. Any minimum is indicated by a slope change from negative to positive (i.e., a concave point or “dent”). Figure 4 shows the slope of the difference $y^*(x)_B - y^*(x)_A$.

$$y^*(x)_B - y^*(x)_A = \frac{1}{6} \cdot (x-0)^0 - \frac{1}{6} \cdot (x-2)^0 - \frac{1}{3} \cdot (x-3)^0 + \frac{2}{3} \cdot (x-5)^0 - \frac{1}{3} \cdot (x-6)^0 + 0 \quad \text{Eq. 9}$$

Taking the second derivative of Equation 8 yields Equation 10, but does not yield any new information, since the entire Equation 10 is already of order zero.

$$y^*(x)_B - y^*(x)_A = 0 \quad \text{Eq. 10}$$
Theoretically the derivative should be evaluated for every value of $x$ to find where the value of the function changes from negative to positive. Fortunately, Macaulay brackets make this process easy – it suffices to calculate the value for each segment in Equation 9, i.e. simply sum up the factors before the equation and record when a change from negative to positive occurs. In our example, the only such change occurs at the amount $x = 5$, which is confirmed as the vertex location in Figure 3. The mathematical algorithm for PSM is therefore independent of the graphical representation of the linear schedule and its derivatives.

Figure 3: Activity Time Difference and Time Buffer

Adding $BT_A = 1.667$ to activity $A$ at $x = 5$ yields a value of $y = 4$ for activity $B$ in Equation 11. This leads to the solution $z = 2$ in Equation 12 and to the final equation for activity $B$ in Equation 13. Note that the normal exponential rule $a^0 = 1$ applies and that all terms at or beyond $x = 5$ are falling out of the calculation. Figure 5 shows the linear schedule with time buffer after activity stacking.

$$4 = z \cdot (5-0)^0 + \frac{1}{2} \cdot (5-0)^1 - \frac{1}{6} \cdot (5-2)^1 + \frac{2}{3} \cdot (5-5)^1 - \frac{3}{3} \cdot (5-5)^1 - (z+3) \cdot (5-5)^0 \quad \text{Eq. 11}$$

$$4 = z + 2.5 - 0.5 \quad \text{Eq. 12}$$

$$y(x)_{st} = 2 \cdot (x-0)^0 + \frac{1}{2} \cdot (x-0)^1 - \frac{1}{6} \cdot (x-2)^1 + \frac{2}{3} \cdot (x-5)^1 - \frac{3}{3} \cdot (x-6)^1 - 5 \cdot (x-6)^0 \quad \text{Eq. 13}$$

The start and finish time $ST$ and $FT$ and the start and finish amount $SA$ and $FA$ shown in Table 1 for the activities in the 2D time-amount coordinate system are obtained by evaluating Equations 4 and 13. These values can also be determined by adding the time and amount distances to the respective start values, in analogy to CPM calculations where the start time plus the duration is equal to the finish time.

**Definition of Coincident Ranges**

To aid the application of the time and amount buffers it is sensible to define a coincident time range $CT_{A,B}$ and a coincident amount range $CT_{A,B}$ between activities $A$ and $B$. Equations 14 and 15 test whether a coincident time or amount range exists between any two neighboring activities $i$ and $j$ or segments $k$ and $l$ thereof.

$$CT_{k(l),i(j)} = \text{not existent if } FT_{k(l)} < ST_{i(j)} \quad \text{Eq. 14}$$
\[ CA_{k(i),l(j)} = \text{not existent if } FA_{k(i)} < SA_{l(j)} \]  \hspace{1cm} \text{Eq. 15}

where \( CT \) is the coincident time range, \( CA \) is the coincident amount range, activity \( i \) precedes activity \( j \), and \( k \) and \( l \) are indices to number segments of activities. Equations 16 and 17 describe the ranges for all possible overlapping combinations of starts and finishes of the activities or their segments. Table 2 shows the coincident time and amount ranges for all segments \( A_k \) and \( B_l \) with their boundaries in swiveled brackets.

\[ CT_{k(i),l(j)} = \min \{ FT_{k(i)}, FT_{l(j)} \} \to \max \{ ST_{k(i)}, ST_{l(j)} \} \]  \hspace{1cm} \text{Eq. 16}

\[ CA_{k(i),l(j)} = \min \{ FA_{k(i)}, FA_{l(j)} \} \to \max \{ SA_{k(i)}, SA_{l(j)} \} \]  \hspace{1cm} \text{Eq. 17}

<table>
<thead>
<tr>
<th>Pair</th>
<th>CT</th>
<th>CA</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A_1,B_1 )</td>
<td>-</td>
<td>{0 - 2}</td>
</tr>
<tr>
<td>( A_1,B_2 )</td>
<td>-</td>
<td>{2 - 3}</td>
</tr>
<tr>
<td>( A_1,B_3 )</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>( A_2,B_1 )</td>
<td>{2 - 3}</td>
<td>-</td>
</tr>
<tr>
<td>( A_2,B_2 )</td>
<td>-</td>
<td>{3 - 5}</td>
</tr>
<tr>
<td>( A_2,B_3 )</td>
<td>-</td>
<td>{5 - 6}</td>
</tr>
</tbody>
</table>

**Figure 5: Time Critical Path**

**Table 2: Coincident Activity Table**

**Time Critical Path and Amount Critical Path**

It is now possible to construct for PSM the equivalent of the critical path under CPM. Shown as a thick line in Figure 5, the following rules apply for constructing it:

- The line overall follows the activity sequence \( \{A, B\} \);
- it runs through the entire linear schedule from earliest start to latest finish;
- if time buffers are considered, it jumps parallel to the time axis between two neighboring activities across their buffers where they are closest in time;
- if amount buffers are considered, it jumps parallel to the amount axis between two neighboring activities across their buffers where they are closest in amount.

Note that if no buffers exist the minimum distance will be zero and the two neighboring activities will touch directly. The analysis shows that critical points can be start, bend, and finish points of activities as well as points induced by these three types of points on neighboring activities. If the two activities \( A \) and \( B \) are diverging their start points are candidates, if they are parallel both their start and finish points are candidates, and if they are converging their finish points are candidates for being critical due to a minimum distance and indicate which initial or induced segments of activities in the linear schedule are critical due to time and amount buffers.

**CONCLUSIONS**

Moving beyond the 1D-nature of CPM network scheduling, LSM with its 2D information, time and amount, allows a deeper analysis of the interrelationship between the temporal and spatial aspects of all activities. Prior to this research, however, the purely graphical LSM was severely limited in its analytical capabilities.
This paper has presented a new method that mathematically models any linear schedule using singularity functions. Singularity functions have numerous mathematical advantages, including their flexible functional form that intuitively contains various characteristics of the shape of each activity and the ability to simply integrate and differentiate them. PSM can accommodate infinitely many activities with infinitely many segments each and yields precise analytical results about starts, finishes, the minimum possible project duration, and how activities might interfere with each other under consideration of time and amount buffers and other possible constraints. Since all concepts for applying singularity functions to LSM mostly require only basic algebra and geometry, the new method can be easily taught to college students and construction professionals and manual checking of calculations is possible for all but the most complex schedules.

Generalized equations for PSM and more advanced scheduling concepts, such as e.g. different types of float and allocating and leveling resources, are being developed under ongoing research at The Catholic University of America. It is hoped that PSM will lead to a strong revival of linear scheduling.

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


