SLIP CHART-INSPIRED PROJECT SCHEDULE DIAGRAMMING:

ORIGINS, BUFFERS, AND EXTENSION TO LINEAR SCHEDULES

By Gunnar Lucko, Ph.D. ¹, Lissa Gomes Araújo ², and Grant R. Cates, Ph.D. ³

¹ Associate Professor of Civil Engineering and Director of Construction Engineering and Management Program, Department of Civil Engineering, Catholic University of America, Washington, DC 20064, lucko@cua.edu.

² Undergraduate Research Assistant, Construction Engineering and Management Program, Department of Civil Engineering, Catholic University of America, Washington, DC 20064, ³gomesaraujo@cardinalmail.cua.edu, affiliated with Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES Foundation), Ministry of Education, CEP 70.040-020 – Brasília-DF, Brazil.

³ Former Flow Director – Columbia, Kennedy Space Center, now Senior Project Leader at The Aerospace Corporation, Exploration Systems Directorate, Eastern Range, FL 32925, grant.r.cates@aero.org.

Abstract

Linear schedules continue to hold significant potential for planning and control in construction project management, where work progress of activities is often repetitive in nature. However, linear schedules are limited in graphical comparisons. A ‘slip chart’ from the aerospace industry, which recorded launch dates on two axes of time, is rediscovered as inspiration for an improved tool. But whereas its original form could only track points in time, schedule activities must be represented as ranges across time. Additional theory is thus needed to extend slip charts toward
being able to handle linear schedules. To create such layout, this research systematically extends its conventions and capabilities. A conversion algorithm for linear schedules is derived. New features are introduced for a comprehensive approach. In particular, buffers that represent minimum and also maximum constraints between activities are investigated and converted. Criticality and float are defined.

Introduction

The fundamental importance of construction lies in being an enabler of modern society. It has a volume of $619.9 billion; with 3.7% of gross domestic product it is an essential sector among all U.S. industries (BEA 2014). Construction projects impact everyday life by providing all of the physical spaces and systems for functions of home, work, travel, etc. They are characterized by their significant size, high capital cost, and lengthy duration. As technology advance and the needs of the population grow, structures become increasingly complex, which in turn amplifies demand on the planning and control of projects. Goals for construction managers therefore are to complete their projects by creating efficient yet realistic schedules, spending money prudently, and economically using the many resources to realize the design intent without incidents; i.e. on time, within budget, to quality, and safely. Schedules can be seen as a basis for all other planning functions, because all events and processes have a time aspect. However, the various existing scheduling techniques are unequally well suited to communicate and compare their intricate data.

Literature Review

Shortcomings of Schedule Diagrams
Current construction scheduling practice is almost entirely based on the critical path method (CPM) of scheduling (Galloway 2006) that was created in 1956 by Kelley and Walker (1959 and 1989). It determines starts and finishes of activities in a network of discrete activities, which is visualized in form of a time-scaled bar chart (Gantt 1910) or a network-style precedence diagram (Fondahl 1962 and 1987). Many studies have explored meanings and uses of the complementary concepts of criticality—activities whose delay will impact successors or the project—versus float—activities that still have sufficient flexibility to absorb some delay (de la Garza et al. 2007).

*Bar charts* display relative durations of activities. Specifically, the length of bars is directly proportional to the duration of the activity that it represents, as measured over a time axis. Less directly, bar charts also indicate the sequence of activities and are often sorted to show earliest to latest activities from top to bottom on the vertical axis, giving a stepped appearance. They are simple and self-explanatory, but also static and fail to show dependency between activities unless arrows are inserted between them, which may lead to crowding (Korman and Daniels 2003). If bar charts are placed side-by-side or on top of each other, schedule updates do not stand out.

*Precedence diagrams* as used for CPM represent each activity as a rectangular box (rarely as a circle, Fondahl 1987), which is divided into sections that contain the value of points in time (start or finish) or ranges across time (duration, float). They are typically measured in work days and may be earliest or latest dates, i.e. minimum or maximum limits for the timing of an activity (because CPM assumes a best and worst case scenario). Boxes are connected by arrows that indicate the dependency structure of the network akin to a non-cyclic flowchart. Arrows may also carry a positive or negative duration to enforce a simple buffer between activity pairs. It is called lead or lag from a predecessor or successor view (Crandall 1973), but was also defined in another manner that a lag is “positive or zero” (Fondahl 1964, p. 64), which in practice is often
interpreted that negative times are leads, so that a successor overlaps its predecessor. A problem of precedence diagrams is that arrows frequently become entangled (Mubarak 2005, Russell and Udaipurwala 2003). Sometimes, they are fused to lighten a chart (Lucko 2005), but may have the opposite effect of reducing clarity. While it was advised to keep arrows short, direct, with few bends, and few crossings, no algorithm exists that could generate an optimal arrangement of the network to minimize the lengths, crossings, and bends (Purchase 2002). In fact, this problem was proven to be NP-complete, i.e. the runtime of algorithms are nondeterministic polynomially related to the number of variables (Garey and Johnson 1983). It therefore becomes extremely complex, if not impossible, to generate clear networks of large schedules. In part due to such issues, delays are difficult to extract, because the network does not explicitly represent them and remains fixed. Only single cell values change, which is less than satisfactory to report delays in a manner that reflects their importance. Comprehensive comparisons between schedule updates are difficult, because dates are merely shown as numbers on the corners of boxes. This problem also affects distinguishing critical from non-critical activities, as networks show the dependency, but bar charts show durations. Float, like starts, durations, and finishes, is merely shown in a cell.

Linear schedules expand the amount of information that can be communicated. Whereas bar charts are essentially one-dimensional (time) and networks completely lack any quantitative axis, linear schedules are truly two-dimensional (time and work). Work quantity has a suitable specific (e.g. cubic meters or kilometers) or general unit (e.g. dollars or percent). Start or finish events, as well as intermediate progress updates (if available) appear as points in the coordinate system. It is commonly assumed that progress is linear between such events (Harris and Ioannou 1998). Yet the method can represent schedule data of any resolution without needing to change the axes. In comparison, precedence diagrams must split into finer sub-activities, so that their number of
elements grows as the product of resolution and initial activities. Each activity is now a line with a progress direction. It is separated from other activities by buffers, which are analogous to lead or lag durations. Float appears as areas between activity curves. Linear schedules can assist in identifying a balanced productivity for repetitive activities by seeking to align activity curves. However, they suffer from similar difficulties as bar charts for comparing initial and changed versions. Moreover, crossing lines confusingly may or may not indicate congestion or conflicts, depending on the nature of activities. And it is unclear if buffers model a time or work constraint.

**Research Need**

Linear and repetitive schedules are an alternative scheduling approach that itself is two-dimensional, tracking work progress over time. Activities or segments thereof (in case different productivities occur during the execution of a single activity) thus are lines that grow from the origin or continue from previous ones. **Buffers** play an even larger role than the analogous lags in networks. Buffers in network schedules are used as contingency to improve the completion time reliability (Park and Peña-Mora 2004). Their needs stems from a multitude of factors, ranging from quality of design documents via complexity of the project itself to like weather and climate (Russell et al. 2013). Another vital factor is obeying safety distances (Buskila and Isaac 2013).

Buffers in linear schedules are expressed in time or work units, the latter can be converted into equivalent time buffers. A special challenge is that they may specify both minimum and maximum constraints between activity pairs in a linear schedule (Kallantzis and Lambropoulos 2004), but no formal algorithm was described. Minimum and maximum constraints in networks remain rare (Hajdu 2015), despite existence of at least an iterative algorithm (Hajdu 1997). While a minimum buffer can always be fulfilled, in some cases a maximum buffer may require
interruptability of an activity or adjusting its productivity. Like in network schedules, criticality and float are vital features of a linear schedule. The critical path traverses a linear schedule from the origin to the finish of the last activity and ‘jumps’ across buffers between activities at the closest proximity. Float occurs only at the starts or finishes of noncritical activities or segments. These vital items must be explored to create a fully functional extension and conversion.

**Slip Chart**

After identifying the need, a literature search for potential improvements or alternatives outside of construction management confirmed that scheduling time-sensitive projects of course is not isolated to the construction industry, but is a key goal in all industries. It was therefore hoped that their approaches may provide inspiration. An intriguing find from aerospace, a so-called *slip chart* (slip indicating delays in events) is shown in Figure 1. It was used to manage launch dates of the Ranger 1 through Ranger 9, an unmanned space program of the National Aeronautics and Space Administration (NASA) between 1961 and 1965 to photograph the moon (NASA 1977). Notably, work schedules at Kennedy Space Center were prepared to “reflect the ambivalence of the Apollo 8 mission. (…) Allowances for a slippage to 15 October were built in for testing the fixes” (NASA 1978a, p. 450). Details of this unusual schedule representation are presented in the following sections as derived from the relatively sparse literature, even within its original area, and the authors’ own testing. To the best of the available knowledge, it has never been applied to construction project planning and control, possibly due to lack of dissemination beyond its area and some currently existing conceptual limitations that will be discussed in a following section.

*Initial Scheduling Techniques*
Before the advent of computer scheduling, aerospace mission planners met with a whiteboard on which magnetic strips were arranged into bar charts; once approved it was photographed “1-2 time per day and copied out to all the departments… to know what was scheduled for the shift / day, week, month” (Madden 2014). This approach mirrors developing construction schedules on large rolls of paper that could be hung onto a wall (Kelley 2005). A somewhat simpler approach with colored note stickers was observed by the first author in a construction field office around 2005. Bar charts showed e.g. environmental conditions and actions of unmanned probes (NASA 1969) and became the common schedule representation of all major space programs, including the Space Shuttle (Madden 2014). Creating launch facilities was documented with lists of dates and a summary bar chart with monthly milestones, while daily and hourly bar charts were used for the countdown of Apollo missions (NASA 1978a). Details with minute or second resolution often were tabulated (NASA 1978b, NASA 1986, NASA 1998). Provisioning the International Space Station was planned in commercial-style bar charts without arrows, but with milestones (NASA 2014). Traditional well-known scheduling techniques were of course also used. A report on manned space flight for example used a plain bar chart without arrows to summarize how the launch complex for Project Mercury was tested (NASA 1962), while another report for the same project used a bar chart with added milestones to display its complete schedule (NASA 1963).

Definitions and Terminology

The graphical layout of the existing slip chart comprises the following elements and meanings:

* Slip chart, also “progress chart”, “time map” (Gangale 2007, pp. 1-3), “milestone trend chart” (ESA 1995, p. 48): Coordinate system with planned and actual time axes to plot events;
• **Event**: A point in time at which the status of one activity changes from one operational level to another (‘milestone’), or start or finish of a range across time, e.g. an ongoing phase or task;

• **Planned time**: One axis, e.g. horizontal, in slip chart on which to indicate projected events;

• **Actual time**: One axis, e.g. vertical, in slip chart on which to measure real historical events;

• **Event horizon**: Diagonal line through the coordinate system of the slip chart at which the “schedule becomes reality, where the future becomes history” (Gangale 2007, p. 12), which gives the slip chart its characteristic triangular appearance with an upper left and lower right half.

  The sequence of activities in the overall project progresses largely parallel to the event horizon;

• **Time path**: A segmented line that tracks planned and actual performance of a single activity;

• **Slip**: Step parallel to the planned time axis when the time path of said activity was modified.

*Composition and Use*

Besides its use in the 1977 report, barely any sources are found that employ slip charts, besides Brooke (1973) who predates it, the European Space Agency (ESA 1995), and Gangale (2007). In manufacturing it is known as the *milestone trend analysis* (Lainati and Arena 2010) and software exists (*Anon.* 2006), but the literature is sparse, notwithstanding a detailed description (Brooke 1973) from the Electricity Council’s Research Centre in Great Britain (now EA Technology).

Slip charts showed a manifest schedule, a “high level schedule with Launch Dates” (Madden 2014) for multiple missions within a program. Data were-planned, as-updated, and as-completed events, i.e. a record of the decision-making itself. Each event – a point in time – has a single path to track both when changes occurred and how they revised the event date. Delays are called slips.
They “thus present a pictorial record of the progress of a project within its planned sequence of activities by revealing the past, present and expected future performance” (Brooke 1973, p. 4).

The launch slip chart had a left vertical axis to measure continuous planned time in months. Its top horizontal axis tracked actual time. Both axes covered the same range. Only an upper left triangle, i.e. half of the square coordinate system was used. The remaining lower right triangle was a legend. Once events touched diagonal “event horizon” (Gangale 2007, p. 12), plan became reality and no further delay could occur. In other words, the event horizon showed the historical record of actual events, whereas any changes to the plan occurred in the triangular area next to it.

Each planned launch was traced as a time path (trajectory) from conceiving it via revisions to its execution at the event horizon. Lines suffered vertical steps (slips) when a date changed due to internal or external factors. Consider the Ranger 5 spacecraft (RA-5) in Figure 1 whose launch changed thrice. Chronologically from left to right, in September ‘60 it was scheduled for July ‘62. But in October ‘60 it was moved to June ‘62 on the vertical axis. In September ‘61 it was postponed to November ‘62. Thankfully, a small gain was realized in March ’62 that advanced it to October ‘62. It actually launched on October 18, 1962 when the line ends at the event horizon.

After the Challenger loss, the space shuttle program manager desired more insight into the progress of STS-26 (the space transport system mission number) toward its launch. In response, Gene Sestile, schedule integration manager at Kennedy Space Center (KSC) created Figure 2 to track five processing milestones and its planned launch. Milestones were (1) on-dock delivery of solid rocket motor (SRM) aft segments from Utah; (2) the start of solid rocket booster (SRB) stacking; (3) mating the external tank (ET) between the twin SRBs; (4) mating the orbiter (ORB) to the ET; and (5) transporting the integrated space shuttle vehicle (SSV) to the pad. The initial slip chart was created in May ‘87, about 9 months prior to the then-planned February 18, 1988
launch; 4.5 months before the then-planned arrival of aft SRM segments that were needed for integration. The root cause of the Challenger accident had been the O-ring in an SRM segment field joint that failed on a freezing Florida morning. In June ‘87 NASA decided to use a J-seal instead. Redesign meant that delivery slipped by 9 weeks, which impacted all other milestones. Also, NASA decided to require a flight readiness firing (FRF), i.e. loading propellant into the ET and firing the three space shuttle main engines (SSMEs) for several seconds. Adding the FRF with its extensive preparation and recovery caused 6 more weeks slip. The payload, a tracking and data relay satellite (TDRS-C), could only be brought to the launch pad after the FRF.

Sestile’s slip chart successfully provided a rich visualization of returning to flight, reasons for milestone slips, and impacts on the planned launch: A 10 day delay for SRM nozzle nose inlet housing and exit cone ablative bonding; 11 days for SRM nozzle dry fit, leak checks, and room temperature vulcanizing adhesive (RTV) backfill, and 8 weeks for nozzle cowl boot failure. A 38-day slip occurred in stacking SRBs and SRB closeout verification took 1 day. Stacking delays caused the ET/SRB Mate milestone and others to slip. Orbiter/ET Mate slipped 4 days due to SRB cable problems and orbiter tile paperwork. Reasons for another 4-day delay of SSV to Pad were the nose landing gear door, the orbiter and ET mating, range safety systems (RSS) on the ET, and the hazardous gas detection system. FRF slipped by 15 more days due to a countdown demonstration test with propellant loading and unloading before FRF, and problems with a bleed valve on SSME #2 and the 17 inch cavity purge. These events delayed the launch by 7 days.

The slip chart was so useful that the program manager ordered its use on all missions. Each slip chart was initiated at the launch site flow review (LSFR), which set processing milestones and indicated the formal commitment by KSC to work toward them. Its status was updated every
Wednesday at 9 a.m. and distributed to senior management. The final version was produced after launch and provided a true record of the events that impacted each milestone. In testament to its value, the program manager signed the final slip chart of each mission from STS-29 onward. While this practice ended in 1992, slip charts continued to be used until the end of the program.

As the historical record grew, it offered rich data for simulation analysis (Lucko et al. 2008) to understand delay causes. Accurate predictions with discrete event simulation could be made in the final decade (Cates and Mollaghasemi 2005, 2005). Such dataset still allows predictions for new flights. ESA (1995, p. 47, p. 48) reports showed a “milestone trend chart” and a “progress chart” in an example with two coordinate axes of how estimated timing evolved. Gangale (2007, p. 1) reproduced the diagram, noting that “[m]uch less known is the slip/progress chart or trend chart” and claiming its invention in 1975. He noted that despite “nine years of space program management for the US Air Force” such chart was “never encountered” (Gangale 2007, p. 6). His five-part ‘time map’ coded the manned Mercury and Gemini missions with colors and icons. Relative change of planned versus actual time was the “expansion rate” (Gangale 2007, p. 12).

Another aerospace chart worth mentioning was the “cyclogram” (Tufte 1997, pp. 94-95) that displayed the three-month-long 1977-1978 Salyut 6 space station mission of the Soviet Union with a horizontal time axis in days that listed planned and actual tasks and a vertical time axis in minutes for each orbit, showing the visible day-night-transitions in the area between the axes.

Temporal Interval Representation

Separate from the aforementioned use, two studies in computer science and artificial intelligence explored how to represent time intervals. A seminal work (Allen 1981; 1983) sorted how discrete starts and finishes of two activities form different constellations. They were intuitively termed
“before”, “overlaps”, “during”, etc. (Allen 1981, p. 222) and led to exploring soft logic (Tamimi and Diekmann 1988). Rit (1986, p. 385) plotted events on a horizontal “begin” and vertical “end” axis and appears to be an originator of this representation. Areas in Figure 3 distinguished where a potential predecessor, parallel activity, or successor could be positioned to generate Allen’s (1981) constellations. Earliest and latest dates were understood as corners of a polygon of potential timing, but were not linked. A generalized chart allowed that durations could vary, giving a distorted shape whose corners had different distances from the event horizon. Said study focused on how a point relates to another in terms of constraints, but overall it remained purely qualitative and fell short of developing a viable approach to model or analyze project schedules.

<PLACE FIGURE 3 HERE>

Research Approach

Research Objectives

Objective 1: Assess capabilities of slip charts as presented in the literature on how quantitative information is recorded and communicated and its limitations to inform potential improvements;

Objective 2: Extend modeling and analytical capabilities of the slip chart toward an application to linear schedules of construction management, including minimum and maximum buffers;

Objective 3: Derive a conversion algorithm of linear schedules, including criticality and float.

Research Methodology

The overall research goal is investigate how concepts related to existing charts can be aggregated into a new approach to represent project schedules efficiently and effectively. Quantitative time data that must be expressed include those of traditional network and linear schedules, i.e. activity
starts, durations, and finishes, any segments within activities, interruptability, general sequence, link types, buffers between them, criticality of activities or segments, and its counterpart float.

To accomplish the research objectives, it will be necessary to first assess the capabilities and limitations of the existing charts. Then concepts will be adopted, adapted, modified, or newly introduced to generate modeling and analytical capabilities that support project managers in their decision making. The approach will generate and test sample schedules of varying complexity on how clearly the aforementioned data and their context are visualized. Emphasis will be placed on expressing minimum and maximum buffers, which are characteristic within linear schedules to separate activities for safety, technical, or administrative requirements. Criteria to guide the development will be the density of communicating data per area and the sparsity of using ink and symbols to generate a rich yet lean chart as advocated by Tufte (2001). Existing representations, specifically network schedules, were found to fare rather poorly on these aspects (Lucko 2005).

The best concepts from its sources of inspiration will be integrated and merged with new ideas. Due to the different nature of the bar charts and network schedules on the one hand, and linear and repetitive schedules on the other hand (Lucko 2007), they will need to be examined separately to derive two detailed algorithms to convert them into the new representation. For clarity, this paper will focus on linear schedules and a future publication will address networks.

**Capabilities and Limitations**

The slip chart has successfully been used in scheduling vital aerospace missions that ranged from the Ranger unmanned lunar exploration spacecrafts of early 1960s to the Space Shuttle reusable manned spacecrafts that were used beyond the late 2000s. Its usefulness in aerospace practice
has thus been proven. From a theoretical view, its modeling and analytical potential stems from
the following features, which are extracted from a review of the published use of the slip chart:

- Each activity is represented as a continuous line (somewhat akin to a linear schedule), which
  shows its performance throughout the entire project; including both planning to execution;
- Two measured axes of time directly convey the quantitative information with its rich context;
- Spacing between activities within the schedule shows the efficiency of using productive time;
- Delays are steps in lines, which clearly shows when they occur and what impact they have;
- Number and size of changes indicate progress, i.e. fewer mean better adherence to the plan;
- Timely accomplishing of all target dates is clearly visualized along the event horizon itself;
- Information is provided for a schedule not just at a single point in time, but how it evolved;
- Compared to networks, real quantitative data are shown, not just numerical values at boxes;
- It efficiently uses only half of its diagram space for information, leaving another half unused.

However, it had various substantial limitations that need to be addressed in order to extend it
into a feasible and valuable addition to the toolkit of professional construction project managers.

- Similar to bar charts and network schedules, it displays time information, not work product;
- Similar to bar charts, by definition it does not explicitly show relationships among activities;
- Only points in time of events were displayed, but activity durations could not be expressed;
- Activity segments of variable performance that may occur in linear schedules were lacking;
- Different link types as permutations of relations among starts and finishes were unsupported;
- Lead or lag durations on links that indicate a mandatory passing of time were unsupported;
- Float, i.e. flexibility of activities to incur delays without harming other activities or the entire
  project, is not shown, which in part is caused by the aforementioned lack of relationships;
- Distinguishing earliest and latest dates for individual activities was not feasible in the chart;
• Representing the equivalent of a critical path versus activities with float was also impossible;

• Comparisons between different possible options of a schedule appear to not have been made;

• While it may be considered a strength or weakness, the chart uses only half of the area that is created within the coordinate system, leaving the remainder open for another chart or legend.

Thus various fundamental limitations exist in the slip chart as it has been used in the aerospace industry. However, its notable use in aerospace and its capabilities attest to the inherent potential, which with appropriate extensions in the definitions and symbols for the first time will become accessible to the construction industry. This conceptual review fulfills Research Objective 1.

Extension to Linear Schedules

Activity Segments

Linear schedules use a coordinate system that plots work and time, where each activity is a line (if it has constant productivity) or collection of line segments (if it has phases of different productivity). Productivity is defined as any measure of work quantity divided by its time duration. Buffers can be expressed in terms of work or time to a distance between sequential activities that must be maintained. Buffers are closely related to float, but whereas obeying buffers is mandatory, using float is optional. Figure 4 illustrates the two-step algorithm that guarantees obtaining the minimum total project duration (Lucko 2009) for an example of a small construction project (Harmelink and Rowings 1998). Figure 4a illustrates how Step 1, stacking, conservatively assumes that the maximum finish of a predecessor buffer, marked as a gray area, becomes the start of a successor, as shown by dotted horizontal lines, in analogy to finish-to-start (FS) links with lags of network schedules. Of course, this does not create an efficient schedule, as potential overlap is prevented. Step 2, consolidation, therefore calculates minimum distances
in Step 1 and removes them. Activities thus touch via the buffers that are located between them.

These critical points are marked with circles in Figure 4b. The continuous chain of critical points forms the critical path. Just like in network schedules, it can split and merge, e.g. in activities E and F. However, whereas network schedules make an assumption of point-to-point links between only starts and finishes, critical points in linear schedules can occur during activities, e.g. in D. They can be caused by productivity changes in predecessors or successors, e.g. in C and E. This induces new segments within activities, as marked with a single quotation mark (’). For example, D has non-critical early and late segments D₁’ and D₃’, but is briefly partially critical in D₂’.

<PLACE FIGURE 4 HERE>

For brevity, Figures 4a and 4a are cropped at a thick dashed line. Initial transferring of such linear schedule into the new chart focused on segmented activities and buffers. Activities with a constant productivity have a distinct start and finish date are shown as a single point in Figure 5 like before. Yet those with changes of productivity can be represented in two ways, as summary activity point or by showing each segment as a separate sub-point, e.g. activity C in Figure 5a.

Buffers in linear schedules were initially modeled as grayed-out areas during which activities may not be performed. Note that the new charts can only express time buffers, because they lack a work axis. But this is not a detriment, as any work buffer can be converted into an equivalent time buffer by dividing it by the productivity (Lucko 2007). Time buffers are adapted in Figure 5a by adding a gray strip with a given height, i.e. duration on the vertical axis, after an activity occurs (assuming that a buffer belongs to the predecessor). Buffer strips are then reflected at the event horizon to displace the successor start along the horizontal axis by the equivalent time. If strips would be omitted, it was found that a particular buffer duration could also be traced across the event horizon and then reflected individually to arrive at the correct successor coordinate.
Figure 5a shows how activity C is composed of two segments; the first has the same start and
the second has the same finish as the activity point. Such segments of activities approximately
form a triangle shape, whose upper left corner is at the activity point. However, it was noted that
buffer areas began to crowd the diagram, so that they were not used in Figure 5b. Instead, they
were summarized by adding a diagonal buffer tail to each activity point. Tails are solid if critical
and dashed if not. By sounding from such tail to the right (as if it cast a shadow) it can be
checked that no direct successors fall into said restricted areas. This means that successors are
moved earlier – parallel to the event horizon toward the origin – to fill the white unoccupied
areas in Figure 5a. For example, activity B comes to rest just on the outside edge of the buffer
strip of A in Figure 5b. Of course, the buffer strips could be shown if it is desired by the user. It
was also envisioned, but is omitted for brevity, that the approximate triangle of activity point and
segment points can be shaded to easily identify constellations that the segmented activities form.

Critical activities and segments thereof are indicated as filled points and connected by a thin
dotted line to explicitly show the critical path as it has emerged after consolidating the previously
stacked activities. It branches and merges for activities E and F, same as in the linear schedule of
Figure 4b. Non-critical activities and segments are shown as hollow points, which the critical
path bypasses. A slight redundancy is retained in Figure 5b by showing activity points (e.g. D) as
well as their more detailed (induced by consolidation) segment points (e.g. D₁’, D₂’, and D₃’). If
desired, activity points could be omitted so that only their segments points are shown. For
information about segmentation, activity polygons are shown, which could be omitted as well.

While FS links are the only type that can occur in stacking per its assumption, consolidation
yields start-to-start (SS), finish-to-finish (FF) or intermediate links between segments, i.e. partial
activities (PP). Unless extra constraints happen to exist, FS links do not typically occur. Start-to-finish links (SF) with lags are very rare, like in network schedules. The can easily be converted into FS links by subtracting both activity durations, and then handled by reflecting beyond the event horizon as has been explained before. The lag duration of each FS link in Figure 5a is the thickness of its buffer strip, which remains the same before and after reflection. In Figure 5b the FF1 link between A and B across the buffer (one day lag) is the vertical distance between A and B. In other words, lags are measured, as could be expected, along the finish axis. The SS1 link between B and C (more specifically C\textsubscript{1}) is measured, analogously, along the start axis. Links between partial activities can be treated as either local SS or FF links for those activity segments.

On the downside, work quantities of activities do not appear in the new chart, which has two time axes. Yet it still contains more information than network schedules, namely segments with possibly different productivities; which segments have float or are critical to form a continuous critical path; what time buffers must be maintained toward any successors; and what exact link types emerge between predecessor and successor segments. The empty triangular half could hold an updated version of this linear schedule, whose axis designations would be switched. Details of such comparison capability to perform a delay analysis will be explored in future research. This theory extension of slip charts to linear schedules fulfills the first half of Research Objective 2.

Conversion of Linear Schedules

Simple Buffers, Emerging Links, Criticality, and Float

Figure 6 shows stacking and consolidation of activities in detail, where Figures 6a through 6d show the converging Case 1 and Figures 6e through 6h show the diverging Case 2. As shown, both cases obey the buffer. The examples of Figure 6 can be verified using the inputs of activity
durations $d_A = 2$, $d_B = 1$ for Case 1 (3 for Case 2), start of $A = 1$, and its buffer $\Delta t = 1.5$, which here is a minimum time. The following rules systematically add activities and buffers of a linear schedule to the new chart, where different link types will emerge as can be identified graphically:

- Same as for network schedules, activities are fundamentally points in the coordinate system;
- Same as for linear schedules, activities are added to the scheduler per the overall sequence;
- Different from network schedules, specific link types are not a required part of the input, but emerge through the aforementioned stacking and consolidation to minimize project duration;
- To indicate that a project is originally conceived as a linear schedule with buffers and not as a traditional network with FS relations, a subtle notation is employed; buffer tails are used;
- Buffer tails are added to activity points, their diagonal length is measured on the buffer axis;
- For stacking, successors are added tentatively with a $FS$ link with a buffer. Said buffer tail is shifted horizontally to the right until it falls onto the event horizon, where its end is marked;
- The successor duration is added upward, which yields its coordinates under this assumption.
- For consolidation, the dotted duration line is shifted diagonally downward to the lower left parallel to the event horizon until its upper tip either crosses the horizontal projection of the shifted buffer tail (Case 1), or until its line touches the top right tip of the buffer tail (Case 2), somewhat akin to a wagon touching a ‘railroad buffer’ at the end of a piece of steel track;
- In Case 1, a FF link will emerge between the two activities, in Case 2 a SS link will emerge. Their locations in the linear schedules are called critical points and are marked with circles.
- Criticality may be partial in that only its start, the entire activity, or only its finish are critical;
- If it happens to be known that both activities perform the same work quantity, their relative productivities can be inferred from whether the predecessor is longer and successor is shorter ($converging$ in Figure 6a, longer then shorter duration as measured perpendicular from event
horizon to activity point in Figure 6b) or \textit{vice versa} (diverging in Figure 6e, shorter then longer duration in Figure 6f), assuming that their overall productivity remains constant.

\begin{figure}
\centering
\caption{Figure 6 Here}
\end{figure}

- In Case 1, the critical path contains all of activity A, the FF link, and only the finish of B. Thus B has \textit{start float}, which Figure 6d shows as a distance on the horizontal dashed line;
- This float can be consumed by increasing the duration of B, but starting as early as possible at 2.5. In this special case, its point shifts left until touching the top right tip of the buffer tail, so that its productivity aligns with that of A, and its buffer forms a single continuous link;
- In Case 2, it contains only the start of activity A, the SS link, and all of B. Thus A has \textit{finish float}, which Figure 6h shows as a distance on the vertical dotted line. Note the mirror symmetry in geometrically modeling FF and SS links with buffers within the new chart;
- It can be consumed by increasing the duration of A, finishing as late as possible at 4. In this special case, its point shifts upward until the top right tip of the buffer tail touches B – the remainder of this modified constellation is analogous to consuming start float in Figure 6d;
- In summary, \textit{criticality} is identified in the new chart if a successor is directly above the tip of the buffer tail (start critical) or directly next to its tip (finish critical) and – conversely – \textit{float} is as the successor distance above the tip (finish float) or next to it (start float). Note that types are identified; start float is parallel to the horizontal axis, finish float to the vertical one.

\begin{strip}
\textbf{Minimum and Maximum Buffers}
\end{strip}

Minimum and maximum buffers pose an additional challenge, which can be resolved in the new chart as follows. Figure 7 shows two new activities after their initial stacking and consolidation based on minimum buffers only (solid tail and dark gray) as described in the previous section.
Yet as Figures 7a and 7e show, the early (late) part of B in Case 3 (4) violates the maximum buffer (dotted tail and light gray). In Figures 7b and 7f this is easily seen, because the point of B falls outside of the boundaries of the light gray square. It is even visible in Case 3 of Figure 7b that B is infeasible at its start, being offset to the right on the start axis. In Case 4 of Figure 7f it is infeasible at its finish, being offset upward on the finish axis. Two solutions could satisfy this constraint, adjusting B so that its slope lies fully within the boundary of the light gray buffer, or inserting deliberate breaks, but keeping the same productivity. The former approach will simply start B earlier in Figure 7a (shift it left in Figure 7b) or finish it earlier in Figure 7e (shift it downward in Figure 7f). Offsets in the new charts are the necessary duration adjustments.

Just slightly more complex is the latter approach of allowing interruptability. Here one must determine the minimum number of breaks that will satisfy the constraint. Figures 7d and 7h show the approach to solve this question. First, A is split into as many segments as B; A₁ and A₂ with their respective minimum and maximum buffer squares. The dashed duration bar of B is split into two pieces that are measured vertically from the event horizon. Then the challenge is simply finding a constellation where the points B₁ and B₂ lie on the edge of the maximum buffer square. Their known length only allows one constellation each, which leads to the solutions of Figures 7d and 7h. Distances to the right (left) in the new chart are the necessary positive (negative) breaks for Case 3 (Case 4). This interruptability approach works analogously for any other number of segments. Linear schedules lack such geometric guidance. The examples of Figure 7 can be verified using the inputs of activity durations \(d_A = 3, d_B = 1\) for Case 3 (5 for Case 4), with the start of A = 1, and its minimum and maximum buffers \(\Delta t_{\text{min}} = 1\) and \(\Delta t_{\text{max}} = 1\) (note that for simplicity the buffers are defined as additive, not concurrently acting). If interruptability with positive or negative breaks (\textit{n.b.} that the latter require more than one crew) had not been allowed,
the productivity rate of B could have been adjusted to fall into the maximum buffer. Its duration would have to be within \{3, 4\}, i.e. \(S_B, F_B = \{2, 5\}\) or \(\{3, 6\}\) for \(d_B = 3\) or \(\{2, 6\}\) for \(d_B = 4\), which in the new chart falls onto corners of the light gray maximum buffer square. This theory extension for minimum and maximum buffers fulfills the second half of Research Objective 2.

<PLACE FIGURE 7 HERE>

Figure 8 summarizes the conversion algorithm as a flowchart, which includes setup, stacking and consolidation steps inspired by linear scheduling calculations (Lucko 2009), how to correctly insert buffers, how different link types emerge, and how to handle activities in linear schedules that feature multiple segments with variable productivities. This fulfills Research Objective 3.

<PLACE FIGURE 8 HERE>

Contributions to the Body of Knowledge

This research has contributed to the body of knowledge in project scheduling in several ways:

- Capabilities and limitations of slip charts have been delineated to prepare for an extension;
- A new approach to perform linear scheduling within a time-scaled chart has been created, which not only preserves all quantitative of the information, but visualizes all of it explicitly;
- A conversion algorithm has been created, which handles even intricacies like different link types that carry lead or lag durations and inserts their equivalent constructs into a new chart;
- Minimum and maximum buffers that are a vital feature within linear schedules are modeled;
- The new approach has various advantages, e.g. that it quantitatively models durations, leads or lags, and float, incorporates buffers that are vital to yield an efficient linear schedule, and allows comparing two versions of a schedule side-by-side, which can reveal delay events.
Conclusions and Recommendations

The new charts hold significant potential of providing an immediate quantitative representation of an entire construction project schedule. Their unique characteristics make them sparing in use of symbols (activities are simply points or lines), intuitively understandable (time passes along the diagonal), and even efficient in communicating vital information on performance (delays in an updated schedule would occur as shifted activity points compared to the original version).

Further research will examine this envisioned ability of the new approach to facilitate delay analysis. It will also extend it to encompass network schedules, which are prevalent in practice.

Acknowledgements

Financial support by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES Foundation), Ministry of Education, Brazil for the second author is gratefully acknowledged.

The first author gratefully acknowledges the interesting conversations on scheduling methods in the aerospace industry with Mr. Michael G. Madden, former Senior Vehicle Engineer for OV-105 Endeavour and OV-104 Atlantis, United Space Alliance, Kennedy Space Center, Florida.

Special thanks to Mr. Eugene M. Sestile, former Space Shuttle Schedule Integration Manager at NASA and Mr. Jeffrey J. Angermeier, Senior Program Planning Manager, Ground Systems Development and Operations Program at NASA for sharing the history of the Space Shuttle slip charts. Input by Dr.-Ing. Andreas Lucko of Airbus Operations on prior use and by Mr. Joseph D. Lombardo of Dysco on the extension is gratefully acknowledged. Input by Mr. Yi Su, doctoral student and graduate research assistant at Catholic University of America on maximum buffers is gratefully acknowledged. Last but not least, thanks to Ms. Kimberly M. Hoffman, Scholarly
Communications Coordinator at Catholic University of America for assistance in obtaining library materials.
References


NASA (1962). *Results of the first United States manned orbital space flight: February 20, 1962*. Report, Manned Spacecraft Center, National Aeronautics and Space Administration,


Figure 1: Slip Chart (public domain, NASA 1977, p. 347)
Figure 2: Critical Path Assessment Summary (with permission from NASA 1988)
Figure 3: Possible Relations (according to Allen 1983) of One Activity with Another (modified from Rit 1986)
Figure 4: Linear Schedule [Adapted from Lucko 2009 and Harmelink and Rowings 1998]
Figure 5: New Chart for Linear Schedule

(a) After Stacking

(b) After Consolidation
Figure 6: Stacking and Consolidation for Converging and Diverging Pair of Activities
Figure 7: Minimum and Maximum Buffers for Pair of Activities
Figure 8: Four Different Link Configurations in Clam Chart

(a) Finish-to-Start Link

(b) Start-to-Finish Link

(c) Start-to-Start-Link

(d) Finish-to-Finish Link
Figure 1: Slip Chart (public domain, NASA 1977, p. 347)

Figure 2: Critical Path Assessment Summary (with permission from NASA 1988)

Figure 3: Possible Relations (according to Allen 1983) of One Activity with Another

(modified from Rit 1986)

Figure 4: Linear Schedule [Adapted from Lucko 2009 and Harmelink and Rowings 1998]

(a) After Stacking

(b) After Consolidation

Figure 5: New Chart for Linear Schedule

(a) After Stacking

(b) After Consolidation

Figure 6: Stacking and Consolidation for Converging and Diverging Pair of Activities

(a): Stacking for Converging Pair

(b): Initial Constellation of Converging Pair

(c): Consolidation of Converging Pair

(d) Start Float in Consolidation

(e): Stacking for Diverging Pair

(f): Initial Constellation of Diverging Pair

(g): Consolidation of Diverging Pair

(h) Finish Float in Consolidation

Figure 7: Minimum and Maximum Buffers for Pair of Activities

(a) Start Violation of Maximum Buffer

(b) Infeasible Start Constellation

(c) Postive Break to Satisfy Buffer
(d) Feasible Positive Break Constellation
(e) Finish Violation of Maximum Buffer
(f) Infeasible Finish Constellation
(g) Negative Break to Satisfy Buffer
(h) Feasible Negative Break Constellation

Figure 8: Flowchart of Conversion Algorithm for Linear Schedules