plexos
Lean Project Management

The Precedence Diagramming Method; History, Principles and Evolution 2

Introduction 2

The origins of PDM networks 3

Criticism of PDM networks 8

Simultaneous relationships between activities. 8

Reverse criticality 9

Unrealistic calendar usage 12

Continuous execution and optimal splitting 15

Space-time conflicts 16

Final comments 20

References 20

The Precedence Diagramming Method; History, Principles and Evolution

Introduction

The Precedence Diagramming Method (PDM networks from now on), as a Critical Path Method (CPM[1]) tool, have faced numerous criticisms since their inception and throughout their history, and nowadays for their ineffectiveness for production planning, especially prevalent in lean construction environments, concerns, as argued by Koskela, Howell, Pikas, & Dave (2014), questioning the use of CPM "*for so long given its inability to produce predictable outcomes?*", treating activities as "black boxes", and asking "*what was the methodological underpinning of the development of the CPM?*.

Behnam, Harfield, & Kenley (2016) states that "*Network-based scheduling methods were not originally developed for managing the production phase in construction projects*", producing an evident "*mis-match when network-based scheduling methods are applied to production management of construction*".

The problem shouldn't be blame on PDM networks, or CPM discipline, but to the available software packages for project management, and the inadequacy of implemented algorithms for production planning in the AEC industry, with "*some peculiar characteristics that are little recognized (and even less well documented), but which can have important implications for the user*" (Wiest, 1978; 1981).

Those issues remain undocumented by software developers, resulting in a lack of awareness among users and practitioners, with the obvious implications for the quality, and confidence, of the schedules delivered.

This article will explore the fundamentals of PDM networks with minimal lags[2], tracing their origins, characteristics, and criticisms, delving into effective strategies for managing these networks and mitigating their potential drawbacks, finishing with brief conclusions and recommendations.

[1] CPM is the discipline of project scheduling that deals with the development of methods and algorithms for project scheduling, as computing the times of the activities.
[2] Minimal lags establish a minimum value for restriction lags. Minimal and maximal lags impose both minimum and maximum limits on restriction lags.

The origins of PDM networks

The first appearance of mother mathematical planning tools and techniques dates to the Cold War with the development of AOA (activity on arrow) networks, instead of the traditional bar charts, such as the Harmonogram and the Gantt chart, where activities, and their durations, are represented by the position and its length.

The Harmonogram (Figure 1) was developed in 1896 by Polish engineer Karol Adamiecki, preceding Henry Gantt's Gantt Chart by a decade. Nevertheless, the latter, introduced in 1906, gained widespread popularity due to its adoption by the US Army during WWI.

In the mid-20th century, driven by the rapid advancement of computer technology and the pressing military needs of the Cold War, AON (Activity on Node) networks emerged as a powerful tool for visually representing and computationally processing projects. These networks utilize directed acyclic graphs, where edges symbolize the activities and nodes the events.

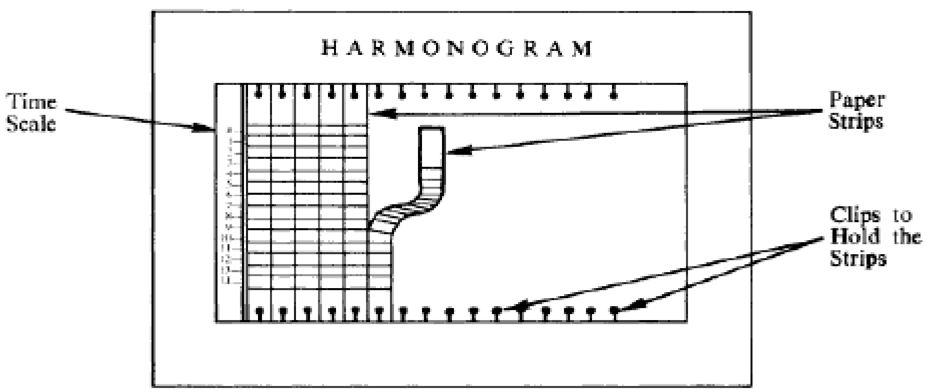

Figure 1 Adamiecki's Harmonogram

The two main AOA techniques are the Program Evaluation and Review Technique (PERT) (Malcolm, D. G., J. H. Roseboom, C. E. Clark, & W. Frazar, 1959) and The Critical Path Method (CPM) (Kelley & Walker, 1959). PERT is a statistical technique developed to deal with the development of the Polaris program, a submarine-launched nuclear-armed ballistic missile by the US Navy. In other line, CPM was developed by DuPont to optimize the maintenance costs of their production plants (Figure 2).

AOA networks can struggle to accurately model certain dependency restrictions between activities. This often necessitates the use of dummy activities (with zero duration), increasing the complexity of project planning (Figure 3). As a result, AOA networks have become less popular in favor of AON networks, which offer greater flexibility and more precise modeling capabilities for a wide range of projects.

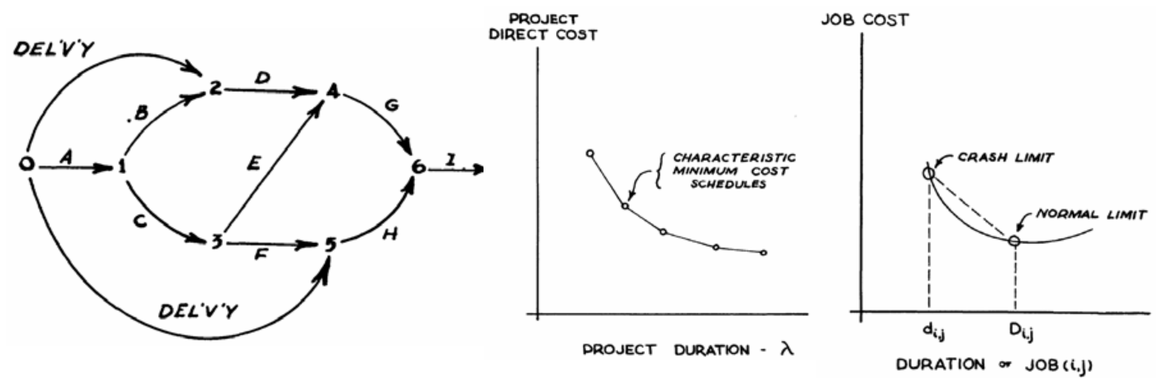

Figure 2 AON Project diagram and job cost curve (Kelley & Walker, 1959)

AON is a method of constructing a project schedule network diagram that uses boxes, referred to as nodes, to represent activities and connects them with arrows that show the dependencies.

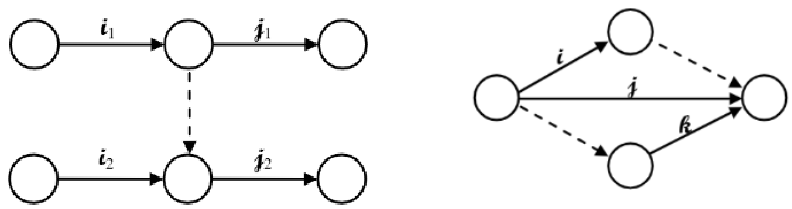

Figure 3 Convergent-divergent relationships with dummy activities

Early approaches to AON graphs emerged in Europe by the French engineer G. B. Roy with the Method of Potential (also known as Metra Potential Method (MPM)) (Roy, 1959; 1962), a novel approach that represents project activities as nodes, incorporating start-to-start relationships with lags (Figure 4).

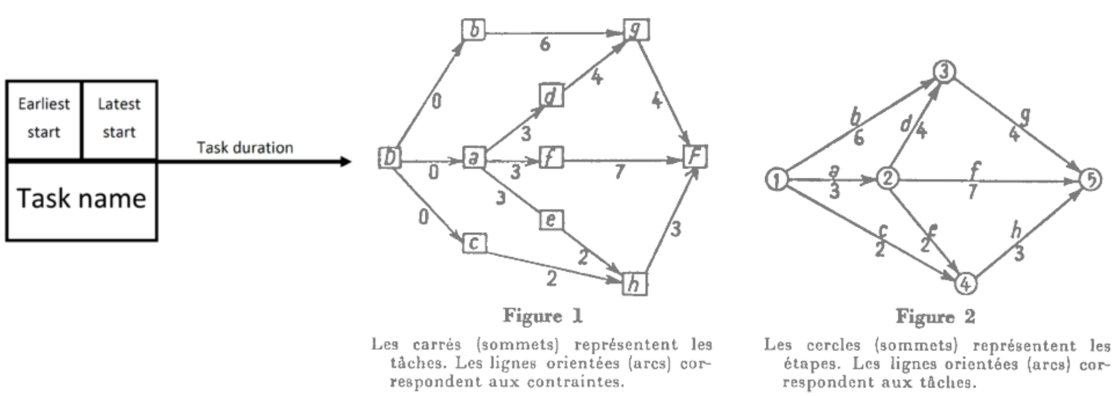

Figure 4 Roy activity and Roy Graph vs Graph PERT by Roy (1962)

In the United States, Fondahl (1962; 1987) introduced a novel Activity-on-Node (AON) methodology, enabling four precedence relationships to define the minimum allowable distances between two related activities. Fondahl's proposal gained widespread recognition through IBM's adoption with the IBM 1440 System and the Project Control System (IBM, 1964), becoming known as the Precedence Diagramming Method PDM (Figure 5).

William H. Linder (1967), from the Department of Civil Engineering at the MIT), proposed a problem-oriented computer language as a subsystem of the Integrated Civil Engineering System (ICES) for AON networks (Figure 6), with a novel philosophy of "*management by exception, in such a way that actual work and cost progress are compared with estimated work and cost progress to indicate those areas of the project which may prevent the project as a whole from finishing on schedule and within its budget*" (MacDermott, 1967).

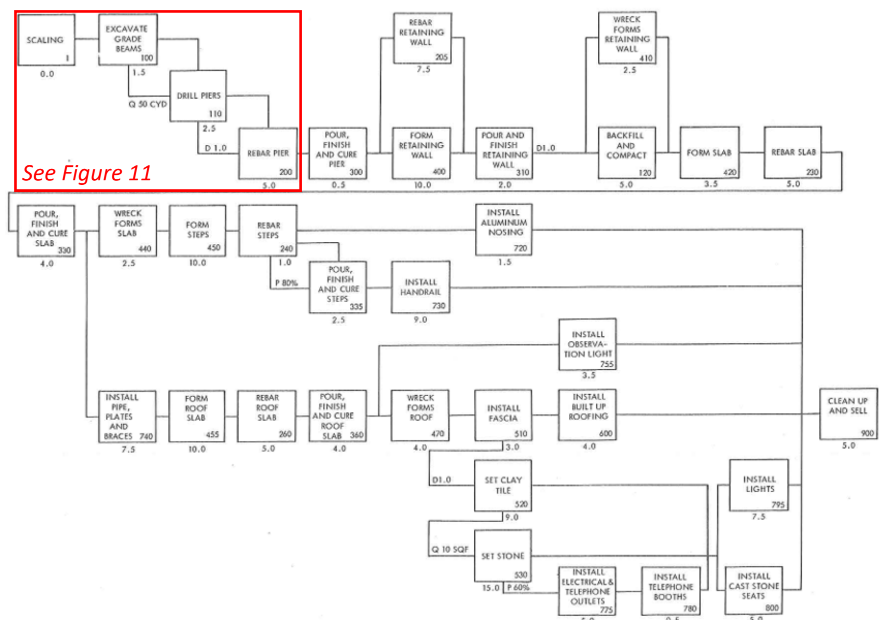

Figure 5 Precedence network (IBM, 1964)

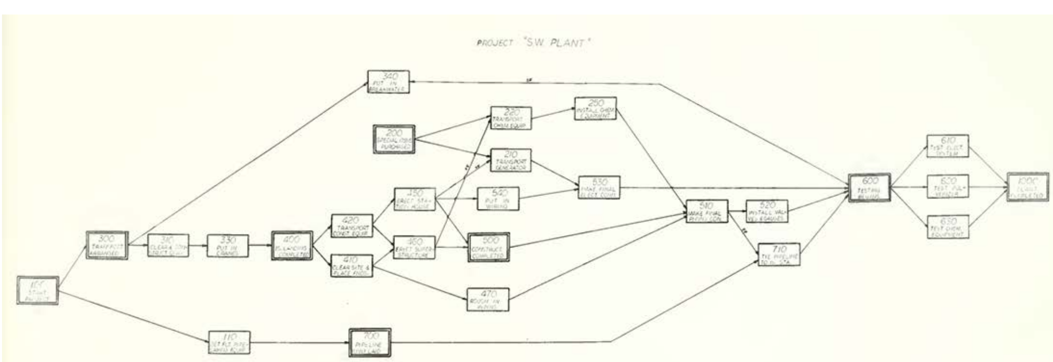

Figure 6 AON Project graph (MacDermott, 1967)

As stated previously, the precedence diagram method (PDM) is a project scheduling technique that visually represents project activities as nodes, and their dependencies (relationships) between them are shown as arrows. These relationships can be of four types.

Finish-to-Start ($FS_{ij}(z)$): An activity j cannot start until its predecessor activity i finishes with a lag of z temporal units (Figure 7).

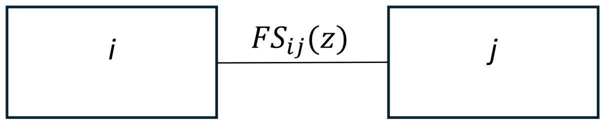

Figure 7 *Finish-to-Start* ($FS_{ij}(z)$) relationship.

Start-to-Start (SS_z): An activity j cannot start until its predecessor activity i starts, with a lag of z temporal units.

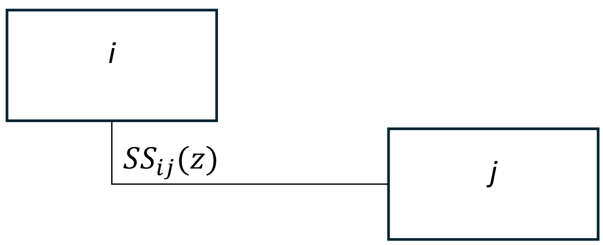

Figure 8 *Start-to-Start* ($SS_{ij}(z)$) relationship.

Finish-to- Finish (FF_z): An activity j cannot finish until its predecessor activity i, finishes, with a lag of z temporal units.

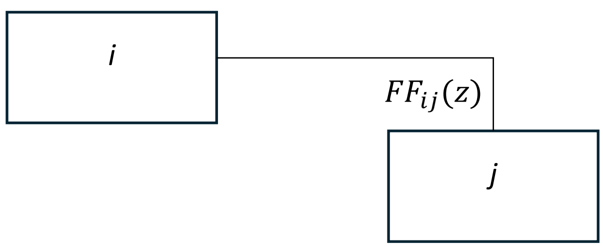

Figure 9 *Finish-to-Finish* ($FF_{ij}(z)$) relationship.

The *Start-to- Finish* (SF_z): A less common relationship where an activity j cannot Finish until its predecessor activity i starts, with a lag of z temporal units.

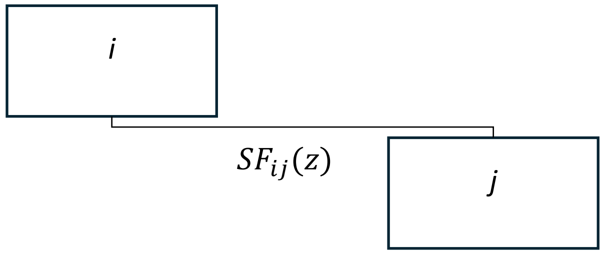

Figure 10 Start-to-Finish ($SF_{ij}(z)$) relationship.

The *IBM Project Control System (PCS)* allowed the use of simultaneous relationships between activities, considering four lag options: a percentage, a quantity, or a time-period in days, providing great flexibility for users and practitioners. In Figure 11 is shown the area highlighted in red of the Precedence network used by IBM in the Users' Manual for IBM 1440 Project Control System (PCS) (Figure 5), where can be seen the representation of Finish-to-start, Start-to-Start, and Finish-to-Finish relationships, and explained in the text as follow:

"The diagram shows that work item 110 (drill piers) may begin after 50 cubic yards of excavation has been completed in work item 100. This logic is represented by the bottom line leading from 100 to 110. The effect of the logic is to show that part of the duration of work item 100 may be overlapped by work item 110. In this particular example the estimated quantity of work item 100 is 150 cubic yards. Since the lag-time factor is 50 cubic yards, work item 110 may start after the excavation operation of work item 100 is one-third complete, or after 0. 5 days has elapsed. Thus, one day of the 1.5-day work item can be overlapped.

The top line leading from work item 100 to work item 110 stipulates that the latter operation may not complete until 0.5 day after the completion of the former."

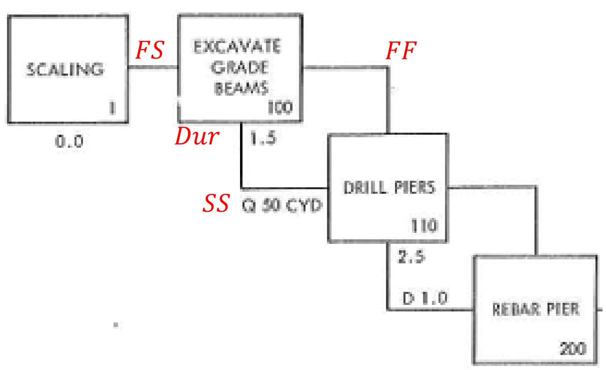

Figure 11 Detail of the Precedence network (IBM, 1964)

PDM networks are the most popular method for scheduling construction projects, despite facing criticisms both technically and on the methodological front for production planning in construction projects. This is largely due to how software packages implement them, relying on relaxed algorithms for time calculation, unrealistic calendar usage, imposed continuity execution of activities, reverse criticality, and a lack of adaptation to modern construction paradigms based on production processes like Lean construction, Takt planning, BIM, and VDC.

The following section will outline the primary criticisms of PDM networks and how commercial software addresses them, delving into effective strategies for managing them, and how to mitigate the potential drawbacks, finishing with brief conclusions and recommendations.

Criticism of PDM networks

As stated previously, PDM networks, and CPM methods, have faced numerous criticisms since their inception, and throughout their history. This section will examine these criticisms, assess their implications for software packages, delve into strategies for their management, and propose methods to mitigate their potential negative effects, but first, it's needed to expose the concept of relaxation, and relaxed algorithm, and how It's involved in the criticisms of PDM networks.

Relaxation is a modeling strategy that involves approximating a difficult problem by a simplified problem that is easier to solve, but consequently, the provided solution is a relaxed solution, or in other words, there's no guarantee that the solution is an optimal solution but a near-optimal solution in the neighborhood of the optimal solution.

Simultaneous relationships between activities.

Prior to the development of PDM networks, overlapping activities were addressed through the fragmentation of activities into splits, and the use of dummy activities.

The first split of the predecessor activity represented the initiation condition for the successor activity, while the second split of the successor activity denoted the portion that could not commence until the completion of the predecessor activity.

Figure 12 shows the details of the fragmentation process of the PDM network exposed in Figure 11 working with an AOA network, where can be seen the first split of the predecessor activity and the second split of the successor activity.

PDM networks can deal effectively with the correct overlapping of activities, using simultaneous use of *Start-to-Start* and *Finish-to-Finish* relationships between activities, considering that the activities must be executed without interruption (View Continuous execution and optimal splitting).

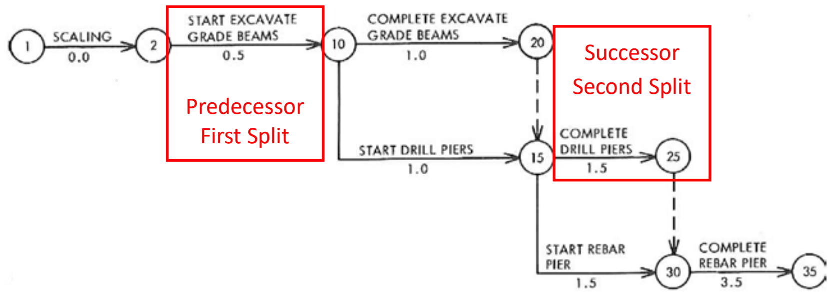

Figure 12 Equivalence of Figure 11 detail as AOA network (IBM, 1964)

The problem arises when the software implements a relaxed algorithm called standardization of relationships, in such a way that all the relationships are "*standardized*" into *Finish-to-Start* relationships (Table 1), being d_i the duration of the predecessor activity, and d_j the duration of the successor one.

Relationship	Initial form	Standardized form
Start-to-Start	$SS_{ij}(z)$	$FS_{ij}(-d_i + z)$
Finish-to-Finish	$FF_{ij}(z)$	$FS_{ij}(-d_j + z)$
Start-to-Finish	$SF_{ij}(z)$	$FS_{ij}(-d_i - d_j + z)$

Table 1 standardized relationships

This issue forces practitioners to prematurely decide which is the more restrictive relationship and being aware of updating the nature of the relationship while resequencing the project.

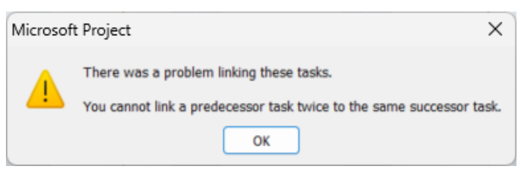

Figure 13 Message warning with Microsoft project

The standardization of relationships imposes all lags to be time delays, as the relaxation into *Finish-to-Start* considers that the successor activity can only begin once the predecessor activity has been fully completed, making impossible the use of these software applications for production planning, and increasing the effects of the Reverse Criticality.

Reverse criticality

The reverse criticality is an effect in such a way shortening an activity has the anomalous effect of lengthening the critical path and lengthening it would shorten the critical path (Wiest, 1978; 1981).

This anomalous effect, known as the "perverse effect" by Wiest) is produced when a critical path passes through an activity from its Finish-to-Start relationship to its Start-to-Start-

Relationship, and increased when the standardization of relationships is implemented because there's not any *Start-to-Start* relationship to limit it (Figure 14).

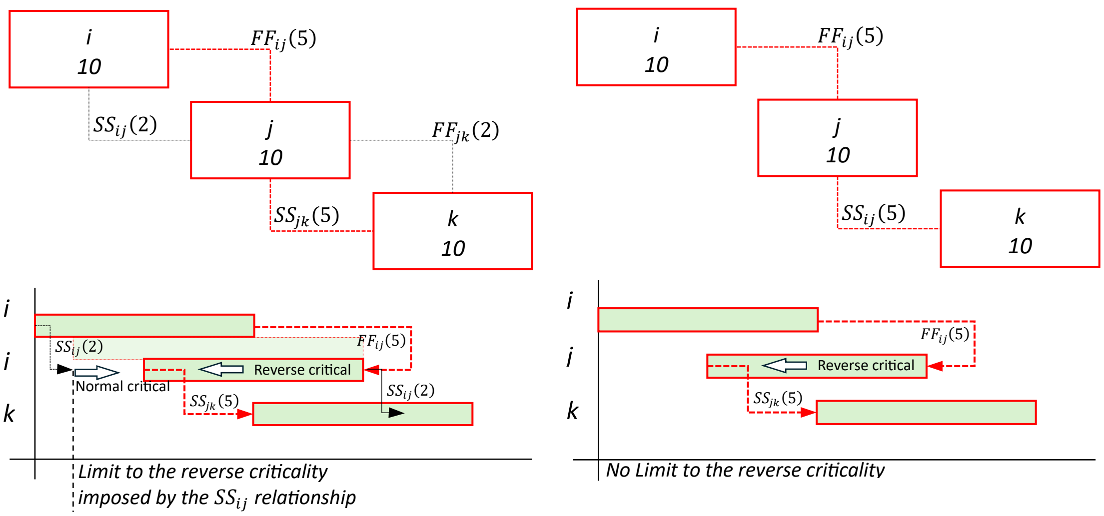

Figure 14 Reverse Critically in Regular and Relaxed Graphs

The effects of reverse criticality with Microsoft Project and *standardized* relationships can be seen in Figure 15, and in Figure 16 the effects of reverse criticality with Primavera P6, limited to when $SS_{ij}(2)$ relationship becomes critical, and activity j normal critical.

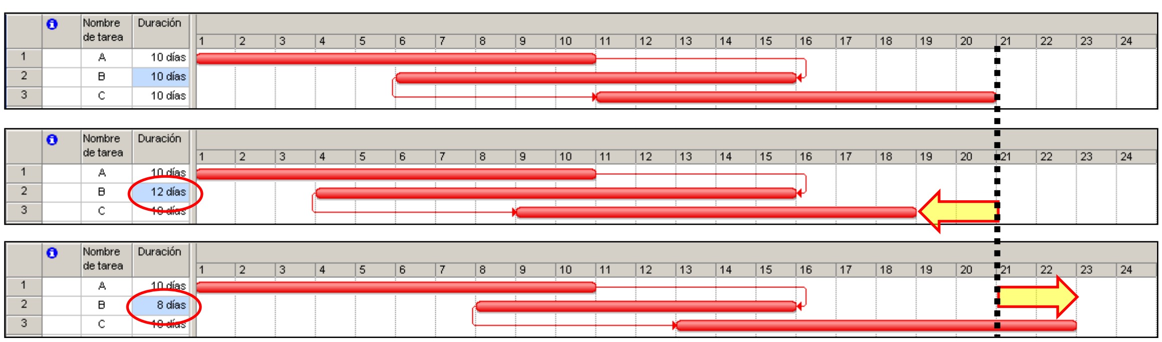

Figure 15 Reverse Criticality with Microsoft Project

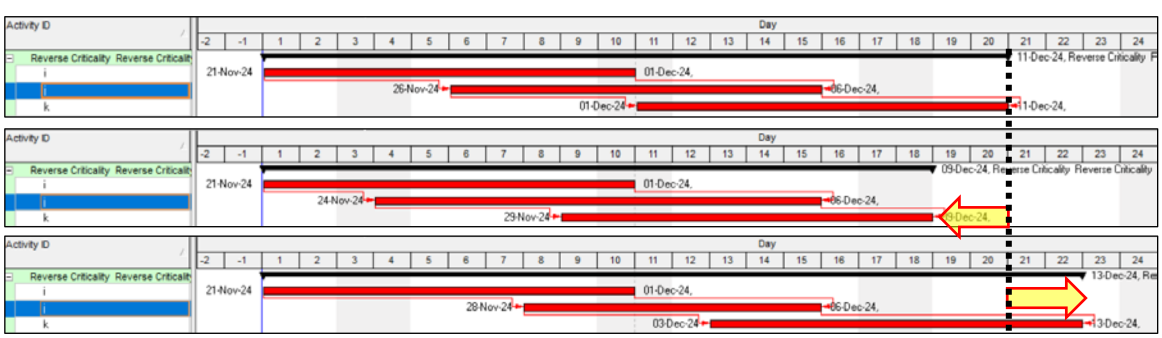

Figure 16 Limited Reverse Criticality with Primavera P6

Wiest (1981), proposed the splitting activities as a potential solution for reverse criticality. This approach involves untying the start by the finishing, of the affected activity. However, Wiest did not provide a solution for the optimal splitting of activities, and raising questions about the feasibility (both technically and managerially) of splitting a job, and, if so, how should the split be made? (View Continuous execution and optimal splitting).

Ponz-Tienda (2010; 2015), agree with Wiest arguing that continuity of activities must be discretional for schedulers, and not imposed by the algorithms, addressing the issue of reverse criticality by two ways: proposing a near-optimal algorithm for the discretionary splitting of activities, and the use of three distinct types of lags for relationships based of a production planning approach: Feeding relationships as a percentage of production; Time lags as effective work periods, and Additional Delays, as idle periods.

Under a production planning approach, relationships are no longer defined by just a delay z but by one, three, or five parameters depending on the type of relationship (Figure 17 and Figure 18), being defined as follows:

- The *Finish-to-Start* ($FS_{ij}(z)$) precedence relationship represents the minimum number of z time-periods that must elapse between the completion of the predecessor activity i, and the start of the follower activity j.
- The *Start-to-Start* ($SS_{ij}(w_i|p_i|z)$) precedence relationship represents the minimum percentage p_i, or w_i effective work-periods, required on the predecessor activity i, prior to the start of the successor activity j, with an additional lag of z time-periods.
- The *Finish-to-Finish* ($FF_{ij}(w_j|p_j|z)$) precedence relationship represents the minimum percentage of production quantity p_j, or w_j effective work-periods required on the follower activity j, after the completion of its predecessor i, with an additional lag of z time-periods.
- The *Start-to-Finish ($SF_{ij}(w_i|p_i|w_j|p_j|z)$)* precedence relationship represents the minimum p_j or/and w_j effective work-periods required on the follower activity j after the minimum number of p_i or/and w_i work-periods on the predecessor activity i has been completed, with an additional lag of z time-periods.

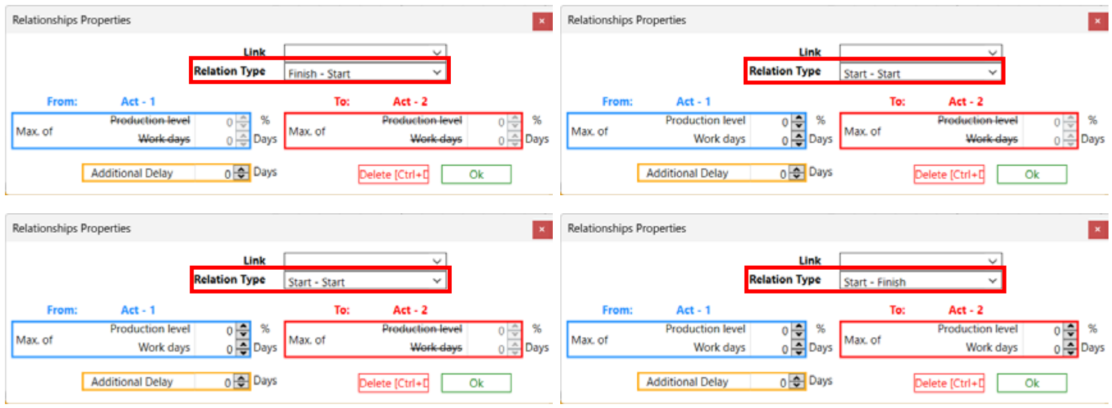

Figure 17 Types of lags for relationships with Plexos Project PPM

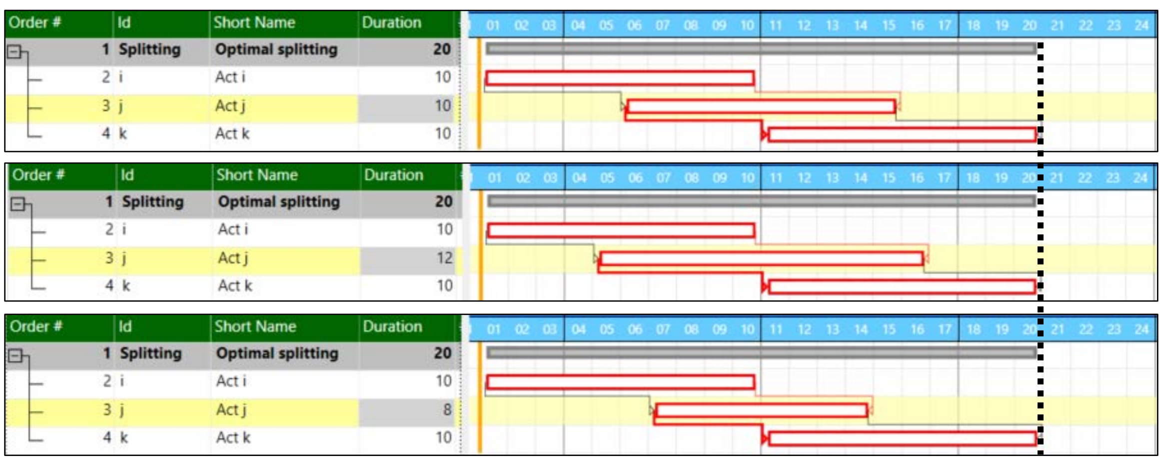

Figure 18 Avoided Reverse Criticality with Feeding relationships and Plexos Project PPM

Unrealistic calendar usage

Project scheduling and management requires the use of multiple calendars assigned to the resources assigned to the activities, and the activities itself, in such a way that the resulting calendar is the Boolean multiplication of both assignments. Note that the result for the resources is the same Boolean multiplication for the resources.

Activity	Resources	Result
0	0	0
0	1	0
1	0	0
1	1	1

Table 2 Boolean multiplication for calendar assignments

The problem arises in the calendar assignment for the relationships, and the unclear criterion used by commercial applications, resulting in different schedules depending on the software used by the practitioner (Kyunghwan & de la Garza, 2005).

There are different criteria on which calendar should be applied to the relationship, the resulting calendar for the predecessor activity, or the successor one instead? It seems that the criterion that should be used to answer this question is asking the relationship: Which is the owner activity of the condition under a production planning approach as stated in the previous section.

The *Start-to-Start* relationship refers to the needs from the predecessor activity for starting the successor one, and consequently its calendar is the calendar of the predecessor activity. In a similar way, The *Finish-to-Finish* relationship refers to restrictions for finishing the successor activity, and consequently its calendar is the calendar of the successor activity.

For the *Finish-to-Start* relationship, there's no owner because there's not any production condition between the predecessor and successor activity, so a natural calendar must be applied, even when negative lags are involved, as exemplified by concrete hardening and curing.

	Calendar		
Relationship	**Predecessor**	**Successor**	**Natural**
Finish-to-Start			✓
Start-to-Start	✓		
Finish-to-Finish		✓	
Start-to-Finish	✓	✓	

Table 3 Calendar assignment to relationships

Let's consider the scheduling of a 10-story building under the following conditions:

1. Slab durations: 3, 5 and 7 days,
2. Column durations: 2 days,
3. Project start days: Monday, Wednesday, and Friday with an official calendar for an arbitrary city,
4. Unshoring periods for each slab, as shown in Figure 19.

This project has been scheduled with 4 different commercial software solutions, and with the following alternatives:

1. Microsoft project, modeled with *Start-to-Start* relationships, and with *Finish-to-Finish* relationships[3],
2. Open project, modeled with *Start-to-Start* relationships, and with *Finish-to-Finish* relationships as with Microsoft Project
3. Primavera P3

[3] Remember that Microsoft uses standardized relationships and doesn't allow simultaneous relationships

4. Primavera P6, computing times using two calendar relationship options:
    a. Applying successor calendar,
    b. Applying predecessor calendar.

Slab	Unshoring Days
Slab 1	12
Slab 2	11
Slab 3	10
Slab 4	9
Slab 5	8
Slab 6	7
Slab 7	6
Slab 8	5
Slab 9	4
Slab 10	3

Element	Duration	Project Start
Slabs	3 days	Monday
	5 days	Wednesday
	7 days	Friday
Columns	2 days	

Figure 19 case study for calendar usage

The durations obtained in calendar days are shown in Figure 20.

Slab Duration	Start Day	Plexos Project	Microsoft Project		Open Project		Primavera P3	Primavera P6	
			SS	FF	SS	FF		Succ.	Predec.
3	Monday	51	47	51	51	65	65	51	59
	Wednesday	49	48	49	49	63	65	49	65
	Friday	50	49	50	50	67	67	50	63
5	Monday	73	72	73	71	80	81	73	81
	Wednesday	76	76	76	76	78	85	76	79
	Friday	76	75	76	76	83	84	76	83
7	Monday	101	101	101	101	106	102	101	102
	Wednesday	104	104	104	104	106	108	104	106
	Friday	104	104	104	104	109	106	104	105

Figure 20 Comparison of Project Durations (Calendar Days)

Figure 21 presents the percentage deviation from the Plexos Project schedule, calculated by applying the appropriate relationship calendar criteria. As can be seen, the different criteria applied for computing the times of the activities have an important effect on the resulting solution, and the substantial differences observed between the resulting scenarios are remarkable, even up to 32%.

Slab Duration	Start Day	Plexos Project	Microsoft Project		Open Project		Primavera P3	Primavera P6	
			SS	FF	SS	FF		Succ.	Predec.
3	Monday		7.84%			-27.45%	-27.45%		-15.69%
	Wednesday		2.04%			-28.57%	-32.65%		-32.65%
	Friday		2.00%			-34.00%	-34.00%		-26.00%
5	Monday		1.37%		2.74%	-9.59%	-10.96%		-10.96%
	Wednesday					-2.63%	-11.84%		-3.95%
	Friday		1.32%			-9.21%	-10.53%		-9.21%
7	Monday					-4.95%	-0.99%		-0.99%
	Wednesday					-1.92%	-3.85%		-1.92%
	Friday					-4.81%	-1.92%		-0.96%

Figure 21 Percentage deviation compared to Plexos Project schedule

A preliminary comparative analysis of project scheduling software reveals that only Microsoft Project with *Finish-to-Finish* relationships, and Primavera P6 with successor's calendar provide correct durations. Microsoft Project's algorithm tends to yield more optimistic estimates, while Primavera P6's often result in more conservative durations.

Nevertheless, it is important to note that the accuracy of duration calculations in Microsoft Project (with SS relationships) and Primavera P6 (with the successor's calendar) can be influenced by specific project parameters.

Continuous execution and optimal splitting

As exposed previously, Wiest (1981) and Ponz-Tienda (2010; 2015) proposed the "discretional fragmentation" of activities to mitigate the effects of reverse criticality. Nevertheless, implementing this strategy is not easy, requiring solving an optimization problem under a production planning approach, considering two types of time periods: work-periods, and idle periods (non-working periods).

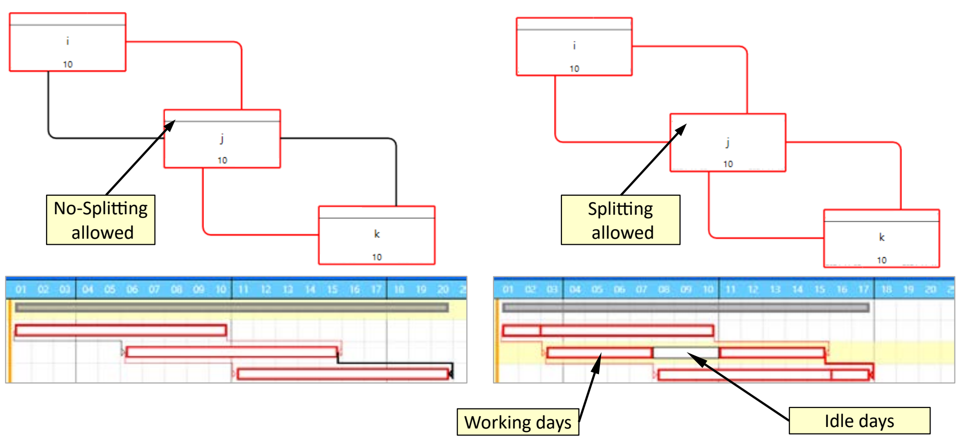

Figure 22 Near-optimal splitting of activities with Plexos Project PPM

Discretional fragmentation for two splits and unlimited relationships can be solved by applying the near-optimal algorithm proposed by Ponz-Tienda (2010; 2015), that is based on stablish the second split of the activity by the most restrictive finishing production restriction (w_j and p_j $parameters$) for all the *Finish-to-Finish* and *Start-to-Finish* relationships. Figure 22 shows the near-optimal splitting of activities with Plexos Project PPM.

In addition to mitigating the effects of reverse criticality, the fragmentation of activities usually provides shorter and more realistic schedules, especially when working with repetitive activities (Figure 29).

Space-time conflicts

Traditional PDM networks provide valuable temporal information, bu do not provide information about where the activity is being executed, arising the problem of checking space-time execution conflicts, even when using simultaneous *Start-to-Start* and *Finish-to-Finish* relationships, and especially complex when non-lineal production rates are involved, as can be seen in Figure 23.

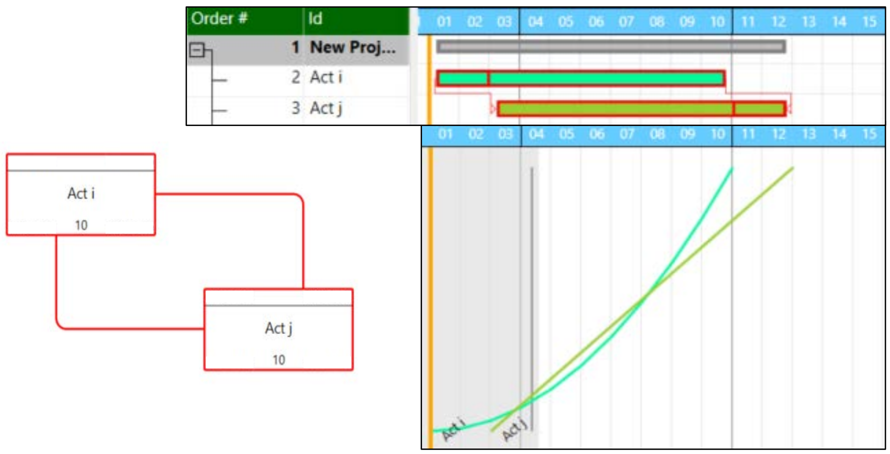

Figure 23 space-time execution conflict with non-linear production rate

Several solutions for this issue have been proposed by researchers and practitioners, outstanding the application of the Line of Balance method in the construction schedule for the “Empire State Building” by Starrett Brothers & Eken, Inc., a prominent construction firm known for their expertise in building skyscrapers (Historic Construction Projects, s.f.). completing the project in just over a year (from 1930 to 1931), which was a remarkable feat at the time.

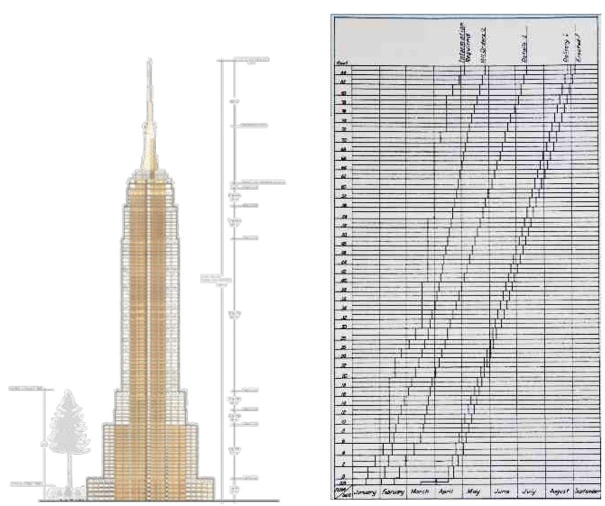

Figure 24 Line of Balance in the schedule for the “Empire State Building”.

The first known application of repetitive activities concept to scheduling networks was proposed by Schoderbek & Digman (1967), showing them on the network as an open box on the activity arrow, and a repetitive event by an X in a circle Figure 25.

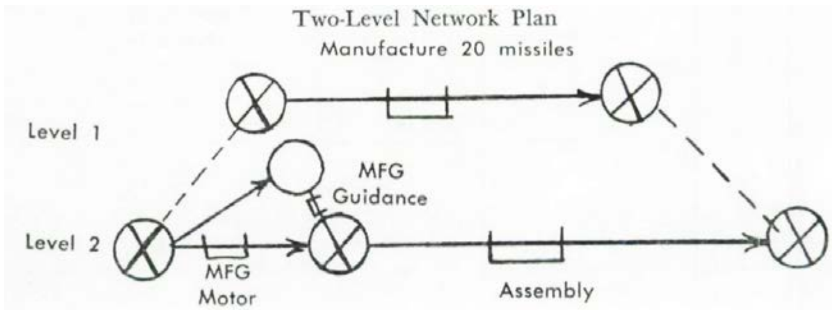

Figure 25 Repetitive activities network by Schoderbek & Digman (1967)

Later, O´Brien (1969; 1975; 1985) developed the Vertical Production Method (VPM) for projects of multi-story buildings, being widely accepted under different names as Linear Scheduling Method (Barrie, 1978), Time-Space Scheduling Method (Stradal & Cacha, 1982), Repetitive Scheduling Method (Harris & Ioannou, 1998); and currently, known as Flow-lines by Ponz-Tienda (2015), and Location-Based Scheduling by Seppänen et al (2014).

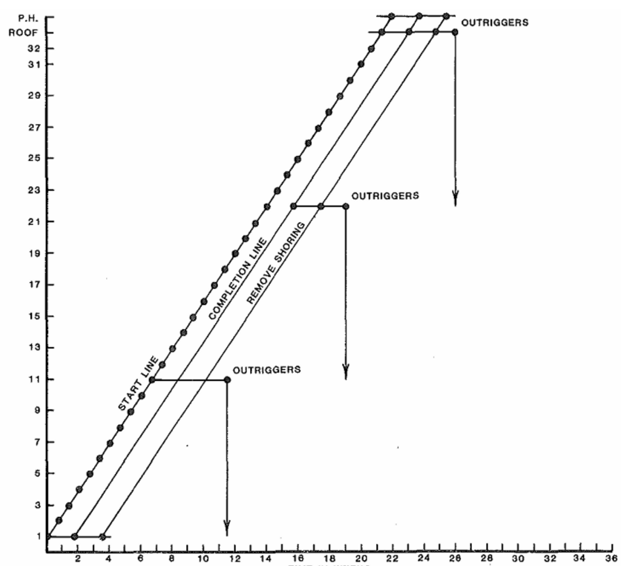

Figure 26 Vertical Production Method or VPM (O'Brien, Kreitzberg, & Mikes, 1985)

The relationships between two activities of repetitive activities are notated by the leading activities i and j, the predecessor and successor sub-activities of the *spatial-buffer* $(i_{SAct} \mid j_{SAct})$, and the relation itself (*temporal-buffer*) can be seen in Equation 1.

$$i_{SAct} | j_{SAct} \left\{ \begin{array}{c} FS\ (z) \\ SS\ (w_i | p_i | z) \\ FF\ (w_i | p_i | z) \\ SF\ (w_i | p_i | w_j | p_j | z) \end{array} \right\} \qquad \text{Equation 1}$$

The representation of the different relationships between activities in a two-dimensional line of balance diagram can be seen in Figure 27.

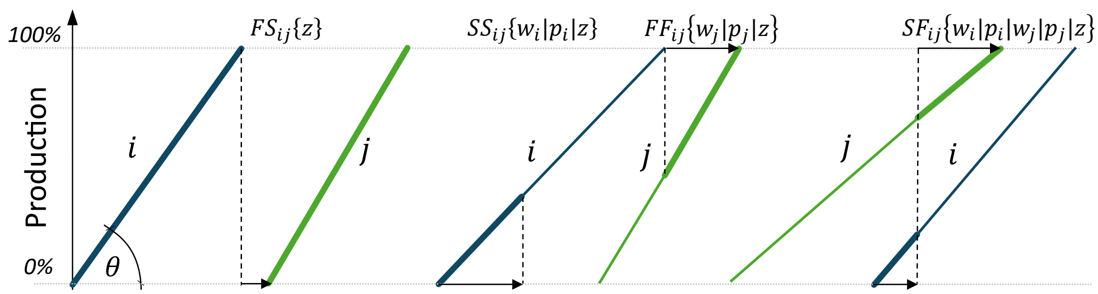

Figure 27 Space-time representation for different relationships.

In Figure 28 is exposed the graphical interpretation of the relationships between activities of repetitive activities for the case of continuity execution of sub-activities (LHS), and non-continuous execution (RHS).

$$i_2 | j_1\ (FS\ (0)) \qquad \text{Equation 2}$$

$$j_1 | k_1\ (FS\ (0)) \qquad \text{Equation 3}$$

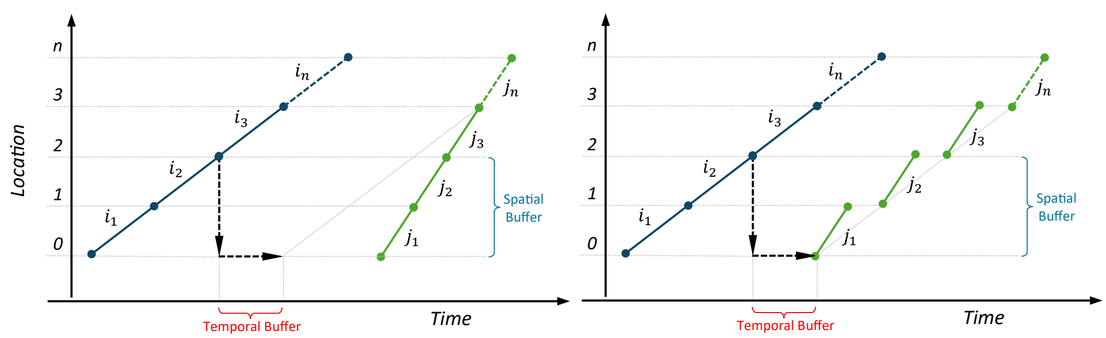

Figure 28 Continuous and non-continuous execution of sub-activities

In figures Figure 29 is shown the continuous and non/continuous cases with Project and the relationships exposed in Equation 2 and Equation 3.

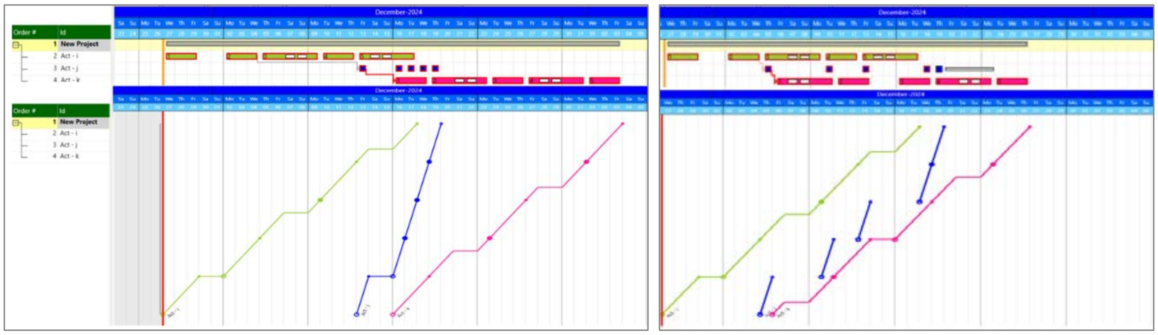

Figure 29 Continuous and non-continuous execution of sub-activities with Plexos Project

The utilization of space-time diagrams enables the clear visualization of production rhythms between activities. As depicted in Figure 29, the blue activity, characterized by a distinct production rate and continuous execution, interrupts the production flow. Nonetheless, by permitting discontinuous execution, the flow can be re-established, providing a more harmonious flow.

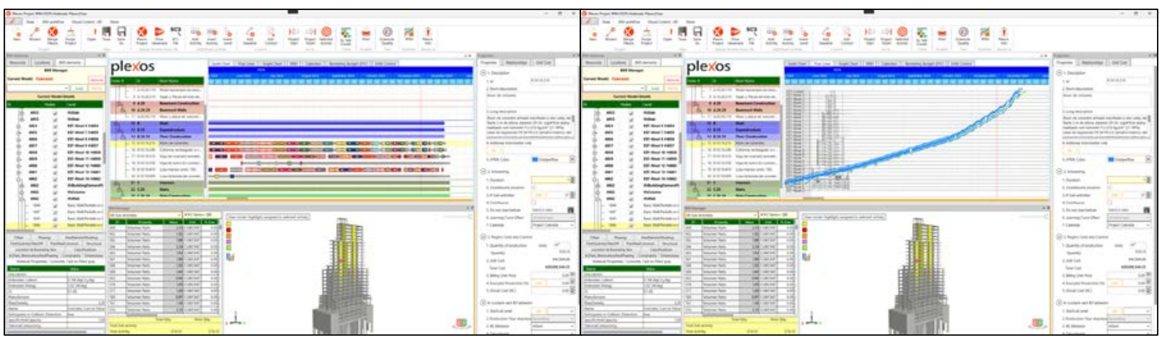

Figure 30 Gantt chart and Flow-lines chart with repetitive activities.

Figure 30 presents a real-world Gantt chart (LHS) and flowchart (RHS), illustrating repetitive activities for the foundation, substructure, and superstructure of a 30-story building. Additionally, Figure 31 displays the PDM graph, and Figure 29 the Takt graph.

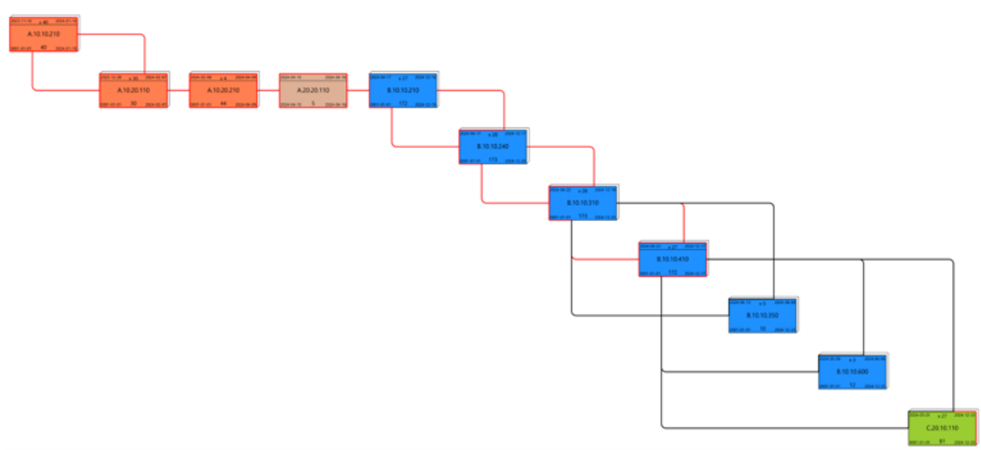

Figure 31 Real-world Project Expanded Graph (IBM Original PDM Graph)

Figure 32 Real-world Project; Takt Graph

Final comments

The state-of-the-art analysis presented in this report indicates that PDM networks are a highly effective method for modeling construction projects and can be readily adapted to accommodate emerging methodologies such as Lean Construction and Virtual Design and Construction.

The inherent weakness of PDM networks doesn't lie in the method itself, but rather in the way they are implemented in software packages and the underlying algorithms. This can lead to limitations for practitioners in construction project planning and budgeting, and in some cases, can even produce incorrect solutions.

Finally, software developers should disclose to users the underlying methodology, technology, and criteria employed for computing project schedules. This transparency would allow users to evaluate the software's suitability for specific AEC industry requirements, and their own unique needs.

References

Barrie, G. &. (1978). *Profesional Construction Manageemnt*. New York: McGraw-Hill Inc.

Behnam, A., Harfield, T., & Kenley, R. (2016). Linear vs. systems approach to production scheduling in construction. *33rd CIB W78 Conference*. Brisbane.

Fondahl, J. (1962). *A Non-Computer Approach to the Critical Path Method for the Construction Industry*. San Francisco: Department of Civil Engineering, Stanford University, Stanford, CA.

Fondahl, J. (1987). The history of modern project management: precedence diagramming method: Origins and early developments. *Project Management Journal, 18*(2), 33-36.

Harris, R., & Ioannou, P. (1998). Scheduling projects with repeating activities. *Journal of construction engineering and management, 124*(4), 269-278. doi:10.1061/(ASCE)0733-9364(1998)124:4(269

Historic Construction Projects. (n.d.). Retrieved from http://constructioncompany.com/historic-construction-projects/empire-state-building/

IBM. (1964). *Users' Manual for IBM 1440 Project Control System (PCS)*. 821 United Nations Plaza, New York, New York 10017: International Business Machines Corporation .

Kelley, J., & Walker, M. (1959). Critical-path planning and scheduling. *IRE-AIEE-ACM computer conference* (pp. 160–173). Boston, Massachusetts: Association for Computing Machinery. New York, NY, USA. doi:10.1145/1460299.1460318

Koskela, L., Howell, G., Pikas, E., & Dave, B. (2014). If CPM is so bad, why have we been using it so long? *The 22th International Group for Lean Construction conference, June 23-27*, (pp. 27-37). Oslo, Norway. Retrieved from https://eprints.hud.ac.uk/id/eprint/22989/

Kyunghwan, K., & de la Garza, J. (2005). Critical Path Method with Multiple Calendars. *Journal of Construction Engineering and Management, 131*(3), 330-342. doi:10.1061/~ASCE!0733-9364~2005!131:3~330

MacDermott, J. T. (1967). *Doctoral Thesis Project control using the problem-oriented language PROJECT.* Cambridge, Massachusetts; Massachusetts Institute of Technology. Retrieved from https://hdl.handle.net/10945/12902

Malcolm, D. G., J. H. Roseboom, C. E. Clark, & W. Frazar. (1959). Critical-path planning and scheduling. *Operations Research, 7*, 646-669.

O´Brien, J. (1969). *Scheduling handbook.* New York, N.Y: McGraw-Hill Inc.

O´Brien, J. (1975). VPM scheduling for high-rise buildings. *Journal of the Construction Division*, 895-905.

O'Brien, J. J., Kreitzberg, F. C., & Mikes, W. F. (1985). Network Scheduling Variations for Repetitive Work. *Journal of Construction Engineering and Management, 111*(2), 105-116. doi:10.1061/(ASCE)0733-9364(1985)111:2(105)

Ponz-Tienda, J. L. (2010). *Doctoral Thesis: GRCPSP Robusto basado en Producción para Proyectos de Edificación y Construcción.* Valencia: Universitat Politècnica de València.

Ponz-Tienda, J. L., Pellicer, E., Benlloch-Marco, J., & Andrés-Romano, C. (2015). The Fuzzy Project Scheduling Problem with Minimal Generalized Precedence Relations. *Computer-Aided Civil and Infrastructure Engineering, 30*(11), 872-891. doi:10.1111/mice.12166

Roy, G. B. (1959). Théorie des Graphes: Contribution de la théorie des graphes á l1 étude de certains problémes linéaries. *Comptes rendus des Séances de l1 Acedémie des Sciences.*, 2437-2449.

Roy, G. B. (1962). Graphes et ordonnancement (Graphs and scheduling). *Revue Française de Recherche Opérationnelle*(25), 323-333. Retrieved from http://ekt.bme.hu/Cikkek/606-2021_Roy_1962.pdf

Schoderbek, P., & Digman, L. (1967). Third Generation in PERT Systems. *Academy of Management*, 195–200.

Seppänen, O., Evinger, J., & Mouflard, C. (2014). Effects of the location-based management system on production rates and productivity. *Construction management and economics, 32*(6), 608-624. doi:10.1080/01446193.2013.853881

Stradal, O., & Cacha, J. (1982). Time Space Scheduling Method. *Journal of the Construction Division, 108*(3), 445-457. doi:10.1061/JCCEAZ.0001056

Wiest, J. (1978). Precedence Diagramming: Some Anomalous Effects of FF and SS Relatioships. *ORSA/TIMS Join National Meeting (November 13-17)*. Los Angeles.

Wiest, J. (1981). Precedence diagramming method: Some unusual characteristics and their implications for project managers. *Journal of Operations Management, 1*(3), 121-130. doi:10.1016/0272-6963(81)90015-2